**Jurnal Sifo Mikroskil (JSM)**
**Volume 27, No 2, Oktober 2026 – Hal 287 - 302**
**DOI: https://doi.org/10.55601/jsm.v27i2.2090**

**e-ISSN: 2622-8130**
**ISSN: 1412-0100**

# Prediksi Harga Saham dengan BiLSTM dan Improved-FA

**Nadiyah Shofa Salsabila[1], Arwin Halim[2], Heru Kurniawan[3]**
[1,2,3] Universitas Mikroskil, Jl. Thamrin No. 112, 124, 140, (061) 4573767
[1,2,3] Fakultas Informatika, Teknik Informatika, Universitas Mikroskil, Medan
e-mail: [1]nadiyah.sho12@gmail.com, [2]arwin@mikroskil.ac.id, [3]heru.kurniawan@mikroskil.ac.id

*Dikirim:* ***24-08-2026*** | *Diterima:* ***22-10-2026*** | *Diterbitkan:* ***31-10-2026***

**Abstrak**

Pergerakan harga saham yang fluktuatif menjadikan prediksi harga saham sebagai salah satu tantangan dalam pengambilan keputusan investasi. Penelitian ini bertujuan melakukan prediksi harga saham dengan menggabungkan model *Bidirectional Long Short-Term Memory* (BiLSTM) dan *Improved Firefly Algorithm* (IFA). Dataset yang digunakan berupa data historis saham PT Aneka Tambang Tbk (ANTM), PT Indofood Sukses Makmur Tbk (INDF), dan PT Unilever Indonesia Tbk (UNVR) yang diperoleh dari Yahoo Finance periode 2 Januari 2020 hingga 30 Desember 2025. Model usulan dioptimasi dengan menggunakan *hyperparameter* terhadap hidden neuron, batch size, dan dropout rate. Data dibagi menjadi 70% data pelatihan, 10% data validasi, dan 20% data pengujian. Evaluasi model menggunakan RMSE, MAE, dan MAPE. Hasil penelitian menunjukkan bahwa model IFA-BiLSTM memperoleh nilai MAPE sebesar 2,47% pada ANTM, 1,46% pada INDF, dan 2,20% pada UNVR. Hasil tersebut menunjukkan bahwa penerapan IFA mampu meningkatkan rata-rata performa model BiLSTM untuk memprediksi harga saham.

**Kata kunci**: Prediksi Harga Saham, BiLSTM, *Improved Firefly Algorithm*, Optimasi *Hyperparameter*, *Time Series*

***Abstract***

*Stock price fluctuations make stock price prediction a challenging task in financial forecasting. This study aims to obtain the optimal hyperparameter configuration for the Bidirectional Long Short-Term Memory (BiLSTM) model using the Improved Firefly Algorithm (IFA) to improve prediction performance. The dataset consists of historical stock price data of PT Aneka Tambang Tbk (ANTM), PT Indofood Sukses Makmur Tbk (INDF), and PT Unilever Indonesia Tbk (UNVR) obtained from Yahoo Finance for the period of January 2, 2020, to December 30, 2025. The optimized hyperparameters include the number of hidden neurons, batch size, and dropout rate. The dataset was divided sequentially into 70% training data, 10% validation data, and 20% testing data. Model performance was evaluated using Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE). The results show that the IFA-BiLSTM model achieved MAPE values of 2.47% for ANTM, 1.46% for INDF, and 2.20% for UNVR. These findings indicate that IFA is effective in obtaining optimal hyperparameter configurations and improving the prediction performance of the BiLSTM model.*

***Keywords***: *Stock Price Prediction, BiLSTM, Improved Firefly Algorithm, Hyperparameter Optimization, Time Series.*

## 1. PENDAHULUAN

Saham menjadi instrumen utama yang banyak diminati di pasar keuangan serta memberikan kontribusi penting dalam pergerakan aktivitas ekonomi [1]. Meskipun investasi saham berpotensi memberikan keuntungan yang besar, instrumen ini juga memiliki risiko yang tinggi akibat fluktuasi nilai yang dapat berubah secara tajam [2]. Penawaran dan permintaan menjadi faktor utama yang menentukan fluktuasi harga saham. Peningkatan minat beli akan mendorong harga naik, sedangkan tingginya keinginan menjual akan menekan harga turun [3]. Kondisi ini membuat lembaga sekuritas maupun investor memiliki ketertarikan yang tinggi dalam mengidentifikasi dan menganalisis tren pergerakan harga saham [4]. Perubahan harga saham umumnya menunjukkan pola yang bersifat non-linier, sehingga proses peramalan menjadi lebih kompleks dan sulit dilakukan secara akurat [5]. Oleh karena itu, diperlukan metode prediksi harga saham yang tepat agar dapat mendukung pengambilan keputusan investasi secara lebih optimal.

Berbagai penelitian telah dilakukan untuk mengembangkan metode prediksi harga saham. Salah satu peneliti membandingkan algoritma *Random Forest* dan *Support Vector Machine* (SVM). Hasil penelitian menunjukkan bahwa *Random Forest* memberikan akurasi yang lebih tinggi dibandingkan SVM, yaitu sekitar 80.8% untuk *Random Forest* dan 78.7% untuk SVM pada data pengujian [6]. Penelitian lainnya menggunakan Long Short Term Memory (LSTM) untuk memprediksi 10 saham acak. Hasil penelitian menunjukkan bahwa rata-rata akurasi prediksi mencapai sekitar 83% pada seluruh saham yang diuji, dengan akurasi tertinggi sebesar 83,33% diperoleh pada saham SBI [7]. Model ini menunjukkan kinerja yang baik dalam menangani beragam permasalahan yang berkaitan dengan prediksi deret waktu pada berbagai bidang [8]. Namun, LSTM tetap memiliki keterbatasan dalam memahami konteks secara menyeluruh, karena hanya memproses data ke arah maju (*forward*) [9]. Hal ini mengakibatkan model tidak dapat memanfaatkan informasi dari arah sebaliknya dalam suatu input historis [10].

Metode *Bidirectional Long Short-Term Memory* (BiLSTM) diterapkan untuk mengatasi keterbatasan LSTM, karena metode ini memproses data ke arah maju (*forward*) dan ke arah mundur (*backward*), sehingga mampu menangkap konteks lebih menyeluruh dan meningkatkan akurasi prediksi [11]. Penelitian Yang dan Wang [12] menggunakan BiLSTM untuk memprediksi harga penutupan indeks keuangan CSI 300 di China. Hasilnya BiLSTM mencapai akurasi prediksi tertinggi dibanding LSTM, SVR dan Arima dengan RMSE sebesar 0.6056 dan MAPE sebesar 0.4482. Selain itu, Puteri [13] menggunakan model BiLSTM untuk memprediksi harga saham syariah dan hasilnya lebih baik dibandingkan LSTM, dengan nilai MAPE masing-masing sebesar 2,59% pada saham ANTM.JK, 1,77% pada UNVR.JK, dan 1,05% pada INDF.JK, yang seluruhnya termasuk dalam kategori sangat akurat. Berdasarkan penelitian tersebut, model BiLSTM menunjukkan kinerja yang lebih unggul dalam melakukan prediksi. Namun, model ini juga rentan terhadap kinerja yang buruk jika konfigurasi *hyperparameter* tidak tepat sehingga dapat berdampak pada hasil prediksi [14].

Untuk mengatasi permasalahan tersebut, teknik optimasi diperlukan sebagai upaya mengefisienkan pencarian *hyperparameter* terbaik [15]. Salah satu pendekatan yang digunakan ialah *Improved Firefly Algorithm* (IFA) yang dimanfaatkan untuk menyempurnakan struktur jaringan saraf tiruan sehingga dapat meningkatkan akurasi dan performa model [16]. Algoritma IFA telah terbukti memiliki performa yang lebih unggul dibandingkan berbagai algoritma optimasi lainnya, *seperti simulated annealing* (SA), *niche genetic algorithm* (NGA*), improved discrete cuckoo search* (IDCS), *mutant firefly algorithm* (MFA), *cuckoo–firefly search* (CSFA), serta *best-neighborhood firefly algorithm* (BNFA) [17]. Pada penelitian Pratama [18], metode IFA diterapkan untuk mengoptimalkan *hyperparameter* model BiLSTM dalam melakukan prediksi harga Bitcoin. Hasil penelitian menunjukkan bahwa model IFA-BiLSTM mencapai RMSE sebesar 2051,55 dan MAPE sebesar 3,48%. Hasil tersebut menunjukkan bahwa penerapan IFA pada BiLSTM berpotensi meningkatkan kinerja model dalam melakukan prediksi data deret waktu finansial.

Penelitian ini bertujuan untuk mengembangkan model BiLSTM dengan penerapan *Improved Firefly Algorithm* (IFA) untuk prediksi harga saham dengan konfigurasi *hyperparameter* yang optimal.

Dengan membandingkan kinerja model BiLSTM sebelum dan setelah optimasi menggunakan IFA, penelitian ini diharapkan dapat menunjukkan efektivitas penerapan IFA dalam memperoleh konfigurasi *hyperparameter* yang menghasilkan kinerja prediksi saham yang lebih optimal.

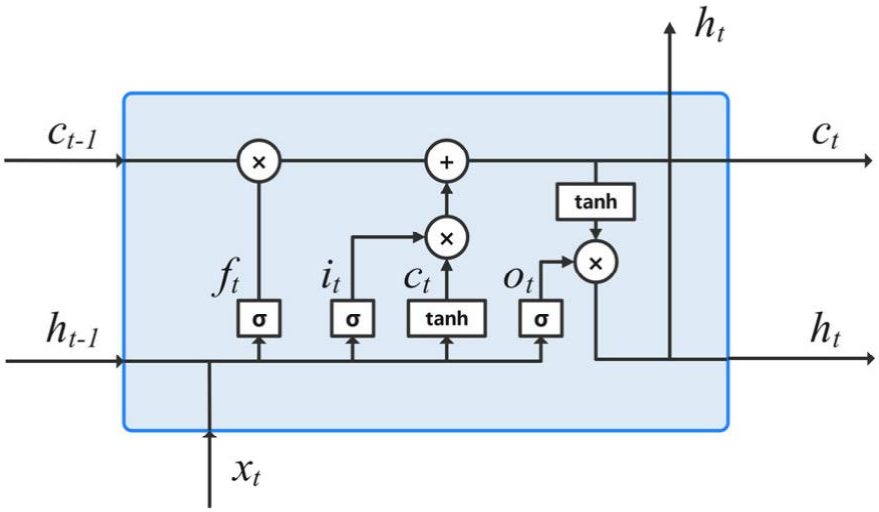
## 2. TINJAUAN PUSTAKA

### 2.1 Prediksi Harga Saham

Prediksi harga saham merupakan proses memperkirakan nilai saham di masa depan dengan memanfaatkan data historis [19]. Investor saham berupaya memilih perusahaan yang tepat untuk berinvestasi untuk memperoleh keuntungan dengan tingkat risiko yang relatif rendah [20]. Prediksi harga saham menjadi aspek penting dalam analisis sistem keuangan, mengingat pergerakan pasar saham dapat merepresentasikan kondisi ekonomi suatu negara hingga perekonomian dunia [21]. Namun, perubahan harga saham umumnya menunjukkan pola yang bersifat non-linier, sehingga proses peramalan menjadi lebih kompleks dan sulit dilakukan secara akurat [5]. Oleh karena itu, Prediksi harga saham yang akurat dapat membantu investor dalam merancang strategi investasi yang optimal serta meminimalkan risiko [22].

### 2.2 LSTM (*Long-Short Term Memory*)

*Long Short-Term Memory* (LSTM) adalah pengembangan dari arsitektur *Recurrent Neural Network* (RNN) yang pertama kali diperkenalkan pada tahun 1997 oleh Sepp Hochreiter dan Jürgen Schmidhuber [13]. Model LSTM memiliki kontribusi utama berupa mekanisme *self-loop* dengan pembaruan bobot secara iteratif yang memungkinkan aliran gradien bertahan dalam jangka panjang sehingga dapat mengatasi masalah *vanishing gradient* pada RNN [23]. Model ini juga memiliki sel memori untuk menyimpan informasi jangka panjang, di mana aliran informasi dalam sel tersebut dikontrol oleh tiga gerbang khusus, yaitu *forget gate, input gate,* dan *output gate* [24]. Model LSTM dapat dilihat pada Gambar 1.




Gambar 1. Model Jaringan *Long-Short Term Memory* [25]

Adapun tahapan dalam algoritma LSTM dapat dijelaskan sebagai berikut [26].

1. Perhitungan *input gate* dilakukan menggunakan persamaan:

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

2. Perhitungan *forget gate* dilakukan menggunakan persamaan:

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_t) \quad (2)$$

3. Hitung pembaruan *cell state* menggunakan persamaan:

$$c_t = f_t * c_{t-1} + i_t * \tanh(W_c[h_{t-1}, x_t] + b_c) \quad (3)$$

4. Perhitungan *output gate* dilakukan menggunakan persamaan:

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \quad (4)$$

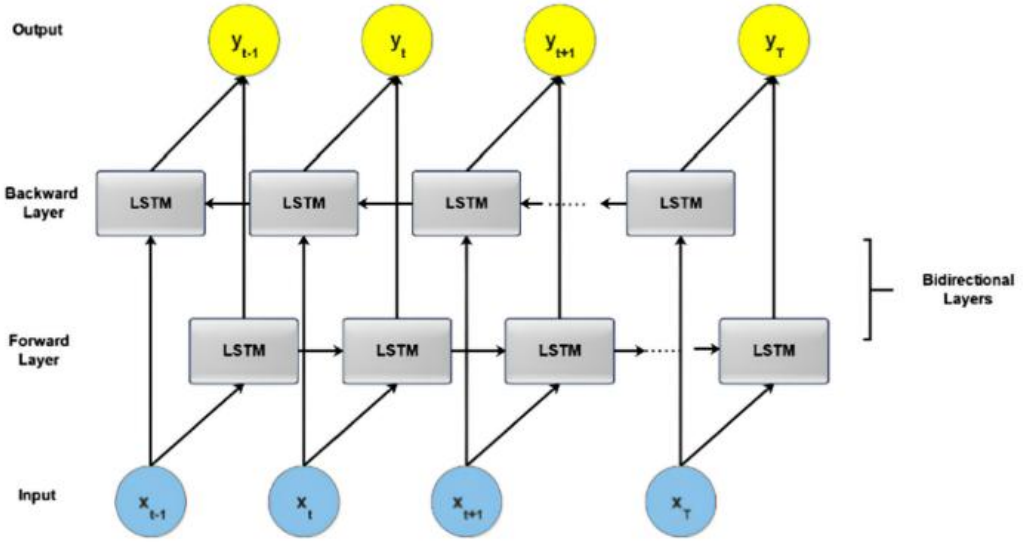
5. Perhitungan *hidden state* dilakukan menggunakan persamaan:

$$h_t = o_t * \tanh(c_t) \quad (5)$$

Dimana, $x_t$ merupakan input pada waktu ke-$t$, sedangkan $h_{t-1}$ dan $c_{t-1}$ masing-masing adalah *hidden state* dan *cell state* dari waktu sebelumnya. Nilai $x_t$ dan $h_{t-1}$ diproses oleh *input gate* ($i_t$), *forget gate* ($f_t$), dan *output gate* ($o_t$) dengan fungsi sigmoid. *forget gate* mengatur informasi dari $c_{t-1}$ yang dipertahankan, sementara *input gate* menentukan informasi baru yang ditambahkan ke *cell state*. *Cell state* diperbarui menjadi $c_t$, dan *hidden state* $h_t$ dihasilkan dari kombinasi $c_t$ melalui fungsi tanh dengan keluaran *output gate*. Parameter $W$ dan $b$ masing-masing merepresentasikan matriks bobot dan vektor bias yang digunakan dalam proses perhitungan pada setiap gerbang [7][26].

### 2.3 BiLSTM (*Bidirectional Long-Short Term Memory*)

LSTM memiliki keterbatasan dalam memahami konteks secara menyeluruh karena hanya memproses data ke *forward* [9]. Untuk mengatasi hal tersebut, dikembangkan *Bidirectional Long Short Term Memory* (BiLSTM) yang terdiri dari dua lapisan tersembunyi LSTM, di mana LSTM pertama mempelajari data dengan memprosesnya ke arah maju (*forward*), sedangkan LSTM kedua memproses informasi ke arah sebaliknya (*backward*) [27]. Output dari LSTM yang memproses maju dan mundur digabungkan pada setiap langkah waktu, sehingga dengan 2 *layer* yang saling berlawanan arah, model dapat mempelajari baik informasi dari masa lalu maupun masa depan untuk setiap urutan input [13]. Pendekatan ini memungkinkan jaringan mempertahankan informasi sebelumnya dan informasi mendatang untuk digunakan sebagai dasar waktu saat ini dari data *time series* [12].




Gambar 2. Model Jaringan BiLSTM [28]

Adapun tahapan dalam algoritma BiLSTM dapat dijelaskan sebagai berikut [9].

1. Tentukan nilai input untuk lapisan tersembunyi di arah maju dengan persamaan berikut.

$$\overrightarrow{h_t} = f(w_1 x_t + w_2 h_{t-1}) \quad (6)$$

2. Tentukan nilai lapisan tersembunyi di arah mundur dengan persamaan berikut.

$$\overleftarrow{h_t} = f(w_3 x_t + w_5 h_{t+1}) \quad (7)$$

3. Tentukan nilai lapisan output dengan mengombinasikan hasil dari lapisan maju dan lapisan mundur sesuai persamaan berikut.

$$o_t = g(w_4\overrightarrow{h_t} + w_6\overleftarrow{h_t}) \quad (8)$$

Dimana, $x_t$ merupakan *input* pada waktu ke- t, $\overrightarrow{h_t}$ *adalah hidden state* pada arah maju, dan $\overleftarrow{h_t}$ adalah *hidden state* pada arah mundur. Variabel $h_{t-1}$ dan $h_{t+1}$ masing-masing menyatakan *hidden state* pada waktu sebelumnya dan waktu berikutnya. Parameter $w_1$, $w_2$, $w_3$, $w_4$, $w_5$, dan $w_6$ merupakan bobot yang digunakan dalam proses pembelajaran model. Fungsi f(·) digunakan pada lapisan tersembunyi, sedangkan g(·) digunakan pada lapisan keluaran. Nilai $o_t$ merupakan output yang diperoleh dari penggabungan *hidden state* arah maju dan arah mundur [9][12].

## 2.4 FA (*Firefly Algorithm*)

*Firefly Algorithm* (FA) merupakan algoritma metaheuristik yang kuat dan efisien, yang telah terbukti memiliki kinerja yang efektif dalam berbagai studi terbaru ketika digunakan untuk menyelesaikan masalah optimasi di bidang teknik [29]. Algoritma ini terinspirasi oleh perilaku berkedip kunang-kunang, di mana kinerja setiap solusi yang dihasilkan secara acak pada fungsi tujuan dinilai berdasarkan intensitas kecerahannya [30].

Parameter masukan untuk *firefly algorithm* meliputi N, yaitu jumlah kunang-kunang dalam populasi, parameter acak α, parameter daya tarik β ketika jarak r dari sumber sama dengan nol, koefisien reduksi cahaya γ, serta jumlah iterasi maksimum yang dijalankan oleh algoritma [18].

Intensitas cahaya dijelaskan pada persamaan (9), di mana, I(r) menyatakan intensitas cahaya pada jarak r, γ merupakan koefisien penyerapan cahaya, dan $I_0$ adalah intensitas cahaya pada sumbernya [31].

$$I(r) = I_0 e^{(-\gamma r^2)} \quad (9)$$

Selain itu, setiap kunang-kunang memanfaatkan daya tarik β, yang sebanding dengan intensitas cahaya dari kunang-kunang tertentu dan juga dipengaruhi oleh jarak [32]. Sebagaimana ditunjukkan dalam persamaan (10) [33].

$$\beta(r) = \beta_0 e^{(-\gamma r^2)} \quad (10)$$

Persamaan (11) digunakan untuk menghitung jarak antara dua kunang-kunang, i dan j, yang berada pada posisi $x_i$ dan $x_j$ masing-masing, dan dapat didefinisikan sebagai *Cartesian distance* [32]. Di sini, $x_{ik}$ merupakan elemen ke-k dari koordinat spasial kunang-kunang ke-i, sedangkan D menunjukkan dimensi dari permasalahan yang sedang dipertimbangkan [34].

$$r_{ij} = \|x_i - x_j\| = \sqrt{\sum_{k=1}^{D}(x_{i,k} - x_{j,k})^2} \quad (11)$$

Kunang-kunang cenderung tertarik pada individu yang lebih terang dan lebih menarik. Perilaku ini dapat dimodelkan melalui persamaan (12), di mana bagian kedua persamaan tersebut berasal dari persamaan daya tarik, yang pada dasarnya melibatkan pengaruh jarak terhadap tingkat daya tarik antara dua kunang-kunang [18][31]. Dimana, α merupakan parameter pengacakan, sedangkan angka acak yang diambil dari distribusi gaussian atau distribusi seragam dilambangkan dengan κ. Simbol $r_{i,j}$ menunjukkan jarak antara dua kunang-kunang yang diamati, i dan j. Nilai tipikal yang umumnya memberikan hasil memuaskan untuk sebagian besar masalah adalah $\beta_0 = 1$ dan α berada dalam rentang [0, 1] [32].

$$x_i^{t+1} = x_i^t + \beta_0 e^{-\gamma r_{i,j}^2}\left(x_j^t - x_i^t\right) + \alpha^t (k - 0.5) \quad (12)$$

Parameter γ memiliki peran penting dalam menentukan laju konvergensi algoritma. Secara teoritis, nilai γ dapat berada dalam rentang $\gamma \in [0, \infty)$. Dalam praktiknya, pemilihan nilai γ biasanya disesuaikan dengan jenis masalah yang dihadapi, dan umumnya berada pada kisaran 0,1 hingga 10 [18].

## 2.5 IFA (*Improved Firefly Algorithm*)

Pada *Firefly Algorithm*, solusi dengan nilai optimum terbaik direpresentasikan oleh kunang-kunang dengan tingkat kecerahan yang sangat tinggi [35]. Namun, pendekatan klasik sering mengalami permasalahan terjebak pada minimum lokal akibat karakteristik perancangan yang bersifat nonlinier [36]. Untuk mengatasi keterbatasan tersebut, dikembangkan *Improved Firefly Algorithm* yang bertujuan meningkatkan kemampuan algoritma dalam menghindari jebakan minimum lokal. Pada metode ini,

ditambahkan suatu komponen baru ke dalam algoritma FA untuk meningkatkan tingkat keacakan dan efisiensi pencarian. Selanjutnya, perbedaan antara posisi kunang-kunang acak dan posisi kunang-kunang ke-X dihitung, sehingga diperoleh mekanisme keacakan yang lebih efektif [35].

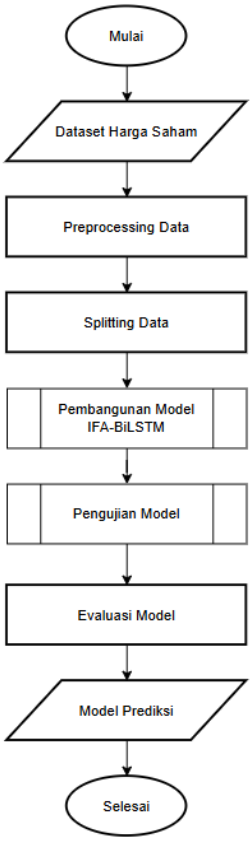
1. *The Minimum Attraction*: Untuk mencegah situasi di mana kunang-kunang bergerak tanpa arah akibat jarak yang terlalu besar dan daya tarik yang menurun, diperkenalkan konsep daya tarik minimum sebagai langkah antisipatif, sebagaimana ditunjukkan pada Persamaan sebagai berikut.
$$\beta_{ij}(r_{ij}) = \beta_{min} + (\beta_0 - \beta_{min})e^{-\gamma r_{ij}^2} \quad (13)$$
Dimana, nilai $\beta_0$ ditetapkan sebesar 1, sedangkan $\beta_k$ berada dalam rentang [0, 1]. Dengan demikian, ketika jarak antara kunang-kunang menjadi sangat besar dan mendekati nol daya tarik efektif, tingkat daya tarik antar individu dapat mencapai $\beta_k$ [18].

2. *Adaptive Step Length Factor*: Untuk mengatasi ketidakseimbangan antara tahap eksplorasi awal dan tahap konvergensi akhir dari keseluruhan proses optimasi, penyesuaian dinamis α diperkenalkan, sebagaimana dalam Persamaan (14), dengan parameter p yang dihitung menurut Persamaan (15).
$$\alpha = \alpha_0 * P \quad (14)$$
$$P = \frac{Max - t}{Max} \quad (15)$$
Di mana Max menyatakan jumlah iterasi maksimum, sedangkan t menunjukkan iterasi saat ini [16].

3. *Inertia Weight*: Dengan mengacu pada pengembangan *Particle Swarm Optimization* yang menggunakan bobot inersia, algoritma dasar ditingkatkan melalui penerapan pengurangan bobot inersia secara linier [16][18]. Sebagaimana ditunjukkan pada Persamaan sebagai berikut.
$$w = w_{max} - (w_{max} - w_{min}) * \frac{t}{MaxT} \quad (16)$$
Pada persamaan tersebut, $w_{max}$ dan $w_{min}$ menunjukkan bobot maksimum dan minimum, sementara t serta $t_{max}$ menyatakan iterasi ke-t saat ini dan jumlah iterasi maksimum [16][18]. Proses pembaruan posisi selanjutnya dilakukan berdasarkan Persamaan sebagai berikut.
$$x_i^{t+1} = wx(t) + \beta\left(x_j^t - x_i^t\right) + \alpha^t\,(k - 0.5) \quad (17)$$

## 3. METODE PENELITIAN

Penelitian ini dilakukan melalui beberapa tahapan, yaitu pengumpulan dataset harga saham, *preprocessing* data, *Spliting* dataset, pembangunan dan optimasi model IFA-BiLSTM, pengujian model, serta evaluasi kinerja prediksi. Tahapan penelitian secara keseluruhan ditunjukkan pada Gambar 3.




Gambar 3. Flowchart Penelitian

### 3.1 Dataset Harga Saham

Dataset yang digunakan dalam penelitian ini berupa data historis harga saham yang diperoleh dari Yahoo Finance. Data yang digunakan terdiri atas tiga saham, yaitu ANTM.JK, INDF.JK, dan UNVR.JK, dengan periode pengamatan 2 Januari 2020 hingga 30 Desember 2025. Masing-masing dataset terdiri atas 1.447 data observasi harian. Atribut yang digunakan meliputi *Open*, *High*, *Low*, dan *Close* (OHLC), dengan harga *Close* digunakan sebagai target prediksi.

### 3.2 *Preprocessing* Data

Data harga saham selanjutnya melalui tahap *preprocessing* berupa normalisasi menggunakan metode *Min-Max Normalization*. Normalisasi dilakukan dengan mengubah nilai setiap atribut ke dalam rentang 0 hingga 1 agar skala data menjadi seragam dan dapat mendukung proses pelatihan model BiLSTM.

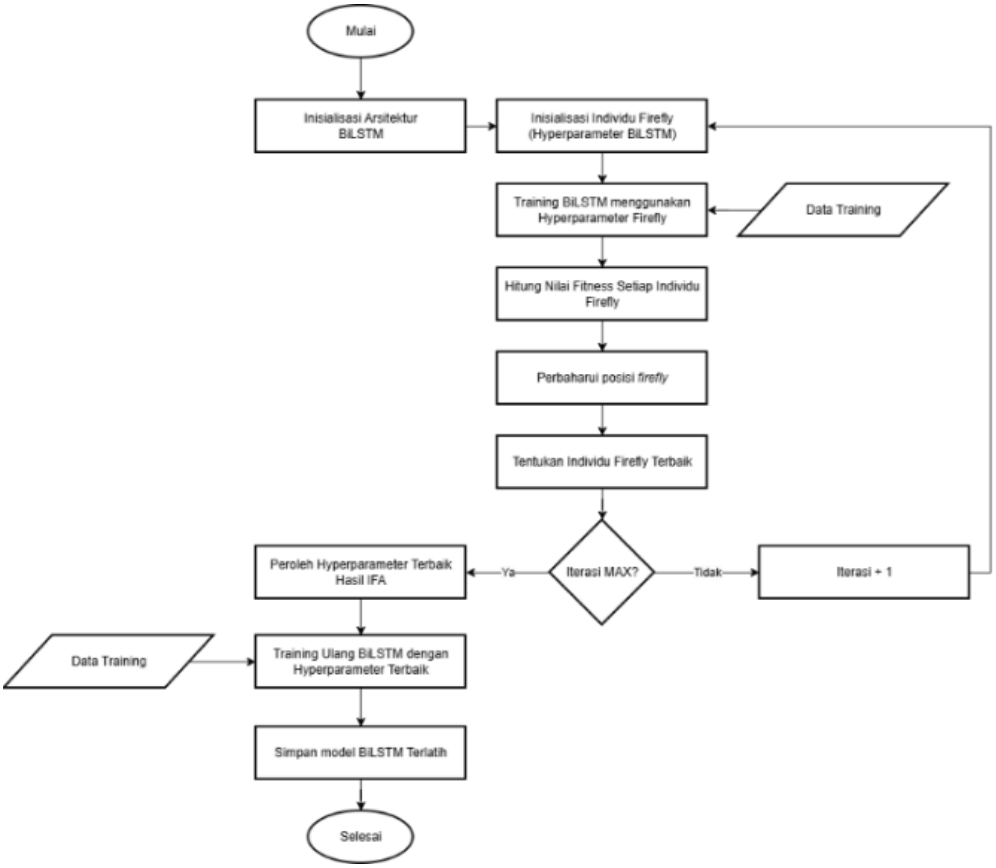
### 3.3 *Spliting* Data

Setelah proses normalisasi, dataset dibagi secara berurutan menjadi data *training*, *validation*, dan *testing* dengan proporsi masing-masing sebesar 70%, 10%, dan 20%. Pembagian dilakukan berdasarkan urutan waktu untuk mempertahankan karakteristik data deret waktu. Data *training* digunakan untuk melatih model, data *validation* digunakan dalam proses optimasi *hyperparameter*, sedangkan data *testing* digunakan untuk menguji kinerja model pada data yang belum digunakan selama proses pelatihan.

### 3.4 Pembangunan Model IFA-BiLSTM

Pembangunan model IFA-BiLSTM dilakukan melalui proses optimasi *hyperparameter* menggunakan *Improved Firefly Algorithm* (IFA). Proses diawali dengan inisialisasi arsitektur BiLSTM dan parameter IFA, kemudian setiap *firefly* merepresentasikan kandidat konfigurasi *hyperparameter*. Kandidat tersebut digunakan untuk membangun dan melatih model BiLSTM menggunakan data *training*, kemudian kinerja setiap kandidat dihitung berdasarkan nilai RMSE sebagai fungsi *fitness*. Selanjutnya, posisi *firefly* diperbarui berdasarkan nilai *fitness* hingga kondisi berhenti tercapai. Konfigurasi dengan nilai *fitness* terbaik ditetapkan sebagai *hyperparameter* optimal dan digunakan untuk membangun model BiLSTM akhir.

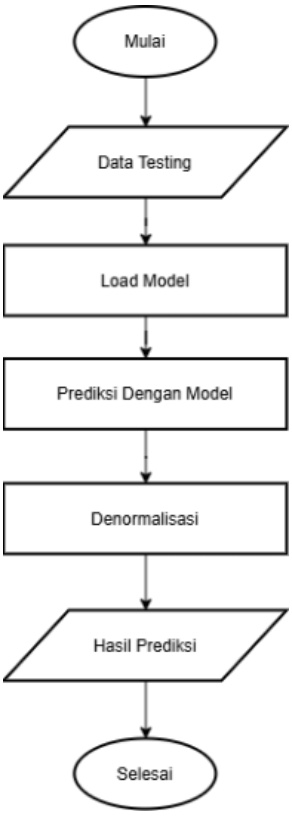
Parameter IFA yang digunakan meliputi jumlah *firefly* sebanyak 5, maksimum 10 iterasi, $\beta_0$ sebesar 1, $\beta_{min}$ sebesar 0,1, $\gamma$ sebesar 0,5, dan $\alpha_0$ sebesar 0,2. *Hyperparameter* yang dioptimalkan terdiri atas *hidden neurons* dengan rentang 10–200, *batch size* 20–200, dan *dropout rate* 0–0,8. Setelah konfigurasi optimal diperoleh, model BiLSTM dilatih kembali menggunakan konfigurasi tersebut dan model hasil pelatihan disimpan untuk digunakan pada tahap pengujian.




Gambar 4. Flowchart Pembangunan Model IFA-BiLSTM

### 3.5 Pengujian Model

Pengujian dilakukan menggunakan data *testing* untuk mengetahui kemampuan model IFA-BiLSTM dalam memprediksi harga saham pada data yang belum digunakan selama proses pelatihan. Tahapan pengujian diawali dengan memasukkan data *testing*, kemudian memuat model BiLSTM terlatih untuk menghasilkan prediksi harga *Close*. Hasil prediksi yang masih berada dalam skala normalisasi selanjutnya dikembalikan ke skala harga aktual melalui proses denormalisasi. Hasil prediksi tersebut kemudian digunakan pada tahap evaluasi model.




Gambar 5. Flowchart Pengujian Model

### 3.6 Evaluasi Model

Evaluasi dilakukan untuk mengukur kinerja model IFA-BiLSTM dalam memprediksi harga saham menggunakan data *testing*. Kinerja model dievaluasi menggunakan *Mean Absolute Error* (MAE), *Root Mean Square Error* (RMSE), dan *Mean Absolute Percentage Error* (MAPE). MAE mengukur rata-rata nilai absolut kesalahan prediksi, RMSE mengukur akar dari rata-rata kuadrat kesalahan sehingga memberikan penalti lebih besar terhadap kesalahan yang tinggi, sedangkan MAPE mengukur rata-rata persentase kesalahan antara nilai aktual dan hasil prediksi. Nilai MAE, RMSE, dan MAPE yang lebih rendah menunjukkan kinerja prediksi yang lebih baik.

## 4. HASIL DAN PEMBAHASAN

### 4.1 Optimasi *Hyperparameter* IFA

Model BiLSTM tanpa optimasi digunakan sebagai *baseline* dengan *timestep* sebesar 7, jumlah *hidden layer* sebanyak 1, *learning rate* sebesar 0,01, dan *epoch* sebanyak 50. Untuk memperoleh konfigurasi *hyperparameter* terbaik, model *baseline* diuji menggunakan tiga konfigurasi *hyperparameter*, yaitu BiLSTM-1, BiLSTM-2, dan BiLSTM-3. Konfigurasi yang digunakan ditunjukkan pada Tabel 1.

Tabel 1. Konfigurasi *Hyperparameter* BiLSTM *Baseline*

| Model | Hidden Neurons | Batch Size | Dropout Rate |
|---|---|---|---|
| BiLSTM-1 | 64 | 128 | 0.01 |
| BiLSTM-2 | 32 | 64 | 0.02 |
| BiLSTM-3 | 128 | 32 | 0.01 |

Hasil pengujian ketiga konfigurasi tersebut digunakan sebagai dasar pemilihan model *baseline* terbaik pada masing-masing saham. Selanjutnya, IFA diterapkan untuk mengoptimasi *hidden neurons*, *batch size*, dan *dropout rate*, sedangkan *timestep*, *hidden layer*, *learning rate*, dan *epoch* tetap menggunakan konfigurasi yang sama dengan model *baseline*.

Tabel 2. Parameter FA dan Ruang Pencarian Hyperparameter

| Hyperparameter | Nilai |
|---|---|
| Hidden Neuron | 10 - 200 |
| Batch Size | 20 - 200 |
| Dropout Rate | 0 - 0.8 |
| Jumlah Firefly (n) | 5 |
| Jumlah Iterasi (MAX) | 10 |
| $\beta_0$ | 1 |
| $\beta$min | 0.1 |
| $\gamma$ | 0.5 |
| $\alpha_0$ | 0.2 |
| Wmin | 0.4 |
| Wmax | 0.9 |

Hasil optimasi IFA menghasilkan konfigurasi *hyperparameter* terbaik yang berbeda pada masing-masing saham. Setiap konfigurasi dipilih berdasarkan nilai *fitness* berupa RMSE, sehingga konfigurasi dengan nilai *fitness* yang lebih rendah menunjukkan tingkat kesalahan prediksi yang lebih kecil. Hasil konfigurasi *hyperparameter* terbaik untuk masing-masing saham ditunjukkan pada Tabel 3.

Tabel 3. *Hyperparameter* Model IFA-BiLSTM Terbaik Setiap Jenis Saham

| Saham | Hidden Neurons | Batch Size | Dropout Rate |
|---|---|---|---|
| ANTM | 26.53 ≈ 27 | 34.31 ≈ 34 | 0.0414 |
| INDF | 25.18 ≈ 25 | 31.20 ≈ 31 | 0.1848 |
| UNVR | 20.72 ≈ 21 | 22.15 ≈ 22 | 0.0454 |

### 4.2 Prediksi Harga Saham

Setelah proses pelatihan menggunakan konfigurasi *hyperparameter* terbaik hasil optimasi IFA, model IFA-BiLSTM digunakan untuk melakukan prediksi pada data *testing* sebesar 20% yang tidak digunakan selama proses pelatihan. Model menggunakan tujuh periode historis (*timestep* = 7) sebagai masukan untuk memprediksi harga penutupan (*Close*) pada periode berikutnya. Hasil prediksi kemudian dikembalikan ke skala harga aktual melalui proses denormalisasi.




Gambar 6. Perbandingan Harga *Close* Aktual dan Prediksi Saham ANTM, INDF, dan UNVR menggunakan IFA-BiLSTM

Berdasarkan hasil visualisasi, prediksi model IFA-BiLSTM secara umum mengikuti pola pergerakan harga *Close* aktual pada ketiga saham. Model mampu menangkap kecenderungan naik dan turun pada data *testing*, meskipun pada beberapa periode masih terdapat perbedaan antara nilai aktual dan prediksi, terutama ketika terjadi perubahan harga yang relatif tajam. Perbedaan tersebut menunjukkan bahwa model belum sepenuhnya menangkap seluruh fluktuasi harga saham, namun secara umum mampu mengikuti pola temporal data historis.

### 4.3 Evaluasi dan Perbandingan Kinerja Model

Evaluasi model dilakukan untuk mengukur dan membandingkan kinerja model BiLSTM dan IFA-BiLSTM dalam memprediksi harga saham ANTM, INDF, dan UNVR. Evaluasi menggunakan tiga metrik, yaitu *Root Mean Squared Error* (RMSE), *Mean Absolute Error* (MAE), dan *Mean Absolute Percentage Error* (MAPE), dengan nilai yang lebih rendah menunjukkan tingkat kesalahan prediksi yang lebih kecil. Pada model BiLSTM, evaluasi dilakukan terhadap tiga konfigurasi *hyperparameter*, yaitu BiLSTM-1, BiLSTM-2, dan BiLSTM-3, sedangkan model IFA-BiLSTM menggunakan konfigurasi *hyperparameter* terbaik yang diperoleh melalui proses optimasi IFA. Hasil evaluasi seluruh konfigurasi model disajikan pada Tabel 4.

Tabel 4. Tabel Evaluasi Seluruh Model Prediksi

| Saham | Model | RMSE | MAE | MAPE |
|---|---|---|---|---|
| ANTM | BiLSTM-1 | 135,80 | 103,09 | 3,95% |
| | BiLSTM-2 | 121,17 | 91,97 | 3,58% |
| | BiLSTM-3 | **86,16** | **62,26** | **2,51%** |
| | **IFA-BiLSTM** | **83,72** | **60,67** | **2,47%** |
| INDF | BiLSTM-1 | 172,55 | 136,20 | 1,78% |
| | BiLSTM-2 | **143,13** | 110,80 | **1,46%** |
| | BiLSTM-3 | 153,24 | 120,21 | 1,60% |
| | **IFA-BiLSTM** | 143,97 | **110,48** | **1,46%** |

| Saham | Model | RMSE | MAE | MAPE |
|---|---|---|---|---|
| UNVR | BiLSTM-1 | 90,85 | **63,80** | **3,61%** |
| | BiLSTM-2 | 92,40 | 72,66 | 4,25% |
| | BiLSTM-3 | **87,64** | 73,05 | 4,17% |
| | **IFA-BiLSTM** | **77,89** | **56,35** | **2,20%** |

Berdasarkan Tabel 4, performa ketiga konfigurasi BiLSTM menunjukkan hasil yang berbeda pada masing-masing saham. Pada ANTM, BiLSTM-3 menghasilkan kinerja terbaik dibandingkan konfigurasi lainnya dengan RMSE 86,16, MAE 62,26, dan MAPE 2,51%. Pada INDF, BiLSTM-2 menghasilkan RMSE dan MAPE terendah masing-masing sebesar 143,13 dan 1,46%, sedangkan BiLSTM-3 memberikan nilai MAE yang lebih tinggi. Pada UNVR, BiLSTM-1 menghasilkan MAE dan MAPE terendah dibandingkan konfigurasi lainnya, meskipun RMSE BiLSTM-3 sedikit lebih rendah.

Setelah diperoleh konfigurasi baseline terbaik pada masing-masing saham, kinerja tersebut dibandingkan dengan model IFA-BiLSTM. Pada ANTM, IFA-BiLSTM menghasilkan penurunan RMSE dari 86,16 menjadi 83,72, MAE dari 62,26 menjadi 60,67, dan MAPE dari 2,51% menjadi 2,47%. Pada UNVR, peningkatan kinerja lebih terlihat dengan penurunan RMSE dari 90,85 menjadi 77,89, MAE dari 63,80 menjadi 56,35, dan MAPE dari 3,61% menjadi 2,20%.

Sementara itu, pada INDF, IFA-BiLSTM menghasilkan RMSE sebesar 143,97, sedikit lebih tinggi dibandingkan BiLSTM-2 sebesar 143,13. Namun, MAE menurun dari 110,80 menjadi 110,48 dan MAPE tetap sebesar 1,46%. Hasil tersebut menunjukkan bahwa optimasi IFA memberikan peningkatan kinerja pada ANTM dan UNVR, sedangkan pada INDF peningkatan hanya terlihat pada sebagian metrik evaluasi karena tingginya data *outlier* pada harga saham INDF. Namun, performa rata-rata algoritma yang diusulkan masih sedikit lebih baik.

## 5. KESIMPULAN

Penelitian ini menerapkan *Improved Firefly Algorithm* (IFA) untuk mengoptimalkan *hyperparameter* model BiLSTM dalam prediksi harga saham ANTM, INDF, dan UNVR. Hasil optimasi menghasilkan konfigurasi *hyperparameter* yang berbeda pada setiap saham, yaitu *hidden neurons*, *batch size*, dan *dropout rate*. Berdasarkan hasil evaluasi, penerapan IFA-BiLSTM meningkatkan kinerja prediksi pada saham ANTM dan UNVR. Pada ANTM, nilai RMSE, MAE, dan MAPE masing-masing menurun dari 86,16 menjadi 83,72, 62,26 menjadi 60,67, dan 2,51% menjadi 2,47%. Pada UNVR, nilai RMSE menurun dari 90,85 menjadi 77,89, MAE dari 63,80 menjadi 56,35, dan MAPE dari 3,61% menjadi 2,20%. Sementara itu, pada INDF, nilai MAE menurun dari 110,80 menjadi 110,48 dan MAPE tetap sebesar 1,46%, tetapi nilai RMSE meningkat dari 143,13 menjadi 143,97. Hasil tersebut menunjukkan bahwa penerapan IFA memberikan peningkatan kinerja pada sebagian dataset, meskipun belum memberikan peningkatan pada seluruh metrik untuk setiap saham.

## 6. SARAN

Penelitian selanjutnya dapat memperluas parameter untuk optimasi *hyperparameter* pada model IFA-BiLSTM dan menguji variasi *timestep* untuk mendapatkan konfigurasi model sesuai karakteristik dataset saham yang berbeda-beda, seperti saham di sektor banking, mining, telekomunikasi dan lainnya. Pengembangan selanjutnya juga dapat mengeksplorasi algoritma optimasi atau kombinasi model lainnya sebagai pembanding untuk memperoleh kinerja prediksi yang lebih optimal.

## DAFTAR PUSTAKA

[1] M. Sharaf, E. E.-D. Hemdan, A. El-Sayed, and N. A. El-Bahnasawy, "StockPred: a framework for stock Price prediction," *Multimed. Tools Appl.*, vol. 80, no. 12, pp. 17923–17954, 2021, doi: 10.1007/s11042-021-10579-8.

[2] W. Budiharto, "Data science approach to stock prices forecasting in Indonesia during Covid - 19 using Long Short - Term Memory ( LSTM )," *J. Big Data*, 2021, doi: 10.1186/s40537-021-00430-0.

[3] M. Diqi, M. Endah, H. Adri, and S. Nur, "StockGAN : robust stock price prediction using GAN algorithm," *Int. J. Inf. Technol.*, vol. 14, no. 5, pp. 2309–2315, 2022, doi: 10.1007/s41870-022-00929-6.

[4] C. Zhao *et al.*, "Progress and prospects of data-driven stock price forecasting research," *Int. J. Cogn. Comput. Eng.*, vol. 4, pp. 100–108, 2023, doi: https://doi.org/10.1016/j.ijcce.2023.03.001.

[5] W. Lu, "A CNN-BiLSTM-AM method for stock price prediction," *Neural Comput. Appl.*, vol. 0123456789, 2020, doi: 10.1007/s00521-020-05532-z.

[6] M. M. Akhtar, A. S. Zamani, S. Khan, A. S. A. Shatat, S. Dilshad, and F. Samdani, "Stock market prediction based on statistical data using machine learning algorithms," *J. King Saud Univ. - Sci.*, vol. 34, no. 4, p. 101940, 2022, doi: https://doi.org/10.1016/j.jksus.2022.101940.

[7] P. S. Sisodia, A. Gupta, Y. Kumar, and G. K. Ameta, "Stock Market Analysis and Prediction for Nifty50 using LSTM Deep Learning Approach," in *2022 2nd International Conference on Innovative Practices in Technology and Management (ICIPTM)*, 2022, pp. 156–161. doi: 10.1109/ICIPTM54933.2022.9754148.

[8] F. P. Simamora, R. Purba, and M. F. Pasha, "Optimisasi Hyperparameter BiLSTM Menggunakan Bayesian Optimization untuk Prediksi Harga Saham," *Jambura J. Math.*, vol. 7, no. 1, pp. 8–13, 2025, doi: 10.37905/jjom.v7i1.27166.

[9] K. Chen, R. Purba, and A. Halim, "Stock Price Prediction Using XCEEMDAN-Bidirectional LSTM -Spline," *Indones. J. Artif. Intell. Data Min.*, vol. 5, no. 1, p. 1, 2022, doi: 10.24014/ijaidm.v5i1.14424.

[10] Y. Jiang, "Emotion Algorithm Analysis and Expression Optimization of Film and Television Drama Lines," *Appl. Math. Nonlinear Sci.*, vol. 9, no. 1, pp. 1–17, 2024, doi: 10.2478/amns-2024-1930.

[11] R. L. Abduljabbar, H. Dia, and P.-W. Tsai, "Unidirectional and Bidirectional LSTM Models for Short-Term Traffic Prediction," *J. Adv. Transp.*, vol. 2021, no. 1, p. 5589075, 2021, doi: https://doi.org/10.1155/2021/5589075.

[12] M. Yang and J. Wang, "Adaptability of Financial Time Series Prediction Based on BiLSTM," *Procedia Comput. Sci.*, vol. 199, pp. 18–25, 2022, doi: https://doi.org/10.1016/j.procs.2022.01.003.

[13] D. I. Puteri, "Implementasi Long Short Term Memory (LSTM) dan Bidirectional Long Short Term Memory (BiLSTM) dalam Prediksi Harga Saham Syariah," *EULER J. Ilm. Mat. Sains dan Teknol.*, vol. 11, no. 1, pp. 35–43, Jun. 2023, doi: 10.34312/euler.v11i1.19791.

[14] H. Dhake, Y. Kashyap, and P. Kosmopoulos, "Algorithms for Hyperparameter Tuning of LSTMs for Time Series Forecasting," *Remote Sens.*, vol. 15, no. 8, 2023, doi: 10.3390/rs15082076.

[15] B. Gülmez, "Stock price prediction with optimized deep LSTM network with artificial rabbits optimization algorithm," *Expert Syst. Appl.*, vol. 227, p. 120346, 2023, doi: https://doi.org/10.1016/j.eswa.2023.120346.

[16] M. Ulina, R. Purba, and A. Halim, "Foreign Exchange Prediction using CEEMDAN and Improved FA-LSTM," *2020 5th Int. Conf. Informatics Comput. ICIC 2020*, 2020, doi: 10.1109/ICIC50835.2020.9288615.

[17] H. Xing, H. Wu, Y. Chen, and K. Wang, "A cooperative interference resource allocation method based on improved fi re fl y algorithm," *Def. Technol.*, vol. 17, no. 4, pp. 1352–1360, 2021, doi: 10.1016/j.dt.2020.07.006.

[18] A. Pratama, "Bitcoin Cryptocurrency Price Prediction Using IFA-BiLSTM," in *Proceedings - 2024 2nd International Conference on Technology Innovation and Its Applications, ICTIIA 2024*, 2024, pp. 1–6. doi: 10.1109/ICTIIA61827.2024.10761776.

[19] T. Phuoc, P. T. K. Anh, P. H. Tam, and C. V. Nguyen, "Applying machine learning algorithms to predict the stock price trend in the stock market – The case of Vietnam," *Humanit. Soc. Sci. Commun.*, vol. 11, no. 1, pp. 1–18, 2024, doi: 10.1057/s41599-024-02807-x.

[20] N. Afrianto, D. H. Fudholi, and S. Rani, "Prediksi Harga Saham Menggunakan BiLSTM dengan Faktor Sentimen Publik," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 6, no. 1, pp. 41–46, 2022, doi: 10.29207/resti.v6i1.3676.

[21] G. Li, A. Zhang, Q. Zhang, D. Wu, and C. Zhan, "Pearson Correlation Coefficient-Based Performance Enhancement of Broad Learning System for Stock Price Prediction," *IEEE Trans. Circuits Syst. II Express Briefs*, vol. 69, no. 5, pp. 2413–2417, 2022, doi: 10.1109/TCSII.2022.3160266.

[22] F. H. Sezgin, Ö. Algorabi, G. Sart, and M. Güler, "Hyperparameter-Optimized RNN, LSTM, and GRU Models for Airline Stock Price Prediction: A Comparative Study on THYAO and PGSUS," *Symmetry (Basel).*, vol. 17, no. 11, 2025, doi: 10.3390/sym17111905.

[23] Z. Hu, Y. Zhao, and M. Khushi, "A Survey of Forex and Stock Price Prediction Using Deep Learning," *Appl. Syst. Innov.*, vol. 4, no. 1, p. 9, Feb. 2021, doi: 10.3390/asi4010009.

[24] S. Jurnalis Pipin and H. Kurniawan, "Analisis Sentimen Kebijakan MBKM Berdasarkan Opini Masyarakat di Twitter Menggunakan LSTM," *J. SIFO Mikroskil*, vol. 23, no. 2, pp. 197–208, 2022, doi: 10.55601/jsm.v23i2.900.

[25] C. Wang and J. Qiao, "Construction Project Cost Prediction Method Based on Improved BiLSTM," *Appl. Sci.*, vol. 14, no. 3, 2024, doi: 10.3390/app14030978.

[26] P. L. Seabe, C. R. B. Moutsinga, and E. Pindza, "Forecasting Cryptocurrency Prices Using LSTM, GRU, and Bi-Directional LSTM: A Deep Learning Approach," *Fractal Fract.*, vol. 7, no. 2, 2023, doi: 10.3390/fractalfract7020203.

[27] F. García, F. Guijarro, J. Oliver, and R. Tamošiūnienė, "Foreign Exchange Forecasting Models: LSTM and BiLSTM Comparison," *Eng. Proc.*, vol. 68, no. 1, 2024, doi: 10.3390/engproc2024068019.

[28] M. Khan, H. Wang, A. Riaz, A. Elfatyany, and S. Karim, "Bidirectional LSTM - RNN - based hybrid deep learning frameworks for univariate time series classification," *J. Supercomput.*, no. 0123456789, 2021, doi: 10.1007/s11227-020-03560-z.

[29] M. Ghasemi, S. kadkhoda Mohammadi, M. Zare, S. Mirjalili, M. Gil, and R. Hemmati, "A new firefly algorithm with improved global exploration and convergence with application to engineering optimization," *Decis. Anal. J.*, vol. 5, no. September, p. 100125, 2022, doi: 10.1016/j.dajour.2022.100125.

[30] A. Abbaszadeh Shahri, M. Khorsand Zak, and H. Abbaszadeh Shahri, "A modified firefly algorithm applying on multi-objective radial-based function for blasting," *Neural Comput. Appl.*, vol. 34, no. 3, pp. 2455–2471, 2022, doi: 10.1007/s00521-021-06544-z.

[31] D. Jovanovic, M. Antonijevic, M. Stankovic, M. Zivkovic, M. Tanaskovic, and N. Bacanin, "Tuning Machine Learning Models Using a Group Search Firefly Algorithm for Credit Card Fraud Detection," *Mathematics*, vol. 10, no. 13, pp. 1–30, 2022, doi: 10.3390/math10132272.

[32] N. Bacanin, R. Stoean, M. Zivkovic, A. Petrovic, T. A. Rashid, and T. Bezdan, "Performance of a novel chaotic firefly algorithm with enhanced exploration for tackling global optimization problems: Application for dropout regularization," *Mathematics*, vol. 9, no. 21, 2021, doi: 10.3390/math9212705.

[33] S. Yu, X. Zuo, X. Fan, Z. Liu, and M. Pei, "An improved firefly algorithm based on personalized step strategy," *Computing*, vol. 103, no. 4, pp. 735–748, 2021, doi: 10.1007/s00607-021-00919-9.

[34] S. Bazi, R. Benzid, Y. Bazi, and M. M. Al Rahhal, "A fast firefly algorithm for function optimization: Application to the control of bldc motor," *Sensors*, vol. 21, no. 16, 2021, doi:

10.3390/s21165267.
[35] B. A. Hassan, "CSCF: a chaotic sine cosine firefly algorithm for practical application problems," *Neural Comput. Appl.*, vol. 33, no. 12, pp. 7011–7030, 2021, doi: 10.1007/s00521-020-05474-6.
[36] A. Liu, P. Li, X. Deng, and L. Ren, "A sigmoid attractiveness based improved firefly algorithm and its applications in IIR filter design," *Conn. Sci.*, vol. 33, no. 1, pp. 1–25, 2021, doi: 10.1080/09540091.2020.1742660.

2622-8130
Jurnal Sifo Mikroskil (JSM) - Volume 27, No 2, Oktober 2026
DOI: https://doi.org/10.55601/jsm.v27i2.2090