

Prediksi Harga Saham BBCA Menggunakan Metode CNN, LSTM dan CNN-LSTM Berbasis Variasi Timestep dan Rasio Split pada Data OHLCV Time Series

Rifki Faiz Azzurananda, Siska Kurnia Gusti*, Elin Haerani, Eka Pandu Cynthia

Fakultas Sains dan Teknologi, Program Studi Teknik Informatika, Universitas Negeri Sultan Syarif Kasim Riau, Pekanbaru, Indonesia

Email: ¹12250113451@students.uin-suska.ac.id, ^{2,*}siskakurniagusti@uin-suska.ac.id, ³elin.haerani@uin-suska.ac.id,

⁴eka.pandu.cynthia@uin-suska.ac.id,

Email Penulis Korespondensi : siskakurniagusti@uin-suska.ac.id

Abstrak—Prediksi harga saham merupakan permasalahan yang menantang karena karakteristik data keuangan yang bersifat dinamis, non-linier, dan mengandung tingkat ketidakpastian yang tinggi. Meskipun berbagai model *deep learning* telah banyak digunakan untuk prediksi harga saham, masih terdapat keterbatasan dalam pemahaman mengenai pengaruh panjang konteks historis (*timestep*) dan proporsi data pelatihan terhadap kemampuan generalisasi model, khususnya pada data saham Indonesia. Penelitian ini mengimplementasikan dan membandingkan tiga model *deep learning*, yaitu *Convolutional Neural Network* (CNN), *Long Short-Term Memory* (LSTM), dan *hybrid CNN-LSTM*, untuk memprediksi harga saham PT Bank Central Asia Tbk (BBCA) menggunakan data OHLCV (*Open, High, Low, Close, Volume*) periode 2015–2025. Pengujian dilakukan menggunakan variasi *timestep* 30, 60, dan 90, serta rasio pembagian data 70:30, 80:20, dan 90:10. Performa model dievaluasi menggunakan *Root Mean Square Error* (RMSE), *Mean Absolute Percentage Error* (MAPE), dan koefisien determinasi (R^2). Hasil pengujian menunjukkan bahwa model CNN memiliki performa paling konsisten di berbagai skenario, sedangkan LSTM mampu menangkap pola temporal meskipun masih menunjukkan kecenderungan *lagging* pada kondisi tertentu. Model *hybrid CNN-LSTM* menghasilkan performa terbaik pada konfigurasi *timestep* 90 dengan pembagian data 80:20, ditunjukkan oleh nilai RMSE 130.54 (IDR), MAPE 1.0669%, dan R^2 0.9581. Temuan ini menunjukkan bahwa kombinasi kemampuan CNN dalam mengekstraksi pola lokal dan kemampuan LSTM dalam menangkap dependensi temporal jangka panjang mampu meningkatkan akurasi prediksi dibandingkan penggunaan model tunggal pada data saham BBCA berbasis OHLCV.

Kata Kunci: Prediksi Saham; CNN; LSTM; CNN-LSTM; Time Series

Abstract—Predicting stock prices is a challenging problem because financial data is dynamic, non-linear, and contains a high level of uncertainty. Although various deep learning models have been widely used for stock price prediction, there remains a limited understanding of the effect of historical context length (*timestep*) and training data proportion on model generalization ability, particularly for Indonesian stock data. This study implements and compares three deep learning models, namely *Convolutional Neural Network* (CNN), *Long Short-Term Memory* (LSTM), and *hybrid CNN-LSTM*, to predict the stock price of PT Bank Central Asia Tbk (BBCA) using OHLCV (*Open, High, Low, Close, Volume*) data from 2015 to 2025. Experiments were carried out using timesteps of 30, 60, and 90, combined with data split ratios of 70:30, 80:20, and 90:10. Model performance was measured using *Root Mean Square Error* (RMSE), *Mean Absolute Percentage Error* (MAPE), and the coefficient of determination (R^2). The results show that CNN gave the most stable performance across all tested scenarios, while LSTM was able to pick up temporal patterns but still showed *lagging* in some conditions. The *hybrid CNN-LSTM* model came out on top at *timestep* 90 with an 80:20 data split, reaching an RMSE of 130.54 (IDR), MAPE of 1.0669%, and R^2 of 0.9581. This shows that the combination of CNN's ability to extract local patterns and LSTM's ability to capture long-term temporal dependencies improves prediction accuracy compared to using single models on BBCA stock data based on OHLCV.

Keywords: Stock Prediction; CNN; LSTM; CNN-LSTM; Time Series

1. PENDAHULUAN

Pertumbuhan ekonomi suatu negara tidak terlepas dari peran aktif pasar modal sebagai penghubung antara investor dengan perusahaan yang memerlukan modal untuk pengembangan usaha [1]. Di antara berbagai instrumen yang tersedia, saham menjadi pilihan utama investor karena menawarkan potensi keuntungan yang kompetitif. Namun demikian, harga saham sangat rentan terhadap perubahan kondisi makroekonomi, kebijakan suku bunga, tekanan inflasi, maupun sentimen pasar dan performa internal perusahaan [2]. Sehingga prediksi harga saham menjadi tantangan tersendiri yang membutuhkan pendekatan analisis yang tepat dan akurat.

Dalam konteks pasar modal Indonesia, sektor perbankan merupakan salah satu sektor utama yang memiliki pengaruh besar terhadap stabilitas pasar. Salah satunya yaitu PT Bank Central Asia Tbk (BBCA), yang merupakan perusahaan perbankan terbesar di Bursa Efek Indonesia (BEI) dengan kapitalisasi pasar tinggi dan kondisi fundamental yang stabil [3]. Kondisi tersebut menjadikan saham BBCA aktif diperdagangkan dan diminati investor. Meskipun memiliki kondisi keuangan yang kuat, harga saham BBCA tetap mengalami fluktuasi dari waktu ke waktu sehingga sulit diprediksi secara konsisten. Hal ini menunjukkan bahwa kestabilan fundamental perusahaan tidak selalu diikuti oleh kestabilan harga saham di pasar.

Pergerakan harga saham termasuk ke dalam data *time series*, yaitu data yang tersusun berdasarkan urutan waktu secara kronologis [4]. Data *time series* memiliki karakteristik kompleks, dinamis, dan bersifat non-linear sehingga sulit diprediksi menggunakan metode statistik konvensional. Selain itu, data saham juga mengandung *noise* yang cukup tinggi akibat pengaruh faktor eksternal maupun internal pasar [5]. Keberadaan *noise* menyebabkan pola pergerakan harga sulit dikenali



karena perubahan harga dapat terjadi secara tiba-tiba dalam waktu singkat. Penelitian sebelumnya menunjukkan bahwa penggunaan data historis jangka panjang yang diproses melalui teknik *windowing* dan normalisasi mampu membantu model mempelajari pola temporal kompleks secara lebih efektif, sehingga menghasilkan akurasi prediksi tinggi pada sektor perbankan Indonesia [6]. Oleh sebab itu, dibutuhkan pendekatan yang mampu mempelajari pola historis secara mendalam untuk menangkap hubungan temporal pada data saham.

Seiring perkembangan kecerdasan buatan, pendekatan *deep learning* telah membuka peluang baru dalam analisis data *time series* finansial. *Long Short-Term Memory* (LSTM), sebagai turunan dari *Recurrent Neural Network* (RNN), dirancang khusus untuk mempertahankan informasi jangka panjang pada data sekuensial [7]. Kemampuan tersebut membuat LSTM efektif digunakan dalam prediksi harga saham karena dapat menangkap hubungan temporal antar data historis. Penelitian sebelumnya menunjukkan bahwa model LSTM mampu menghasilkan performa yang baik pada prediksi harga saham BBCA dengan nilai MAPE sebesar 0,96% dan nilai R^2 sebesar 0,98 [6]. Namun, model ini memiliki kelemahan berupa kecenderungan prediksi yang lebih datar dan mengalami *lagging* [6]. Secara teknis, kondisi ini berkaitan dengan cara kerja *cell state* pada LSTM yang cenderung mempertahankan representasi historis yang dominan [8], sehingga ketika terjadi perubahan harga yang tajam dan mendadak, model tidak dapat merespons secara cepat karena masih terikat pada konteks temporal sebelumnya. Pada data saham seperti BBCA yang pergerakannya dapat berubah tiba-tiba akibat sentimen pasar dan kondisi makroekonomi Indonesia, karakteristik ini menjadikan *lagging* sebagai kelemahan yang relevan dan perlu diatasi. Temuan Zhang juga mengonfirmasi bahwa meskipun LSTM memiliki kemampuan prediksi lebih baik dibandingkan ARIMA pada data saham, model ini masih menunjukkan kecenderungan *lagging* terhadap perubahan tren yang terjadi secara drastis dalam jangka pendek, sehingga diperlukan tambahan fitur atau kombinasi arsitektur untuk meningkatkan responsivitas prediksi [9]. Di sisi lain, *Convolutional Neural Network* (CNN) juga semakin banyak diaplikasikan dalam analisis data *time series*.

Kemampuan CNN dalam mengekstraksi pola lokal dari data sekuensial menjadikannya efektif untuk menangkap pergerakan harga jangka pendek [10]. Pada data saham, CNN mengekstraksi pola pergerakan harga jangka pendek melalui proses konvolusi yang mampu mengidentifikasi fitur-fitur penting secara otomatis. Menurut Kiranyaz dkk., arsitektur 1D-CNN memiliki keunggulan dalam memproses data sekuensial secara langsung tanpa memerlukan transformasi ke bentuk dua dimensi, sehingga lebih efisien dalam mengenali pola lokal pada data *time series* dengan kompleksitas komputasi yang relatif rendah [11]. Temuan Mohan menunjukkan bahwa model CNN secara mandiri memiliki performa yang lebih unggul dibandingkan metode konvensional seperti ARIMA dan Prophet dalam konteks prediksi data finansial [10].

Berdasarkan karakteristik masing-masing metode, beberapa penelitian mulai mengombinasikan CNN dan LSTM ke dalam model *hybrid* CNN-LSTM. Studi eksperimental oleh Lara-Benítez dkk. menunjukkan bahwa LSTM dan CNN merupakan dua arsitektur *deep learning* yang paling konsisten menghasilkan performa tinggi pada *forecasting time series*, dengan LSTM unggul dalam akurasi sedangkan CNN lebih efisien secara komputasi [12]. Temuan tersebut menunjukkan bahwa kedua model memiliki keunggulan yang saling melengkapi sehingga integrasinya berpotensi menghasilkan model prediksi yang lebih optimal. Pendekatan *hybrid* ini bertujuan untuk menggabungkan kemampuan CNN dalam mengekstraksi pola lokal dan kemampuan LSTM dalam menangkap hubungan temporal jangka panjang [13]. Dengan kombinasi tersebut, model diharapkan mampu menghasilkan prediksi yang lebih adaptif terhadap fluktuasi harga saham. Penelitian Wibowo menunjukkan bahwa model *hybrid* CNN-LSTM mampu menghasilkan performa prediksi yang lebih baik dibandingkan model tunggal pada prediksi harga Bitcoin [14]. Selain itu, penelitian yang dilakukan oleh Handayani dkk. menegaskan bahwa integrasi arsitektur CNN-LSTM terbukti efektif dalam memprediksi fluktuasi harga emas sebagai data *time series*. Dalam studinya, kombinasi tersebut mampu menghasilkan tingkat akurasi yang tinggi karena kombinasi antara CNN dalam mengekstraksi fitur spasial dan LSTM dalam menangkap pola temporal jangka panjang [15].

Selain pemilihan model, penggunaan variabel *input* juga mempengaruhi performa prediksi. Banyak penelitian sebelumnya hanya menggunakan harga penutupan sebagai variabel utama [6]. Padahal, data saham memiliki informasi yang lebih lengkap melalui variabel *Open, High, Low, Close, Volume* (OHLCV). Variabel OHLCV banyak digunakan dalam penelitian prediksi harga saham karena menyediakan informasi pergerakan harga dan aktivitas perdagangan yang lebih lengkap dibandingkan penggunaan satu variabel harga saja, sehingga berpotensi meningkatkan kemampuan model dalam mempelajari pola data [16]. Oleh karena itu, penelitian ini menggunakan data OHLCV agar model memperoleh informasi yang lebih lengkap selama proses pelatihan.

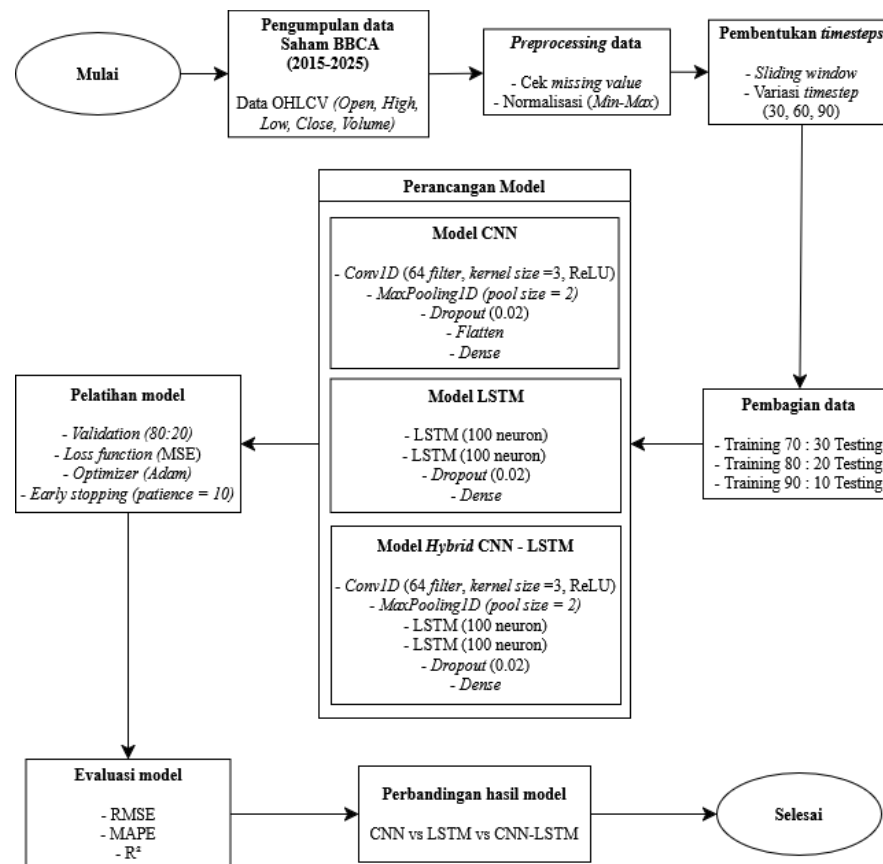
Berdasarkan uraian tersebut, penelitian ini mengimplementasikan tiga model *deep learning*, yaitu CNN, LSTM, dan *hybrid* CNN-LSTM untuk memprediksi harga saham PT Bank Central Asia Tbk (BBCA) menggunakan data historis periode 2015–2025. Perbandingan dilakukan pada berbagai konfigurasi eksperimen dengan evaluasi menggunakan metrik RMSE, MAPE, dan R^2 untuk mengevaluasi dan membandingkan performa ketiga model dalam menghasilkan prediksi harga saham yang akurat pada dinamika pasar saham Indonesia.

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian



Tahapan penelitian yang dilakukan pada penelitian ini ditunjukkan pada Gambar 1.



Gambar 1. Flowchart Tahapan Penelitian

Gambar 1 menunjukkan alur tahapan penelitian yang berawal dari pengumpulan data historis saham, dilanjutkan dengan *preprocessing* data untuk memastikan kualitas dan konsistensi dataset, kemudian pembentukan *timesteps* sebagai representasi sekuensial data. Setelah itu, data dibagi menjadi data latih dan data uji, lalu digunakan untuk merancang tiga model *deep learning* yaitu CNN, LSTM, dan *hybrid* CNN-LSTM. Ketiga model tersebut dilatih secara bersamaan, kemudian dievaluasi menggunakan metrik performa, dan terakhir dilakukan perbandingan hasil antar model untuk menentukan model dengan akurasi prediksi terbaik.

2.2 Pengumpulan Data

Penelitian ini menggunakan data historis harga saham PT Bank Central Asia Tbk (BBCA). Data diperoleh dari Yahoo Finance. Data yang digunakan merupakan data harian dengan rentang waktu dari tahun 2015 hingga 2025. Penelitian ini menggunakan data OHLCV untuk merepresentasikan pergerakan harga saham secara lebih lengkap. Tabel 1 menunjukkan variabel yang digunakan pada penelitian ini.

Tabel 1. Deskripsi Variabel Dataset

Variabel	Keterangan
<i>Open</i>	Harga pembukaan saham
<i>High</i>	Harga tertinggi saham
<i>Low</i>	Harga terendah saham
<i>Close</i>	Harga penutupan saham
<i>Volume</i>	Volume transaksi saham

Data historis saham yang diperoleh kemudian digunakan sebagai dataset dalam proses pelatihan dan pengujian model CNN, LSTM, dan *hybrid* CNN-LSTM.

2.3 Preprocessing Data



Tahap *preprocessing* dilakukan untuk mempersiapkan data sebelum digunakan pada proses pelatihan model. Proses pertama yaitu pengecekan *missing value* untuk memastikan kualitas dataset yang digunakan. Selanjutnya, data akan dinormalisasi menggunakan metode *MinMaxScaler* yang nantinya seluruh fitur berada pada rentang nilai 0 hingga 1. Normalisasi dilakukan untuk mengurangi perbedaan skala antar fitur dan membantu proses pelatihan model menjadi lebih optimal. Pendekatan normalisasi menggunakan *MinMaxScaler* merupakan tahap *preprocessing* yang umum diterapkan dalam prediksi harga saham berbasis data OHLCV pada pasar saham Indonesia [17]. Persamaan normalisasi *MinMaxScaler* ditunjukkan pada Persamaan (1).

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (1)$$

Pada Persamaan (1), X_{norm} menyatakan data hasil normalisasi, sedangkan X merupakan data asli sebelum dilakukan proses normalisasi. Nilai X_{min} dan X_{max} masing-masing merepresentasikan nilai minimum dan maksimum dari data yang digunakan. Dengan demikian, proses normalisasi dilakukan dengan memetakan setiap nilai data asli ke dalam rentang tertentu berdasarkan nilai minimum dan maksimum pada dataset. Tahap normalisasi dilakukan terhadap seluruh variabel OHLCV sehingga data memiliki distribusi nilai yang lebih seragam sebelum digunakan pada model *deep learning*.

2.4 Pembentukan *Timestep*

Setelah proses normalisasi selesai, data kemudian disusun menjadi format *time series* menggunakan metode *sliding window*. Teknik *sliding window* dengan variasi ukuran *window* terbukti secara signifikan memengaruhi performa prediksi, di mana pemilihan ukuran *window* yang tepat berperan penting dalam menentukan seberapa banyak konteks historis yang dimanfaatkan model untuk menghasilkan prediksi yang akurat [18]. Pada penelitian ini digunakan variasi *timestep* 30, 60, dan 90 hari untuk menganalisis pengaruh panjang konteks historis secara sistematis terhadap performa masing-masing model. Nilai *timestep* 30 dipilih untuk merepresentasikan pergerakan harga dalam rentang satu bulan perdagangan, *timestep* 60 untuk rentang dua bulan, dan *timestep* 90 untuk rentang tiga bulan, sehingga ketiga variasi ini mencakup spektrum konteks historis jangka pendek hingga menengah secara bertahap. *Timestep* digunakan untuk menentukan jumlah data historis yang digunakan sebagai input model dalam memprediksi harga saham berikutnya. Sebagai contoh, pada *timestep* 30, model menggunakan 30 hari data historis untuk memprediksi harga saham pada hari berikutnya.

2.5 Pembagian Data

Kemudian dataset yang telah dibentuk menjadi data *time series* akan dibagi lagi menjadi data latih dan data uji dengan menggunakan tiga skenario pembagian data, yaitu 70:30, 80:20, dan 90:10. Hal ini dilakukan untuk mengetahui apakah jumlah data pelatihan memiliki pengaruh terhadap kemampuan model dalam melakukan prediksi.

Tabel 2. Split Data

Split Data	Data Latih	Data Uji
70:30	1732	743
80:20	1980	495
90:10	2227	248

Pembagian data dilakukan secara kronologis (*chronological split*), bukan secara acak (*random split*). Data diurutkan berdasarkan urutan waktu sejak tahap *loading* data, kemudian dipisahkan berdasarkan indeks posisi sehingga data latih seluruhnya berasal dari periode lebih awal dan data uji dari periode lebih akhir. Tidak ada proses pengacakan (*shuffle*) di seluruh *pipeline* penelitian ini. Pendekatan ini merupakan syarat mutlak dalam pemodelan *time series* karena penggunaan *random split* dapat menyebabkan *data leakage* yang membuat hasil evaluasi tidak valid. Data latih digunakan untuk proses pelatihan model, sedangkan data uji digunakan untuk mengevaluasi performa model dalam memprediksi harga saham.

2.6 Perancangan Arsitektur Model

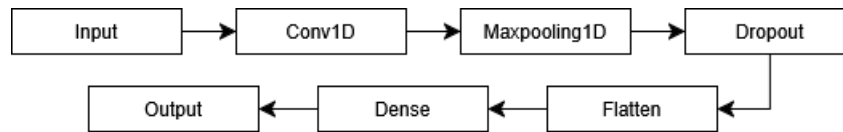
Penelitian ini menggunakan tiga model *deep learning*, yaitu CNN, LSTM, dan *hybrid* CNN-LSTM. Setiap model dirancang untuk mempelajari pola data *time series* dengan karakteristik yang berbeda.

2.6.1 Arsitektur CNN

Model CNN digunakan untuk mengekstraksi pola lokal pada data saham melalui proses konvolusi. Arsitektur CNN terdiri dari lapisan *Conv1D*, *MaxPooling1D*, *Dropout*, *Flatten*, dan *Dense*. Kemampuan 1D CNN dalam mengekstraksi pola dari data *time series* keuangan telah dibuktikan dalam klasifikasi pola grafik saham, di mana 1D CNN mengungguli metode lain seperti SVM, LSTM, dan *dynamic time warping* dalam mengenali pola pada data finansial [19].



Gambar 2 menunjukkan alur arsitektur model CNN yang digunakan pada penelitian ini.



Gambar 2. Diagram Arsitektur Model CNN

Berdasarkan Gambar 2, data input diproses secara berurutan melalui lapisan *Conv1D* untuk ekstraksi pola lokal, dilanjutkan dengan *MaxPooling1D* untuk reduksi dimensi, *Dropout* untuk regularisasi, *Flatten* untuk mengubah data menjadi satu dimensi, hingga akhirnya lapisan *Dense* menghasilkan nilai prediksi akhir.

Tabel 3 menunjukkan arsitektur model CNN yang digunakan pada penelitian ini secara lebih rinci.

Tabel 3. Arsitektur Model CNN

Layer	Parameter
<i>Conv1D</i>	64 filter, kernel size = 3, stride = 1, padding = same, ReLU
<i>MaxPooling1D</i>	pool size = 2, stride = 2 (default)
<i>Dropout</i>	rate = 0.02
<i>Flatten</i>	-
<i>Dense</i>	1 neuron, aktivasi linear

Pemilihan 64 filter dengan *kernel size* 3, *stride* 1, dan *padding* = same pada lapisan *Conv1D* bertujuan untuk mengekstraksi pola lokal dari data *time series* secara efisien, di mana *kernel size* 3 memungkinkan model menangkap hubungan antar tiga titik waktu yang berdekatan, *stride* 1 memastikan seluruh posisi sekuensial dipindai tanpa melewati data, dan *padding* = same menjaga dimensi output tetap sama dengan input sehingga tidak ada informasi yang hilang di ujung sekuens. Lapisan *MaxPooling1D* dengan *pool size* 2 dan *stride* 2 mereduksi dimensi fitur sekaligus mempertahankan informasi penting. *Dropout* dengan *rate* 0.02 diterapkan sebagai regularisasi untuk mencegah *overfitting*, sedangkan lapisan *Dense* dengan aktivasi *linear* menghasilkan nilai prediksi kontinu.

Proses konvolusi pada CNN digunakan untuk mengekstraksi fitur lokal dari data *time series*. Operasi konvolusi ditunjukkan pada Persamaan (2).

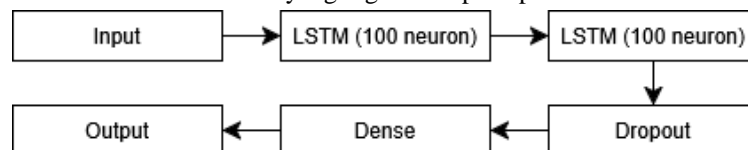
$$y_i = f \left(\sum_{k=1}^K \omega_k \cdot x_{i+k-1} + b \right) \quad (1)$$

Pada Persamaan (2), x menyatakan data input yang menjadi masukan pada operasi konvolusi, sedangkan ω_k merupakan bobot kernel yang berfungsi untuk mengekstraksi pola atau fitur lokal dari data *time series*. Parameter b menyatakan nilai bias yang ditambahkan untuk meningkatkan kemampuan model dalam mempelajari karakteristik data. Adapun y_i merupakan hasil konvolusi pada posisi ke- i , yang diperoleh melalui penjumlahan berbobot antara data input dan kernel, kemudian diteruskan ke fungsi aktivasi f untuk menghasilkan keluaran akhir.

2.6.2 Arsitektur LSTM

Model LSTM digunakan untuk mempelajari hubungan temporal antar data historis saham. LSTM dirancang dengan mekanisme *memory cell* yang memungkinkannya untuk menyimpan dan mempertahankan informasi dari data-data sebelumnya dalam jangka waktu yang lebih panjang.

Gambar 3 menunjukkan alur arsitektur model LSTM yang digunakan pada penelitian ini.



Gambar 3. Diagram Arsitektur Model LSTM

Berdasarkan Gambar 3, data input diproses secara berurutan melalui dua lapisan LSTM untuk menangkap dependensi temporal jangka panjang, dilanjutkan dengan *Dropout* untuk regularisasi, hingga akhirnya lapisan *Dense* menghasilkan nilai prediksi akhir. Tabel 4 menunjukkan arsitektur model LSTM yang digunakan pada penelitian ini secara lebih rinci.

Tabel 4. Arsitektur Model LSTM

Layer	Parameter
LSTM	100 neuron, <i>return_sequences</i> = True, aktivasi tanh (default)
LSTM	100 neuron, <i>return_sequences</i> = False, aktivasi tanh (default)
Dropout	rate = 0.02
Dense	1 neuron, aktivasi linear

Dua lapisan LSTM dengan masing-masing 100 unit digunakan untuk memberikan kapasitas yang cukup dalam menangkap dependensi temporal jangka panjang. Konfigurasi *return_sequences* = True pada lapisan pertama memungkinkan keluaran sekuensial di setiap *timestep* diteruskan ke lapisan LSTM berikutnya, sehingga lapisan kedua dapat memproses seluruh urutan temporal secara penuh. Sebaliknya, *return_sequences* = False pada lapisan kedua dikonfigurasi agar hanya keluaran pada *timestep* terakhir yang diteruskan ke lapisan berikutnya, karena pada tahap ini seluruh informasi temporal telah terangkum dalam satu vektor representasi akhir yang siap digunakan untuk prediksi. *Dropout* 0.02 dan lapisan *Dense* dengan aktivasi *linear* diterapkan untuk regularisasi dan menghasilkan nilai prediksi kontinu.

LSTM terdiri atas *forget gate*, *input gate*, dan *output gate* yang bertugas mengendalikan informasi apa saja yang perlu disimpan, diperbarui, maupun dikeluarkan dalam *memory cell*. Mekanisme gating ini memungkinkan LSTM untuk menangkap pola temporal dan hubungan jangka panjang dalam data sekuensial, sehingga efektif digunakan dalam prediksi harga saham [20]. Persamaan *forget gate* ditunjukkan pada Persamaan (3).

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{3}$$

Persamaan input gate ditunjukkan pada Persamaan (4).

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{4}$$

Persamaan *cell state* ditunjukkan pada Persamaan (5).

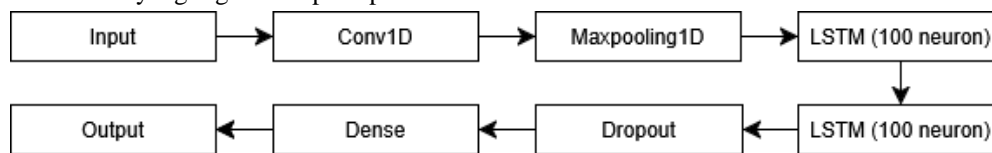
$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \tag{5}$$

Persamaan *output gate* ditunjukkan pada Persamaan (6).

$$h_t = o_t \odot \tanh(C_t) \tag{6}$$

2.6.3 Arsitektur CNN-LSTM

Model *hybrid* CNN-LSTM menggabungkan kemampuan CNN untuk mengenali pola-pola lokal pada data serta kemampuan LSTM dalam mempelajari keterkaitan antar data dalam rentang waktu yang panjang. Gambar 4 menunjukkan alur arsitektur model *hybrid* CNN-LSTM yang digunakan pada penelitian ini.



Gambar 4. Diagram Arsitektur Model CNN-LSTM

Berdasarkan Gambar 4, data input pertama-tama diproses melalui lapisan *Conv1D* dan *MaxPooling1D* untuk mengekstraksi pola lokal, kemudian hasil ekstraksi tersebut diteruskan ke dua lapisan *LSTM* untuk menangkap dependensi temporal jangka panjang, dilanjutkan dengan *Dropout* untuk regularisasi, hingga akhirnya lapisan *Dense* menghasilkan nilai prediksi akhir.

Tabel 5 menunjukkan arsitektur model *hybrid* CNN-LSTM yang digunakan pada penelitian ini secara lebih rinci.

Tabel 5. Arsitektur Model CNN-LSTM

Layer	Parameter
<i>Conv1D</i>	64 filter, kernel size = 3, stride = 1, padding = same, ReLU
<i>MaxPooling1D</i>	pool size = 2, stride = 2 (default)
<i>LSTM</i>	100 neuron, <i>return_sequences</i> = True, aktivasi tanh (default)
<i>LSTM</i>	100 neuron, <i>return_sequences</i> = False, aktivasi tanh (default)
<i>Dropout</i>	rate = 0.02
<i>Dense</i>	1 neuron, aktivasi linear

Pada arsitektur ini, lapisan CNN berfungsi mengekstraksi pola lokal dari data *time series*, kemudian hasil ekstraksi tersebut diteruskan ke lapisan LSTM untuk mempelajari dependensi temporal jangka panjang. Kombinasi tersebut memungkinkan model memanfaatkan keunggulan kedua pendekatan secara bersamaan sehingga meningkatkan kemampuan prediksi pada data saham yang bersifat acak dan tidak berpola tetap. Keunggulan model *hybrid* CNN-LSTM dibandingkan model tunggal ini didukung oleh berbagai temuan, di mana *hybrid* CNN-LSTM secara umum mengungguli model tunggal CNN maupun LSTM pada prediksi *time series* keuangan [21], dan terkonfirmasi pula pada studi pasar saham India menggunakan enam metrik performa sekaligus [22].

2.7 Pelatihan Model

Setiap model dilatih menggunakan data latih melalui proses pembaruan bobot secara iteratif dengan algoritma *backpropagation*. Proses kompilasi model menggunakan *optimizer* Adam dengan *loss function* *Mean Squared Error* (MSE). Eksperimen pelatihan mencakup seluruh kombinasi *timestep* 30, 60, dan 90 dan juga tiga skenario pembagian data, yaitu 70:30, 80:20, dan 90:10. Sebagian dari data latih pada setiap skenario dipisahkan sebagai data validasi menggunakan rasio 80:20 guna memantau perkembangan performa model selama pelatihan berlangsung.

Konfigurasi *hyperparameter* yang diterapkan meliputi *epoch* maksimum sebesar 50, *batch size* 32, dan *learning rate* 0.001. Untuk mencegah *overfitting*, mekanisme *early stopping* diterapkan dengan nilai *patience* sebesar 10, sehingga proses pelatihan akan dihentikan secara otomatis apabila tidak terdapat peningkatan performa pada data validasi. Konfigurasi ini sejalan dengan pendekatan yang diterapkan oleh Suprpto dan Andryana dalam membandingkan CNN, LSTM, dan CNN-LSTM untuk prediksi harga saham di pasar modal Indonesia, di mana penggunaan *optimizer* Adam, MSE sebagai *loss function*, dan *early stopping* terbukti efektif dalam menghasilkan model yang stabil dan tidak *overfitting* [23].

2.8 Evaluasi Model

Performa setiap model diukur menggunakan tiga metrik evaluasi, yaitu RMSE, MAPE, dan R^2 . RMSE mengukur besarnya rata-rata *error* prediksi. MAPE menghitung rata-rata persentase penyimpangan hasil prediksi terhadap nilai aktualnya. Sementara itu, R^2 digunakan untuk mengetahui seberapa mampu model menjelaskan variabilitas yang terdapat pada data aktual. Penggunaan RMSE, MAPE, dan R^2 sebagai metrik evaluasi prediksi harga saham merupakan pendekatan yang umum dan direkomendasikan dalam literatur, sebagaimana diulas secara komprehensif oleh Vuong et al. dalam tinjauan bibliometrik model prediksi harga saham dari pendekatan statistik hingga *deep learning* [24].

Persamaan RMSE ditunjukkan pada Persamaan (7).

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (7)$$

Persamaan MAPE ditunjukkan pada Persamaan (8).

$$MAPE = \frac{100\%}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right| \quad (8)$$

Persamaan R^2 ditunjukkan pada Persamaan (9).

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (9)$$

Semakin mendekati 0 nilai RMSE dan MAPE nya, maka semakin baik pula performa modelnya, sedangkan untuk nilai R^2 , semakin mendekati 1 nilainya maka semakin baik pula performanya.

2.9 Perbandingan Hasil

Hasil dari ketiga model, yaitu CNN, LSTM, dan *hybrid* CNN-LSTM, dibandingkan satu sama lain berdasarkan nilai RMSE, MAPE, dan R^2 . Tujuan perbandingan ini adalah untuk menentukan model mana yang paling baik dalam memprediksi harga saham BBKA di berbagai kombinasi *timestep* dan skema pembagian data. Model yang menghasilkan nilai RMSE dan MAPE paling kecil serta nilai R^2 paling tinggi dianggap sebagai model dengan performa terbaik.

3. HASIL DAN PEMBAHASAN



3.1 Preprocessing Data

Data historis saham BBCA dari Yahoo Finance periode 2015–2025 berhasil dimuat sebanyak ± 2.475 baris dengan variabel OHLCV. Hasil pengecekan *missing value* menunjukkan bahwa tidak ditemukan nilai kosong pada dataset sehingga data langsung dapat diproses tanpa tindakan tambahan. Statistik deskriptif data setelah normalisasi disajikan pada Tabel 6.

Tabel 6. Statistik deskriptif data setelah normalisasi

	Open	High	Low	Close	Volume
Count	2475.0000	2475.0000	2475.0000	2475.0000	2475.0000
Mean	0.4203	0.4188	0.4254	0.4188	0.0765
Std	0.2785	0.2814	0.2780	0.2764	0.0520
Min	0.0000	0.0000	0.0000	0.0000	0.0000
25%	0.1614	0.1564	0.1676	0.1614	0.0496
50%	0.4203	0.4198	0.4222	0.4177	0.0668
75%	0.6435	0.6453	0.6472	0.6447	0.0914
Max	1.0000	1.0000	1.0000	1.0000	1.0000

Tabel 6 menunjukkan karakteristik statistik data saham BBCA setelah proses normalisasi menggunakan MinMaxScaler. Seluruh variabel telah berada pada rentang 0 hingga 1 sehingga perbedaan skala antar fitur berhasil diminimalkan. Kondisi ini membantu proses pelatihan model *deep learning* menjadi lebih stabil dan memungkinkan setiap fitur memberikan kontribusi yang lebih seimbang dalam pembelajaran pola data.

Nilai rata-rata (*mean*) untuk variabel harga (*Open, High, Low, Close*) berada di kisaran 0.42, yang berarti secara rata-rata harga saham BBCA selama periode 2015–2025 berada di sekitar pertengahan rentang nilainya. Sementara itu, variabel *Volume* memiliki nilai rata-rata yang jauh lebih rendah, yaitu sekitar 0.077, yang mencerminkan bahwa sebagian besar aktivitas perdagangan harian berada di kisaran bawah dari *volume* tertinggi yang pernah terjadi selama periode tersebut. Perbedaan ini wajar karena lonjakan *volume* biasanya hanya terjadi pada momen-momen tertentu seperti saat ada pengumuman dividen atau peristiwa besar lainnya yang memengaruhi pasar. Nilai standar deviasi (*std*) untuk variabel harga berkisar 0.276–0.281, menunjukkan variasi data yang cukup merata dan tidak terlalu condong ke salah satu ujung rentang, yang mengindikasikan bahwa data saham BBCA relatif terdistribusi dengan baik setelah normalisasi.

3.2 Hasil Pengujian dan Evaluasi Model

Tahap pelatihan dan pengujian melibatkan tiga model *deep learning*, yakni CNN, LSTM, dan hybrid CNN-LSTM, yang masing-masing diuji pada seluruh kombinasi *timestep* 30, 60, 90 dengan skenario pembagian data 70:30, 80:20, dan 90:10. Evaluasi performa model dilakukan menggunakan RMSE, MAPE, dan R². Tabel 7 menunjukkan hasil evaluasi seluruh model pada berbagai skenario pengujian.

Tabel 7. Perbandingan Hasil Evaluasi Skenario Pengujian

<i>Timesteps</i>	<i>Split</i>	Model	RMSE	MAPE(%)	R ²
30	70:30	CNN	141,161	1,2422	0,9767
30	70:30	LSTM	280,5337	2,5809	0,9078
30	70:30	CNN-LSTM	424,9638	3,9183	0,7884
30	80:20	CNN	176,7557	1,5003	0,9246
30	80:20	LSTM	136,1428	1,0936	0,9553
30	80:20	CNN-LSTM	200,0129	1,6663	0,9034
30	90:10	CNN	157,2294	1,2509	0,8527
30	90:10	LSTM	135,6561	1,0769	0,8904
30	90:10	CNN-LSTM	188,0195	1,5087	0,7894
60	70:30	CNN	151,7307	1,3413	0,9723
60	70:30	LSTM	208,7119	1,8898	0,9477
60	70:30	CNN-LSTM	149,5391	1,2777	0,9731
60	80:20	CNN	169,58	1,438	0,93
60	80:20	LSTM	568,2475	5,4266	0,2142
60	80:20	CNN-LSTM	399,6718	3,6868	0,6113
60	90:10	CNN	169,9232	1,3309	0,8235
60	90:10	LSTM	169,0644	1,3649	0,8253
60	90:10	CNN-LSTM	194,3399	1,5521	0,7692
90	70:30	CNN	162,9377	1,4729	0,9674



<i>Timesteps</i>	<i>Split</i>	Model	RMSE	MAPE(%)	R ²
90	70:30	LSTM	233,4636	2,0604	0,933
90	70:30	CNN-LSTM	345,6279	3,1852	0,8531
90	80:20	CNN	137,726	1,1416	0,9534
90	80:20	LSTM	142,6659	1,1371	0,95
90	80:20	CNN-LSTM	130,5432	1,0669	0,9581
90	90:10	CNN	161,3452	1,2608	0,838
90	90:10	LSTM	142,0278	1,1435	0,8745
90	90:10	CNN-LSTM	164,9095	1,3032	0,8307

Tabel 7 menyajikan hasil evaluasi lengkap dari seluruh kombinasi skenario pengujian yang melibatkan tiga model, tiga variasi timestep, dan tiga skenario pembagian data. Secara umum, performa masing-masing model sangat dipengaruhi oleh kombinasi antara panjang timestep dan proporsi data latih yang digunakan. Analisis lebih rinci mengenai karakteristik performa masing-masing model diuraikan pada subbagian berikut.

3.2.1 Analisis Performa per Model

Hasil evaluasi menunjukkan bahwa Model CNN memiliki performa yang paling konsisten di antara ketiga model pada hampir seluruh skenario pengujian. Nilai RMSE model CNN di sebagian besar kombinasi berkisar antara 137 hingga 177, dengan pengecualian pada beberapa skenario split 90:10 yang sedikit lebih tinggi. Konsistensi ini mencerminkan karakteristik dasar CNN yang bekerja dengan cara mengekstraksi pola lokal dari data melalui proses konvolusi, sehingga tidak terlalu bergantung pada panjang konteks historis. Berbeda dengan LSTM yang menggunakan mekanisme memori untuk menyimpan informasi jangka panjang, CNN cukup melihat pola pada *sliding window* yang ada dan langsung menghasilkan prediksi. Keunggulan ini membuat CNN lebih stabil dan tidak mudah terganggu oleh perubahan konfigurasi, meskipun kelemahannya adalah model ini kurang responsif saat terjadi pergerakan harga yang tajam dan mendadak karena tidak memiliki kemampuan untuk mengingat konteks temporal yang lebih panjang.

Sedangkan Model LSTM memperlihatkan karakteristik yang berbeda. Pada beberapa konfigurasi, LSTM mampu bersaing bahkan melampaui CNN, seperti pada timestep 30 split 80:20 dengan RMSE 136.1428, di mana LSTM bahkan melampaui CNN yang memperoleh RMSE 176.7557. Namun, LSTM juga menunjukkan kerentanan yang cukup jelas pada konfigurasi tertentu, terutama pada *timestep* 60 split 80:20 di mana nilai RMSE melonjak drastis menjadi 568.2475, MAPE mencapai 5.43%, dan R² turun ke 0.2142. Penurunan yang sangat tajam ini menunjukkan bahwa model LSTM mengalami masalah konvergensi pada konfigurasi tersebut, kemungkinan akibat kombinasi antara panjang timestep dan jumlah data latih yang tidak sesuai dengan karakteristik pola data pada periode pengujian. Kondisi ini diperparah oleh mekanisme *early stopping* yang aktif ketika tidak ada peningkatan pada *validation loss* selama 10 *epoch* berturut-turut, sehingga proses pelatihan terhenti sebelum model sempat keluar dari kondisi konvergensi yang buruk tersebut dan menemukan bobot yang lebih optimal. Selain itu, *timestep* 60 menciptakan sekuens yang cukup panjang sehingga gradien yang dipropagasi balik harus melewati lebih banyak *gate* LSTM, yang meningkatkan risiko *vanishing gradient* dan membuat proses pelatihan menjadi lebih tidak stabil. Perlu dicatat bahwa penelitian ini tidak melakukan *hyperparameter tuning* ulang pada skenario anomali ini, karena tujuan eksperimen adalah membandingkan performa ketiga model pada konfigurasi yang seragam dan tetap (*fixed configuration*). Pendekatan ini konsisten dengan desain eksperimen yang adil (*fair comparison*), di mana setiap model diuji pada kondisi yang identik tanpa penyesuaian khusus per skenario, sehingga anomali yang muncul justru mengungkap kelemahan inheren arsitektur LSTM yang sensitif terhadap kombinasi panjang *timestep* dan proporsi data latih tertentu.

Di sisi lain, Model *hybrid* CNN-LSTM menunjukkan pola yang menarik, tidak selalu menjadi model terbaik, tetapi menghasilkan performa tertinggi secara keseluruhan ketika diberikan konfigurasi yang tepat. Pada *timestep* 30 split 70:30, CNN-LSTM justru menghasilkan performa terburuk di antara ketiga model dengan RMSE 424.9638 dan R² hanya 0.7884. Ini menunjukkan bahwa model *hybrid* memerlukan data historis yang cukup panjang agar kedua komponennya, yaitu CNN dan LSTM, dapat bekerja secara optimal. Ketika *timestep* diperpanjang menjadi 90 dan data latih ditingkatkan ke rasio 80:20, CNN-LSTM menghasilkan performa terbaik di antara semua model dengan RMSE 130.54, MAPE 1.0669%, dan R² 0.9581. Hal ini menunjukkan bahwa keunggulan model *hybrid* baru dapat muncul ketika jumlah data historis cukup panjang untuk CNN mengekstraksi pola lokal yang bermakna, dan LSTM mendapatkan konteks yang cukup untuk membangun hubungan temporal yang kuat.

Untuk memberikan interpretasi yang lebih praktis, nilai RMSE sebesar 130,54 menunjukkan bahwa rata-rata selisih prediksi terhadap harga aktual berada pada kisaran Rp130 per lembar saham. Jika dibandingkan dengan rentang harga saham BBCA selama periode pengujian yang berada pada kisaran Rp8.000–Rp11.000 per lembar saham, maka tingkat kesalahan tersebut relatif kecil, yaitu sekitar 1–1,5% dari harga aktual. Hal ini mengindikasikan bahwa model CNN-LSTM mampu menghasilkan prediksi yang cukup akurat untuk merepresentasikan pergerakan harga saham BBCA.

Nilai R^2 yang tinggi pada beberapa skenario perlu dipahami lebih hati-hati, karena dalam prediksi data *time series* finansial, nilai R^2 yang tinggi tidak selalu berarti model benar-benar belajar pola yang bermakna. Kekhawatiran yang umum muncul adalah kemungkinan model hanya meniru harga hari sebelumnya tanpa menangkap pola yang sesungguhnya, yang dikenal sebagai fenomena *random walk*. Namun hasil yang diperoleh menunjukkan beberapa indikasi bahwa model tidak sekadar meniru nilai sebelumnya, di mana nilai MAPE pada skenario terbaik berkisar 1.07%–1.14% dan variasi R^2 antar skenario yang cukup besar (0.21 hingga 0.97) menunjukkan bahwa performa model sangat dipengaruhi oleh konfigurasi eksperimen.

Untuk mempermudah perbandingan, Tabel 8 merangkum konfigurasi terbaik dari masing-masing model berdasarkan nilai RMSE terendah yang diperoleh di seluruh skenario pengujian.

Tabel 8. Konfigurasi Terbaik Masing-Masing Model

Model	Timestep	Split	RMSE	MAPE (%)	R^2
CNN	90	80:20	137,726	1,1416	0,9534
LSTM	30	80:20	136,1428	1,0936	0,9553
CNN-LSTM	90	80:20	130,5432	1,0669	0,9581

Berdasarkan Tabel 8, model *hybrid* CNN-LSTM menghasilkan performa terbaik secara keseluruhan dengan RMSE sebesar 130.54, MAPE sebesar 1.0669%, dan R^2 sebesar 0.9581 pada konfigurasi *timestep* 90 dengan pembagian data 80:20. Model LSTM mencapai performa terbaiknya pada konfigurasi *timestep* 30 dengan split 80:20, menghasilkan RMSE 136.1428 dan R^2 0.9553. Sementara itu, model CNN menghasilkan performa terbaiknya pada konfigurasi *timestep* 90 dengan split 80:20, dengan nilai RMSE 137.726 dan R^2 0.9534. Menariknya, ketiga model sama-sama mencapai performa optimalnya pada skenario pembagian data 80:20, yang mengindikasikan bahwa rasio ini memberikan keseimbangan terbaik antara jumlah data pelatihan yang memadai dan kemampuan generalisasi model terhadap data uji. Untuk memahami mengapa CNN-LSTM mampu menghasilkan performa terbaik, Tabel 9 menyajikan perbandingan kemampuan masing-masing model secara lebih rinci.

Tabel 9. Perbandingan Kemampuan Antar Model

Kemampuan	CNN	LSTM	CNN-LSTM
Ekstraksi pola lokal	Mampu mengekstraksi pola lokal jangka pendek melalui proses konvolusi	Tidak dirancang untuk ekstraksi pola lokal	Mewarisi kemampuan CNN dalam mengekstraksi pola lokal
Penangkapan dependensi temporal	Tidak memiliki mekanisme memori untuk menangkap hubungan antar waktu	Mampu menangkap dependensi temporal jangka panjang melalui mekanisme <i>memory cell</i>	Mewarisi kemampuan LSTM dalam menangkap dependensi temporal jangka panjang
Hasil prediksi	Stabil namun kurang responsif terhadap perubahan harga yang tajam	Mampu mengikuti pola historis namun cenderung <i>lagging</i>	Paling responsif dan akurat terhadap fluktuasi harga saham

Tabel 9 menunjukkan bahwa model CNN-LSTM berhasil menggabungkan keunggulan CNN dalam mengekstraksi pola lokal dan keunggulan LSTM dalam menangkap dependensi temporal, sehingga menghasilkan prediksi yang lebih akurat dan responsif dibandingkan model tunggal. Perbedaan nilai RMSE antara model terbaik yaitu CNN-LSTM dengan RMSE sebesar 130.5432 dan model dengan performa terendah di antara ketiganya yaitu CNN dengan RMSE 137.726 tidak terlalu besar, namun perbedaan tersebut menjadi lebih signifikan pada pergerakan harga yang tajam.

3.2.2 Analisis Pengaruh Timestep

Variasi *timestep* memberikan pengaruh terhadap performa ketiga model. Pada *timestep* 30, CNN menjadi model yang paling unggul, khususnya pada split 70:30 dengan RMSE 141.161. Dengan *sliding window* yang pendek, CNN cukup efektif karena hanya perlu mengenali pola pergerakan harga dalam 30 hari terakhir. Sementara LSTM dan CNN-LSTM justru kurang optimal pada *timestep* ini, karena window yang terlalu pendek membuat mekanisme memori LSTM belum mendapatkan cukup informasi untuk membangun representasi temporal yang baik.

Pada *timestep* 60, hasil yang diperoleh cukup bervariasi dan tidak stabil, terutama pada model LSTM dan CNN-LSTM. Penurunan yang tidak biasa dengan nilai terbesar terjadi pada LSTM dengan split 80:20, di mana model mengalami kegagalan yang sangat signifikan. Satu-satunya model yang tetap stabil pada *timestep* 60 adalah CNN, yang kembali membuktikan ketangguhannya. Sementara CNN-LSTM pada split 70:30 justru menunjukkan hasil yang cukup baik dengan RMSE 149.5391 dan R^2 0.9731, namun tidak konsisten pada split lainnya.

Pada *timestep* 90, ketiga model secara umum menunjukkan performa yang lebih baik dibandingkan *timestep* sebelumnya, terutama pada split 80:20. Penggunaan *sliding window* yang lebih panjang memungkinkan model mempelajari



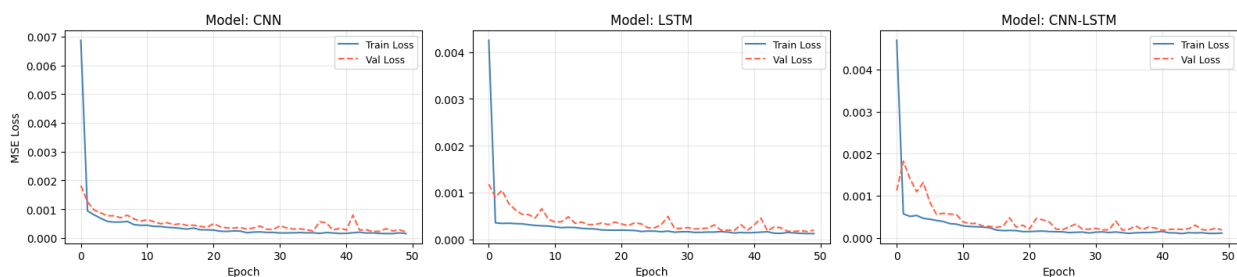
lebih banyak pola historis sebelum membuat prediksi, sehingga representasi data yang dipelajari menjadi lebih kaya. Hal ini memberikan penjelasan mengapa *timestep* 90 secara umum menghasilkan performa yang lebih tinggi, terutama untuk model yang bergantung pada konteks temporal panjang seperti LSTM dan CNN-LSTM.

3.2.3 Analisis Pengaruh Pembagian Data

Skenario pembagian data juga memberikan pengaruh yang terlihat pada hasil evaluasi. Secara umum, split 80:20 menghasilkan keseimbangan terbaik antara jumlah data latih dan kemampuan generalisasi model. Split 70:30 memberikan lebih banyak data uji sehingga evaluasi lebih representatif, namun data latih yang lebih sedikit membuat beberapa model kesulitan mempelajari pola data secara optimal. Sebaliknya, split 90:10 memberikan data latih paling banyak, tetapi data ujinya yang sangat sedikit membuat nilai R^2 pada skenario ini cenderung lebih rendah, mengindikasikan bahwa model kurang mampu menggeneralisasi data yang belum pernah dilihat sebelumnya meskipun performa pada data latih baik.

3.3 Grafik Training Loss dan Validation Loss

Gambar 2 menunjukkan grafik *training loss* dan *validation loss* dari ketiga model, yaitu CNN, LSTM, dan CNN-LSTM, pada konfigurasi terbaik masing-masing.



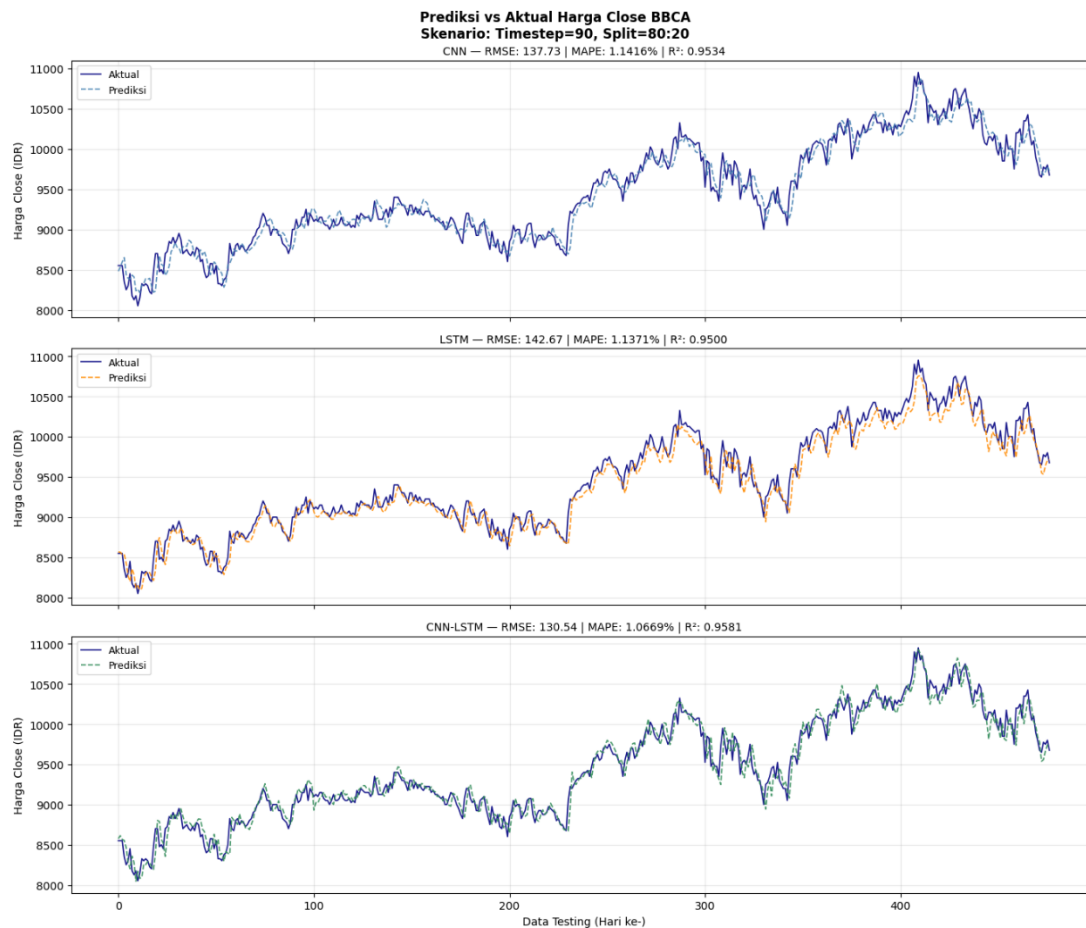
Gambar 5. Grafik Training Loss dan Validation Loss

Grafik pada Gambar 2 menampilkan perkembangan *training loss* dan *validation loss* dari ketiga model CNN, LSTM, dan CNN-LSTM selama proses pelatihan berlangsung hingga 50 *epoch*. Secara keseluruhan, ketiga model memperlihatkan pola yang serupa, penurunan *loss* yang sangat tajam terjadi pada *epoch-epoch* awal, kemudian melambat dan mulai stabil seiring bertambahnya *epoch*. Ini merupakan tanda yang baik bahwa model berhasil menangkap pola utama dari data sejak awal pelatihan.

Secara umum, jarak antara *training loss* dan *validation loss* pada ketiga model tidak terlalu jauh, sehingga tidak ada indikasi *overfitting* yang serius. Model masih mampu menggeneralisasi pola yang dipelajari dari data latih ke data validasi dengan cukup baik. Nilai *learning rate* sebesar 0.001 yang digunakan juga tampak sesuai, karena tidak ada lonjakan *loss* yang ekstrem yang biasanya muncul akibat *learning rate* yang terlalu besar. Meski demikian, *validation loss* pada model CNN dan CNN-LSTM terlihat mengalami fluktuasi di pertengahan pelatihan, yang menunjukkan bahwa kemampuan generalisasi kedua model tersebut belum sepenuhnya stabil pada setiap *epoch*.

3.4 Visualisasi Prediksi Saham

Visualisasi hasil prediksi dilakukan sebagai pembandingan pergerakan harga saham aktual dengan hasil prediksi dari model CNN, LSTM, dan CNN-LSTM pada konfigurasi terbaik, yaitu *timestep* 90 dengan pembagian data 80:20. Gambar 3 menunjukkan perbandingan antara data aktual dan hasil prediksi ketiga model.



Gambar 6. Perbandingan Data Aktual dan Hasil Prediksi Ketiga Model

Berdasarkan Gambar 3, ketiga model berhasil mengikuti tren pergerakan harga saham BBCA dengan tingkat kesesuaian yang cukup baik, meskipun tingkat kedekatan prediksi terhadap data aktual berbeda pada masing-masing model. CNN menghasilkan prediksi yang stabil namun masih menunjukkan selisih pada beberapa perubahan harga yang tajam. LSTM mampu mengikuti pola historis dengan baik, tetapi pada beberapa bagian terlihat kecenderungan *lagging* saat terjadi perubahan tren secara cepat. Hal ini terlihat jelas pada sekitar hari ke-200 hingga ke-350, di mana LSTM menunjukkan keterlambatan respons saat terjadi kenaikan harga yang tajam.

Sementara itu, *hybrid* CNN-LSTM menghasilkan kurva prediksi yang paling mendekati data aktual dan lebih responsif terhadap fluktuasi harga saham. Hal ini sejalan dengan evaluasi sebelumnya, di mana *hybrid* CNN-LSTM memperoleh nilai RMSE dan MAPE terendah serta nilai R^2 tertinggi pada konfigurasi *timestep* 90 dan *split* data 80:20. Pada rentang harga yang bergerak relatif stabil, ketiga model menunjukkan kinerja yang hampir serupa. Perbedaan yang paling mencolok muncul pada periode di mana harga bergerak secara tajam dalam waktu singkat, di mana CNN-LSTM secara konsisten lebih cepat menyesuaikan prediksinya dengan perubahan tersebut dibandingkan CNN maupun LSTM secara mandiri. Hal ini menunjukkan bahwa kombinasi kemampuan CNN dalam mengenali pola lokal dan kemampuan LSTM dalam mempertahankan memori jangka panjang memberikan keunggulan nyata dalam menghadapi kondisi pasar yang berubah-ubah. Secara keseluruhan, kombinasi CNN dan LSTM mampu meningkatkan akurasi prediksi dibandingkan model tunggal.

3.5 Pembahasan

Dibandingkan dengan penelitian Kurniansyah dkk. yang menggunakan LSTM tunggal dengan variabel *close* saja pada saham BBCA dan memperoleh RMSE 114.17 pada *timestep* 60 hari [6], hasil LSTM pada penelitian ini sedikit lebih tinggi yaitu RMSE 136.14 pada *timestep* 30. Selisih ini wajar terjadi karena variabel input yang dipakai berbeda. Penelitian ini menggunakan variabel OHLCV yang lebih lengkap, sementara penelitian Kurniansyah dkk. hanya mengandalkan harga penutupan. Menariknya, kecenderungan *lagging* pada LSTM yang ditemukan di penelitian ini juga konsisten dengan temuan Kurniansyah dkk. [6] dan Zhang [9], yang sama-sama melaporkan bahwa LSTM cenderung lambat merespons perubahan harga yang tajam dan mendadak. Konsistensi pola ini pada dataset dan periode yang berbeda mengindikasikan bahwa

karakteristik tersebut bukan hanya dampak dari pengaturan *hyperparameter* tertentu, melainkan lebih merupakan kelemahan yang melekat pada mekanisme kerja *memory cell* LSTM itu sendiri.

Sementara itu, perbandingan dengan model *hybrid* justru menunjukkan hasil yang berlawanan. Wibowo dkk. menemukan bahwa CNN-LSTM pada data Bitcoin punya RMSE dan MAE yang lebih tinggi dibanding CNN maupun LSTM tunggal, meskipun MAPE tetap yang paling rendah [14]. Hal ini bertolak belakang dengan penelitian ini, di mana CNN-LSTM unggul di ketiga metrik sekaligus. Begitu juga dengan penelitian Joddy pada lima saham IDX30, yang justru mendapati LSTM tunggal sebagai model terbaik, bukan CNN-LSTM [17]. Hal yang sebanding juga dilaporkan oleh Suprpto dan Andryana pada saham PT Astra International Tbk (ASII) dengan periode data yang serupa, yaitu 2015–2025, di mana LSTM tunggal kembali tampil sebagai model dengan RMSE terkecil, hanya sebesar 77.86 saja, sementara itu CNN-LSTM hanya unggul dalam hal kestabilan performa antar skema pembagian data, bukan dari sisi akurasi prediksi [24]. Perbedaan-perbedaan ini menunjukkan bahwa keunggulan model *hybrid* tidak bersifat mutlak, sangat tergantung pada karakteristik volatilitas aset yang diprediksi mengingat Bitcoin jauh lebih fluktuatif dibanding BBCA, dan juga pada ada ataupun tidaknya eksplorasi konfigurasi seperti *timestep* dan rasio pembagian data secara sistematis.

Dari perbandingan tersebut, dapat dipahami bahwa tidak terdapat satu model yang secara konsisten mendominasi pada seluruh studi sejenis. Performa CNN, LSTM, maupun CNN-LSTM tampak sangat dipengaruhi oleh jenis aset yang diteliti, variabel input yang digunakan, serta kombinasi *timestep* dan skema pembagian data yang diterapkan. Penelitian ini memberikan kontribusi dengan menunjukkan bahwa keunggulan CNN-LSTM yang tampak secara optimal ketika model diberikan konteks historis yang cukup panjang, yaitu pada *timestep* 90, serta proporsi data latih yang seimbang melalui pembagian data 80:20, suatu aspek yang belum banyak dieksplorasi secara sistematis pada penelitian-penelitian sebelumnya.

4. KESIMPULAN

Penelitian ini telah berhasil menerapkan dan menguji performa model CNN, LSTM, dan *hybrid* CNN-LSTM dalam memprediksi harga saham BBCA menggunakan data *time series* berbasis OHLCV periode 2015–2025. Pengujian dilakukan menggunakan variasi *timestep* 30, 60, dan 90 serta pembagian data 70:30, 80:20, dan 90:10 dengan evaluasi menggunakan RMSE, MAPE, dan R^2 . Hasil penelitian menunjukkan bahwa model CNN menghasilkan performa yang paling konsisten pada berbagai skenario pengujian, sedangkan model LSTM mampu memahami hubungan temporal data historis dengan cukup baik meskipun pada beberapa kondisi masih menunjukkan kecenderungan *lagging*. Sementara itu, model *hybrid* CNN-LSTM berhasil menghasilkan performa terbaik pada konfigurasi *timestep* 90 dan *split* data 80:20 dengan nilai RMSE sebesar 130.54 (IDR), MAPE sebesar 1.0669%, dan R^2 sebesar 0.9581. Keunggulan CNN-LSTM secara fundamental bersumber pada kemampuannya menangani dua karakteristik utama data saham Indonesia yang bersifat *volatile*, yaitu pola pergerakan harga jangka pendek yang ditangkap CNN dan dependensi temporal jangka panjang yang dikelola LSTM, sehingga model lebih adaptif terhadap fluktuasi pasar modal Indonesia. Untuk penelitian selanjutnya, disarankan untuk menerapkan teknik *walk-forward validation* sebagai pengganti *fixed split* guna menghasilkan evaluasi yang lebih *robust*, serta mengeksplorasi mekanisme *attention* pada arsitektur CNN-LSTM untuk memberikan bobot yang berbeda pada setiap titik waktu dalam data historis, sehingga model dapat lebih selektif dalam memanfaatkan informasi yang paling relevan untuk prediksi.

REFERENCES

- [1] F. P. Sholikah, W. Putri, and R. M. Djangi, "Peranan Pasar Modal Dalam Perekonomian Negara Indonesia," *Arbitr. J. Econ. Account.*, vol. 3, no. 2, pp. 341–345, 2022, doi: 10.47065/arbitrase.v3i2.496.
- [2] D. E. S. Ririn Nur Fadila, "Pengaruh Inflasi, Suku Bunga, dan Nilai Tukar terhadap Harga Saham Perbankan di BEI Periode 2020–2024," *Ekon. Keuangan, Investasi dan Syariah*, vol. 7, no. 3, pp. 753–761, 2026, doi: 10.47065/ekuitas.v7i3.9261.
- [3] M. Lathif Arafat, Williya Meta, "Performance Comparison of Indonesia's Big Banks Using The Competitive Profile Matrix Approach," *InFestasi*, vol. 21, no. 1, pp. 57–70, 2025, doi: 10.21107/infestasi.v21i1.29836.
- [4] B. Zhang, "The Stock Price Forecasting Based on Time Series Model and Neural Network," *BCP Bus. Manag.*, vol. 38, pp. 3423–3428, 2023, doi: 10.54691/bcpbm.v38i.4319.
- [5] H. Malaikah and J. F. Alabdali, "Analysis of Noise on Ordinary and Fractional-Order Financial Systems," *Fractal Fract.*, vol. 9, no. 2, pp. 1–9, 2025, doi: 10.3390/fractalfract9050316.
- [6] S. K. G. Juliandi Kurniansyah and M. A., Febi Yanto, "Implementasi Model Long Short Term Memory pada Prediksi Harga Saham Bank BCA," *Bull. Inf. Technol.*, vol. 6, no. 2, pp. 79–86, 2025, doi: 10.47065/bit.v5i2.1783.
- [7] M. Barua, T. Kumar, and K. Raj, "Comparative Analysis of Deep Learning Models for Stock Price Prediction in the Indian Market," *FinTech (MDPI)*, vol. 3, no. 4, pp. 551–568, 2024, doi: 10.3390/fintech3040029.
- [8] M. Mukhlis, A. Kustiyo, and A. Suharso, "Peramalan Produksi Pertanian Menggunakan Model Long Short-Term Memory," *BINA Insa. ICT J.*, vol. 8, no. 1, pp. 22–32, 2021, doi: 10.51211/biict.v8i1.1492.
- [9] Y. Zhang, "Stock Price Prediction Using ARIMA and LSTM Models: An Application to CSI 300 Closing Prices," *Highlights Sci. Eng. Technol.*, vol. 88, no. 3, pp. 286–293, 2024, doi: 10.54097/7d9x1c14.
- [10] M. D. B. H. K. Mohan, "A Convolutional Neural Network Based Approach to Financial Time Series Prediction," *Neural Comput. Appl.*, vol. 34, no. 16, pp. 13319–13337, 2022, doi: 10.1007/s00521-022-07143-2.



- [11] S. Kiranyaz, O. Avci, O. Abdeljaber, T. Ince, M. Gabbouj, and D. J. Inman, “1D Convolutional Neural Networks and Applications : A Survey,” *Mech. Syst. Signal Process.*, vol. 151, pp. 1–21, 2021, doi: 10.1016/j.ymsp.2020.107398.
- [12] P. Lara-ben and M. Carranza-garc, “An Experimental Review on Deep Learning Architectures for Time Series Forecasting,” *Int. J. Neural Syst.*, vol. 31, no. 3, pp. 1–28, 2021, doi: 10.1142/S0129065721300011.
- [13] M. Abumohsen, A. Yousef, M. Owda, and A. Abumihsan, “Hybrid Machine Learning Model Combining of CNN-LSTM-RF for Time Series Forecasting of Solar Power Generation,” *e-Prime - Adv. Electr. Eng. Electron. Energy*, vol. 9, pp. 1–21, 2024, doi: 10.1016/j.prime.2024.100636.
- [14] C. Wibowo, R. Purba, and M. F. Pasha, “Implementation of the CNN-LSTM Hybrid Model in Predicting Bitcoin Price Fluctuations,” *J. Teknol. dan Manaj. Inform.*, vol. 11, no. 2, pp. 201–211, 2025, doi: 10.26905/jtmi.v11i2.16239.
- [15] D. T. Handayani, Susi, Taslim, “Convolutional Neural Network – Long Short Term Memory Untuk Prediksi Harga Emas Indonesia,” *Indones. J. Comput. Sci.*, vol. 11, no. 3, pp. 901–911, 2022, doi: 10.33022/ijcs.v11i3.3074.
- [16] O. B. Sezer, M. U. Gudelek, and A. M. Ozbayoglu, “Financial Time Series Forecasting With Deep Learning : A Systematic Literature Review : 2005 – 2019,” *Appl. Soft Comput. J.*, vol. 90, pp. 1–27, 2020, doi: 10.1016/j.asoc.2020.106181.
- [17] S. Joddy, “Comparative Analysis of CNN , LSTM , and CNN – LSTM for Indonesian Stock Prediction,” *EMACS (Engineering, Math. Comput. Sci.)*, vol. 7, no. 3, pp. 283–289, 2025, doi: 10.21512/emacsjournal.v7i3.14326.
- [18] Z. Zhan and S. Kyoo, “Versatile Time - Window Sliding Machine Learning Techniques for Stock Market Forecasting,” *Artif. Intell. Rev.*, vol. 57, no. 209, pp. 1–20, 2024, doi: 10.1007/s10462-024-10851-x.
- [19] L. Liu and Y. Whar, “1D Convolutional Neural Networks for Chart Pattern Classification in Financial Time Series,” *J. Supercomput.*, vol. 78, no. 12, pp. 14191–14214, 2022, doi: 10.1007/s11227-022-04431-5.
- [20] A. Ritonga, A. Ma, and I. Suwarno, “Stock Price Forecasting with Multivariate Time Series Long Short-Term Memory : A Deep Learning Approach,” *J. Robot. Control*, vol. 5, no. 5, pp. 1322–1335, 2024, doi: 10.18196/jrc.v5i5.22460.
- [21] W. Chen, W. Hussain, F. Cauteruccio, and X. Zhang, “Deep Learning for Financial Time Series Prediction : A State-of-the-Art Review of Standalone and Hybrid Models Auto encoder,” *C. - Comput. Model. Eng. Sci.*, vol. 139, no. 1, pp. 187–224, 2023, doi: 10.32604/cmescs.2023.031388.
- [22] M. A. El-meligy, “Harnessing a Hybrid CNN-LSTM Model for Portfolio Performance : A Case Study on Stock Selection and Optimization,” *IEEE Access*, vol. 11, pp. 104000–104015, 2023, doi: 10.1109/ACCESS.2023.3317953.
- [23] M. F. Suprpto and S. Andryana, “Optimizing CNN-LSTM Models for Stock Price Prediction in a Multi-Sector Holding Company,” *Int. J. Comput. Sci. Humanit. AI*, vol. 3, no. 1, pp. 1–8, 2026, doi: 10.21512/ijcsjournal.v3i1.14936.
- [24] P. H. Vuong and L. H. Phu, “A Bibliometric Literature Review of Stock Price Forecasting : From Statistical Model to Deep Learning Approach,” *Sci. Prog.*, vol. 107, no. 1, pp. 1–31, 2024, doi: 10.1177/00368504241236557.

