Implementasi Dekomposisi Matriks dalam Prediksi Tren Harga Bitcoin

Adinda Putri and 13523071[1,2]
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
[1]13523071@std.stei.itb.ac.id, [2]adindaputri0517@gmail.com

***Abstrak*—*Cryptocurrency*, khususnya Bitcoin, telah menjadi salah satu aset yang paling populer di pasar global saat ini. Mengingat fluktuasi harga yang tajam, prediksi tren harga Bitcoin menjadi sangat penting bagi investor dan analisis pasar. Makalah ini membahas penerapan *Singular Value Decomposition* (SVD) untuk mereduksi dimensi data dengan tujuan meningkatkan akurasi model dalam memprediksi tren harga Bitcoin. Kombinasi SVD dan *Long Short Term Memory* (LSTM) terbukti meningkatkan efisiensi pelatihan model, mengurangi risiko *overfitting*, dan menghasilkan prediksi harga Bitcoin yang lebih akurat dibanding metode biasa. Pendekatan ini menunjukkan potensi SVD dan LSTM dalam menganalisis data yang kompleks untuk memprediksi tren pasar kripto.**

***Kata Kunci*—*Cryptocurrency*, Bitcoin, *Singular Value Decomposition*, *Long Short-Term Memory*.**

I. PENDAHULUAN

Investasi *cryptocurrency* kini menjadi salah satu bidang yang semakin populer dalam dunia bisnis modern. Popularitas pasar digital dan kemajuan teknologi baru telah meningkatkan pertumbuhan permintaan terhadap aset kripto. Nilai investasi di pasar *cryptocurrency* global terus meningkat setiap tahun. Pada tahun 2020, nilai pasar *cryptocurrency* mencapai 827 juta USD, dengan proyeksi tingkat pertumbuhan tahunan sebesar 11,1%, yang diperkirakan akan mencapai 1,9 miliar USD pada tahun 2028. [2]

Bitcoin menjadi *cryptocurrency* yang paling terkenal dan bernilai hingga saat ini. [2] Diperkenalkan pada tahun 2008 oleh sosok anonim bernama Satoshi Nakamoto, Bitcoin dirancang sebagai alat transaksi digital yang tidak membutuhkan perantara. Tidak seperti mata uang konvensional yang didukung oleh pemerintah, Bitcoin adalah mata uang kripto terdesentralisasi yang tidak bergantung pada otoritas apa pun. Hal ini membuat harga Bitcoin sangat fluktuatif, sehingga menarik perhatian dan harapan para investor untuk meraih keuntungan besar dari pergerakan harganya. [1]

Meskipun banyak jenis mata uang kripto lainnya, Bitcoin tetap menjadi yang paling populer, dengan kapitalisasi pasar yang mencapai lebih dari 40% dari total kapitalisasi pasar mata uang kripto saat ini. [1] Dengan popularitas tersebut, Bitcoin menjadi fokus utama dalam dunia investasi dan perdagangan aset digital. Maka dari itu, kemampuan memprediksi tren harga Bitcoin sangat penting bagi investor untuk membuat keputusan yang tepat.

Makalah ini menghadirkan pendekatan inovatif untuk memprediksi tren harga Bitcoin dengan menggunakan algoritma *Long Short Term Memory* (LSTM) dan dekomposisi matriks *Singular Value Decomposition* (SVD). Pendekatan ini bertujuan untuk meningkatkan akurasi prediksi serta mengatasi kompleksitas *dataset* pengujian harga Bitcoin.

II. LANDASAN TEORI

A. Singular Value Decomposition (SVD)

Dekomposisi matriks berarti memfaktorkan matriks, misalnya A, menjadi hasil kali dari sejumlah matriks lain, $P_1, P_2, \ldots, P_k$.

$$A = P_1 \times P_2 \times \ldots \times P_k \quad (1)$$

Dekomposisi matriks dapat dilakukan metode dekomposisi LU, dekomposisi QR, dan dekomposisi nilai singular (SVD). [3] *Singular Value Decomposition* (SVD) merupakan dekomposisi matriks menjadi tiga buah matriks. Matriks persegi A berukuruan n x n dapat difaktorkan menjadi,

$$A = PDP^{-1}, \quad (2)$$

dengan P adalah matirks yang kolom-kolomnya adalah basis ruang eigen dari matriks A,

$$P = (p_1 \mid p_2 \mid \ldots \mid p_n), \quad (3)$$

D adalah matriks diagonal sedemikian sehingga,

$$D = P^{-1}AP \quad (4)$$

Untuk matriks non persegi berukuran m x n dan tidak memiliki nilai eigen, pemfaktoran dilakukan dengan metode *singular value decomposition* (SVD). Metode ini memfaktorkan matriks A berukuran m x n menjadi matriks U, ∑, dan V sedemikian sehingga,

$$A = U\sum V^T \quad (5)$$

Dengan:

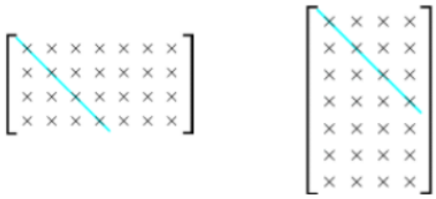
- U adalah matriks ortogonal m x m
- V adalah matriks ortogonal n x n
- ∑ adalah matriks berukuran m x n yang elemen-elemen diagonal utamanya adalah nilai-nilai singular dari matriks A sedangkan elemen lainnya 0 [3].

Matriks ortogonal adalah matriks yang kolom-kolomnya adalah vektor yang saling ortogonal satu sama lain atau hasil kali titik sama dengan 0. Jika vektor-vektor kolom tersebut adalah vektor satuan, maka matriks tersebut dinamakan juga matriks ortonormal. Jika Q merupakan matriks ortogonal berukuran n x n maka,

$$Q^TQ = QQ^T = I \quad (6)$$

dengan I merupakan matriks identitas berukuran n x n.

Diagonal utama pada matriks nonpersegi berukuran m x n, didefinisikan pada garis yang dimulai dari sudut kiri atas terus ke bawah matriks sejauh mungkin.



Gambar 1. Diagonal utama matriks nonpersegi
Sumber:https://informatika.stei.itb.ac.id/~rinaldi.munir/AljabarGeometri/2022-2023/Algeo-20-Singular-value-decomposition-Bagian1-2022.pdf

Misalkan A adalah matriks m x n. Jika $\lambda_1, \lambda_2, ..., \lambda_n$ merupakan nilai-nilai eigen dari A^TA, maka

$$\sigma_1 = \sqrt{\lambda_1}, \sigma_2 = \sqrt{\lambda_2}, ..., \sigma_n = \sqrt{\lambda_n} \quad (7)$$

disebut nilai-nilai singular dari matriks A. Diasumsikan bahwa $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n \geq 0$ sehingga $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_n \geq 0$.

Jika A adalah matriks m x n dengan *rank* k, maka A dapat difaktorkan menjadi

$$A = U\sum V^T =$$

$$[\boldsymbol{u}_1 \quad \boldsymbol{u}_2 \quad \ldots \quad \boldsymbol{u}_k | \boldsymbol{u}_{k+1} \quad \ldots \quad \boldsymbol{u}_m] \left[\begin{array}{cccc|c} \sigma_1 & 0 & \cdots & 0 & \\ 0 & \sigma_2 & \ddots & 0 & \sigma_{kx(n-k)} \\ \vdots & \vdots & \vdots & \vdots & \\ 0 & 0 & \cdots & \sigma_k & \\ \hline & 0_{(m-k)xk} & & & \sigma_{(m-k)x(n-k)} \end{array}\right] \left[\begin{array}{c} v_1^T \\ v_2^T \\ \vdots \\ v_k^T \\ \hline v_{k+1}^T \\ \vdots \\ v_n^T \end{array}\right]$$

(7)

Dengan:

- U adalah matriks m x m
- ∑ adalah matriks m x n, dan V adalah matriks n x n.
- $u_1, u_2 \ldots, u_k$disebut vektor-vektor singular kiri matriks A.
- $v_1, v_2 \ldots, v_k$disebut vektor-vektor singular kanan dari matriks A.
- $\sigma_1 = \sqrt{\lambda_1}, \sigma_2 = \sqrt{\lambda_2}, \ldots, \sigma_k = \sqrt{\lambda_k}$ adalah nilai-nilai singular dari matriks A, dan $\lambda_1, \lambda_2, \ldots, \lambda_k$ adalah nilai-nilai eigen dari A^TA.
- $V = [v_1, v_2 \ldots, v_n]$ secara ortogonal mendiagonalisasi A^TA.
- Vektor-vektor kolom di V diurutkan sedemikian sehingga $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_k > 0$.
- $u_i = \frac{Av_i}{\|Av_i\|} = \frac{1}{\sigma_i} Av_i \,.\, i = 1, 2, \ldots, k$
- $\{u_1, u_2 \ldots, u_k\}$ adalah basis ortonormal untuk col(A).
- $\{u_1, u_2 \ldots, u_k, u_{k+1}, \ldots, u_m\}$ adalah oerluassasn dari $\{u_1, u_2 \ldots, u_k\}$ untuk membentuk basis ortonormal ruang vektor R^m.

B. Bitcoin

Bitcoin merupakan mata uang digital terdesentralisasi pertama di dunia. Bitcoin diciptakan pada 2008 oleh orang atau kelompok dengan nama samaran Satoshi Nakamoto. Mata uang digital terdesentralisasi tersebut disebut juga *criptocurrency* atau mata uang kripto. Investasi kripto menjadi cakupan yang relevan di era bisnis modern. Jumlah uang yang diinvestasikan di kripto global meningkat setiap tahun. Sebagai contoh, saat 2020 pasar kripto bernilai 827 juta USD, dengan proyeksi tingkat kenaikan gabungan (CAGR) sebesar 11.1%, dan diperkirakan akan mencapai 1903 juta USD pada akhir tahun 2028.

Pada awalnya, kripto pertama (Bitcoin) dibuat sebagai *peer-to-peer payment system*, dan dengan begitu dikategorikan sebagai mata uang. Akan tetapi, karena volatilitasnya yang tinggi, beberapa peneliti percaya bahwa Bitcoin tidak sepenuhnya berperilaku seperti mata uang. Hal ini menuju ke kesimpulan bahwa terdapat beberapa faktor tersembunyi yang memengaruhi perubahan harganya [1].

Dalam dunia analisis Bitcoin, terdapat beberapa istilah kunci yang digunakan untuk memahami pergerakan harga dan aktivitas pasar, yaitu:

1. *Close*
   Close merupakan harga penutupan Bitcoin pada akhir periode tertentu, biasanya di akhir hari perdagangan. Harga ini digunakan untuk mengevaluasi kinerja harian Bitcoin.
2. *Open*
   Open merupakan harga pembukaan Bitcoin pada awal periode tertentu. Harga ini memberikan gambaran awal posisi harga sebelum pasar bergerak.
3. *Market Capitalization*
   Market capitalization adalah total nilai pasar Bitcoin yang dihitung dengan mengalikan harga saat ini dengan jumlah Bitcoin yang beredar. *Market capitalization* menunjukkan skala pasar dan popularitas Bitcoin.
4. *Volume*
   Volume yang dimaksud adalah jumlah total Bitcoin yang diperdagangkan dalam suatu periode, yang mencerminkan tingkat aktivitas pasar. *Volume* tinggi menunjukkan minat yang besar.

C. Long Short-Term Memory

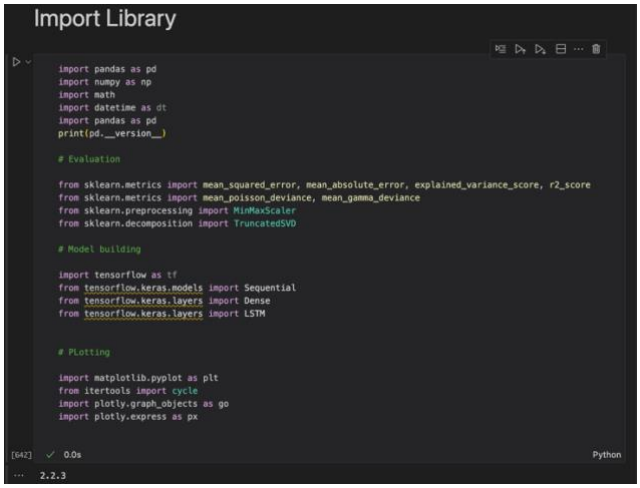
Terdapat berbagai cara untuk menganalisis pola pada data finansial. Salah satunya adalah melalui pendekatan prediksi *multivariate time series*. Metode ini mengkaji perubahan dari setiap variabel terhadap waktu beserta hubungannya dengan variabel lain yang diprediksi. Cara yang paling efisien untuk mengatasi data *time series* seperti itu adalah dengan *recurrent neural network* (RNN) [1]. Salah satu contoh dari RNN yang dapat digunakan untuk memprediksi harga Bitcoin adalah *long short-term memory* (LSTM) [2]. Cara ini melibatkan input terdahulu dalam memerhitungkan prediksi [1].

Long short-term memory merupakan versi perbaikan dari RNN. *Recurrent neural network* memiliki sebuah keadaan tersembunyi yang membuat sebuah *network* kesulitan dalam memperlajari hubungan jangka panjang. Model LSTM mengatasi hal tersebut dengan memerkenalkan *memory cell*, yang merupakan penyimpan informasi untuk periode yang cukup lama [4].

Pada data *time series* dengan volatilitas tinggi seperti harga Bitcoin, sulit untuk menghitung fluktuasi yang cepat dan menentukan korelasi yang tepat antarinput. Salah satu cara yang dapat dilakukan untuk mengurangi kompleksitas tersebut adalah dekomposisi. Dekomposisi berperan untuk menyederhanakan data *series* [3]. Pada makalah ini, dibahas penggunaan dekomposisi *singular value decomposition* (SVD) pada penyederhanaan data *series* dalam bentuk matriks. Lalu, dibahas perbandingan performansi LSTM dengan dan tanpa SVD.

III. Implementasi Program

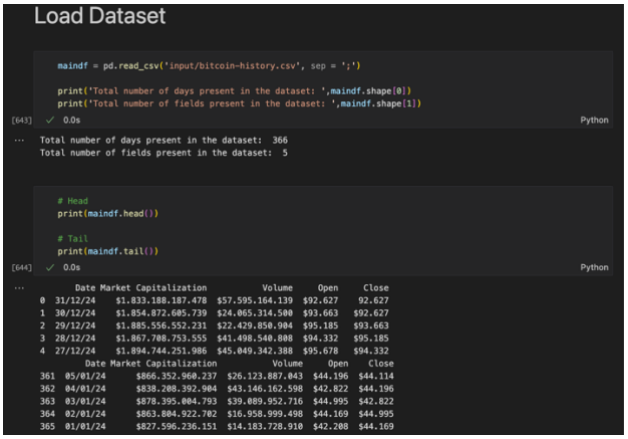
Untuk memprediksi harga Bitcoin yang fluktuatif, penulis membangun sebuah program Python yang menggunakan kombinasi algoritma *Long Short-Term Memory* (LSTM) dan *Singular Value Decomposition* (SVD). Penggunaan kombinasi ini bertujuan untuk mengurangi *noise* dan kompleksitas data. SVD digunakan untuk menyederhanakan dimensi data, sedangkan LSTM digunakan untuk melatih mesin dalam mengenali pola-pola temporal. Proses implementasi dilakukan secara terstruktur dengan bantuan berbagai *library* pendukung, dimulai dari tahap pengumpulan dan pengolahan data hingga pelatihan serta evaluasi model untuk menghasilkan prediksi yang lebih akurat.



Gambar 2. Penggunaan *library* dalam kode
Sumber: Penulis

A. Pengumpulan Data

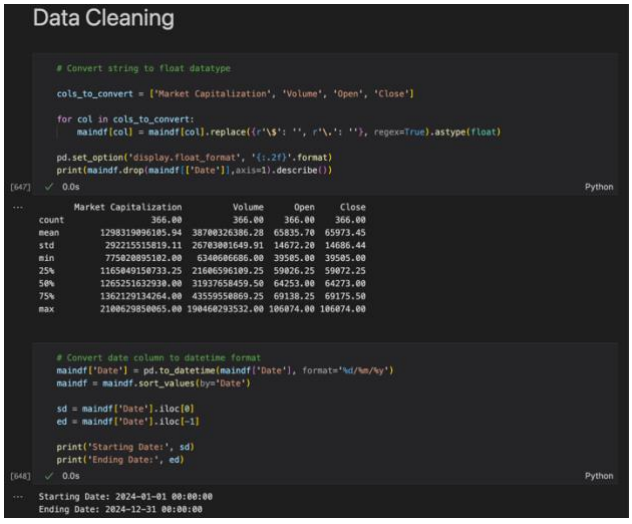
Langkah pertama dari implementasi program adalah mengumpulkan data harga Bitcoin dari CoinGecko. Data ini mencakup berbagai kolom yang relevan dengan prediksi tren harga Bitcoin, seperti tanggal (*Date*), harga pembukaan (*Open*), harga penutupan (*Close*), volume perdagangan (*Volume*), dan kapitalisasi pasar (*Market Capitalization*). Dalam penelitian ini, data yang digunakan mencakup rentang waktu 1 tahun terakhir (2024).



Gambar 3. Kode untuk memuat *dataset* dengan format CSV
Sumber: Penulis

B. Prapemrosesan Data

Prapemrosesan data dilakukan untuk membersihkan dan mempersiapkan data sebelum langkah analisis. Langkah pertama adalah pembersihan data, yang bertujuan untuk memastikan tidak ada data yang hilang atau tidak valid. Kolom yang mengandung simbol seperti $ atau tanda titik disesuaikan dengan mengubah nilainya menjadi format angka desimal (*float*) agar lebih mudah dianalisis. Selanjutnya, dilakukan perubahan kolom *Date* menjadi format *datetime* untuk memastikan kompatibilitasnya dengan analisis berbasis waktu.



Gambar 4. Kode untuk prapemrosesan data
Sumber: Penulis

C. Exploratory Data Analysis (EDA)

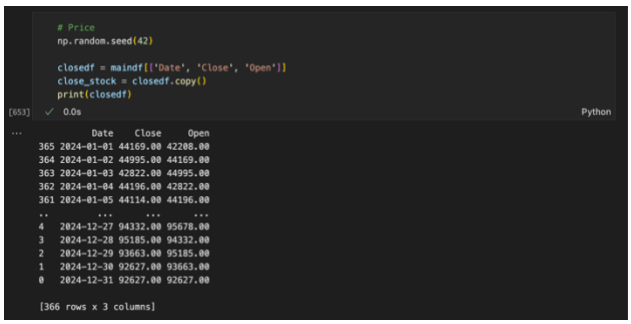
Tahap *Exploratory Data Analysis* (EDA) dilakukan untuk memberikan gambaran awal kepada pengguna mengenai pola dalam data. Analisis ini mencakup perbandingan harga pembukaan dan penutupan berdasarkan bulan, yang bertujuan untuk mengidentifikasi

pola yang mungkin terjadi. Selain itu, EDA juga menggunakan visualisasi data dalam bentuk grafik untuk membantu pengguna memahami tren dan fluktuasi harga Bitcoin.

Gambar 5. Kode untuk memvisualisasikan data dalam EDA
Sumber: Penulis

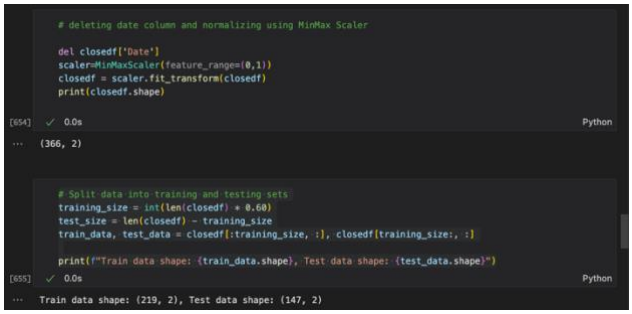
D. Pemodelan Data dan Pelatihan Model

Pada tahap ini, model prediksi dirancang untuk mengolah data secara efisien dan menghasilkan prediksi harga Bitcoin yang akurat. Pemodelan data dilakukan menggunakan kolom *Open* dan *Close* karena kedua kolom ini mencerminkan pergerakan harga harian Bitcoin, yang menjadi fokus utama dalam proses prediksi. Dengan menggunakan kolom-kolom tersebut, model dapat lebih fokus mengenali pola penting tanpa terpengaruh informasi yang kurang relevan.



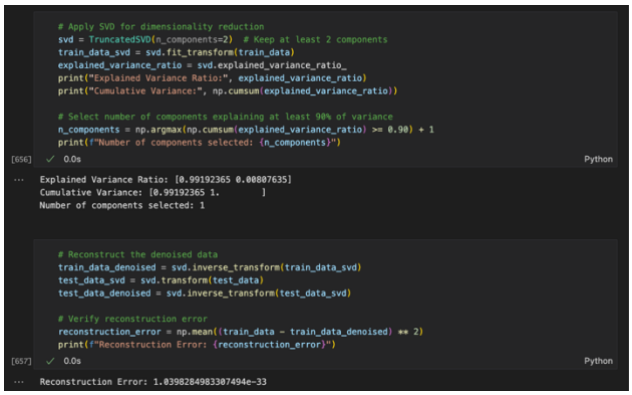
Gambar 6. Kode untuk memilah data dengan kolom *Date*, *Close* dan *Open*.
Sumber: Penulis

Selanjutnya, data dinormalisasi menggunakan MinMaxScaler agar semua nilai berada dalam rentang [0, 1]. Proses normalisasi dilakukan untuk meningkatkan stabilitas dan efisiensi selama proses pelatihan model. Setelah itu, *dataset* dibagi menjadi dua bagian, yaitu 60% untuk pelatihan dan 40% untuk pengujian, dengan tujuan memastikan bahwa model dapat belajar dari sebagian besar data yang tersedia.



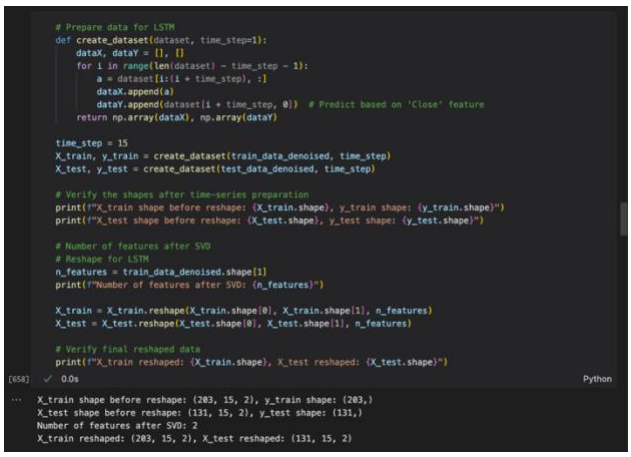
Gambar 7. Kode untuk menormalisasi data dengan kolom *Close* dan *Open*, serta membaginya untuk pelatihan dan pengujian.
Sumber: Penulis

Untuk mengatasi *noise* dan kompleksitas data, diterapkan metode *Singular Value Decomposition* (SVD). SVD mereduksi dimensi *dataset* dengan memilih komponen utama yang menjelaskan lebih dari 90% variansi dalam data. Setelah proses reduksi, data direkonstruksi kembali untuk menghilangkan *noise* dan menghasilkan *dataset* yang lebih bersih dan mudah diolah. Dengan cara ini, data yang digunakan menjadi lebih stabil dan berkualitas tinggi untuk proses pelatihan.



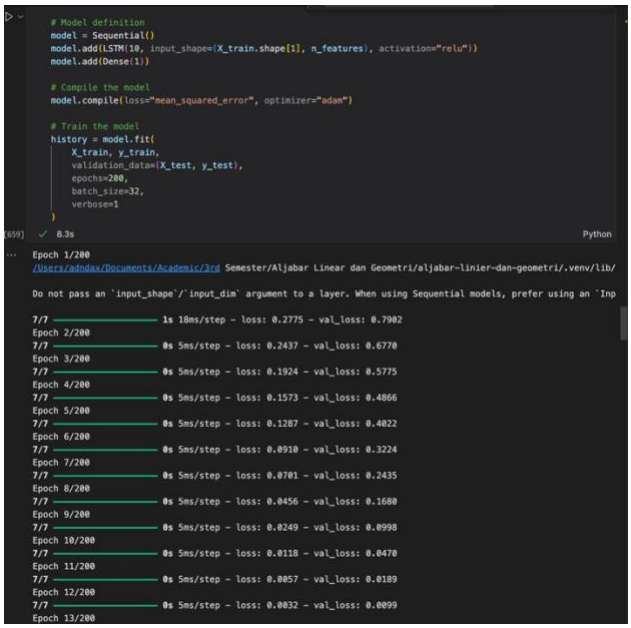
Gambar 8. Kode untuk mereduksi *dataset* dengan *Singular Value Decomposition* (SVD)
Sumber: Penulis

Dataset hasil rekonstruksi ini kemudian diubah menjadi format *time-series*. Format ini dirancang untuk memanfaatkan data harga sebelumnya sebagai *input*, dan memprediksi harga penutupan (*Close*) pada langkah waktu berikutnya sebagai *output*. Pendekatan *time-series* ini memungkinkan model untuk memahami pola yang ada dalam data historis Bitcoin.

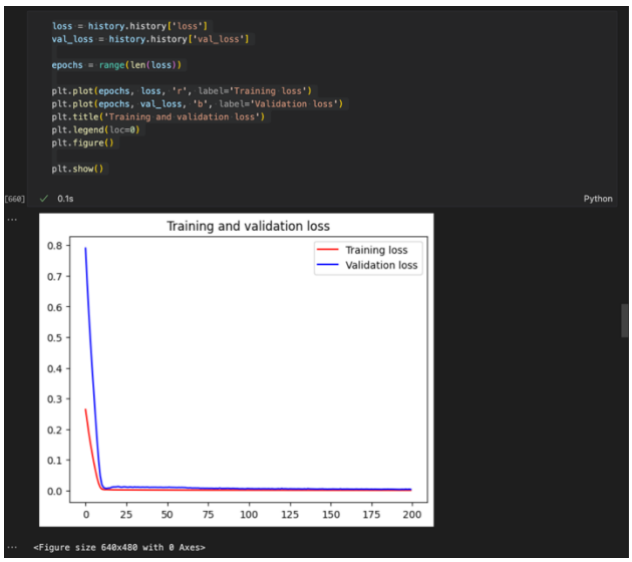


Gambar 9. Kode untuk merubah *dataset* dalam format *time-series*
Sumber: Penulis

Langkah terakhir adalah membangun model prediksi menggunakan algoritma *Long Short-Term Memory* (LSTM). Data dalam format *batch_size*, *time_steps*, *features* dimasukkan ke dalam model. Model ini terdiri dari lapisan LSTM untuk mempelajari pola temporal dan lapisan Dense untuk menghasilkan prediksi harga. Model dilatih menggunakan *train data* dengan validasi pada *test data*. *Optimizer* Adam digunakan untuk mempercepat pelatihan, sementara fungsi *loss Mean Squared Error* (MSE) digunakan untuk meminimalisasi kesalahan prediksi.



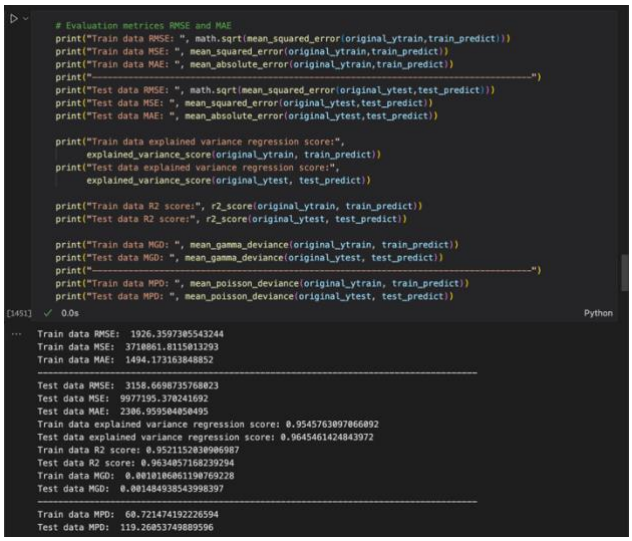
Gambar 10. Kode pemodelan data dan pelatihan model LSTM
Sumber: Penulis



Gambar 11. Kode visualisasi performa pelatihan model LSTM
Sumber: Penulis

D. Evaluasi Model

Untuk memastikan model mampu memberikan prediksi yang akurat, dilakukan evaluasi menggunakan beberapa indikator. Indikator seperti *Mean Absolute Error* (MAE), *Root Mean Squared Error* (RMSE), dan *Mean Absolute Percentage Error* (MAPE) digunakan untuk menilai sejauh mana prediksi model mendekati nilai aktual. Selain itu, selama proses pelatihan, grafik training loss dan validation loss juga dibuat untuk memantau kinerja model. Hal tersebut bertujuan untuk memastikan model tidak *overfitting* maupun *underfitting*, sehingga menghasilkan prediksi yang akurat.



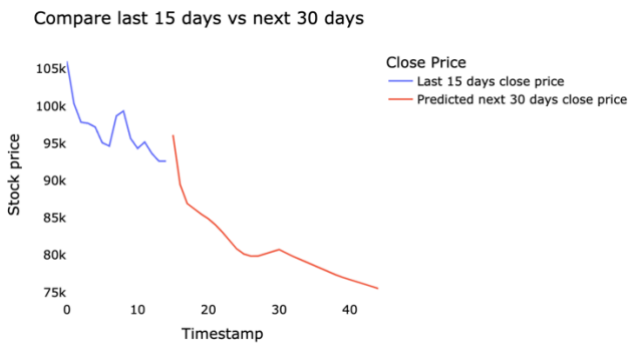
Gambar 12. Kode untuk mengevaluasi model LSTM
Sumber: Penulis

E. Prediksi Model

Pada tahap ini, model digunakan untuk memprediksi harga Bitcoin di masa depan, misalnya selama 30 hari ke depan. Untuk prediksi *multistep*, diterapkan pendekatan *rolling window*, yaitu pendekatan yang menggunakan hasil prediksi pada langkah sebelumnya sebagai *input* untuk langkah berikutnya. Selain itu, hasil prediksi divisualisasikan dalam bentuk grafik dengan membandingkan harga penutupan (*Close)* aktual dengan prediksi untuk data pelatihan dan pengujian. Grafik ini juga digunakan untuk menampilkan prediksi harga di masa depan secara visual. Pendekatan ini menunjukkan kemampuan model untuk menangkap pola dan tren harga Bitcoin dengan baik, sekaligus menyediakan informasi yang berguna sebagai referensi dalam pengambilan keputusan investasi.

```
# Correctly initialize temp_mat with the combined length
temp_mat_length = time_step + pred_days
temp_mat = np.empty(temp_mat_length)
temp_mat[:] = np.nan

# Copy temp_mat for original and predicted values
last_original_days_value = temp_mat.copy()
next_predicted_days_value = temp_mat.copy()

# Prepare data for last original days
placeholder = np.zeros((time_step, closedf.shape[1]))
placeholder[:, 0] = closedf[len(closedf) - time_step:, 0]
last_original_days_value[:time_step] = scaler.inverse_transform(placeholder)[:, 0]

# Prepare data for next predicted days
placeholder_pred = np.zeros((pred_days, closedf.shape[1]))
placeholder_pred[:, 0] = np.array(lst_output).reshape(-1)
next_predicted_days_value[time_step:] = scaler.inverse_transform(placeholder_pred)[:, 0]

# Ensure lengths match
assert len(last_original_days_value) == len(next_predicted_days_value), "Lengths do not match!"

# Create DataFrame
new_pred_plot = pd.DataFrame({
    'last_original_days_value': last_original_days_value,
    'next_predicted_days_value': next_predicted_days_value
})

# Prepare the plot
names = cycle(['Last 15 days close price', 'Predicted next 30 days close price'])
fig = px.line(
    new_pred_plot,
    x=new_pred_plot.index,
    y=['last_original_days_value', 'next_predicted_days_value'],
    labels={'value': 'Stock price', 'index': 'Timestamp'}
)
fig.update_layout(
    title_text='Compare last 15 days vs next 30 days',
    plot_bgcolor='white',
    font_size=15,
    font_color='black',
    legend_title_text='Close Price'
)
fig.for_each_trace(lambda t: t.update(name=next(names)))
fig.update_xaxes(showgrid=False)
fig.update_yaxes(showgrid=False)
fig.show()
```

Gambar 13. Kode untuk visualisasi prediksi model LSTM
Sumber: Penulis



Gambar 14. Visualisasi prediksi model LSTM
Sumber: Penulis

IV. ANALISIS DAN PEMBAHASAN

```
Train data RMSE:  1957.2262059518837
Train data MSE:  3830734.421264806
Train data MAE:  1517.7100729983251
-------------------------------------------------------------------------
Test data RMSE:  4282.623691093927
Test data MSE:  18340865.679518968
Test data MAE:  3173.418368240122
Train data explained variance regression score: 0.9507642616586462
Test data explained variance regression score: 0.9498942277201502
Train data R2 score: 0.9505683722290047
Test data R2 score: 0.9327295088986183
Train data MGD:  0.0010330499321472449
Test data MGD:  0.002568691918711917 4
-------------------------------------------------------------------------
Train data MPD:  62.41133577847785
Test data MPD:  214.72525055113493
```

Gambar 15. Hasil evaluasi model LSTM tanpa SVD
Sumber: Penulis

```
Train data RMSE:  1926.3597305543244
Train data MSE:  3710861.8115013293
Train data MAE:  1494.173163848852
-------------------------------------------------------------------------
Test data RMSE:  3158.6698735768023
Test data MSE:  9977195.370241692
Test data MAE:  2306.959504050495
Train data explained variance regression score: 0.9545763097066092
Test data explained variance regression score: 0.9645461424843972
Train data R2 score: 0.9521152030906987
Test data R2 score: 0.9634057168239294
Train data MGD:  0.0010106061190769228
Test data MGD:  0.001484938543998397
-------------------------------------------------------------------------
Train data MPD:  60.721474192226594
Test data MPD:  119.26053749889596
```

Gambar 16. Hasil evaluasi model LSTM dengan SVD
Sumber: Penulis

Berdasarkan hasil pengujian yang ditampilkan pada Gambar 15 (kode pertama) dan Gambar 16 (kode kedua), terlihat bahwa model pada Gambar 16, yang menggunakan teknik *Singular Value Decomposition* (SVD), menunjukkan kinerja yang lebih baik dibandingkan model pada Gambar 15, yang tidak menggunakan SVD. Pada data pelatihan, model pada Gambar 16 memiliki nilai RMSE, MSE, dan MAE yang lebih rendah, Hal tersebut menunjukkan bahwa model ini lebih efektif dalam mempelajari pola dari data historis Bitcoin. Selain itu, nilai *explained variance regression score* dan R2 *score* yang lebih tinggi menunjukkan bahwa model pada gambar 16 lebih mampu menjelaskan variasi dalam data pelatihan. Stabilitas model pada Gambar 16 juga terlihat dari nilai mean *gamma deviance* (MGD) dan *mean poisson deviance* (MPD) yang lebih rendah dibandingkan model pada Gambar 15.

Keunggulan model pada Gambar 16, yang memanfaatkan SVD membuktikan efektivitas teknik dekomposisi matriks dalam meningkatkan kualitas prediksi. Dengan mereduksi dimensi data dan menghilangkan *noise*, SVD menghasilkan data yang lebih relevan untuk pelatihan model. Hasil ini menunjukkan bahwa SVD tidak hanya meningkatkan efisiensi pelatihan, tetapi juga memberikan prediksi yang lebih akurat. Oleh karena itu, model dengan SVD menjadi pendekatan yang lebih unggul untuk menganalisis data kompleks seperti harga Bitcoin.

V. KESIMPULAN

Makalah ini menunjukkan bagaimana kombinasi antara dekomposisi matriks, *Singular Value Decomposition* (SVD), dan *Long Short-Term Memory* (LSTM) dapat diterapkan untuk memprediksi tren harga Bitcoin dengan lebih akurat. Dengan SVD, dimensi data yang kompleks berhasil direduksi, sehingga menghilangkan *noise* dan mempertahankan informasi yang paling relevan. Data hasil reduksi ini kemudian diolah dengan algoritma LSTM, yang memanfaatkan pola temporal dalam data untuk menghasilkan prediksi yang akurat.

Hasil penelitian dalam makalah ini menunjukkan bahwa penggunaan dekomposisi matriks melalui SVD secara signifikan meningkatkan efisiensi pelatihan model, mengurangi risiko *overfitting*, dan menghasilkan prediksi yang lebih presisi dibandingkan metode biasa. Pendekatan ini dapat diterapkan pada aset finansial lainnya atau diperluas dengan menambahkan data eksternal, seperti sentimen pasar atau indikator ekonomi global untuk meningkatkan akurasi lebih lanjut.

VI. UCAPAN TERIMA KASIH

Sebagai penulis makalah ini, saya ingin mengucapkan terima kasih kepada Tuhan Yang Maha Esa atas rahmat dan karunia-Nya sehingga makalah yang berjudul *"Implementasi Dekomposisi Matriks dalam Prediksi Tren Harga Bitcoin"* dapat diselesaikan dengan baik. Saya juga ingin menyampaikan rasa terima kasih kepada berbagai pihak yang telah mendukung dan membantu selama proses penulisan makalah ini, khususnya kepada:

1. Bapak Dr. Rila Mandala, Bapak Dr. Rinaldi Munir, serta Bapak Dr. Judhi Santoso, M.T., selaku dosen pengajar mata kuliah IF2123 Aljabar Linier dan Geometri, atas bimbingan, pengajaran, dan ilmu yang telah diberikan selama perkuliahan dan menjadi landasan penting dalam penyusunan makalah ini.
2. Kedua orang tua saya yang selalu memberikan doa restu dan dukungan moral sehingga saya dapat menyelesaikan makalah ini dengan baik.
3. Pacar saya, Fajar Asyraf R., serta teman-teman saya, Rana dan Heleni, yang telah memberikan dorongan, motivasi, serta masukan yang mendukung kelancaran penyusunan makalah ini.
4. Para penulis jurnal, artikel, dan referensi lainnya yang menjadi sumber inspirasi dan memberikan kontribusi penting melalui wawasan yang relevan untuk penulisan makalah ini.

Saya menyadari bahwa makalah ini tidak akan terwujud tanpa kontribusi dan dukungan dari seluruh pihak yang telah disebutkan di atas. Untuk itu, saya mengucapkan terima kasih yang sebesar-besarnya atas segala bantuan, inspirasi, dan ilmu yang telah diberikan. Saya berharap makalah ini dapat memberikan manfaat bagi para pembaca serta mendukung pengembangan ilmu pengetahuan, khususnya di bidang teknologi.

REFERENSI

[1] E. Koo dan G. Kim, "Centralized decomposition approach in LSTM for Bitcoin price prediction," *Expert Systems with Applications*, vol. 237, art. no. 121401, 2024. [Daring]. Tersedia di:https://www.sciencedirect.com/science/article/pii/S0957417423019036?fr=RR-2&ref=pdf_download&rr=8f922bf0fba887b7. [Diakses: 28 Desember 2024].

[2] V. Mizdrakovic, M. Kljajic, M. Zivkovic, N. Bacanin, L. Jovanovic, M. Deveci, dan W. Pedrycz, "Forecasting bitcoin: Decomposition aided long short-term memory based time series modeling and its explanation with Shapley values," *Knowledge-Based Systems*, vol. 299, art. no. 112026, 2024. [Daring]. Tersedia di:https://www.sciencedirect.com/science/article/pii/S0950705124006609?ref=pdf_download&fr=RR-2&rr=8f928f8c1a156038. [Diakses: 28 Desember 2024].

[3] R. Munir, "Singular Value Decomposition (SVD), Bagian 1," *Bahan Kuliah IF2123 Aljabar Linier dan Geometri*, Institut Teknologi Bandung, 2022. [Daring]. Tersedia: https://informatika.stei.itb.ac.id/~rinaldi.munir/AljabarGeometri/2022-2023/Algeo-20-Singular-value-decomposition-Bagian1-2022.pdf. [Diakses: 29 Desember 2024].

[4] "What is LSTM – Long Short Term Memory," GeeksforGeeks. [Daring]. Tersedia: https://www.geeksforgeeks.org/deep-learning-introduction-to-long-short-term-memory/. [Diakses: 29 Desember 2024].

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 30 Desember 2024

Adinda Putri
13523071