

SKRIPSI

**PREDIKSI HARGA BERAS MENGGUNAKAN
*LONG SHORT-TERM MEMORY (LSTM)***

**Disusun Oleh :
Muhamad Yuda
065119014**



**PROGRAM STUDI ILMU KOMPUTER
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS PAKUAN
BOGOR
2024**

HALAMAN PENGESAHAN

Judul : Prediksi Harga Beras Menggunakan *Long Short-Term Memory (LSTM)*
Nama : Muhamad Yuda
Npm : 065119014

Mengesahkan,

Pembimbing Pendamping
Program Studi Ilmu Komputer
FMIPA – UNPAK



Dr. Fajar Delli, M.Kom.

Pembimbing Utama
Program Studi Ilmu Komputer
FMIPA - UNPAK



Dr. Herfina., M.Pd., M.Kom.

Mengetahui,

Ketua Program Studi Ilmu Komputer
FMIPA – UNPAK



Arie Qur'ania, M.Kom.

Dekan
FMIPA -UNPAK



The official stamp of the Faculty of Mathematics and Computer Science (FMIPA) at Unswatani University (UNPAK) is visible. It features a circular emblem with a book and a lamp, surrounded by the text 'FACULTAS MATEMATIKA DAN ILMU PENGETIKAN A' and 'UNIVERSITAS PAUJAN'. A handwritten signature in black ink is written over the stamp.

Asep Denih, S.Kom., M.Sc., Ph.D.

PERNYATAAN KEASLIAN KARYA TULIS SKRIPSI

Dengan ini, saya yang bertanda tangan dibawah ini:

Nama : Muhamad Yuda
NPM : 065119014
Program Studi : Ilmu Komputer
Fakultas : Matematika dan Ilmu Pengetahuan Alam Universitas Pakuan
Bogor

Menyatakan bahwa sejauh yang saya ketahui, karya tulis ini bukan merupakan karya tulis yang pernah dipublikasikan atau sudah pernah dipakai untuk mendapatkan gelar sarjana di Universitas lain, kecuali pada bagian-bagian dimana sumber informasinya dicantumkan dengan cara referensi yang semestinya.

Demikian pernyataan ini saya buat dengan sebenar-benarnya. Apabila kelak ditemukan hari terdapat gugatan, penulis bersedia dikenakan sanksi sesuai dengan peraturan yang berlaku.

Bogor, 10 Juli 2024



Muhamad Yuda

PERNYATAAN PELIMPAHAN SKRIPSI DAN SUMBER INFORMASI SERTA PELIMPAHAN HAK CIPTA

Dengan ini, saya yang bertanda tangan dibawah ini :

Nama : Muhamad Yuda
NPM : 065119014
Judul Skripsi : Prediksi Harga Beras Menggunakan *Long Short-Term Memory (LSTM)*

Dengan ini saya menyatakan bahwa Paten dan Hak Cipta dari produk Skripsi dan Tugas Akhir di atas adalah benar karya saya dengan arahan dari komisi pembimbing dan belum diajukan dalam bentuk apapun kepada perguruan tinggi manapun.

Sumber informasi yang berasal atau dikutip dari karya yang diterbitkan maupun tidak diterbitkan dari penulis lain telah disebutkan dalam teks dan dicantumkan dalam Daftar Pustaka di bagian akhir skripsi ini.

Dengan ini saya melimpahkan Paten, hak cipta dari karya tulis saya kepada Universitas Pakuan.

Bogor, 10 Juli 2024



Muhamad Yuda

RIWAYAT HIDUP



Muhamad Yuda, Dilahirkan di Cianjur pada tanggal 18 Juni 2001 dari pasangan Bapak Misbah SW Bin Mamad dan Ibu Yanti sebagai anak pertama dari tiga bersaudara.

Penulis memulai Pendidikan sekolah dasar pada tahun 2007 di SD Negeri Pandansari 02 Ciawi dan lulus pada tahun 2013. Di tahun yang sama penulis melanjutkan Pendidikan ke sekolah menengah pertama di SMP Addina Ciawi dan lulus pada tahun 2016, lalu pada tahun yang sama penulis melanjutkan Pendidikan ke sekolah menengah kejuruan di SMK Amaliah 1 Ciawi Bogor dengan jurusan Teknik Komputer Dan Jaringan, dan lulus pada tahun 2019. Di tahun yang sama penulis melanjutkan Pendidikan ke Universitas Pakuan Bogor dengan Program Studi Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam.

Selama menjadi mahasiswa di Universitas Pakuan Bogor, pada bulan Agustus 2022 penulis melaksanakan Praktik Lapang di DPRD Kabupaten Bogor. Pada bulan April 2023 penulis memulai penelitian tugas akhir dan selesai pada bulan Juli 2024 dengan judul “Prediksi Harga Beras Menggunakan *Long Short-Term Memory (LSTM)*” untuk menuntaskan jenjang perkuliahan dan mendapatkan gelar Sarjana Ilmu Komputer.

Penulis aktif pada organisasi eksternal kampus yang tergabung dalam Pergerakan Mahasiswa Islam Indonesia (PMII) Universitas Pakuan, Selain itu penulis juga tergabung dalam organisasi Gerakan Mahasiswa Advokasi Masyarakat (GERMAT), Serta tergabung dan ikut serta menjadi pelopor berdirinya organisasi ke-daerahan tingkat Kecamatan Ciawi yaitu Asosiasi Mahasiswa Ciawi, Selain aktif dalam berorganisasi, penulis juga membuat beberapa website diantaranya kotak karya yang merupakan sebuah wadah untuk pecinta literasi, Dan penulis juga membuat website sobatscamp untuk para pecinta alam.

RINGKASAN

Muhamad Yuda, 2024. Prediksi Harga Beras Menggunakan *Long Short-Term Memory* (LSTM). Dibawah bimbingan Ibu **Dr. Herfina., M.Pd., M.Kom.** dan **Dr. Fajar Delli, M.Kom.**

Penelitian ini dilatar belakangi oleh beras yang merupakan makanan pokok bagi sebagian besar populasi dunia, terutama di Asia. Stabilitas harga beras diperlukan untuk memastikan ketahanan pangan dan kesejahteraan masyarakat. Harga beras dipengaruhi oleh berbagai faktor ekonomi seperti inflasi, tingkat pengangguran, dan daya beli masyarakat, serta faktor produksi seperti cuaca, teknologi pertanian, dan distribusi. Kebijakan pemerintah, seperti subsidi, tarif impor, dan program bantuan pangan, juga berperan dalam menentukan harga beras. Dengan prediksi yang akurat, pemerintah dan pelaku pasar dapat mengambil langkah yang tepat untuk mengelola stok, mengatur distribusi, dan menetapkan kebijakan yang mendukung stabilitas harga beras.

Prediksi harga beras memiliki berbagai manfaat penting, termasuk menjaga stabilitas ekonomi dengan meminimalkan fluktuasi harga yang dapat mempengaruhi inflasi dan daya beli masyarakat. Selain itu, prediksi harga membantu dalam perencanaan produksi dan distribusi, memastikan ketahanan pangan dengan menghindari kekurangan atau kelebihan stok. Prediksi yang akurat juga memungkinkan pemerintah untuk merumuskan kebijakan yang tepat, seperti subsidi atau tarif impor, guna mengatur pasar dan melindungi petani serta konsumen. Dengan informasi prediktif, pelaku pasar dapat membuat keputusan bisnis yang lebih baik, mengurangi risiko, dan meningkatkan efisiensi operasional.

Pengujian dimulai dari tahap pengumpulan data berupa data harga beras yang di ambil dari *website* PIHPS (Pusat Informasi Harga Pangan Strategis) pada ada sembilan kota dan kabupaten yang tersebar di Jawa Barat, data berjumlah 1414 dari bulan Januari 2019 hingga bulan Februari 2024, Dan setelah dilakukan perbandingan metrik evaluasi dari kedua metode menggunakan RMSE dan MAPE, dapat disimpulkan bahwa model menggunakan metode LSTM dengan MAPE dan RMSE sebesar 0.02% dan 0.153073, Model dengan pembagian data latih sebesar 90% dan data uji sebesar 10% menggunakan parameter yang meliputi *batch size* sebanyak 64 dan *epoch* sebanyak 250.

KATA PENGANTAR

Puji syukur kehadiran Allah SWT karena telah memberikan kesempatan pada penulis untuk menyelesaikan laporan Penelitian ini. Atas rahmat dan hidayah-Nya penulis dapat menyelesaikan laporan Penelitian yang berjudul *Prediksi Harga Beras Menggunakan Long Short-Term Memory (LSTM)*.

Dalam penulisan laporan Penelitian ini penulis dengan senang hati ingin mengucapkan terimakasih yang sebesar-besarnya kepada :

1. Dr. Herfina, M.Pd., M.Kom. selaku pembimbing utama Program Studi Ilmu Komputer FMIPA Universitas Pakuan Bogor.
2. Dr. Fajar Delli M.Kom. selaku pembimbing pendamping Program Studi Ilmu Komputer FMIPA Universitas Pakuan Bogor.
3. Arie Qur'ania, M.Kom. selaku Program Studi Ilmu Komputer FMIPA Universitas Pakuan Bogor.
4. Asep Denih, S.Kom., M.Sc., Ph. D. selaku Dekan FMIPA Universitas Pakuan Bogor.
5. Kedua orang tua tercinta yang telah memberikan dorongan, semangat dan doa yang tiada henti agar penulis dapat menyelesaikan laporan ini.
6. Teman-teman seangkatan yang telah membantu dan mendukung saya dalam mengerjakan laporan penelitian ini.

Menyadari keterbatasan waktu dan kemampuan dalam penulisan laporan Penelitian ini masih jauh dari kata sempurna. Oleh karena itu, segala kritik dan saran yang membangun akan diterima dengan senang hati. Mudah-mudahan Allah SWT akan membalas semua kebaikan kepada semua pihak yang membantu. Akhir kata, Semoga laporan hasil ini dapat bermanfaat bagi kita semua. Aamiin.

Bogor, 10 Juli 2024

Muhamad Yuda
065119014

DAFTAR ISI

	Halaman
HALAMAN PENGESAHAN	i
PERNYATAAN KEASLIAN KARYA TULIS SKRIPSI	ii
PERNYATAAN PELIMPAHAN SKRIPSI DAN SUMBER INFORMASI SERTA PELIMPAHAN HAK CIPTA	iii
RIWAYAT HIDUP	iv
RINGKASAN	v
KATA PENGANTAR	vi
DAFTAR ISI	vii
DAFTAR GAMBAR	ix
DAFTAR TABEL	x
DAFTAR LAMPIRAN	xi
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Tujuan	2
1.3 Ruang Lingkup.....	2
1.4 Manfaat Penelitian	2
BAB II TINJAUAN PUSTAKA	3
2.1 Landasan Teori.....	3
2.1.1 Prediksi	3
2.1.2 Harga.....	3
2.1.3 Beras	3
2.1.4 <i>Time Series</i>	3
2.1.5 <i>Long Short Term Memory (LSTM)</i>	4
2.1.6 PIHPS Nasional	5
2.1.7 <i>Python</i>	5
2.1.8 RMSE.....	6
2.1.9 MAPE	6
2.2 Penelitian Terdahulu	7
2.3 Tabel Perbandingan	8
BAB III METODE PENELITIAN	9
3.1 Metodologi Penelitian.....	9
3.1.1 Pembersihan Data dan Integrasi Data	9
3.1.2 Seleksi Data dan Transformasi Data.....	10
3.1.3 <i>Data Mining</i>	10
3.1.4 <i>Evaluasi Data</i>	10
3.1.5 Presentase Pengetahuan	10
3.2 Alat dan Bahan.....	10
3.2.1 Alat Penelitian.....	10
3.2.2 Bahan Penelitian	11
BAB IV PERANCANGAN DAN IMPLEMENTASI	12
4.1 Tahap Analisis Kebutuhan.....	12

4.1.1 Sumber Data.....	12
4.1.2 Lingkup Pengambilan Data.....	12
4.1.3 Teknik Pengumpulan Data.....	13
4.1.4 Sampel Data	14
4.2 Perancangan Sistem	14
4.2.1 <i>Flowchart</i> Sistem.....	14
4.3 Seleksi Data	15
4.4 <i>Preprocessing</i> Data.....	16
4.4.1 Transformasi Data.....	17
4.4.2 <i>Cleaning</i> Data	17
4.5 Pembuatan Model	18
4.6 Evaluasi Model	18
4.6 Presentase Pengetahuan	19
BAB V HASIL DAN PEMBAHASAN.....	20
5.1 Pengambilan Data	20
5.2 Pembersihan Data dan Integrasi Data	20
5.3 <i>Preprocessing</i> Data	21
5.3.1 Transformasi Data.....	22
5.3.2 <i>Data Cleaning</i>	22
5.4 <i>Data Mining</i>	25
5.5 Evaluasi Data	25
5.6 Presentase Pengetahuan	27
BAB VI KESIMPULAN DAN SARAN	32
6.1 Kesimpulan.....	32
6.2 Saran	32
DAFTAR PUSTAKA.....	33
LAMPIRAN.....	35

DAFTAR GAMBAR

Gambar 1. Arsitektur LSTM	4
Gambar 2. Metode penelitian <i>Knowledge Discovery In Databases</i>	9
Gambar 3. Titik Pengambilan Data Harga Beras Di Jawa Barat	12
Gambar 4. Sample Data Harga Beras	14
Gambar 5. <i>Flowchart</i> Sistem.....	15
Gambar 6. <i>Sample</i> Visualisasi Pengetahuan	19
Gambar 7. Daaset Harga Beras	20
Gambar 9. Sebelum data harga beras di <i>cleaning</i>	23
Gambar 10. Sesudah data harga beras di <i>cleaning</i>	23
Gambar 11. <i>Visualisasi Missing Values</i>	24
Gambar 12. <i>Visualisasi</i> Setelah Proses <i>Interpolasi</i>	24
Gambar 13. Prediksi Harga Beras Daerah Kota Bandung	27
Gambar 14. Prediksi Harga Beras Daerah Kota Cirebon.....	28
Gambar 15. Prediksi Harga Beras Daerah Kota Tasikmalaya	28
Gambar 16. Prediksi Harga Beras Daerah Kota Bekasi.....	29
Gambar 17. Prediksi Harga Beras Daerah Kota Bogor.....	29
Gambar 18. Prediksi Harga Beras Daerah Kota Depok.....	30
Gambar 19. Prediksi Harga Beras Daerah Kota Sukabumi.....	30
Gambar 20. Prediksi Harga Beras Daerah Kabupaten Cirebon	31
Gambar 21. Prediksi Harga Beras Daerah Kabupaten Tasikmalaya.....	31

DAFTAR TABEL

Tabel 1. Tabel Perbandingan.....	8
Tabel 2. Daftar Pasar Yang Ada Pada Kabupaten dan Kota Di Jawa Barat	13
Tabel 3. <i>Sample</i> Seleksi Data untuk Daerah Bogor	16
Tabel 4. <i>Sample</i> Transformasi Data	17
Tabel 5. <i>Sample Cleaning</i> Data.....	18
Tabel 6. Rentang Nilai MAPE	18
Tabel 7 Seleksi Data.....	21
Tabel 8. Transformasi Data	22
Tabel 9. Pembagian Data	25
Tabel 10. Hasil Evaluasi Data Dengan pembagian data 90% Training 10% Testing.....	25
Tabel 11. Hasil Evaluasi Data Dengan pembagian data 80% Training 20% Testing.....	26
Tabel 12. Hasil Evaluasi Data Dengan pembagian data 70% Training 30% Testing.....	26
Tabel 13. Hasil Evaluasi Data Dengan pembagian data 60% Training 40% Testing.....	27

DAFTAR LAMPIRAN

Lampiran 1. Kartu Bimbingan Laporan Proposal	36
Lampiran 2. Kartu Bimbingan Laporan Hasil Penelitian.....	37
Lampiran 3. Kartu Bimbingan Laporan Skripsi.....	38
Lampiran 4. <i>Source Code</i>	39

BAB I

PENDAHULUAN

1.1 Latar Belakang

Prediksi adalah suatu proses memperkirakan secara sistematis tentang sesuatu yang paling mungkin terjadi di masa depan berdasarkan informasi masa lalu dan sekarang yang dimiliki, agar kesalahannya (selisih antara sesuatu yang terjadi dengan hasil perkiraan) dapat diperkecil. Prediksi tidak harus memberikan jawaban secara pasti kejadian yang akan terjadi, melainkan berusaha untuk mencari jawaban sedekat mungkin yang akan terjadi (Kaffil, 2019). Beras adalah butir padi yang telah dibuang kulit luarnya (sekamnya) yang menjadi dasar dedak kasar. Dedak halus berasal dari lapisan-lapisan permukaan biji beras, misalnya lapisan aleuron, lembaga dan beberapa lapis sel biji beras yang terlepas waktu proses penggilingan (Aminah, 2019).

Metode yang digunakan yaitu *Long Short Term Memory (LSTM)*, *Long Short Term Memory (LSTM)*, merupakan pengembangan salah satu algoritma *deep learning* yaitu *Recurrent Neural Network (RNN)* yang mengatasi salah satu kekurangan *RNN* yaitu kemampuan pengelolaan informasi dalam periode yang lama (Zahara, 2019). *LSTM* memiliki kelebihan untuk memproses keterkaitan antar konteks di dalam teks dan dapat menyimpan informasi dalam jangka waktu panjang (Riswinda, 2023). Kekurangan *LSTM* tidak dapat secara simultan mendeskripsikan interaksi spasial antar kendaraan serta relasi sewaktu antara trajektori deret waktu (Talita & Wiguna, 2019).

Penelitian sebelumnya tentang analisis harga yang dilakukan oleh Muhammad Ikhsan dan Sri Ngudi Wahyuni yang berjudul “Sentimen Terhadap Kenaikan Harga Bahan Bakar Minyak Menggunakan *Long Short-Term Memory*” (Ikhsan dan Wahyuni, 2023), Penelitian selanjutnya tentang analisis harga yang dilakukan oleh Roby Julian dan Muhammad Rizky Pribadi yang berjudul “Peramalan Harga Saham Pertambangan Pada Bursa Efek Indonesia (BEI) Menggunakan *Long Short Term Memory (LSTM)*” (Julian dan Pribadi, 2021). Penelitian selanjutnya tentang analisis harga yang dilakukan oleh Rizki Mugi Setya Adi dan Sudianto yang berjudul “Prediksi Harga Komoditas Pangan Menggunakan Algoritma *Long Short-Term Memory (LSTM)*” (Adi dan Sudianto, 2022). Penelitian selanjutnya tentang analisis sentimen pada review produk skincare oleh Rona Guines Purnasiwi, Kusriani dan Muhammad Hanafi yang berjudul “Analisis Sentimen Pada Review Produk Skincare Menggunakan Word Embedding dan Metode *Long Short-Term Memory (LSTM)*” (Purnasiwi, dkk, 2023). Penelitian selanjutnya tentang analisis harga yang dilakukan oleh Lailan Sahrina Hasibuani dan Yanda Novialdi yang berjudul “Prediksi Harga Minyak Goreng Curah dan Kemasan Menggunakan *Algoritme Long Short-Term Memory (LSTM)*” (Hasibuani & Novialdi, 2023).

Berdasarkan uraian di atas maka peneliti akan melakukan penelitian yang berjudul “Prediksi harga beras menggunakan *Long Short-Term Memory (LSTM)*” guna untuk mengetahui harga beras di masa mendatang dengan data yang ada pada saat ini.

1.2 Tujuan

Adapun tujuan dari penelitian ini adalah untuk memprediksi harga beras menggunakan metode *Long Short-Term Memory (LSTM)*, Guna untuk mengetahui harga beras di masa mendatang berdasarkan data-data harga beras di sejumlah pasar yang tersebar pada wilayah Jawa Barat yang dikumpulkan oleh Pusat Informasi Harga Pangan Strategis (PIHPS) Nasional.

1.3 Ruang Lingkup

Ruang lingkup dalam penelitian ini adalah sebagai berikut :

1. Objek Penelitian adalah data harga beras yang sudah di kumpulkan oleh Pusat Informasi Harga Pangan Strategis (*PIHPS*).
2. Penelitian ini hanya berfokus pada penerapan metode *Long Short-Term Memory (LSTM)* untuk memprediksi harga beras berdasarkan data harga beras dari data sebelumnya (*Forecasting Time Series*).
3. Output yang di hasilkan adalah prediksi data harga beras yang sudah di proses menggunakan metode *Long Short-Term Memory (LSTM)* dan di jadikan parameter untuk mengetahui harga beras di masa mendatang.
4. *Text Editor* yang di gunakan adalah *Google Colab* dan Bahasa yang digunakan adalah *Python*.

1.4 Manfaat Penelitian

Adapun manfaat dari penelitian ini adalah :

1. Memberikan informasi terkait prediksi harga beras di masa mendatang agar masyarakat bisa untuk merencanakan dan mengambil tindakan ketika telah di prediksi terjadinya kenaikan harga beras yang telah di proses menggunakan metode *Long Short-Term Memory (LSTM)*.
2. Menjadikan penelitian ini sebagai referensi untuk pihak-pihak yang memerlukan penelitian yang sama atau yang hampir sama dengan penelitian terkait.

BAB II

TINJAUAN PUSTAKA

2.1 Landasan Teori

2.1.1 Prediksi

Prediksi adalah suatu proses memperkirakan secara sistematis tentang sesuatu yang paling mungkin terjadi di masa depan berdasarkan informasi masa lalu dan sekarang yang dimiliki, agar kesalahannya (selisih antara sesuatu yang terjadi dengan hasil perkiraan) dapat diperkecil. Prediksi tidak harus memberikan jawaban secara pasti kejadian yang akan terjadi, melainkan berusaha untuk mencari jawaban sedekat mungkin yang akan terjadi (Kaffil, 2019).

Prediksi adalah suatu kegiatan untuk memperkirakan kejadian yang akan terjadi dimasa depan dengan menggunakan data yang sudah ada atau suatu proses memperkirakan secara sistematis tentang sesuatu yang paling mungkin terjadi di masa depan berdasarkan informasi masa lalu dan sekarang yang dimiliki (Orpa, dkk, 2019).

2.1.2 Harga

Harga adalah jumlah uang yang ditukarkan untuk sebuah produk atau jasa. Harga sebagai nilai suatu barang atau jasa yang dinyatakan dengan uang. Harga juga merupakan unsur bauran pemasaran yang sifatnya fleksibel di mana setiap saat dapat berubah berdasarkan waktu dan tempat. Harga bukan hanya angka atau nominal yang tertera dilabel suatu kemasan, akan tetapi harga mempunyai banyak bentuk dan memiliki banyak fungsi di antaranya, ongkos, upah, tarif, sewa tempat, pembayaran jasa, dan gaji. Semuanya merupakan harga yang harus dibayar untuk mendapatkan barang dan jasa. Dan harga juga bisa di bilang sejumlah uang yang dibebankan atas suatu produk atau jasa, atau jumlah dari nilai yang ditukar konsumen atas manfaat-manfaat karena menggunakan produk atau jasa tersebut (Saputri, 2019).

2.1.3 Beras

Beras adalah hasil utama dari proses penggilingan gabah hasil tanaman padi yang seluruh lapisan sekamnya terkelupas dan seluruh atau sebagian lembaga dan lapisan bekatulnya telah dipisahkan. Dalam peraturan ini juga disebutkan bahwa beras yang diperoleh melalui penggilingan padi (Aminah, 2019).

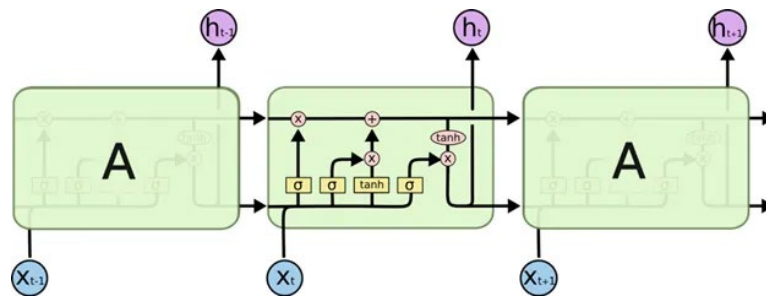
2.1.4 Time Series

time series adalah teknik statistik yang menggunakan data historis terakumulasi selama periode waktu. Metode time series berasumsi bahwa apa yang telah terjadi di masa lalu akan terus terjadi di masa depan. Metode ini berhubungan hanya dengan satu faktor waktu. Metode ini berasumsi bahwa pola historis diidentifikasi atau tren permintaan dari waktu ke waktu akan berulang (Yuliana, 2019).

2.1.5 Long Short Term Memory (LSTM)

Long Short-Term Memory (LSTM) merupakan turunan dari *Recurrent Neural Network* (RNN) yang telah terbukti berhasil digunakan untuk melakukan prediksi pada data time series. RNN mampu menggunakan informasi yang telah direkam sebelumnya yang memiliki urutan atau *sequence* yang panjang dan beragam. Namun, RNN memiliki masalah *vanishing gradient* yang dapat menghambat kinerja sistem. Dengan kemampuannya untuk mempelajari pola yang rumit dari data berurutan, LSTM dapat melatih model dan mengatasi kendala *vanishing gradient* yang sering ditemukan pada RNN. (Ashari & Sadiki, 2020)

RNN memiliki bentuk rangkaian modul- modul jaringan syaraf yang berulang. LSTM juga memiliki struktur yang serupa dengan RNN tetapi memiliki fitur tambahan berupa gerbang pada sel. Arsitektur dari LSTM ditunjukkan pada Gambar 1.



Gambar 1. Arsitektur LSTM

Sel memori (*cell state*) berfungsi untuk menyimpan informasi yang sudah di proses *forget gate*, *input gate* dan *output gate*, persamaan dari sel memori (*cell state*) ditunjukkan pada Persamaan 1.

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \quad (1)$$

Forget Gate berfungsi untuk menghapus informasi yang tidak relevan dari sel memori (*cell state*). hasil dari gerbang ini adalah angka 0 dan 1, angka 1 menunjukkan bahwa informasi harus disimpan sedangkan angka 0 menunjukkan bahwa informasi sudah tidak dibutuhkan sehingga dapat dihapus, persamaan dari *forget gate* ditunjukkan pada Persamaan 2.

$$f_t = \sigma(W_{hf} \cdot h_{t-1} + W_{xf} \cdot x_t + b_f) \quad (2)$$

Input Gate berfungsi untuk menentukan dan mengontrol berapa banyak informasi baru yang akan ditambahkan ke dalam sel memori (*cell state*), hal ini perlu dilakukan untuk mencegah sel memori menyimpan data yang tidak relevan, persamaan dari *input gate* ditunjukkan pada Persamaan 3.

$$i_t = \sigma(W_{hi} \cdot h_{t-1} + W_{xi} \cdot x_t + b_i) \quad (3)$$

Output Gate berfungsi untuk menggabungkan informasi dari sel memori (*cell state*) saat ini dengan informasi dari *input gate* lalu meneruskannya kepada sel memori (*cell state*) yang baru, persamaan dari *output gate* ditunjukkan pada Persamaan 4.

$$o_t = \sigma(W_{ho} \cdot h_{t-1} + W_{xo} \cdot x_t + b_o) \quad (4)$$

Keterangan:

- w adalah bobot (*weight*) dari matriks
- h adalah *hidden state* pada *time step* sebelumnya
- x adalah *input time step*
- b adalah bias dari vector

2.1.6 PIHPS Nasional

Pusat Informasi Harga Pangan Strategis (PIHPS) Nasional adalah sistem informasi perangkat lunak yang dikembangkan oleh Bank Indonesia untuk memberikan akses kepada masyarakat mengenai informasi harga pangan terkini dan membantu proses perumusan kebijakan pengendalian inflasi.

Dengan adanya akses yang mudah terhadap informasi harga pangan, permasalahan kesenjangan informasi harga dapat diatasi dan stabilitas harga pangan dapat terjaga. Sistem Informasi ini juga menjadi tambahan pilihan bagi masyarakat untuk memperoleh informasi pangan baik di level nasional maupun daerah.

2.1.7 Python

Python adalah bahasa pemrograman komputer tingkat tinggi berbasis teks, kode *python* ditulis instruksinya menggunakan gabungan kata-kata, singkatan, angka, dan simbol. (Santoso, 2022).

Python dikembangkan oleh Guido van Rossum seorang programmer Belanda dan dirilis pada 20 Februari 1991. Nama *python* berasal dari grup komedi favorit Guido yaitu Monty Python yang merupakan kelompok komedi Inggris yang populer pada tahun 1970-an. Sintaksnya yang intuitif, sifatnya yang *open source* dan pustakanya yang beragam membuat *python* dipakai secara luas dalam berbagai industri.

Saat ini *Python* banyak digunakan dalam komputasi ilmiah dan telah digunakan secara luas dalam berbagai bidang seperti *data science*, statistik, *machine learning*, *deep learning*, analisis prediktif dan lain-lain.

2.1.8 RMSE

RMSE (*Root Mean Square Error*) adalah akar kuadrat dari rata-rata kesalahan yang dihasilkan dari perhitungan (Prasetyo et al., 2021) Persamaan RMSE ditunjukkan pada Persamaan 5.

$$\text{RMSE} = \left(\frac{\sum (y_i - \hat{y}_i)}{n} \right)^{1/2} \quad (5)$$

Keterangan:

- n adalah ukuran sampel
- y_i adalah nilai data aktual
- \hat{y}_i adalah nilai data prediksi

Semakin kecil nilai RMSE, maka prediksi tersebut semakin akurat.

2.1.9 MAPE

MAPE (*Mean Absolute Percentage Error*) adalah nilai tengah kesalahan persentase absolut ramalan. MAPE menunjukkan permintaan aktual dengan rata-rata persentase nilai absolut kesalahan yang terjadi selama periode peramalan (Edita, 2023). Persamaan MAPE ditunjukkan pada Persamaan 6.

$$M = \frac{1}{n} \sum_{t=1}^n \left| \frac{A_t - F_t}{A_t} \right| \quad (6)$$

Keterangan:

- n adalah ukuran sampel
- A_t adalah nilai data actual
- F_t adalah nilai data prediksi

Semakin kecil persentase dari MAPE, maka prediksi tersebut semakin akurat.

2.2 Penelitian Terdahulu

1. Penulis : Deden Martia N., Tacbir Hendro P., & Puspita Nurul S.
Judul : Metode *K-Nearest Neighbor (KNN)* dalam Memprediksi Curah Hujan di Kota Bandung
Tahun : 2022
Isi : Penelitian ini berisi hasil prediksi curah hujan di kota bandung menggunakan metode *K-Nearest Neighbor (KNN)*. Berdasarkan penelitian yang dilaksanakan didapatkan hasil nilai K yang telah dilakukan pelatihan *K-Cross Validation* maka didapatkan hasil terbaik sebanyak 5 dengan nilai akurasi sebesar 86.199%.
2. Penulis : Roby Julian dan Muhammad Rizky Pribadi
Judul : Peramalan Harga Saham Pertambangan Pada Bursa Efek Indonesia (*BEI*) Menggunakan *Long Short Term Memory (LSTM)*
Tahun : 2021
Isi : Isi dari penelitian ini yaitu identifikasi masalah yang berkaitan dengan peramalan atau forecasting harga saham yang terdaftar pada Bursa Efek Indonesia (*BEI*) serta mengidentifikasi fenomena-fenomena yang terjadi pada pasar bursa sahan..
3. Penulis : Dwiasnati Saruni & Devianto Yudo
Judul : Optimasi Prediksi Bencana Banjir menggunakan Algoritma *SVM* untuk penentuan Daerah Rawan Bencana Banjir.
Tahun : 2021
Isi : Memberikan peningkatan yang sangat tajam dengan nilai selisih untuk *accuracy* sebesar 14.69 dan 0.227 untuk *AUC*. Sehingga dapat disimpulkan penggunaan feature selection *Weight by Corellation* sangat baik apabila digunakan pada algoritma *Support Vector Machine (SVM)*.
4. Penulis : Rona Guines Purnasiwi, Kusriani dan Muhammad Hanafi
Judul : Analisis Sentimen Pada *Review* Produk *Skincare* Menggunakan *Word Embedding* dan Metode *Long Short-Term Memory (LSTM)*
Tahun : 2023
Isi : Isi dari penellitian ini adalah analisis sentimen *review* produk *skincare female daily* menggunakan *LSTM* dengan menggunakan *word embedding Word2Vec*.
5. Penulis : Laia Luter Marthin dan Setyawan Yudi
Judul : Perbandingan Hasil Klasifikasi Curah Hujan Menggunakan Metode *SVM* Dan *NBC*
Tahun : 2020
Isi : Isi dari Penelitian ini yaitu Prakiraan curah hujan, Metode ini menghasilkan tingkat akurasi yang berbeda, metode *Naïve Bayes Classifier* menghasilkan tingkat akurasi 65,75% dengan prakiraan angka kejadian hujan sebanyak 27 dari 73 hari pada data testing. Sedangkan metode *Support Vector Machine* menghasilkan prakiraan hujan sebanyak 58 dari 73 hari pada data testing dengan tingkat akurasi 79,45%.

2.3 Tabel Perbandingan

Tabel perbandingan berfungsi untuk membandingkan penelitian terdahulu yang kita gunakan sebagai bahan acuan untuk penelitian prediksi harga beras. Dapat dilihat pada Tabel 1.

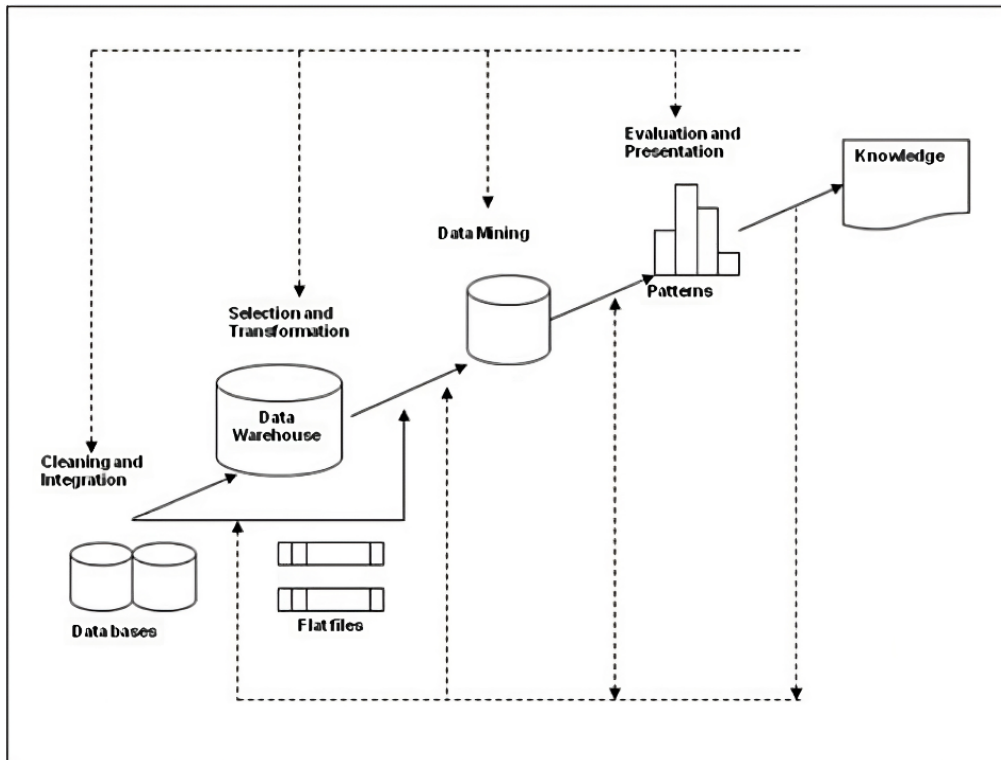
Tabel 1. Tabel Perbandingan

No.	Peneliti dan Tahun	Judul	Perbandingan Metode			
			KNN	LSTM	SVM	NBC
1.	N Martia.Deden, P Hendro Tacbir., & S Nurul Puspita. (2022)	Metode <i>K-Nearest Neighbor (KNN)</i> dalam Memprediksi Curah Hujan di Kota Bandung	√	---	---	---
2.	Julian Roby dan Pribadi Rizky Muhammad (2021)	Peramalan Harga Saham Pertambangan Pada Bursa Efek Indonesia (<i>BEI</i>) Menggunakan <i>Long Short Term Memory (LSTM)</i>	---	√	---	---
3.	Dwiasnati Saruni & Devianto Yudo (2021)	Optimasi Prediksi Bencana Banjir menggunakan Algoritma <i>SVM</i> untuk penentuan Daerah Rawan Bencana Banjir	---	---	√	---
4.	Purnasiwi Guines Rona, Kusriani dan Hanafi Muhammad (2023)	Analisis Sentimen Pada Review Produk <i>Skincare</i> Menggunakan <i>Word Embedding</i> dan Metode <i>Long Short-Term Memory (LSTM)</i>	---	√	---	---
5.	Laia Luter Marthin dan Setyawan Yudi (2020)	Perbandingan Hasil Klasifikasi Curah Hujan Menggunakan Metode <i>SVM</i> Dan <i>NBC</i>	---	---	√	√
6.	Yuda Muhamad (2023)	Prediksi Harga Beras Menggunakan <i>Long Short-Term Memory (LSTM)</i>	---	√	---	---

BAB III METODE PENELITIAN

3.1 Metodologi Penelitian

Metode yang dipakai pada penelitian ini yaitu *Knowledge Discovery In Databases (KDD)*. *Knowledge Discovery in Database (KDD)* memiliki tujuan yakni menggunakan data yang tersedia pada basis data kemudian mengolah data untuk mendapatkan sebuah informasi baru yang bermanfaat. Dapat dilihat pada Gambar 2.



Gambar 2. Metode penelitian *Knowledge Discovery In Databases*

Berikut adalah tahap-tahap yang di gunakan dalam metode Penelitian *Knowledge Discovery In Databases (KDD)* :

3.1.1 Pembersihan Data dan Integrasi Data

Pada tahap awal, data yang akan digunakan dalam proses KDD dikumpulkan dan dibersihkan dari data yang tidak relevan dan data yang hilang (*missing values*). Selain itu, berbagai berkas data juga diintegrasikan menjadi satu kesatuan. Tahap ini sangat penting karena data yang tidak bersih dan tidak terintegrasi dengan baik dapat menghambat proses data mining dan juga menghasilkan hasil yang tidak akurat.

3.1.2 Seleksi Data dan Transformasi Data

Seleksi data dilakukan untuk mengurangi kompleksitas dan waktu pemrosesan data dengan hanya fokus pada data yang benar-benar diperlukan untuk proses prediksi dan Transformasi data dilakukan untuk mengubah data mentah menjadi format yang lebih mudah untuk dianalisis oleh algoritma *data mining*.

3.1.3 Data Mining

Pada tahapan ini dilakukan validation yakni proses membagi dataset menjadi beberapa bagian yang mana 1 diantara bagian lainnya menjadi data *testing*, dan yang lainnya menjadi data training. Kemudian dimasukkan kedalam *Long Short-Term Memory (LSTM)*. Ini dilakukan bergantian pada tiap bagian data sampai mendapat nilai terbaik dari model ini.

3.1.4 Evaluasi Data

Setelah tahapan data *mining* selesai maka selanjutnya adalah melakukan *evaluasi* terhadap hasil dari pemodelan tersebut menggunakan metode *LSTM*. *Evaluasi* merupakan suatu pola informasi yang dihasilkan dalam data mining untuk menampilkan suatu bentuk yang dapat mudah di mengerti oleh pihak yang berkepentingan.

3.1.5 Presentase Pengetahuan

Presentase pengetahuan adalah *visualisasi* dan penyajian pengetahuan terhadap hasil data prediksi dan metode yang digunakan untuk memproses hasil prediksi tersebut, presentase pengetahuan dapat ditampilkan dalam bentuk grafik atau tabel untuk memudahkan pemahaman dan interpretasi hasil prediksi.

3.2 Alat dan Bahan

3.2.1 Alat Penelitian

Alat yang di butuhkan pada penelitian ini berupa perangkat lunak (*software*) dan perangkat keras (*hardware*) yaitu :

a. Perangkat Lunak (Software)

Perangkat Lunak yang digunakan adalah :

1. Sistem Operasi *Windows 11 64bit*
2. *Google Chrome*
3. *Google Colab*
4. *Microsoft Excel 2021*
5. *Microsoft Word 2021*
6. *Mendeley*

b. Perangkat Keras (*Hardware*)

Perangkat Keras yang digunakan adalah :

1. Laptop Hp
2. *Procesor Intel core i3*
3. RAM 12GB DDR3I

3.2.2 Bahan Penelitian

Bahan yang diperlukan pada penelitian ini adalah :

1. Jurnal, media cetak, dan internet sebagai penunjang referensi dalam pelaksanaan penelitian dan pembuatan laporan proposal penelitian.
2. Data harga beras yang disediakan oleh Pusat Informasi Harga Pangan Strategis (PIHPS) Nasional.

BAB IV PERANCANGAN DAN IMPLEMENTASI

4.1 Tahap Analisis Kebutuhan

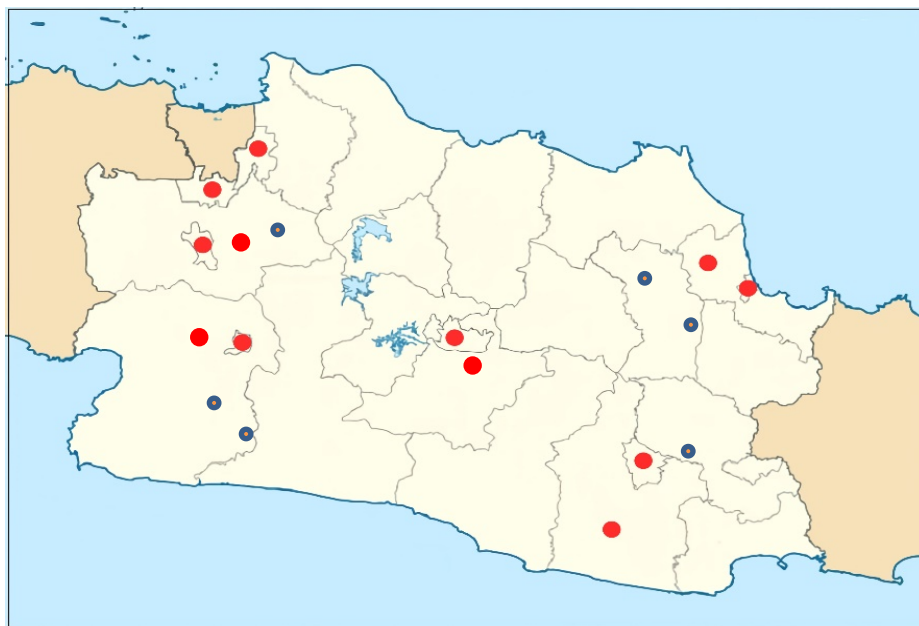
Proses yang dilakukan pertama kali dalam penelitian ini adalah pengumpulan data, data harga beras di wilayah Jawa barat yang merupakan objek utama dalam penelitian ini.

4.1.1 Sumber Data

Data yang digunakan di penelitian ini merupakan data sekunder yang didapat pada Pusat Informasi Harga Pangan Strategis (PIHPS) Nasional yang merupakan sistem informasi yang dikembangkan oleh Bank Indonesia. Data harga telur dikumpulkan dalam format data perhari dan harga perkilogram pada sejumlah pasar yang tersebar di Jawa Barat. Data-data tersebut dapat diakses untuk umum pada situs PIHPS Nasional dengan alamat www.bi.go.id/hargapangan.

4.1.2 Lingkup Pengambilan Data

Lingkup pengambilan data mencakup data yang diperoleh pada 18 pasar tradisional yang tersebar pada sembilan kota dan kabupaten yang terdapat di Jawa Barat. Pasar tersebut adalah pasar utama yang tergolong besar, baik dari segi ukuran maupun jumlah pengunjung. Adapun titik pengambilan data harga ditunjukkan pada gambar 4, dan daftar pasar yang ada dalam titik tersebut ditunjukkan pada tabel 2.



Gambar 3. Titik Pengambilan Data Harga Beras Di Jawa Barat

Adapun daftar pasar yang terdaftar pada website PIHPS yang ada pada Kabupaten dan Kota Di Jawa Barat ditunjukkan pada Tabel 2.

Tabel 2. Daftar Pasar Yang Ada Pada Kabupaten dan Kota Di Jawa Barat

No.	Kabupaten / Kota	Nama Pasar
1	Kota Bandung	Pasar Kiara Condong
		Pasar Kosambi
2	Kota Cirebon	Pasar Jagasatru
		Pasar Harjamukti
3	Kota Tasikmalaya	Pasar Induk Cikurubuk
		Pasar Cibauti
4	Kota Bekasi	Pasar Krani
		Pasar Pondok Gede
5	Kota Bogor	Pasar Anyar
		Pasar Warung Jambu Dua
6	Kota Depok	Pasar Kemiri Muka
		Pasar Negeri
7	Kota Sukabumi	Pasar Pelita
		Pasar Dewi Sartika
8	Kab Cirebon	Pasar Mundu
		Pasar Weru
9	Kab. Tasikmalaya	Pasar Manojaya
		Pasar Singaparna

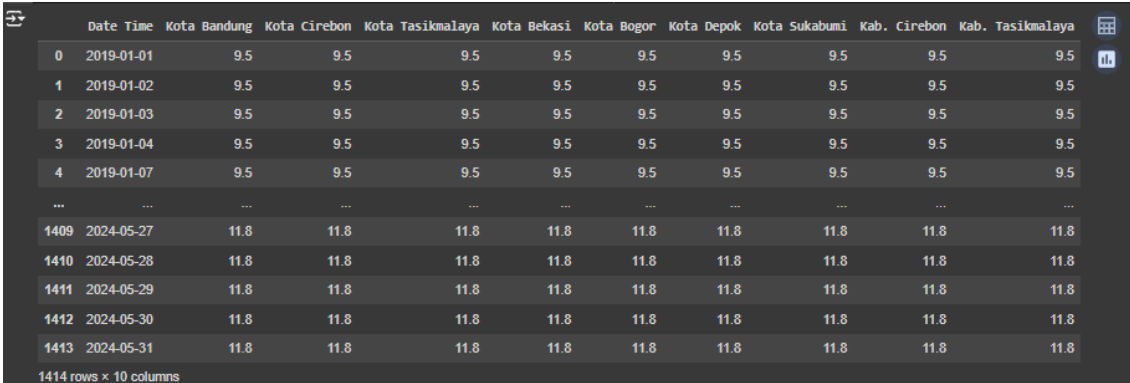
4.1.3 Teknik Pengumpulan Data

Data yang digunakan pada penelitian ini merupakan data sekunder dari PIHPS Nasional. PIHPS Nasional memperoleh data harga melalui proses survei pada transaksi antara penjual dan pembeli. Survei ini dilakukan dengan kriteria berikut:

1. Survei dilakukan pada dua pedagang yang berjarak cukup jauh antar satu sama lain di pasar yang sama untuk menghindari homogenitas harga.
2. Pedagang yang dipilih bersifat tetap pada setiap survei untuk menjamin kontinuitas survei.
3. Pedagang tersebut berjualan pada tempat yang tetap dengan penjualan yang relatif besar dan menjual pangan untuk kebutuhan konsumsi, bukan untuk dijual kembali (grosir).
4. Jika pedagang tersebut tidak berjualan sementara, maka dapat diganti dengan pedagang lain dengan kriteria persyaratan yang sama.

4.1.4 Sampel Data

Sampel data mencakup informasi mengenai harga beras per-liter yang dikumpulkan dari berbagai pasar yang tersebar di seluruh Jawa Barat, data mencakup periode harian yang menggambarkan fluktuasi harga beras di wilayah Jawa Barat. Adapun contoh sampel data harga beras yang ada di kota dan kabupaten di Jawa Barat, Ditunjukkan pada Gambar 4.



	Date Time	Kota Bandung	Kota Cirebon	Kota Tasikmalaya	Kota Bekasi	Kota Bogor	Kota Depok	Kota Sukabumi	Kab. Cirebon	Kab. Tasikmalaya
0	2019-01-01	9.5	9.5	9.5	9.5	9.5	9.5	9.5	9.5	9.5
1	2019-01-02	9.5	9.5	9.5	9.5	9.5	9.5	9.5	9.5	9.5
2	2019-01-03	9.5	9.5	9.5	9.5	9.5	9.5	9.5	9.5	9.5
3	2019-01-04	9.5	9.5	9.5	9.5	9.5	9.5	9.5	9.5	9.5
4	2019-01-07	9.5	9.5	9.5	9.5	9.5	9.5	9.5	9.5	9.5
...
1409	2024-05-27	11.8	11.8	11.8	11.8	11.8	11.8	11.8	11.8	11.8
1410	2024-05-28	11.8	11.8	11.8	11.8	11.8	11.8	11.8	11.8	11.8
1411	2024-05-29	11.8	11.8	11.8	11.8	11.8	11.8	11.8	11.8	11.8
1412	2024-05-30	11.8	11.8	11.8	11.8	11.8	11.8	11.8	11.8	11.8
1413	2024-05-31	11.8	11.8	11.8	11.8	11.8	11.8	11.8	11.8	11.8

1414 rows x 10 columns

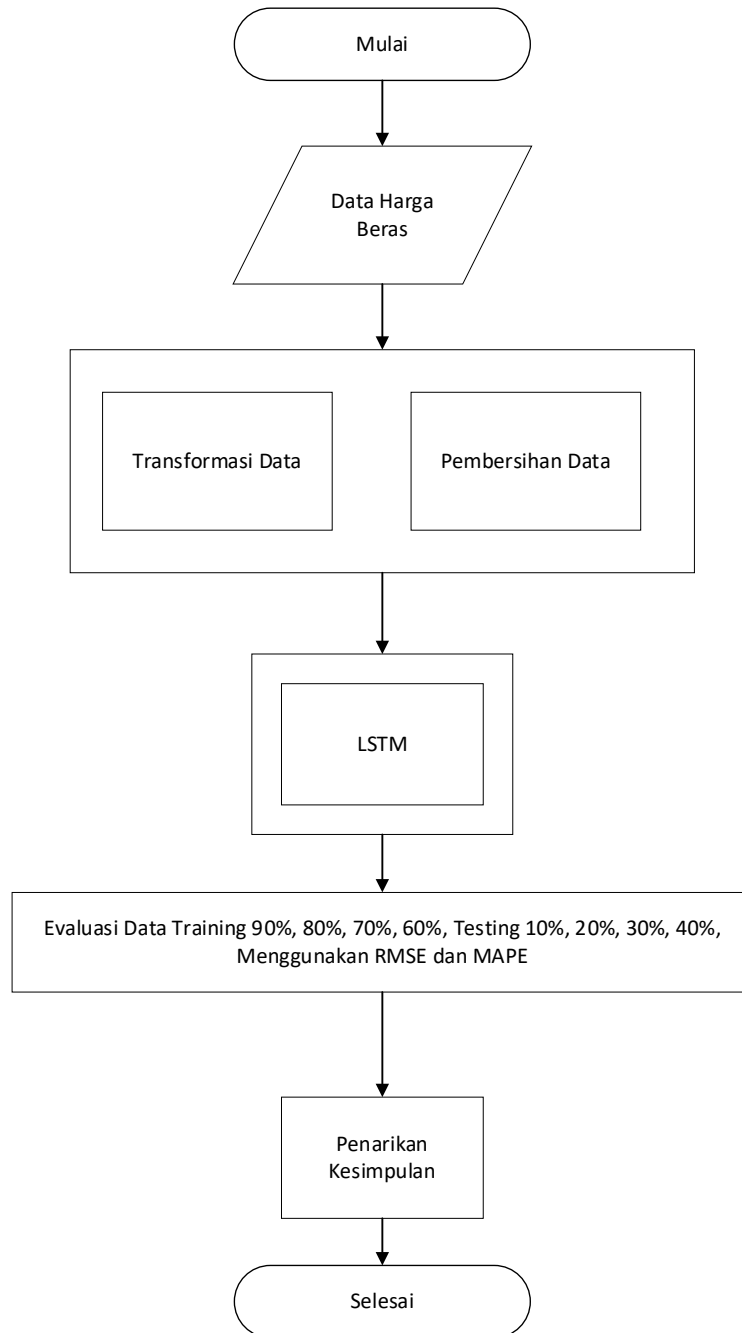
Gambar 4. Sample Data Harga Beras

Data yang akan digunakan pada penelitian ini merupakan data harga beras di sembilan kota dan kabupaten yang ada di Jawa Barat pada periode Januari 2019 hingga Februari 2024 dengan total data berjumlah 1414.

4.2 Perancangan Sistem

4.2.1 Flowchart Sistem

Flowchart merupakan penggambaran secara grafik dari langkah-langkah dan urutan prosedur suatu program. Biasanya mempengaruhi penyelesaian masalah yang khususnya perlu dipelajari dan dievaluasi lebih lanjut, Adapun *flowchart* yang digunakan dalam metode *LSTM* ditunjukkan pada gambar 5.



Gambar 5. *Flowchart* Sistem

4.3 Seleksi Data

Seleksi data dilakukan untuk mengurangi kompleksitas dan waktu pemrosesan data dengan hanya fokus pada data yang benar-benar diperlukan untuk proses prediksi, Seperti ditunjukkan pada tabel 3.

Tabel 3. *Sample* Seleksi Data untuk Daerah Bogor

Sebelum Di Seleksi		Sesudah Di Seleksi	
Bogor 2019		Bogor 2019	
Tanggal	Januari	Tanggal	Harga
1	-	03/01/2019	9500
2	-	04/01/2019	9500
3	9500	05/012019	9500
4	9500	06/01/2019	9500
5	9500	07/01/2019	9500
6	9500	08/01/2019	9500
7	9500	09/01/2019	9500
8	9500	10/01/2019	9500
9	9500	11/01/2019	9300
10	9500	12/01/2019	9300
....
....

4.4 Preprocessing Data

Data harga beras yang diunduh dari situs PIHPS Nasional tidak bisa langsung diproses untuk membuat model, akan tetapi perlu dilakukan pra-pemrosesan data terlebih dahulu. Prapemrosesan data adalah proses mengubah data mentah menjadi bentuk yang lebih mudah dipahami dan siap untuk diproses oleh algoritma, dengan melakukan prapemrosesan data, proses prediksi menjadi lebih efisien dan hasil prediksi menjadi lebih akurat. Prapemrosesan data perlu dilakukan untuk memperbaiki kesalahan pada data mentah seperti data yang hilang, data yang tidak valid maupun format data yang tidak teratur.

Terdapat beberapa tahapan dalam prapemrosesan data atau data *preprocessing* yang umumnya dilakukan dalam data mining, tahapan-tahapan tersebut adalah integrasi data, pembersihan data, transformasi data dan seleksi data.

4.4.1 Transformasi Data

Transformasi Data adalah proses mengubah data mentah menjadi bentuk yang lebih sesuai untuk proses *data mining*, beberapa teknik transformasi data yang umum dilakukan dalam prapemrosesan data adalah *normalization*, *discretization* dan lain-lain.

Teknik transformasi data yang akan digunakan pada penelitian ini adalah normalisasi menggunakan *min-max scaler* untuk menyesuaikan skala pada data menjadi rentang yang sama, *min-max scaler* dilakukan agar rentang pada data tidak terlalu besar maupun tidak terlalu kecil, Seperti ditunjukkan pada tabel

Tabel 4. *Sample* Transformasi Data

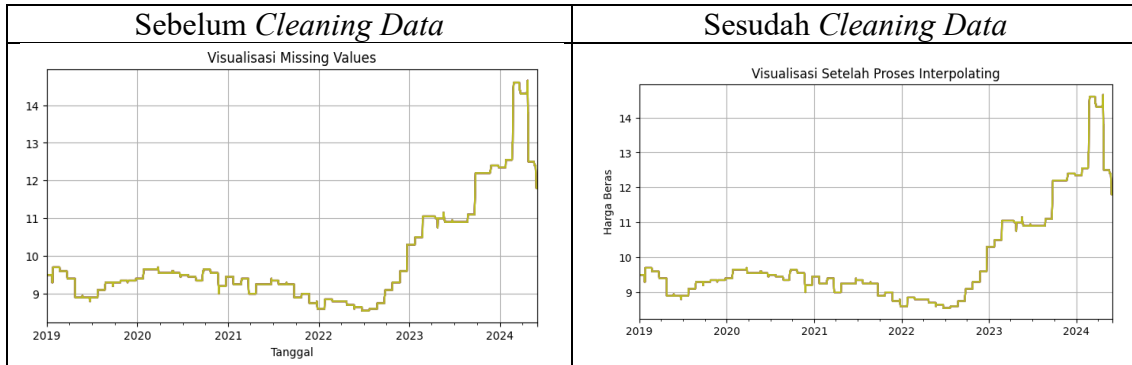
Tanggal	Sebelum Harga Beras Di Transformasi	Sesudah Harga Beras Di Transformasi
01/01/2020	9400	9.491.364
02/01/2020	9400	9.490.711
03/01/2020	9400	9.490.356
04/01/2020	9400	9.491.161
05/01/2020	9400	9.490.483
06/01/2020	9400	9.490.369
07/01/2020	9400	9.902.725
08/01/2020	9400	9.490.626
09/01/2020	9400	9.490.291
10/01/2020	9400	9.490.291
....
....

4.4.2 *Cleaning* Data

Tahap kedua yang dilakukan adalah pembersihan data, pembersihan data dilakukan untuk menghapus data yang tidak valid, data yang tidak relevan dan data yang tidak sesuai format, pembersihan data dilakukan untuk memastikan bahwa data yang digunakan untuk proses pemodelan adalah data yang berkualitas.

Data yang diunduh dari situs PIHPS Nasional masih terdapat data yang tidak valid, yaitu data dengan nilai (-) pada beberapa baris data, oleh karena itu proses pembersihan data perlu dilakukan sebelum data diproses lebih lanjut. Seperti ditunjukkan pada tabel 5.

Tabel 5. Sample *Cleaning Data*



4.5 Pembuatan Model

Sebelum pemodelan dari masing masing metode dilakukan, beberapa parameter yang diperlukan oleh metode tersebut harus diinisialisasi terlebih dahulu. Parameter sendiri adalah suatu nilai yang mengatur perilaku dari suatu algoritma, dengan mengoptimalkan nilai dari parameter, model akan memiliki kinerja dan akurasi yang baik.

4.6 Evaluasi Model

Setelah model selesai dilatih sesuai beberapa skenario yang ditentukan, model-model tersebut perlu dievaluasi untuk menentukan model terbaik untuk digunakan pada proses prediksi. Pada penelitian ini, evaluasi model dilakukan dengan cara menghitung nilai *Root Mean Squared Error* (RMSE) dan *Mean Absolute Percentage Error* (MAPE) dari masing-masing model, nilai RMSE digunakan untuk mengukur selisih rata-rata antara data prediksi dengan data uji, nilai MAPE digunakan untuk mengategorikan akurasi model.

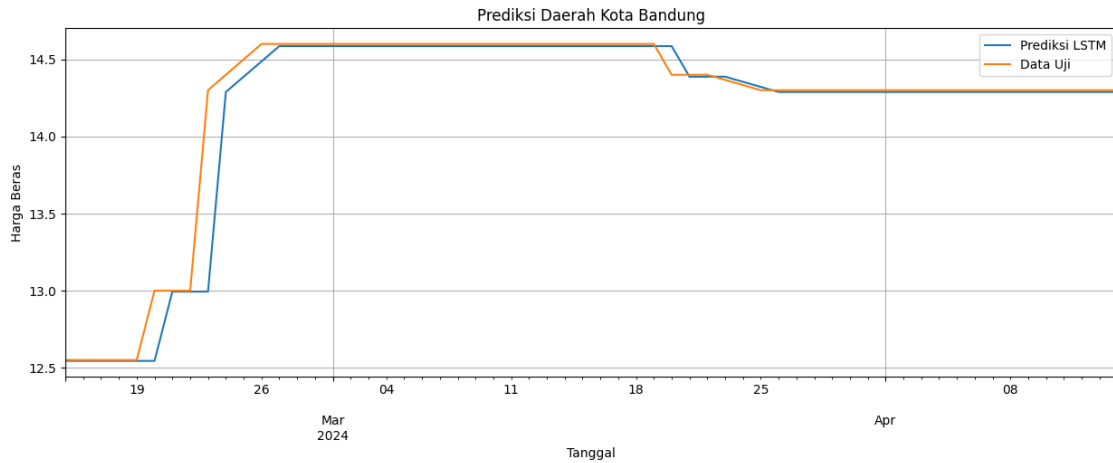
Rentang kategori MAPE ditunjukkan pada Tabel 6, nilai MAPE kurang dari 10% memiliki tingkat akurasi prediksi yang sangat baik. Nilai MAPE antara 10% sampai 20% memiliki tingkat akurasi prediksi baik. Nilai MAPE antara 20% sampai 50% memiliki tingkat akurasi prediksi cukup sedangkan nilai MAPE lebih dari 50% memiliki tingkat akurasi prediksi buruk (Nurfadilah et al., 2022)

Tabel 6. Rentang Nilai MAPE

No	Nilai MAPE	Akurasi Prediksi
1	<10 %	Sangat Baik
2	10% - 20%	Baik
3	20% - 50%	Cukup
4	>50%	Buruk

4.6 Presentase Pengetahuan

Presentase pengetahuan adalah visualisasi dan penyajian pengetahuan terhadap hasil data prediksi dan metode yang digunakan untuk memproses hasil prediksi tersebut, presentase pengetahuan dapat ditampilkan dalam bentuk grafik atau tabel untuk memudahkan pemahaman dan interpretasi hasil prediksi, seperti ditunjukkan pada gambar 6.

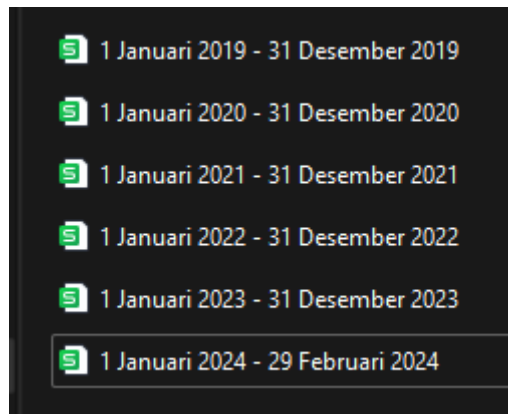


Gambar 6. *Sample* Visualisasi Pengetahuan

BAB V HASIL DAN PEMBAHASAN

5.1 Pengambilan Data

Penelitian ini menggunakan data harga beras harian di enam kota dan kabupaten yang tersebar di Jawa Barat, data harga beras tersebut diperoleh dari situs Pusat Informasi Harga Pangan Strategis Nasional (PIHPS Nasional, <https://www.bi.go.id/hargapangan>). Data yang digunakan untuk penelitian ini dimulai dari tanggal 1 Januari 2019 hingga 29 Februari 2024 berupa beberapa dokumen *excel* dalam format XLSX. Ditunjukkan pada gambar 7.



Gambar 7. Daaset Harga Beras

5.2 Pembersihan Data dan Integrasi Data

Tabel 7 kolom sebelah kirimerupakan data yang belum di seleksi atau belum di pilih untuk diolah dalam data *minning* dan pada kolom sebelah kanan merupakan data hasil seleksinya.

Tabel 7 Seleksi Data

Sebelum Di Seleksi		Sesudah Di Seleksi	
Bogor 2019		Bogor 2019	
Tanggal	Januari	Tanggal	Harga
1	-	03/01/2019	9500
2	-	04/01/2019	9500
3	9500	05/01/2019	9500
4	9500	06/01/2019	9500
5	9500	07/01/2019	9500
6	9500	08/01/2019	9500
7	9500	09/01/2019	9500
8	9500	10/01/2019	9500
9	9500	11/01/2019	9300
10	9500	12/01/2019	9300
....
....

5.3 Preprocessing Data

Data harga beras yang diunduh di situs PIHPS tidak bisa langsung diproses untuk membuat model, akan tetapi perlu dilakukan prapemrosesan data terlebih dahulu. Prapemrosesan data adalah proses mengubah data mentah menjadi bentuk yang lebih mudah dipahami dan siap untuk diproses oleh algoritma, dengan melakukan prapemrosesan data, proses prediksi menjadi lebih efisien dan hasil prediksi menjadi lebih akurat. Prapemrosesan data perlu dilakukan untuk memperbaiki kesalahan pada data mentah seperti data yang hilang, data yang tidak valid maupun format data yang tidak teratur. Terdapat beberapa tahapan dalam prapemrosesan data atau data preprocessing yang umumnya dilakukan dalam data mining, tahapan-tahapan tersebut adalah integrasi data, pembersihan data, transformasi data dan seleksi data.

5.3.1 Transformasi Data

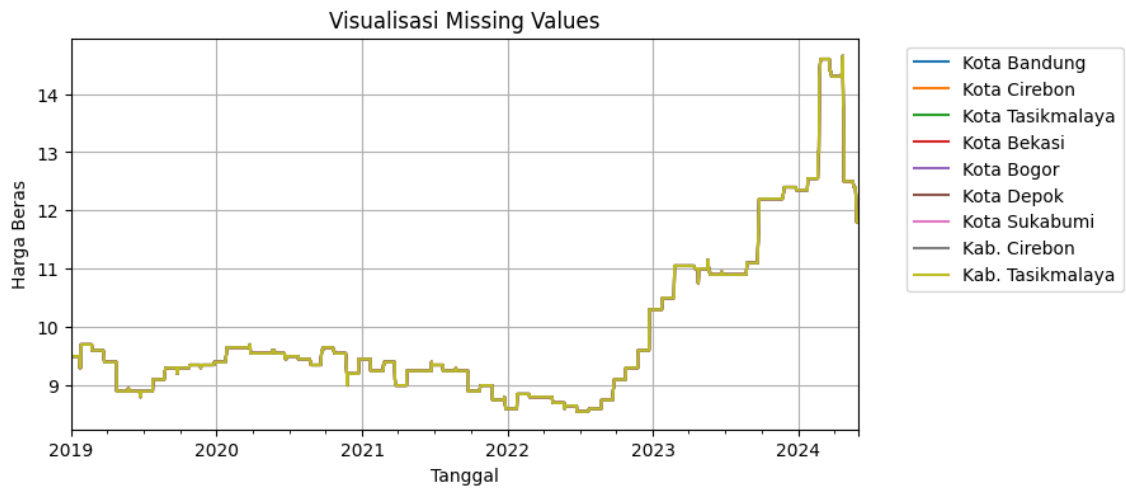
Tabel 7 merupakan transformasi data yang akan digunakan pada penelitian ini adalah normalisasi menggunakan min-max scaler untuk menyesuaikan skala pada data menjadi rentang yang sama, min-max scaler dilakukan agar rentang pada data tidak terlalu besar maupun tidak terlalu kecil

Tabel 8. Transformasi Data

Tanggal	Sebelum Harga Beras Di Transformasi	Sesudah Harga Beras Di Transformasi
01/01/2020	9400	9.491.364
02/01/2020	9400	9.490.711
03/01/2020	9400	9.490.356
04/01/2020	9400	9.491.161
05/01/2020	9400	9.490.483
06/01/2020	9400	9.490.369
07/01/2020	9400	9.902.725
08/01/2020	9400	9.490.626
09/01/2020	9400	9.490.291
10/01/2020	9400	9.490.291
....
....

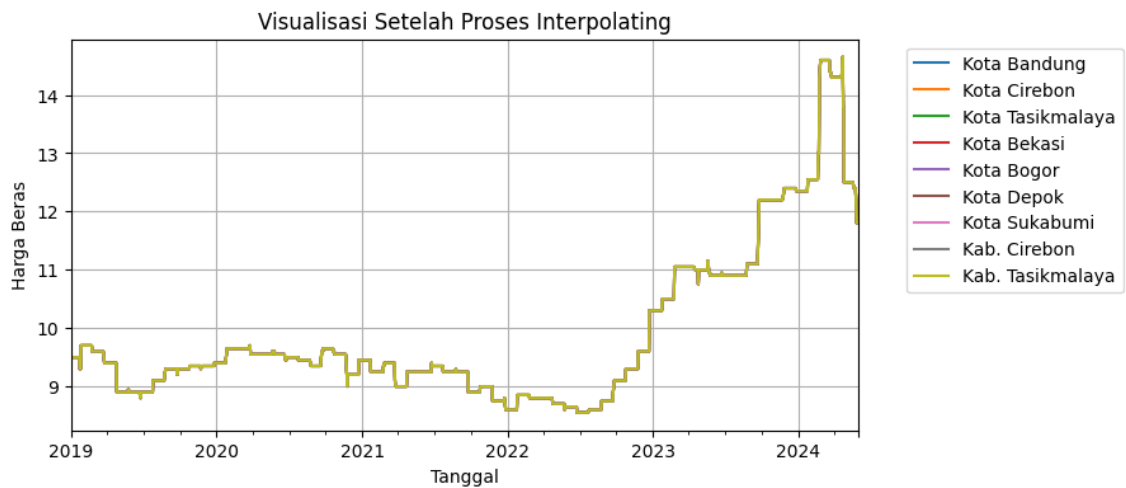
5.3.2 Data Cleaning

Gambar 9 merupakan grafik data harga beras yang awalnya ada kekosongan data yang digambarkan dengan garis yang putus (sebelum di *cleaning*).



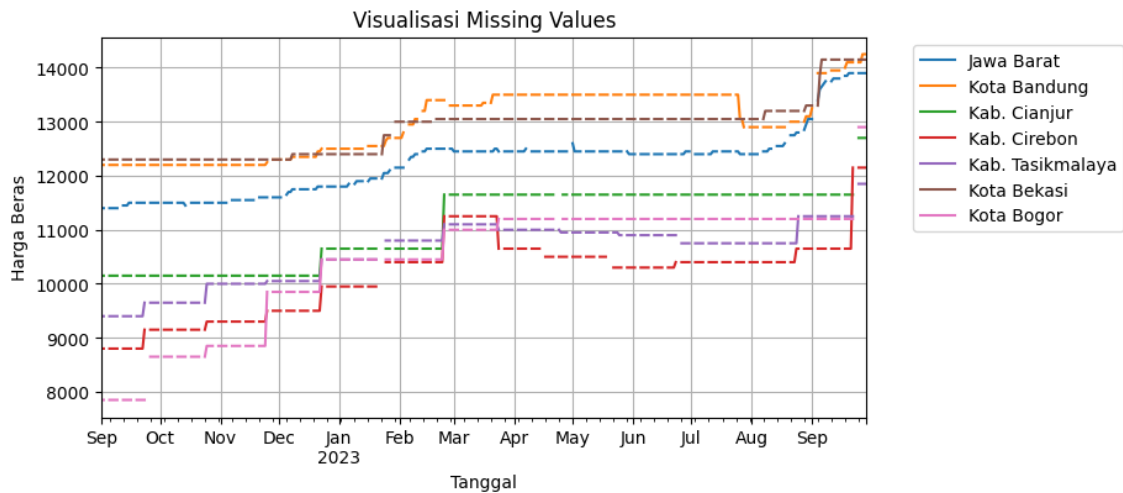
Gambar 8. Sebelum data harga beras di *cleaning*

Gambar 10 merupakan grafik data harga beras yang sudah di *cleaning* garisnya menjadi terhubung yang berarti kekosongan datanya berhasil terisi (sesudah data di *cleaning*).



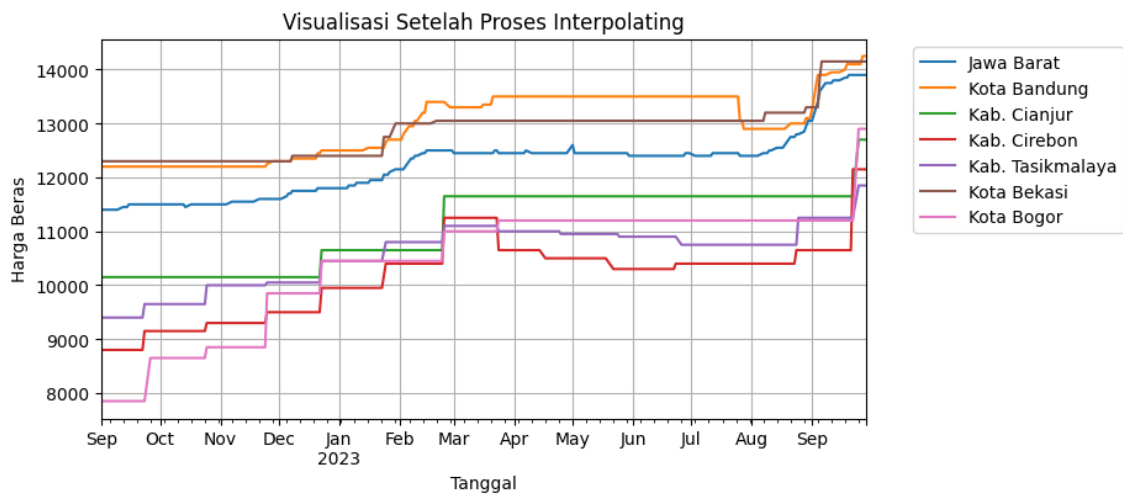
Gambar 9. Sesudah data harga beras di *cleaning*

Dari data yang diunduh dari situs PIHPS Nasional, masih terdapat data yang hilang terutama data pada hari libur dan juga data tidak valid yang telah dibersihkan di tahap sebelumnya, oleh karena itu data-data tersebut dianggap *missing value* seperti yang ditunjukkan pada Gambar 11.



Gambar 10. *Visualisasi Missing Values*

Missing value adalah data yang hilang maupun tidak tercatat dalam *dataset*, data yang hilang dapat mempengaruhi hasil analisis dan prediksi, sehingga perlu dilakukan pengisian data yang hilang dengan menggunakan metode interpolasi. Interpolasi adalah teknik yang digunakan untuk memperkirakan dan mengisi nilai yang hilang diantara rentang titik data yang diketahui berdasarkan pola atau tren yang terdapat pada data tersebut. Tampilan data sesudah proses interpolasi ditunjukkan pada Gambar 12.



Gambar 11. *Visualisasi Setelah Proses Interpolasi*

5.4 Data Mining

Pada tahapan ini dilakukan *validation* yakni proses membagi dataset menjadi beberapa bagian yang mana 1 diantara bagian lainnya menjadi data *testing*, dan yang lainnya menjadi data *Training*. Kemudian dimasukkan kedalam algoritma yaitu LSTM.

Penelitian ini menggunakan 1 input layer dan 1 dense layer dengan jumlah neuron 100, *optimizer* yang digunakan adalah *adaptive moment estimation* (Adam), *batch size* yang digunakan sebanyak 32 dan 64, dengan epoch sebanyak 100, 250 dan 500. Ditunjukkan pada tabel 9 yang merupakan pembagian data *training* dan data *testing* yang dipakai dalam proses data *minning* dalam penelitian ini.

Tabel 9. Pembagian Data

No	Data Training	Data Testing
1	90%	10%
2	80%	20%
3	70%	30%
4	60%	40%

5.5 Evaluasi Data

Berikut ini merupakan hasil pengujian dari metode LSTM yang telah dievaluasi menggunakan MAPE dan RMSE dengan empat kali pengujian. Ditunjukkan pada tabel 10 sampai 13.

Tabel 10. Hasil Evaluasi Data Dengan pembagian data 90% Training 10% Testing

	MAPE LSTM	RMSE LSTM
Kota Bandung	0,02%	0,15%
Kota Cirebon	0,02%	0,15%
Kota Tasikmalaya	0,03%	0,15%
Kota Bekasi	0,03%	0,15%
Kota Bogor	0,02%	0,15%
Kota Depok	0,03%	0,15%
Kota Sukabumi	0,02%	0,15%
Kab. Cirebon	0,02%	0,15%
Kab. Tasikmalaya	0,02%	0,15%

Tabel 10 merupakan hasil perbandingan dari pengujian yang sudah dievaluasi dengan RMSE dan MAPE dengan pembagian data 90% data Training dan 10% data testing.

Tabel 11. Hasil Evaluasi Data Dengan pembagian data 80% Training 20% Testing

	MAPE LSTM	RMSE LSTM
Kota Bandung	0,02%	0,11%
Kota Cirebon	0,02%	0,11%
Kota Tasikmalaya	0,02%	0,11%
Kota Bekasi	0,02%	0,11%
Kota Bogor	0,02%	0,11%
Kota Depok	0,02%	0,11%
Kota Sukabumi	0,02%	0,11%
Kab. Cirebon	0,02%	0,11%
Kab. Tasikmalaya	0,02%	0,11%

Tabel 11 merupakan hasil perbandingan dari pengujian yang sudah dievaluasi dengan RMSE dan MAPE dengan pembagian data 80% data Training dan 20% data testing,

Tabel 12. Hasil Evaluasi Data Dengan pembagian data 70% Training 30% Testing

	MAPE LSTM	RMSE LSTM
Kota Bandung	0,03%	0,10%
Kota Cirebon	0,02%	0,10%
Kota Tasikmalaya	0,03%	0,10%
Kota Bekasi	0,03%	0,11%
Kota Bogor	0,02%	0,10%
Kota Depok	0,01%	0,10%
Kota Sukabumi	0,02%	0,11%
Kab. Cirebon	0,03%	0,10%
Kab. Tasikmalaya	0,03%	0,11%

Tabel 12 merupakan hasil perbandingan dari pengujian yang sudah dievaluasi dengan RMSE dan MAPE dengan pembagian data 70% data Training dan 30% data testing,

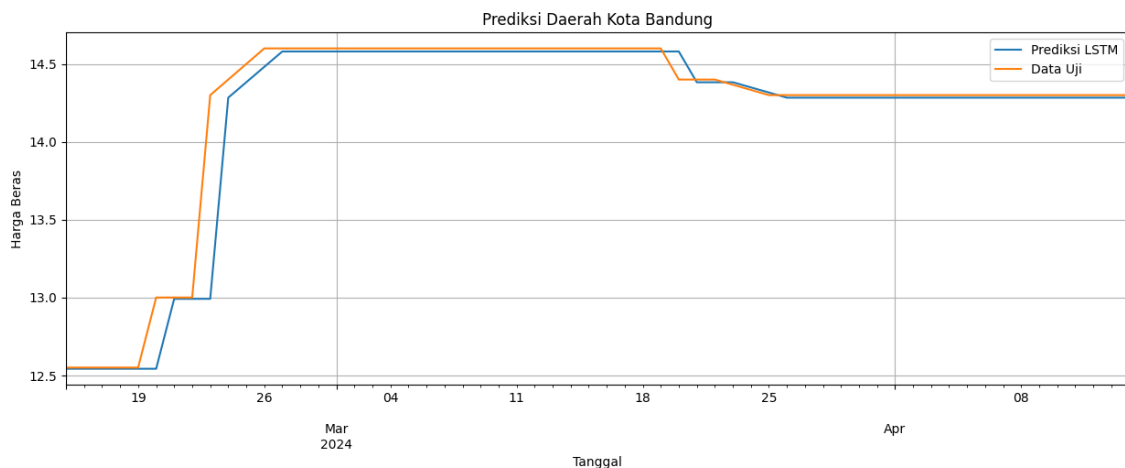
Tabel 13. Hasil Evaluasi Data Dengan pembagian data 60% Training 40% Testing

	MAPE LSTM	RMSE LSTM
Kota Bandung	0,05%	0,13%
Kota Cirebon	0,05%	0,12%
Kota Tasikmalaya	0,06%	0,14%
Kota Bekasi	0,07%	0,15%
Kota Bogor	0,06%	0,14%
Kota Depok	0,06%	0,15%
Kota Sukabumi	0,04%	0,11%
Kab. Cirebon	0,05%	0,12%
Kab. Tasikmalaya	0,06%	0,15%

Tabel 13 merupakan hasil perbandingan dari pengujian yang sudah dievaluasi dengan RMSE dan MAPE dengan pembagian data 60% data Training dan 40% data testing,

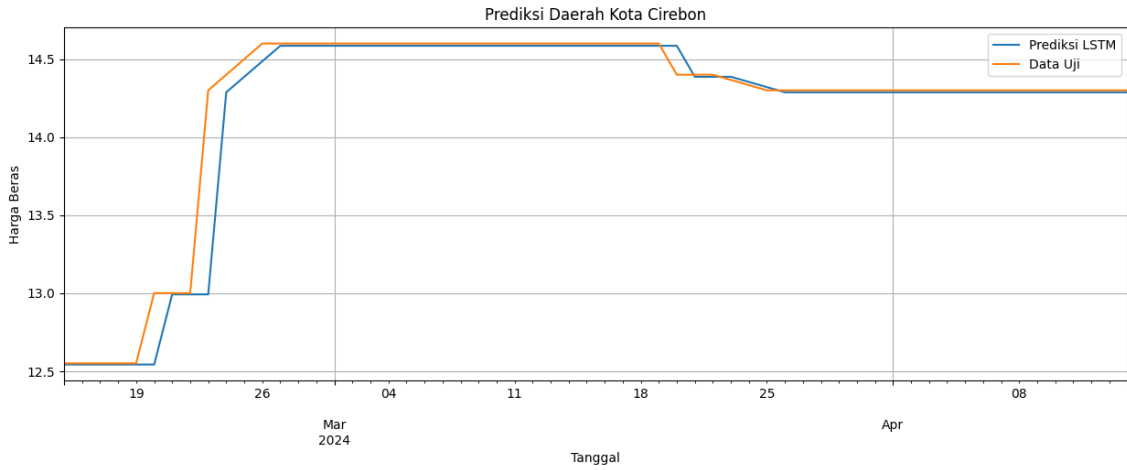
5.6 Presentase Pengetahuan

Prediksi dilakukan pada sembilan kota dan kabupaten yang tersebar di Jawa Barat menggunakan menggunakan model terbaik yang didapatkan pada tahap sebelumnya Hasil Prediksi tersebut ditampilkan dalam plot grafik diagram garis. hingga Gambar 14. Berikut ini merupakan grafik hasil pengujian dari metode LSTM, dan Data Uji yang telah dievaluasi menggunakan MAPE dan RMSE dengan empat kali pengujian dengan pembagian data Training 90%, 80%, 70%, 60 dan Testing 10%, 20%, 30%, 40% yang ditunjukkan pada Gambar 13 sampai 20.



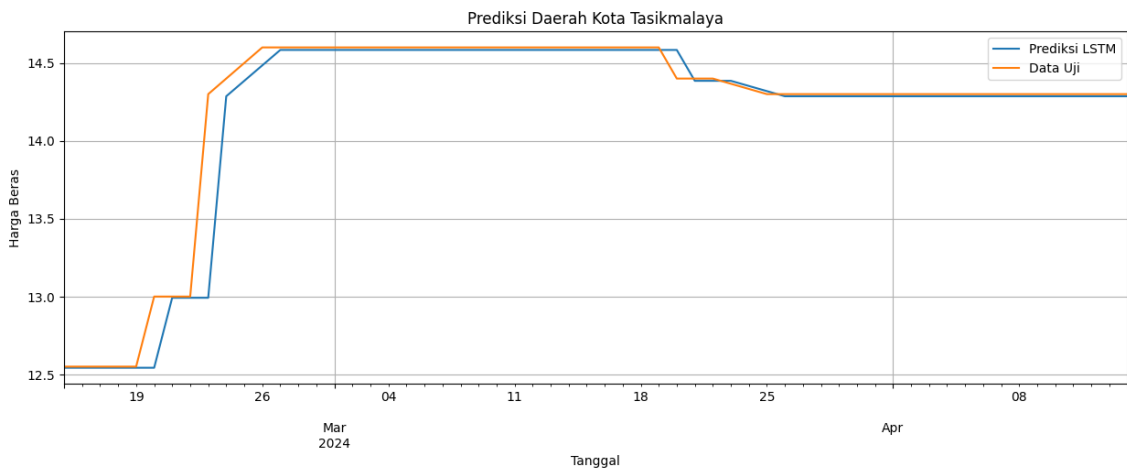
Gambar 12. Prediksi Harga Beras Daerah Kota Bandung

Gambar 13 adalah grafik prediksi Kota Bandung menggunakan metode LSTM yang ditandai dengan garis biru, dan yang ditandai dengan garis *orange* adalah data ujinya.



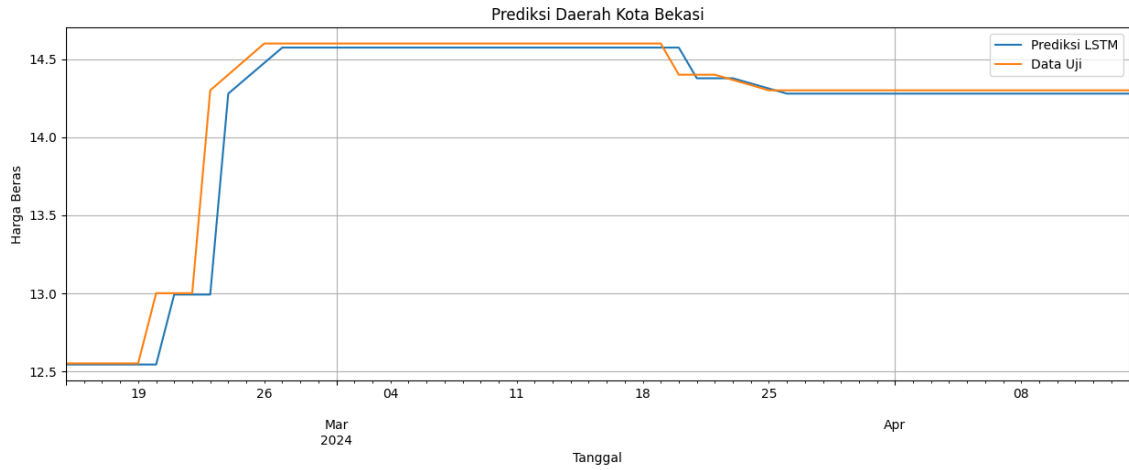
Gambar 13. Prediksi Harga Beras Daerah Kota Cirebon

Gambar 14 adalah grafik prediksi Kota Cirebon menggunakan metode LSTM yang ditandai dengan garis biru, dan yang ditandai dengan garis *orange* adalah data ujinya.



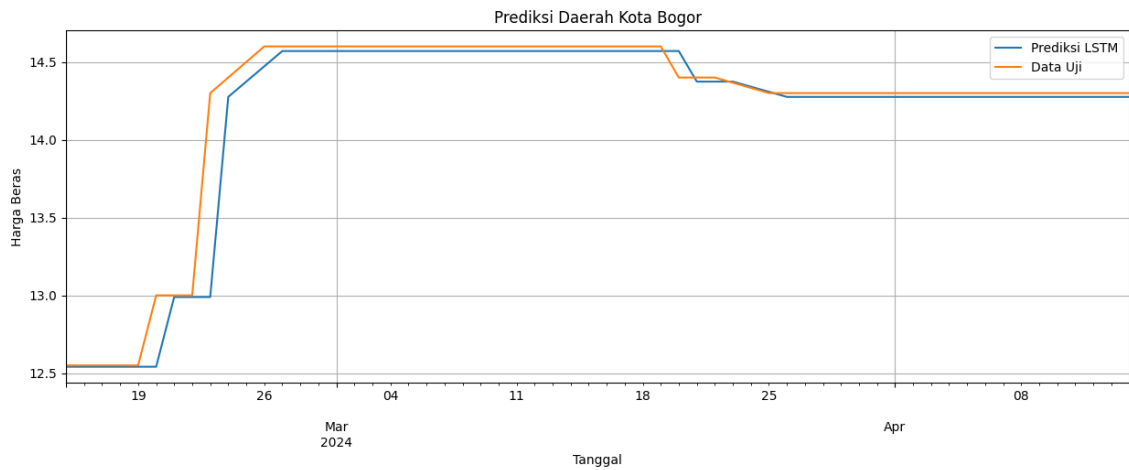
Gambar 14. Prediksi Harga Beras Daerah Kota Tasikmalaya

Gambar 15 adalah grafik prediksi Kota Tasikmalaya menggunakan metode LSTM yang ditandai dengan garis biru, dan yang ditandai dengan garis *orange* adalah data ujinya.



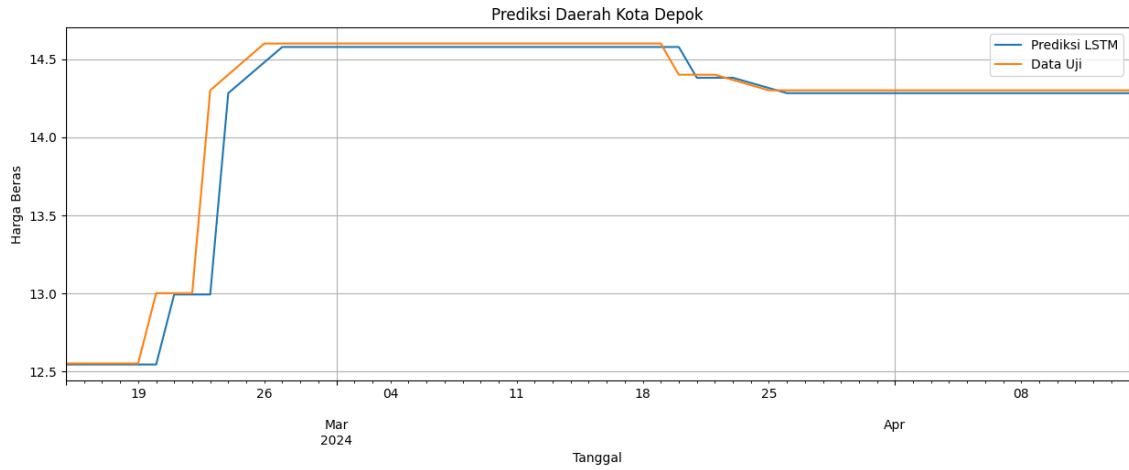
Gambar 15. Prediksi Harga Beras Daerah Kota Bekasi

Gambar 16 adalah grafik prediksi Kota Bekasi menggunakan metode LSTM yang ditandai dengan garis biru, dan yang ditandai dengan garis *orange* adalah data ujinya.



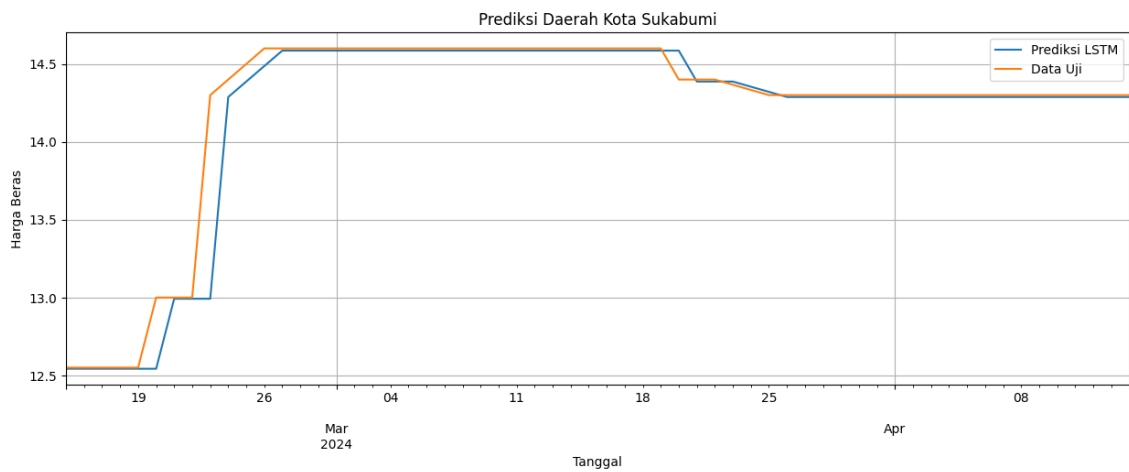
Gambar 16. Prediksi Harga Beras Daerah Kota Bogor

Gambar 17 adalah grafik prediksi Kota Bogor menggunakan metode LSTM yang ditandai dengan garis biru, dan yang ditandai dengan garis *orange* adalah data ujinya.



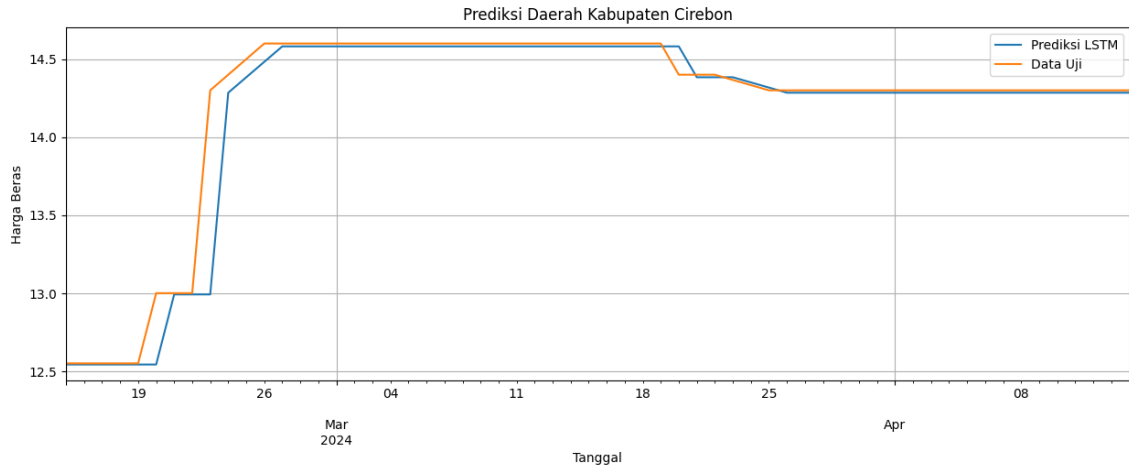
Gambar 17. Prediksi Harga Beras Daerah Kota Depok

Gambar 18 adalah grafik prediksi Kota Depok menggunakan metode LSTM yang ditandai dengan garis biru, dan yang ditandai dengan garis *orange* adalah data ujinya.



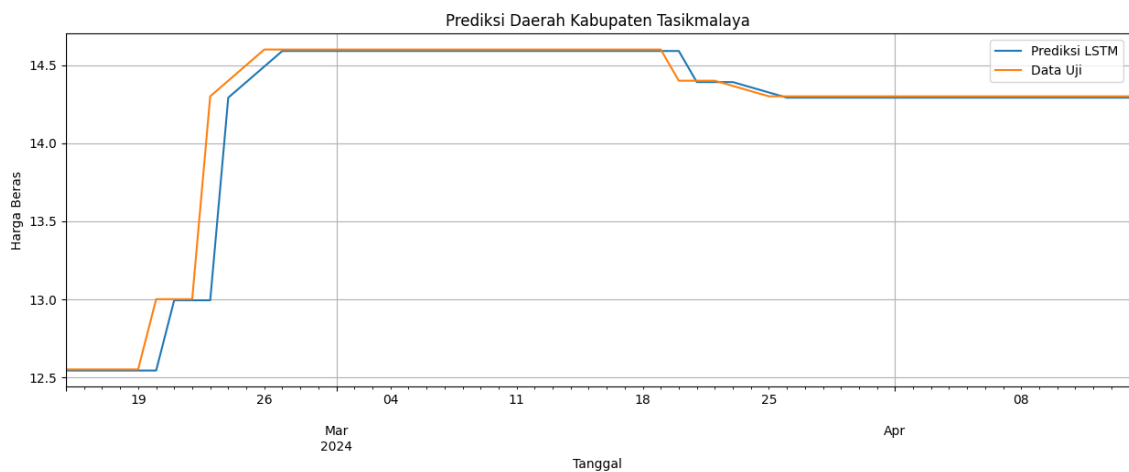
Gambar 18. Prediksi Harga Beras Daerah Kota Sukabumi

Gambar 19 adalah grafik prediksi Kota Sukabumi menggunakan metode LSTM yang ditandai dengan garis biru, dan yang ditandai dengan garis *orange* adalah data ujinya.



Gambar 19. Prediksi Harga Beras Daerah Kabupaten Cirebon

Gambar 20 adalah grafik prediksi Kabupaten Cirebon menggunakan metode LSTM yang ditandai dengan garis biru, dan yang ditandai dengan garis *orange* adalah data ujinya.



Gambar 20. Prediksi Harga Beras Daerah Kabupaten Tasikmalaya

Gambar 21 adalah grafik prediksi Kabupaten Tasikmalaya menggunakan metode LSTM yang ditandai dengan garis biru, dan yang ditandai dengan garis *orange* adalah data ujinya.

BAB VI

KESIMPULAN DAN SARAN

6.1 Kesimpulan

Berdasarkan pembahasan dan implementasi serta hasil pengujian dari penerapan *Long Short-Term Memory* (LSTM) untuk memprediksi harga beras di Jawa Barat, menunjukkan hasil prediksi yang diperoleh tergolong baik. Berikut pembahasan tentang penelitian yang telah dilakukan, Penelitian menggunakan data beras harian dari Pusat Informasi Harga Pangan Strategis (PIHPS) Nasional pada sembilan kota dan kabupaten yang tersebar di Jawa Barat, data berjumlah 1414 dari bulan Januari 2019 hingga bulan Februari 2024, Dan setelah dilakukan perbandingan metrik evaluasi dari kedua metode menggunakan RMSE dan MAPE, dapat disimpulkan bahwa model menggunakan metode LSTM dengan MAPE dan RMSE sebesar 0.02% dan 0.153073, Model dengan pembagian data latih sebesar 90% dan data uji sebesar 10% menggunakan parameter yang meliputi *batch size* sebanyak 64 dan *epoch* sebanyak 250

6.2 Saran

Setelah mengetahui hasil akhir dari penelitian ini, ada beberapa hal yang dapat dikembangkan peneliti lain di masa yang akan datang, diantaranya:

1. Arsitektur dan parameter yang digunakan pada penelitian ini dapat diubah atau ditambah untuk mendapatkan model yang lebih optimal.
2. Jumlah dan rentang data dapat ditambah untuk mendapatkan nilai prediksi yang lebih baik.

DAFTAR PUSTAKA

- Adhib Arfan dan Lussiana. (2020).** Perbandingan Algoritma Long Short-Term Memory dengan SVR Pada Prediksi Harga Saham di Indonesia. *PETIR*, 13(1), 33–43. <https://doi.org/10.33322/petir.v13i1.858>
- Azzahra Zulfa Riswinda. (2023).** *Klasifikasi penilaian produk nutrimax food supplement menggunakan metode hybrid convolutional neural network (cnn)-long short term memory (lstm).*
- Elsa Paskalis Krisda Orpa, E. F. R. dan T. (2019).** *Model Prediksi Awal Masa Studi Mahasiswa.*
- Lingga Yuliana. (2019).** Analisis perencanaan penjualan dengan metode time series (studi kasus pada pd. Sumber jaya aluminium) lingga yuliana stie media nusantara citra informasi artikel abstrak *Jurnal Mitra Manajemen (JMM Online).*
- Mohammad Kaffil. (2019).** Penerapan Metode K-Nearest Neighbors Untuk Prediksi Penjualan Berbasis Web Pada Boutiq Dealove Bondowoso. In *Jurnal Mahasiswa Teknik Informatika* (Vol. 3, Issue 2).
- Muhammad Ikhsan dan Sri Ngudi Wahyuni. (2023).** Analisis Sentimen Terhadap Kenaikan Harga Bahan Bakar Minyak Menggunakan Long Short-Term Memory. *IJCSR: The Indonesian Journal of Computer Science Research*, 1(1). <https://doi.org/10.37905>
- Rini sugiarsih duki saputri. (2019).** Pengaruh kualitas pelayanan dan harga terhadap loyalitas pelanggan grab semarang rini sugiarsih duki saputri.
- Rizki Mugi Setya Adi dan Sudianto. (2022).** Prediksi Harga Komoditas Pangan Menggunakan Algoritma Long Short-Term Memory (LSTM). *Building of Informatics, Technology and Science (BITS)*, 4(2). <https://doi.org/10.47065/bits.v4i2.2229>
- Roby Julian dan Muhammad Rizky Pribadi. (2021).** Peramalan Harga Saham Pertambangan Pada Bursa Efek Indonesia (BEI) Menggunakan Long Short Term Memory (LSTM). *Jurnal Teknik Informatika Dan Sistem Informasi*, 8(3). <http://jurnal.mdp.ac.id>
- Rona Guines Purnasiwi, K. dan M. H. (2023).** *Analisis Sentimen Pada Review Produk Skincare Menggunakan Word Embedding.*
- Sahrina Hasibuan, L., & Novialdi, Y. (2023).** *Prediction of Bulk and Packaged Cooking Oil Prices Using the Long Short-Term Memory (LSTM) Algorithm.* <https://jurnal.ipb.ac.id/index.php/jika>
- Siti Aminah, I. M. & A. R. (2019).** *Analisis Kandungan Klorin pada Beras yang Beredar Di Pasar Tradisional Makassar Dengan Metode Argentometri Volhard.*

Talita, A. S., & Wiguna, A. (2019). Implementasi Algoritma Long Short-Term Memory (LSTM) Untuk Mendeteksi Ujaran Kebencian (Hate Speech) Pada Kasus Pilpres 2019. *MATRIK : Jurnal Manajemen, Teknik Informatika Dan Rekayasa Komputer*, 19(1), 37–44. <https://doi.org/10.30812/matrik.v19i1.495>





Zahara, S. & B. I. (2019). Terakreditasi SINTA Peringkat 2 Prediksi Indeks Harga Konsumen Menggunakan Metode Long Short Term Memory (LSTM) Berbasis Cloud Computing. *1(3)*, 357–363.

LAMPIRAN

Lampiran 1. Kartu Bimbingan Laporan Proposal

**Kartu Bimbingan Mahasiswa
Program Studi Ilmu Komputer FMIPA – UNPAK**

Nama Mahasiswa : Muhamad Yuda
NPM : 065119014
Judul : Prediksi Harga Beras Menggunakan *Long Short-Term Memory (LSTM)*.
Pembimbing I : Dr. Herfina.,M.Pd.,M.Kom.
Pembimbing II : Dr. Fajar Delli, M.Kom.

No	Hari, Tanggal	Catatan	Tanda Tangan	
			Pembimbing.I	Pembimbing.II
1	Selasa, 27 Juni 2023	Memperbaiki Laporan Proposal Penelitian sesuai arahan		
2	Senin, 3 Juli 2023	Menambahkan pada bab III <i>flowchart</i> dan contoh studi kasus beserta perhitungannya		
3	Rabu, 5 Juli 2023	Acc oleh dosen pembimbing utama		
4.	Kamis. 6 Juli 2023	Memperbaiki Laporan Proposal Penelitian sesuai arahan dari dosen pembimbing pendamping		

Bogor, Juni 2023
Program Studi Ilmu Komputer
Fakultas MIPA-UNPAK
Ketua




Arie Qur'ania, M. Kom

Lampiran 2. Kartu Bimbingan Laporan Hasil Penelitian

**Kartu Bimbingan Mahasiswa
Program Studi Ilmu Komputer FMIPA – UNPAK**

Nama Mahasiswa : Muhamad Yuda
NPM : 065119014
Judul : Prediksi Harga Beras Menggunakan *Long Short-Term Memory (LSTM)*.
Pembimbing I : Dr. Herfina.,M.Pd.,M.Kom.
Pembimbing II : Dr. Fajar Delli, M.Kom.

No	Hari, Tanggal	Catatan	Tanda Tangan	
			Pembimbing.I	Pembimbing.II
1	Jum'at, 29 Desember 2023	Memperbaiki Laporan Hasil sesuai arahan		

Bogor, Desember 2023
Program Studi Ilmu Komputer
Fakultas MIPA-UNPAK
Ketua




Arie Qur'ania, M. Kom

Lampiran 3. Kartu Bimbingan Laporan Skripsi

Kartu Bimbingan Mahasiswa
Program Studi Ilmu komputer FMIPA – UNPAK

Nama Mahasiswa : Muhamad Yuda
NPM : 065119014
Judul : Prediksi Harga Beras Menggunakan *Long Short-Term Memory (LSTM)*.
Pembimbing I : Dr. Herfina.,M.Pd.,M.Kom.
Pembimbing II : Dr. Fajar Delli, M.Kom.

No	Hari, Tanggal	Catatan	Tanda Tangan	
			Pembimbing.I	Pembimbing.II
1	Jum'at, 06 Juni 2024	Memperbaiki Laporan Hasil sesuai arahan		

Bogor, Juni 2024
Program Studi Ilmu Komputer
Fakultas MIPA-UNPAK
Ketua



Arie Qur'ania, M. Kom

Lampiran 4. *Source Code*

```
from google.colab import files
from sklearn.preprocessing import MinMaxScaler
import tensorflow as tf
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import glob
import os
import io
import copy

# Upload Files Harga Beras ke Google Collab
uploaded = files.upload()

# Proses Integrasi
dfs = []
for filename in uploaded.keys():
    df = pd.read_excel(filename)
    df = df.iloc[:, 1:]
    df_transposed = df.transpose()

# Reset index of transposed dataframe
df_transposed = df_transposed.reset_index()

# Rename columns of transposed dataframe
df_transposed.columns = df_transposed.iloc[0]
```

```

# Drop first row of transposed dataframe
df_transposed = df_transposed.drop(0)

# Reset index of transposed dataframe again
df_transposed = df_transposed.reset_index(drop=True)

df_transposed = df_transposed.rename(columns={'Komoditas (Rp)': 'Date Time'})
df_transposed['Date Time'] = pd.to_datetime(df_transposed['Date Time'], format='%d/
%m/ %Y')
df_transposed = df_transposed.replace(',', '', regex=True)
dfs.append(df_transposed)

df = pd.concat(dfs, ignore_index=True)
df = df.drop('Semua Provinsi', axis=1)
df = df.sort_values(by='Date Time')
df = df.reset_index(drop=True)
print(df.to_string())

df = df.set_index('Date Time')
df = df.resample('D').asfreq()
df[df.select_dtypes(include='object').columns] =
df.select_dtypes(include='object').apply(pd.to_numeric, errors='coerce')

# Visualisasi Missing Values
df.plot(kind='line', title='Visualisasi Missing Values', xlabel='Tanggal', ylabel='Harga
Beras', legend=True, grid=True, figsize=(8, 4))
plt.legend(bbox_to_anchor=(1.05, 1), loc='upper left')
# Show the plot
plt.show()

```

```

# Proses transformasi kolom dari string menjadi integer agar bisa diproses

# df = df[df.select_dtypes(include='object').apply(lambda x: x.str.contains('-')).sum(axis=1) == 0]

# df[df.select_dtypes(include='object').columns] = df.select_dtypes(include='object').astype(int)

df = df.resample('D').asfreq()
df.interpolate(method='linear', inplace=True)

df.plot(kind='line', title='Visualisasi Setelah Proses Interpolating', xlabel='Tanggal',
ylabel='Harga Beras', legend=True, grid=True, figsize=(8, 4))
plt.legend(bbox_to_anchor=(1.05, 1), loc='upper left')
plt.show()

# Simpan tanggal untuk proses visualisasi setelah prediksi
tanggal = df.index.copy()
tanggal = tanggal.to_numpy()

df.reset_index(inplace=True)
df.drop(['Date Time'], axis=1, inplace=True)
# dataset = df.values
# dataset = dataset.astype('float32')

datasetJawaBarat = df['Jawa Barat'].values.astype('float32')
datasetKotaBandung = df['Kota Bandung'].values.astype('float32')
datasetKabCianjur = df['Kab. Cianjur'].values.astype('float32')
datasetKabCirebon = df['Kab. Cirebon'].values.astype('float32')
datasetKabTasikmalaya = df['Kab. Tasikmalaya'].values.astype('float32')

```

```

datasetKotaBekasi = df['Kota Bekasi'].values.astype('float32')
datasetKotaBogor = df['Kota Bogor'].values.astype('float32')

# Scaling data menjadi rentang antara 0 dan 1
scaler = MinMaxScaler(feature_range=(0, 1))
datasetJawaBarat = scaler.fit_transform(datasetJawaBarat.reshape(-1, 1))
datasetKotaBandung = scaler.fit_transform(datasetKotaBandung.reshape(-1, 1))
datasetKabCianjur = scaler.fit_transform(datasetKabCianjur.reshape(-1, 1))
datasetKabCirebon = scaler.fit_transform(datasetKabCirebon.reshape(-1, 1))
datasetKabTasikmalaya = scaler.fit_transform(datasetKabTasikmalaya.reshape(-1, 1))
datasetKotaBekasi = scaler.fit_transform(datasetKotaBekasi.reshape(-1, 1))
datasetKotaBogor = scaler.fit_transform(datasetKotaBogor.reshape(-1, 1))

print(datasetKabCianjur)

# Membuat fungsinya untuk membuat data dalam bentuk array 2D (sample, timestep)
def create_dataset(dataset, look_back=1):
    dataX, dataY = [], []
    for i in range(len(dataset) - look_back - 1):
        a = dataset[i:(i + look_back), 0]
        dataX.append(a)
        dataY.append(dataset[i + look_back, 0])
    return np.array(dataX), np.array(dataY)

tanggal = tanggal.reshape(-1, 1)

# Membagi data menjadi training dan testing
train_size = int(len(datasetJawaBarat) * 0.9) # Length setiap dataset sama, jadi hanya
perlu memilih salah satu saja

```

```
test_size = len(datasetJawaBarat) - train_size
```

```
tanggalTrain, tanggalTest = tanggal[0:train_size,:], tanggal[train_size:len(tanggal),:]
```

```
trainJawaBarat, testJawaBarat = datasetJawaBarat[0:train_size:],  
datasetJawaBarat[train_size:len(datasetJawaBarat),:]
```

```
trainKotaBandung, testKotaBandung = datasetKotaBandung[0:train_size:],  
datasetKotaBandung[train_size:len(datasetKotaBandung),:]
```

```
trainKabCianjur, testKabCianjur = datasetKabCianjur[0:train_size:],  
datasetKabCianjur[train_size:len(datasetKabCianjur),:]
```

```
trainKabCirebon, testKabCirebon = datasetKabCirebon[0:train_size:],  
datasetKabCirebon[train_size:len(datasetKabCirebon),:]
```

```
trainKotaBekasi, testKotaBekasi = datasetKotaBekasi[0:train_size:],  
datasetKotaBekasi[train_size:len(datasetKotaBekasi),:]
```

```
trainKabTasikmalaya, testKabTasikmalaya = datasetKabTasikmalaya[0:train_size:],  
datasetKabTasikmalaya[train_size:len(datasetKabTasikmalaya),:]
```

```
trainKotaBekasi, testKotaBekasi = datasetKotaBekasi[0:train_size:],  
datasetKotaBekasi[train_size:len(datasetKotaBekasi),:]
```

```
trainKotaBogor, testKotaBogor = datasetKotaBogor[0:train_size:],  
datasetKotaBogor[train_size:len(datasetKotaBogor),:]
```

```
# Membuat fungsinya untuk membuat data dalam bentuk array 2D (sample, timestep)
```

```
def create_dataset(dataset, look_back=1):
```

```
    dataX, dataY = [], []
```

```
    for i in range(len(dataset) - look_back - 1):
```

```
        a = dataset[i:(i + look_back), 0]
```

```
        dataX.append(a)
```

```
        dataY.append(dataset[i + look_back, 0])
```

```
    return np.array(dataX), np.array(dataY)
```

```
# Menentukan jumlah timestep (look_back)
```

```
look_back = 1
```

```
trainX_JawaBarat, trainY_JawaBarat = create_dataset(trainJawaBarat, look_back)
```

```
testX_JawaBarat, testY_JawaBarat = create_dataset(testJawaBarat, look_back)
```

```
trainX_KotaBandung, trainY_KotaBandung = create_dataset(trainKotaBandung,  
look_back)
```

```
testX_KotaBandung, testY_KotaBandung = create_dataset(testKotaBandung,  
look_back)
```

```
trainX_KabCianjur, trainY_KabCianjur = create_dataset(trainKabCianjur, look_back)
```

```
testX_KabCianjur, testY_KabCianjur = create_dataset(testKabCianjur, look_back)
```

```
trainX_KabCirebon, trainY_KabCirebon = create_dataset(trainKabCirebon, look_back)
```

```
testX_KabCirebon, testY_KabCirebon = create_dataset(testKabCirebon, look_back)
```

```
trainX_KabTasikmalaya, trainY_KabTasikmalaya = create_dataset(trainKabTasikmalaya, look_back)
```

```
testX_KabTasikmalaya, testY_KabTasikmalaya = create_dataset(testKabTasikmalaya,  
look_back)
```

```
trainX_KotaBekasi, trainY_KotaBekasi = create_dataset(trainKotaBekasi, look_back)
```

```
testX_KotaBekasi, testY_KotaBekasi = create_dataset(testKotaBekasi, look_back)
```

```
trainX_KotaBogor, trainY_KotaBogor = create_dataset(trainKotaBogor, look_back)
```

```
testX_KotaBogor, testY_KotaBogor = create_dataset(testKotaBogor, look_back)
```

```
from keras.models import Sequential
```

```
from keras.layers import LSTM, Dense
```



```
# Membuat model LSTM
modelJawaBarat = Sequential()
modelJawaBarat.add(LSTM(100, input_shape=(look_back, 1)))
modelJawaBarat.add(Dense(1))

modelKotaBandung = Sequential()
modelKotaBandung.add(LSTM(100, input_shape=(look_back, 1)))
modelKotaBandung.add(Dense(1))

modelKabCianjur = Sequential()
modelKabCianjur.add(LSTM(100, input_shape=(look_back, 1)))
modelKabCianjur.add(Dense(1))

modelKabCirebon = Sequential()
modelKabCirebon.add(LSTM(100, input_shape=(look_back, 1)))
modelKabCirebon.add(Dense(1))

modelKabTasikmalaya = Sequential()
modelKabTasikmalaya.add(LSTM(100, input_shape=(look_back, 1)))
modelKabTasikmalaya.add(Dense(1))

modelKotaBekasi = Sequential()
modelKotaBekasi.add(LSTM(100, input_shape=(look_back, 1)))
modelKotaBekasi.add(Dense(1))

modelKotaBogor = Sequential()
modelKotaBogor.add(LSTM(100, input_shape=(look_back, 1)))
modelKotaBogor.add(Dense(1))
```

```

# Compile model menggunakan optimisasi ADAM dan fungsi kerugian Mean Squared
Error
modelJawaBarat.compile(loss='mean_squared_error', optimizer='adam')
modelKotaBandung.compile(loss='mean_squared_error', optimizer='adam')
modelKabCianjur.compile(loss='mean_squared_error', optimizer='adam')
modelKabCirebon.compile(loss='mean_squared_error', optimizer='adam')
modelKabTasikmalaya.compile(loss='mean_squared_error', optimizer='adam')
modelKotaBekasi.compile(loss='mean_squared_error', optimizer='adam')
modelKotaBogor.compile(loss='mean_squared_error', optimizer='adam')

# Reshape data menjadi bentuk 3D (sample, timestep, feature)
trainX_JawaBarat = np.reshape(trainX_JawaBarat, (trainX_JawaBarat.shape[0],
look_back, 1))
testX_JawaBarat = np.reshape(testX_JawaBarat, (testX_JawaBarat.shape[0], look_back,
1))

trainX_KotaBandung = np.reshape(trainX_KotaBandung,
(trainX_KotaBandung.shape[0], look_back, 1))
testX_KotaBandung = np.reshape(testX_KotaBandung, (testX_KotaBandung.shape[0],
look_back, 1))

trainX_KabCianjur = np.reshape(trainX_KabCianjur, (trainX_KabCianjur.shape[0],
look_back, 1))
testX_KabCianjur = np.reshape(testX_KabCianjur, (testX_KabCianjur.shape[0],
look_back, 1))

trainX_KabCirebon = np.reshape(trainX_KabCirebon, (trainX_KabCirebon.shape[0],
look_back, 1))
testX_KabCirebon = np.reshape(testX_KabCirebon, (testX_KabCirebon.shape[0],
look_back, 1))

```

```
trainX_KabTasikmalaya = np.reshape(trainX_KabTasikmalaya,
(trainX_KabTasikmalaya.shape[0], look_back, 1))
```

```
testX_KabTasikmalaya = np.reshape(testX_KabTasikmalaya,
(testX_KabTasikmalaya.shape[0], look_back, 1))
```

```
trainX_KotaBekasi = np.reshape(trainX_KotaBekasi, (trainX_KotaBekasi.shape[0],
look_back, 1))
```

```
testX_KotaBekasi = np.reshape(testX_KotaBekasi, (testX_KotaBekasi.shape[0],
look_back, 1))
```

```
trainX_KotaBogor = np.reshape(trainX_KotaBogor, (trainX_KotaBogor.shape[0],
look_back, 1))
```

```
testX_KotaBogor = np.reshape(testX_KotaBogor, (testX_KotaBogor.shape[0],
look_back, 1))
```

```
modelJawaBarat.fit(trainX_JawaBarat, trainY_JawaBarat, epochs=250, batch_size=64,
verbose=0)
```

```
modelKotaBandung.fit(trainX_KotaBandung, trainY_KotaBandung, epochs=250,
batch_size=64, verbose=0)
```

```
modelKabCianjur.fit(trainX_KabCianjur, trainY_KabCianjur, epochs=250,
batch_size=64, verbose=0)
```

```
modelKabCirebon.fit(trainX_KabCirebon, trainY_KabCirebon, epochs=250,
batch_size=64, verbose=0)
```

```
modelKabTasikmalaya.fit(trainX_KabTasikmalaya, trainY_KabTasikmalaya,
epochs=250, batch_size=64, verbose=0)
```

```
modelKotaBekasi.fit(trainX_KotaBekasi, trainY_KotaBekasi, epochs=250,
batch_size=64, verbose=0)
```

```
modelKotaBogor.fit(trainX_KotaBogor, trainY_KotaBogor, epochs=250,
batch_size=64, verbose=0)
```

```
modelKotaBandung.summary()
```

```
# Melakukan prediksi pada data testing
```

```

trainPredictJawaBarat = modelJawaBarat.predict(trainX_JawaBarat)
testPredictJawaBarat = modelJawaBarat.predict(testX_JawaBarat)
trainPredictKotaBandung = modelKotaBandung.predict(trainX_KotaBandung)
testPredictKotaBandung = modelKotaBandung.predict(testX_KotaBandung)
trainPredictKabCianjur = modelKabCianjur.predict(trainX_KabCianjur)
testPredictKabCianjur = modelKabCianjur.predict(testX_KabCianjur)
trainPredictKabCirebon = modelKabCirebon.predict(trainX_KabCirebon)
testPredictKabCirebon = modelKabCirebon.predict(testX_KabCirebon)
trainPredictKabTasikmalaya = modelKabTasikmalaya.predict(trainX_KabTasikmalaya)
testPredictKabTasikmalaya = modelKabTasikmalaya.predict(testX_KabTasikmalaya)
trainPredictKotaBekasi = modelKotaBekasi.predict(trainX_KotaBekasi)
testPredictKotaBekasi = modelKotaBekasi.predict(testX_KotaBekasi)
trainPredictKotaBogor = modelKotaBogor.predict(trainX_KotaBogor)
testPredictKotaBogor = modelKotaBogor.predict(testX_KotaBogor)

trainPredictScaledJawaBarat = scaler.inverse_transform(trainPredictJawaBarat)
trainYScaledJawaBarat = scaler.inverse_transform(trainY_JawaBarat.reshape(-1, 1))
testPredictScaledJawaBarat = scaler.inverse_transform(testPredictJawaBarat)
testYScaledJawaBarat = scaler.inverse_transform(testY_JawaBarat.reshape(-1, 1))

trainPredictScaledKotaBandung = scaler.inverse_transform(trainPredictKotaBandung)
trainYScaledKotaBandung = scaler.inverse_transform(trainY_KotaBandung.reshape(-1, 1))
testPredictScaledKotaBandung = scaler.inverse_transform(testPredictKotaBandung)
testYScaledKotaBandung = scaler.inverse_transform(testY_KotaBandung.reshape(-1, 1))

trainPredictScaledKabCianjur = scaler.inverse_transform(trainPredictKabCianjur)
trainYScaledKabCianjur = scaler.inverse_transform(trainY_KabCianjur.reshape(-1, 1))

```

```

testPredictScaledKabCianjur = scaler.inverse_transform(testPredictKabCianjur)
testYScaledKabCianjur = scaler.inverse_transform(testY_KabCianjur.reshape(-1, 1))

trainPredictScaledKabCirebon = scaler.inverse_transform(trainPredictKabCirebon)
trainYScaledKabCirebon = scaler.inverse_transform(trainY_KabCirebon.reshape(-1, 1))
testPredictScaledKabCirebon = scaler.inverse_transform(testPredictKabCirebon)
testYScaledKabCirebon = scaler.inverse_transform(testY_KabCirebon.reshape(-1, 1))

trainPredictScaledKabTasikmalaya = scaler.inverse_transform(trainPredictKabTasikmalaya)
trainYScaledKabTasikmalaya = scaler.inverse_transform(trainY_KabTasikmalaya.reshape(-1, 1))
testPredictScaledKabTasikmalaya = scaler.inverse_transform(testPredictKabTasikmalaya)
testYScaledKabTasikmalaya = scaler.inverse_transform(testY_KabTasikmalaya.reshape(-1, 1))

trainPredictScaledKotaBekasi = scaler.inverse_transform(trainPredictKotaBekasi)
trainYScaledKotaBekasi = scaler.inverse_transform(trainY_KotaBekasi.reshape(-1, 1))
testPredictScaledKotaBekasi = scaler.inverse_transform(testPredictKotaBekasi)
testYScaledKotaBekasi = scaler.inverse_transform(testY_KotaBekasi.reshape(-1, 1))

trainPredictScaledKotaBogor = scaler.inverse_transform(trainPredictKotaBogor)
trainYScaledKotaBogor = scaler.inverse_transform(trainY_KotaBogor.reshape(-1, 1))
testPredictScaledKotaBogor = scaler.inverse_transform(testPredictKotaBogor)
testYScaledKotaBogor = scaler.inverse_transform(testY_KotaBogor.reshape(-1, 1))

tanggalTest = tanggalTest[1:-1]

```

```

dates = pd.to_datetime(tanggalTest)

def drawPlot(predict, actual, title):
    dataPlot = {'Hasil Prediksi': predict.flatten(), 'Data Asli': actual.flatten()}
    dfPlot = pd.DataFrame(dataPlot)
    dfPlot['Date Time'] = tanggalTest.flatten()
    dfPlot['Date Time'] = pd.to_datetime(dfPlot['Date Time'])
    dfPlot.set_index("Date Time", inplace=True)
    dfPlot.plot(kind='line', title='Prediksi Daerah ' + title, xlabel='Tanggal', ylabel='Harga
Beras', legend=True, grid=True, figsize=(8, 4))
    plt.show()

drawPlot(testPredictScaledJawaBarat, testYScaledJawaBarat, "Seluruh Jawa Barat")
drawPlot(testPredictScaledKotaBandung, testYScaledKotaBandung, "Kota Bandung")
drawPlot(testPredictScaledKabCianjur, testYScaledKabCianjur, "Kabupaten Cianjur")
drawPlot(testPredictScaledKabCirebon, testYScaledKabCirebon, "Kabupaten Cirebon")
drawPlot(testPredictScaledKabTasikmalaya, testYScaledKabTasikmalaya, "Kabupaten
Tasikmalaya")
drawPlot(testPredictScaledKotaBekasi, testYScaledKotaBekasi, "Kota Bekasi")
drawPlot(testPredictScaledKotaBogor, testYScaledKotaBogor, "Kota Bogor")

from sklearn.metrics import mean_squared_error
from sklearn.metrics import mean_absolute_percentage_error
from sklearn.metrics import mean_absolute_error
from sklearn.metrics import mean_squared_error

import numpy as np

def evaluasi(actual, prediction):

```

```

MSE = mean_squared_error(actual, prediction)
MAPE = mean_absolute_percentage_error(actual, prediction)
MAE = mean_absolute_error(actual, prediction)
RMSE = np.sqrt(MSE)
return {'MSE':MSE, 'MAPE':MAPE, 'MAE':MAE, 'RMSE': RMSE}

```

```

evaluasi = {
    # "Jawa Barat": {
    #     "MSE": evaluasi(testPredictScaledJawaBarat, testYScaledJawaBarat)['MSE'],
    #     "MAPE": evaluasi(testPredictScaledJawaBarat, testYScaledJawaBarat)['MAPE'],
    #     "MAE": evaluasi(testPredictScaledJawaBarat, testYScaledJawaBarat)['MAE'],
    #     "RMSE": evaluasi(testPredictScaledJawaBarat, testYScaledJawaBarat)['RMSE']
    # },
    "Kota Bandung": {
        "MSE": evaluasi(testPredictScaledKotaBandung,
testYScaledKotaBandung)['MSE'],
        "MAPE": evaluasi(testPredictScaledKotaBandung,
testYScaledKotaBandung)['MAPE'],
        "MAE": evaluasi(testPredictScaledKotaBandung,
testYScaledKotaBandung)['MAE'],
        "RMSE": evaluasi(testPredictScaledKotaBandung,
testYScaledKotaBandung)['RMSE']
    },
    "Kabupaten Cianjur": {
        "MSE": evaluasi(testPredictScaledKabCianjur, testYScaledKabCianjur)['MSE'],
        "MAPE": evaluasi(testPredictScaledKabCianjur,
testYScaledKabCianjur)['MAPE'],
        "MAE": evaluasi(testPredictScaledKabCianjur, testYScaledKabCianjur)['MAE'],
        "RMSE": evaluasi(testPredictScaledKabCianjur, testYScaledKabCianjur)['RMSE']
    },
}

```

```

"Kabupaten Cirebon": {
  "MSE": evaluasi(testPredictScaledKabCirebon, testYScaledKabCirebon)['MSE'],
  "MAPE": evaluasi(testPredictScaledKabCirebon,
testYScaledKabCirebon)['MAPE'],
  "MAE": evaluasi(testPredictScaledKabCirebon, testYScaledKabCirebon)['MAE'],
  "RMSE": evaluasi(testPredictScaledKabCirebon,
testYScaledKabCirebon)['RMSE']
},
"Kabupaten Tasikmalaya": {
  "MSE": evaluasi(testPredictScaledKabTasikmalaya,
testYScaledKabTasikmalaya)['MSE'],
  "MAPE": evaluasi(testPredictScaledKabTasikmalaya,
testYScaledKabTasikmalaya)['MAPE'],
  "MAE": evaluasi(testPredictScaledKabTasikmalaya,
testYScaledKabTasikmalaya)['MAE'],
  "RMSE": evaluasi(testPredictScaledKabTasikmalaya,
testYScaledKabTasikmalaya)['RMSE']
},
"Kota Bekasi": {
  "MSE": evaluasi(testPredictScaledKotaBekasi, testYScaledKotaBekasi)['MSE'],
  "MAPE": evaluasi(testPredictScaledKotaBekasi,
testYScaledKotaBekasi)['MAPE'],
  "MAE": evaluasi(testPredictScaledKotaBekasi, testYScaledKotaBekasi)['MAE'],
  "RMSE": evaluasi(testPredictScaledKotaBekasi, testYScaledKotaBekasi)['RMSE']
},
"Kota Bogor": {
  "MSE": evaluasi(testPredictScaledKotaBogor, testYScaledKotaBogor)['MSE'],
  "MAPE": evaluasi(testPredictScaledKotaBogor, testYScaledKotaBogor)['MAPE'],
  "MAE": evaluasi(testPredictScaledKotaBogor, testYScaledKotaBogor)['MAE'],
  "RMSE": evaluasi(testPredictScaledKotaBogor, testYScaledKotaBogor)['RMSE']
},

```



```
}
```

```
df_evaluasi = pd.DataFrame.from_dict(evaluasi)
```

```
df_evaluasi
```

```
from matplotlib import pyplot as plt
```

```
df_evaluasi['Kota Bandung'].plot(kind='hist', bins=20, title='Kota Bandung')
```

```
plt.gca().spines[['top', 'right']].set_visible(False)
```

```
# calculate the mean horizontally
```

```
mean_horizontal = df_evaluasi.mean(axis=1)
```

```
df_mean = pd.DataFrame.from_dict(mean_horizontal)
```

```
df_mean
```