

BAB 2

LANDASAN TEORI

2.1 Machine Learning

Machine Learning merupakan salah satu cabang kecerdasan buatan yang memungkinkan sistem mempelajari pola dari data tanpa memerlukan aturan yang dituliskan secara khusus untuk setiap kondisi. Dalam masalah prediksi, pola yang diperoleh dari data historis digunakan untuk memperkirakan nilai pada periode berikutnya.

Secara umum, proses pembelajaran dilakukan dengan mencari fungsi pemetaan f yang menghubungkan variabel input X dengan variabel target y . Hubungan tersebut dapat dituliskan sebagai berikut:

$$\hat{y} = f(X) \quad (2.1)$$

Nilai $\hat{y}$ merupakan hasil prediksi model, sedangkan y merupakan nilai aktual. Selama proses pelatihan, parameter model diperbarui untuk meminimalkan fungsi *loss* yang mengukur selisih antara nilai prediksi dan nilai aktual [4].

Pada prediksi *time series*, metode *Machine Learning* dapat digunakan untuk mempelajari hubungan non-linear antarvariabel. Beberapa algoritma yang umum digunakan untuk masalah regresi antara lain *Random Forest*, *Gradient Boosting*, dan *Support Vector Regression* (SVR). Setiap algoritma memiliki cara yang berbeda dalam mempelajari pola dan menghasilkan prediksi [4].

2.2 Deep Learning

Deep Learning merupakan bagian dari *Machine Learning* yang menggunakan jaringan saraf tiruan dengan beberapa lapisan untuk mempelajari pola pada data. Lapisan-lapisan tersebut mengolah informasi secara bertahap sehingga model dapat membentuk representasi data yang lebih kompleks. Berbeda dengan metode *Machine Learning* konvensional yang sering membutuhkan pemilihan fitur secara manual, *Deep Learning* dapat mempelajari representasi fitur selama proses pelatihan [2].

Model *Deep Learning* banyak digunakan pada data yang memiliki urutan, termasuk data *time series*. Salah satu arsitektur yang dikembangkan untuk data

berurutan adalah *Recurrent Neural Network* (RNN). RNN memproses data pada setiap langkah waktu dan membawa informasi dari langkah sebelumnya melalui *hidden state*.

RNN konvensional dapat mengalami masalah *vanishing gradient* ketika digunakan untuk mempelajari urutan yang panjang. Pada kondisi ini, nilai gradien menjadi semakin kecil selama proses propagasi balik sehingga informasi dari langkah waktu yang jauh lebih sulit dipertahankan. *Long Short-Term Memory* (LSTM) dikembangkan sebagai salah satu varian RNN untuk mengurangi masalah tersebut melalui penggunaan *cell state* dan mekanisme gerbang [2].

2.3 Time Series

Data *time series* merupakan sekumpulan pengamatan yang disusun berdasarkan urutan waktu. Data ini dapat memiliki beberapa karakteristik, seperti tren, pola musiman, dan perubahan acak. Secara matematis, *time series* dapat dinyatakan sebagai barisan $\{y_1, y_2, \dots, y_t\}$, dengan t menunjukkan indeks waktu. Nilai pada suatu waktu juga dapat memiliki hubungan dengan nilai pada waktu sebelumnya, yang dikenal sebagai autokorelasi [9].

Dalam pasar saham, data harga pembukaan, harga tertinggi, harga terendah, harga penutupan, dan volume perdagangan termasuk data *time series*. Urutan waktu perlu dipertahankan selama proses pengolahan data. Oleh karena itu, pembagian data *training* dan *testing* dilakukan secara berurutan tanpa pengacakan. Cara tersebut digunakan agar data dari periode pengujian tidak masuk ke dalam proses pelatihan dan menyebabkan *data leakage* [9].

Data *time series* dapat diubah menjadi bentuk *supervised learning* menggunakan *look back window*. Sejumlah pengamatan sebelumnya digunakan sebagai input untuk memprediksi nilai pada periode berikutnya. Sebagai contoh, *look back* 60 menggunakan data selama 60 hari sebelumnya untuk memprediksi nilai pada hari ke-61. Pembentukan input dilakukan secara bergeser sampai seluruh data pada periode yang digunakan selesai diproses.

2.4 Indeks Harga Saham Gabungan (IHSG)

Indeks Harga Saham Gabungan (IHSG) merupakan indeks yang menggambarkan pergerakan harga saham yang tercatat di Bursa Efek Indonesia (BEI). IHSG digunakan sebagai salah satu indikator untuk melihat perubahan

kondisi pasar saham Indonesia secara umum. Nilai indeks yang meningkat menunjukkan bahwa secara agregat nilai pasar saham mengalami kenaikan, sedangkan nilai indeks yang menurun menunjukkan kondisi sebaliknya.

Secara umum, perhitungan IHSG membandingkan nilai pasar pada periode berjalan dengan nilai dasar indeks. Nilai pasar diperoleh dari hasil perkalian harga saham dengan jumlah saham yang digunakan dalam perhitungan indeks. Rumus perhitungannya dapat dituliskan sebagai berikut:

$$IHSG_t = \frac{\sum_{i=1}^n (P_{i,t} \times S_{i,t})}{\text{Nilai Dasar}_t} \times 100 \quad (2.2)$$

Keterangan:

- $IHSG_t$ adalah nilai IHSG pada waktu t ;
- $P_{i,t}$ adalah harga saham ke- i pada waktu t ;
- $S_{i,t}$ adalah jumlah saham ke- i yang digunakan dalam perhitungan indeks pada waktu t ;
- n adalah jumlah saham yang menjadi konstituen indeks; dan
- Nilai Dasar _{t} adalah nilai dasar yang digunakan sebagai pembanding dan dapat disesuaikan apabila terjadi perubahan tertentu pada konstituen atau jumlah saham.

IHSG juga digunakan sebagai acuan dalam menilai kinerja pasar dan membandingkan hasil suatu portofolio dengan kondisi pasar secara umum. Pergerakannya dapat dipengaruhi oleh berbagai faktor, seperti kondisi ekonomi, tingkat suku bunga, nilai tukar, kebijakan pemerintah, kondisi politik, dan sentimen investor. Penelitian ini hanya menggunakan data historis harga dan indikator teknikal. Oleh karena itu, pengaruh faktor eksternal tersebut tidak dimasukkan secara langsung ke dalam model.

2.5 Support Vector Regression (SVR)

Support Vector Regression (SVR) merupakan pengembangan dari *Support Vector Machine* (SVM) untuk menyelesaikan masalah regresi. SVR mencari fungsi yang dapat mendekati nilai aktual dengan batas toleransi kesalahan sebesar ϵ .

Model juga berusaha mempertahankan fungsi yang relatif sederhana agar tidak terlalu menyesuaikan diri dengan data pelatihan [10].

Untuk sekumpulan data pelatihan $\{(x_i, y_i)\}_{i=1}^n$, fungsi regresi SVR dapat dituliskan sebagai berikut:

$$(2.3) \quad f(x) = w^T \phi(x) + b$$

Pada persamaan tersebut, w merupakan vektor bobot, b merupakan *bias*, dan $\phi(x)$ merupakan fungsi yang memetakan data input ke ruang fitur. Kerataan fungsi dikendalikan melalui nilai $\|w\|^2$. Permasalahan optimisasi pada SVR dirumuskan sebagai berikut:

$$\min_{w, b, \xi_i, \xi_i^*} \frac{1}{2} \|w\|^2 + C \sum_{i=1}^n (\xi_i + \xi_i^*) \quad (2.4)$$

dengan batasan:

$$y_i - w^T \phi(x_i) - b \leq \varepsilon + \xi_i, w^T \phi(x_i) + b - y_i \leq \varepsilon + \xi_i^*, \xi_i, \xi_i^* \geq 0 \quad (2.5)$$

Parameter C mengatur keseimbangan antara kerataan fungsi dan penalti terhadap kesalahan. Parameter ε menentukan lebar daerah toleransi di sekitar fungsi regresi. Kesalahan yang masih berada di dalam daerah tersebut tidak dikenai penalti. Sementara itu, ξ_i dan ξ_i^* merupakan variabel *slack* yang menunjukkan besarnya kesalahan di luar batas toleransi.

Fungsi kerugian *epsilon-insensitive* dapat dituliskan sebagai berikut:

$$L_\varepsilon(y_i, f(x_i)) = 0, \text{ jika } |y_i - f(x_i)| \leq \varepsilon, |y_i - f(x_i)| - \varepsilon, \text{ jika } |y_i - f(x_i)| > \varepsilon \quad (2.6)$$

Penelitian ini menggunakan kernel *Radial Basis Function* (RBF) untuk mempelajari hubungan non-linear pada data. Kernel RBF dirumuskan sebagai

berikut:

$$K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2) \quad (2.7)$$

Parameter γ mengatur luas pengaruh setiap data pelatihan. Nilai γ yang besar menghasilkan pengaruh yang lebih sempit sehingga model lebih peka terhadap perubahan lokal. Sebaliknya, nilai γ yang kecil memberikan pengaruh yang lebih luas dan menghasilkan fungsi yang lebih halus.

Dengan menggunakan kernel, fungsi keputusan SVR dapat dituliskan dalam bentuk berikut:

$$f(x) = \sum_{i=1}^n (\alpha_i - \alpha_i^*) K(x_i, x) + b \quad (2.8)$$

Nilai α_i dan α_i^* merupakan pengali Lagrange yang diperoleh melalui proses optimisasi. Data dengan nilai $(\alpha_i - \alpha_i^*)$ yang tidak sama dengan nol menjadi *support vector* dan berpengaruh terhadap hasil prediksi [10]. SVR dapat memberikan hasil yang baik pada data berukuran terbatas, tetapi penambahan data tidak selalu meningkatkan hasil secara langsung. Jumlah data yang semakin besar juga menambah beban komputasi karena perhitungan kernel dilakukan terhadap lebih banyak data pelatihan [1].

2.6 Long Short-Term Memory (LSTM)

Long Short-Term Memory (LSTM) merupakan salah satu pengembangan dari RNN yang dirancang untuk mempelajari hubungan pada data berurutan. LSTM menggunakan *cell state* sebagai jalur penyimpanan informasi dan tiga gerbang untuk mengatur informasi yang dipertahankan, ditambahkan, dan dikeluarkan [2].

Pada langkah waktu t , LSTM menerima input x_t , *hidden state* sebelumnya h_{t-1} , dan *cell state* sebelumnya c_{t-1} . Perhitungan dimulai dari *forget gate*, yaitu gerbang yang menentukan bagian informasi lama yang masih dipertahankan:

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \quad (2.9)$$

Selanjutnya, *input gate* menentukan bagian informasi baru yang akan disimpan. Kandidat informasi baru dibentuk menggunakan fungsi aktivasi tanh:

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i), \tilde{c}_t = \tanh(W_c[h_{t-1}, x_t] + b_c) \quad (2.10)$$

Nilai *cell state* diperbarui dengan menggabungkan informasi lama yang dipertahankan dan informasi baru yang ditambahkan:

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \quad (2.11)$$

Setelah itu, *output gate* menentukan informasi yang digunakan untuk menghasilkan *hidden state* pada langkah waktu t :

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o), h_t = o_t \odot \tanh(c_t) \quad (2.12)$$

Pada rangkaian persamaan tersebut, W_f , W_i , W_c , dan W_o merupakan matriks bobot, sedangkan b_f , b_i , b_c , dan b_o merupakan *bias*. Notasi $[h_{t-1}, x_t]$ menunjukkan penggabungan antara *hidden state* sebelumnya dan input pada waktu t . Fungsi σ merupakan fungsi sigmoid yang menghasilkan nilai antara 0 dan 1, sedangkan $\odot$ menunjukkan perkalian elemen per elemen.

Melalui rangkaian tersebut, LSTM dapat mempertahankan informasi yang masih dibutuhkan dan mengurangi pengaruh informasi yang tidak relevan. Kemampuan ini membuat LSTM sesuai untuk mempelajari ketergantungan temporal pada data *time series* [2, 11]. Penelitian terdahulu juga menunjukkan bahwa LSTM dapat digunakan untuk mempelajari pola non-linear pada data saham dan data finansial [12, 13].

Walaupun demikian, kinerja LSTM dipengaruhi oleh jumlah dan karakteristik data, rancangan arsitektur, serta pemilihan hiperparameter. LSTM memiliki lebih banyak parameter yang perlu dipelajari sehingga umumnya membutuhkan data dan waktu pelatihan yang lebih besar dibandingkan model yang lebih sederhana. Oleh karena itu, jumlah data yang bertambah dapat membantu model mempelajari pola temporal, tetapi tidak menjamin hasil prediksi selalu menjadi lebih baik [5, 14].

2.7 Evaluasi Model

Untuk mengukur seberapa baik performa model dalam menghasilkan prediksi, penelitian ini menggunakan beberapa matrik evaluasi yang umum dipakai pada studi sejenis [15], yaitu:

1. *Root Mean Squared Error* (RMSE)

Matrik ini menghitung akar dari rata-rata kuadrat selisih antara nilai aktual dan nilai prediksi. Semakin kecil nilai RMSE, semakin baik performa model yang bersangkutan. Perhitungan RMSE mengikuti persamaan berikut:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (2.13)$$

di mana y_i adalah nilai aktual, $\hat{y}_i$ adalah nilai prediksi, dan n adalah jumlah data.

2. *Mean Absolute Error* (MAE)

Berbeda dengan RMSE, MAE menghitung rata-rata nilai absolut dari selisih antara nilai aktual dan nilai prediksi, tanpa melalui proses pengkuadratan. Karena itu, MAE relatif lebih *robust* terhadap *outlier* dibandingkan RMSE. Persamaan yang digunakan adalah sebagai berikut:

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (2.14)$$

di mana y_i adalah nilai aktual, $\hat{y}_i$ adalah nilai prediksi, dan n adalah jumlah data.

3. *Mean Absolute Percentage Error* (MAPE)

MAPE menyatakan rata-rata kesalahan prediksi dalam bentuk persentase terhadap nilai aktual, sehingga hasilnya lebih mudah diinterpretasikan secara proporsional. Perhitungannya mengikuti persamaan berikut:

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right| \times 100\% \quad (2.15)$$

di mana y_i adalah nilai aktual, $\hat{y}_i$ adalah nilai prediksi, dan n adalah jumlah data.

4. Koefisien Determinasi (R^2)

Matrik ini menunjukkan seberapa besar proporsi variasi data aktual yang mampu dijelaskan oleh model. Rentang nilainya berada antara 0 hingga 1, di mana nilai yang semakin mendekati 1 mengindikasikan kemampuan prediksi model yang semakin baik. Apabila nilai R^2 bernilai negatif, hal itu menandakan bahwa prediksi model justru lebih buruk dibandingkan jika hanya menggunakan nilai rata-rata data sebagai prediksi. R^2 dihitung dengan persamaan berikut:

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (2.16)$$

di mana y_i adalah nilai aktual, $\hat{y}_i$ adalah nilai prediksi, $\bar{y}$ adalah rata-rata nilai aktual, dan n adalah jumlah data.

2.8 Penelitian Terdahulu

Topik prediksi harga saham menggunakan metode *Machine Learning* dan *Deep Learning* sudah banyak dieksplorasi melalui berbagai pendekatan, dengan tujuan utama meningkatkan akurasi prediksi pada data *time series* finansial. Bathla [3], misalnya, mendapati bahwa SVR dan LSTM sama-sama menunjukkan performa yang kompetitif, tergantung pada karakteristik dataset yang digunakan. Mehtab et al. [4] memperkuat hal tersebut dengan menunjukkan bahwa model berbasis LSTM secara konsisten mengungguli sejumlah model *Machine Learning* konvensional, seperti Random Forest dan SVM, dalam memprediksi indeks NIFTY 50. Senada dengan itu, Pradasyah dan Baita [16] melakukan perbandingan SVR dan LSTM pada prediksi harga saham Indonesia menggunakan *sliding window* 60 hari, dan mendapati bahwa SVR justru lebih unggul dibandingkan LSTM untuk prediksi jangka pendek.

Namun, hasil yang berbeda diperoleh Fischer dan Krauss [5], yang menemukan bahwa LSTM justru mampu mengungguli metode *Machine Learning* tradisional dalam prediksi pasar saham. Temuan ini sejalan dengan hasil Qiu et al. [11], yang menyatakan bahwa LSTM efektif dalam menangkap pola non-linear pada data *time series* finansial.

Sejalan dengan itu, Nelson et al. [12] membuktikan bahwa LSTM mampu mempelajari pola pergerakan harga saham dengan memanfaatkan data historis serta indikator teknikal, dan berhasil mencapai akurasi rata-rata sebesar 55,9% dalam memprediksi arah pergerakan harga saham. Pada konteks pasar saham Indonesia

secara spesifik, penelitian juga telah dilakukan oleh Haryono et al. [7] yang membandingkan beberapa model *deep learning* pada Bursa Efek Indonesia, serta Budiharto [13] yang menerapkan LSTM untuk memprediksi harga saham Indonesia selama masa pandemi COVID-19. Studi yang lebih baru oleh Amalia et al. [17] justru menunjukkan hasil sebaliknya pada data saham Indonesia, yaitu ARIMA secara signifikan mengungguli SVR (MAPE 2,01% berbanding 3,07%), yang mengindikasikan bahwa keunggulan model tidak selalu berpihak pada pendekatan *Machine Learning*. Di sisi lain, Dwiandiyanta et al. [8] mengembangkan model *hybrid deep learning* untuk prediksi saham di Bursa Efek Indonesia dan berhasil memperoleh peningkatan akurasi dibandingkan model tunggal.

Dari rangkaian penelitian tersebut, dapat ditarik benang merah bahwa belum ada model yang secara konsisten unggul pada seluruh kondisi. Performa suatu model tampaknya lebih banyak ditentukan oleh karakteristik data, periode pengamatan, fitur yang digunakan, serta konfigurasi parameter model itu sendiri. Atas dasar itulah, penelitian ini berupaya menganalisis lebih jauh perbandingan performa SVR dan LSTM pada data IHSG dengan mempertimbangkan variasi panjang *window* yang berbeda-beda. Ringkasan penelitian terdahulu disajikan pada Tabel 2.1.

Tabel 2.1. Penelitian Terdahulu

No	Peneliti & Tahun	Judul Penelitian	Metode	Hasil
1	Bathla (2020) [3]	Stock Price Prediction Using LSTM and SVR	LSTM, SVR	LSTM dan SVR menunjukkan performa kompetitif; LSTM unggul pada sebagian besar indeks saham yang diuji.
2	Mehtab et al. (2021) [4]	Stock Price Prediction Using Machine Learning and LSTM-Based Deep Learning Models	LSTM, Random Forest, Ridge Regression, SVM	Model berbasis LSTM secara konsisten mengungguli model <i>Machine Learning</i> konvensional dalam memprediksi indeks NIFTY 50.

Berlanjut ke halaman berikutnya

No	Peneliti & Tahun	Judul Penelitian	Metode	Hasil
3	Fischer & Krauss (2018) [5]	Deep Learning with Long Short-Term Memory Networks for Financial Market Predictions	LSTM, Random Forest, DNN	LSTM mengungguli metode <i>Machine Learning</i> tradisional dalam prediksi pasar saham S&P 500.
4	Qiu et al. (2020) [11]	Forecasting Stock Prices with Long-Short Term Memory Neural Network Based on Attention Mechanism	LSTM dengan <i>Attention Mechanism</i>	LSTM dengan <i>attention mechanism</i> efektif menangkap pola non-linear pada data <i>time series</i> finansial.
5	Amalia et al. (2025) [17]	Comparative Forecasting of Indonesian Stock Prices Using ARIMA and Support Vector Regression: A Statistical Learning Approach	ARIMA, SVR	ARIMA secara signifikan mengungguli SVR (MAPE 2,01% vs 3,07%) dalam memprediksi harga saham TLKM bulanan.
6	Henrique et al. (2018) [1]	Stock Price Prediction Using Support Vector Regression on Daily and Up to the Minute Prices	SVR dengan berbagai <i>kernel</i>	SVR dengan kernel RBF menghasilkan prediksi akurat pada harga saham harian menggunakan indikator teknikal.

Berlanjut ke halaman berikutnya

No	Peneliti & Tahun	Judul Penelitian	Metode	Hasil
7	Dwiandiyanta et al. (2025) [8]	Optimization of Stock Predictions on Indonesia Stock Exchange: A New Hybrid Deep Learning Method	Hybrid Deep Learning	Pendekatan hybrid deep learning meningkatkan akurasi prediksi saham pada Bursa Efek Indonesia dibandingkan model tunggal.
8	Oukhouya dan El Himdi (2023) [6]	Comparing Machine Learning Methods—SVR, XGBoost, LSTM, and MLP—for Forecasting the Moroccan Stock Market	SVR, XGBoost, LSTM, MLP	SVR mengungguli LSTM dan XGBoost dalam prediksi indeks saham harian MSI 20 menggunakan GridSearch.
9	Haryono et al. (2024) [7]	Stock Price Forecasting in Indonesia Stock Exchange Using Deep Learning: A Comparative Study	LSTM, BiLSTM, CNN-LSTM	Model <i>deep learning</i> mampu memprediksi harga saham di BEI dengan akurasi tinggi; BiLSTM menunjukkan performa terbaik.
10	Pradasyah & Baita (2025) [16]	Comparative Study of Support Vector Regression and Long Short-Term Memory for Stock Price Prediction	SVR, LSTM	SVR mengungguli LSTM dalam prediksi jangka pendek saham Indonesia (BBRI) menggunakan <i>sliding window</i> 60 hari.

Berlanjut ke halaman berikutnya

No	Peneliti & Tahun	Judul Penelitian	Metode	Hasil
11	Idrees et al. (2025) [14]	Forecasting Stock Prices Using Long Short-Term Memory Involving Attention Approach	LSTM dengan Attention Mechanism	Integrasi attention mechanism meningkatkan kemampuan LSTM dalam menangkap pola kompleks pada data pasar saham.

