

**PERBANDINGAN PREDIKSI RADIASI SINAR MATAHARI
MENGUNAKAN ALGORITMA GATED RECURRENT
UNIT (GRU), LONG SHORT-TERM MEMORY
(LSTM) DAN DEEP Q-NETWORK (DQN)**

SKRIPSI

IQBAL FAKHRIZA

191402097



**PROGRAM STUDI S1 TEKNOLOGI INFORMASI
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA
MEDAN
2024**

PERBANDINGAN PREDIKSI RADIASI SINAR MATAHARI MENGGUNAKAN
ALGORITMA GATED RECURRENT UNIT (GRU), LONG SHORT-TERM
MEMORY (LSTM) DAN DEEP Q-NETWORK (DQN)

SKRIPSI

Diajukan untuk melengkapi dan memenuhi syarat memperoleh ijazah
Sarjana Teknologi Informasi

IQBAL FAKHRIZA

191402097



**PROGRAM STUDI S1 TEKNOLOGI INFORMASI
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA**

MEDAN

2024

PERSETUJUAN

Judul : Perbandingan Prediksi Radiasi Sinar Matahari
Menggunakan Algoritma *Gated Recurrent Unit (GRU)*,
Long Short-Term Memory (LSTM) Dan *Deep Q-*
Network (DQN)
Kategori : Skripsi
Nama Mahasiswa : Iqbal Fakhriza
Nomor Induk Mahasiswa : 191402097
Program Studi : Sarjana (S-1) Teknologi Informasi
Fakultas : Ilmu Komputer dan Teknologi Informasi
Universitas Sumatera Utara

Medan, 20 Mei 2024

Komisi Pembimbing:

Pembimbing 2,

Pembimbing 1,

Sarah Purnamawati, S.T., M.Sc.
NIP. 198302262010122003

Romi Fadillah Rahmat B.Comp.Sc., M.Sc.
NIP. 198603032010121004

Diketahui/disetujui oleh
Program Studi S-1 Teknologi Informasi
Ketua,

Dedy Arisandi S.T., M.Kom.
NIP. 197908312009121002

PERNYATAAN

**PERBANDINGAN PREDIKSI RADIASI SINAR MATAHARI MENGGUNAKAN
ALGORITMA GATED RECURRENT UNIT (GRU), LONG SHORT-TERM
MEMORY (LSTM) DAN DEEP Q-NETWORK (DQN)**

SKRIPSI

Saya mengakui bahwa skripsi ini adalah hasil karya saya sendiri, kecuali beberapa kutipan dan ringkasan yang masing-masing telah disebutkan sumbernya.

Medan, 20 Mei 2024

Iqbal Fakhriza

191402097

UCAPAN TERIMA KASIH

Segala puji syukur disampaikan pada Allah SWT, Tuhan semesta alam, untuk limpahan berkah, anugerah, serta ridho-Nya telah memungkinkan penyelesaian penyusunan skripsi ini baik bentuk dan isinya oleh penulis untuk memenuhi persyaratan memperoleh gelar Sarjana Komputer di Program Studi S1 Teknologi Informasi, Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara.

Penulis tidak akan bisa selesai menyusun skripsi ini tanpa didukung doa, bantuan, dan motivasi semua orang yang berpengaruh pada proses perkuliahan serta pengerjaan tugas akhir hingga selesai. Oleh karena itu, penulis mengungkapkan rasa terima kasih yang sebesar-besarnya pada kesempatan ini kepada:

1. Allah SWT. yang selalu memudahkan, memberikan kekuatan serta rezeki yang berlimpah karenanya skripsi ini bisa diselesaikan oleh penulis insyaallah dengan jiwa dan raga yang sehat.
2. Keluarga penulis, Abi, Umi, serta adik dan kakak, yang selalu menemani dan menanyakan perihal pengerjaan skripsi. Terima kasih untuk semua dukungannya terutama umi yang selalu menjadi penyemangat saat bercerita dan berdiskusi.
3. Dosen Pembimbing 1 penulis, Bapak Romi Fadillah Rahmat B.Comp.Sc., M.Sc. yang selalu mengingatkan dan dengan sabar membimbing penulisan skripsi ini dari awal pengerjaan hingga semua detailnya
4. Dekan Fasilkom-TI Universitas Sumatera Utara Ibu Dr. Maya Silvi Lydia, M.Sc.
5. Ketua Program Studi S1 Teknologi Informasi Universitas Sumatera Utara Bapak Dedy Arisandi S.T., M.Kom.
6. Seluruh Dosen Program Studi S1 Teknologi Informasi yang sudah menyampaikan pengetahuan serta memberi pelajaran kepada penulis dalam masa kuliah.
7. Seluruh staff dan pegawai Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara yang menolong dalam segala urusan administrasi selama masa perkuliahan penulis.
8. Bapak Emerson Pascawira Sinulingga ST., M.Sc. Ph.D yang selalu menyemangati untuk segera menuntaskan perkuliahan. Saya ucapkan terima kasih untuk segala

saran dan diskusi yang telah terjadi dengan bapak sangat membantu saya dalam proses penulisan skripsi ini.

9. Sahabat penulis yang selalu menjadi pemacu untuk segera menuntaskan pengerjaan skripsi ini, Muhammad Iqbal Tanjung. Atas seluruh inspirasi dan semangat yang tercurah saya ucapkan banyak terima kasih, semoga dapat berjumpa lagi segera.
10. Teman-teman dari stambuk 2019 S1 Teknologi Informasi yang banyak memberi semangat, berdiskusi, bertanya, memberi saran, serta menemani penulis selama kuliah hingga kegiatan tugas akhir: Alwi, Andrea, Saddam, Meina, Karvin, Masayu, Wanda, Andi, Tazrian, Zen dan teman-teman lain yang tidak dapat dituliskan satu persatu.
11. Seluruh teman-teman part-timer Direktorat Digitalisasi dan Integrasi Sistem USU yang selalu menjadi tempat curhat dan keluh-kesah skripsi: Bang Rasyid, Bang Habib, Kak Andini, Kak Feni, Wildan, Arief dan teman-teman lain
12. Seluruh pihak yang secara langsung ataupun tidak langsung turut membantu dalam penyelesaian skripsi ini dan tidak dapat dituliskan satu per satu.

Medan, 20 Mei 2024

Penulis,

Iqbal Fakhriza

191402097

ABSTRAK

Prediksi radiasi matahari dalam jangka panjang sangat penting dalam memberikan informasi detail tentang sumber daya matahari dalam suatu wilayah. Bagi pengusaha, investor hingga pemerintahan, hal ini sangat bermanfaat untuk pengembangan energi surya dengan skala besar. Informasi tersebut memungkinkan mereka untuk membuat keputusan yang lebih baik terkait investasi dan kebijakan energi surya di suatu wilayah. Beberapa penelitian sebelumnya telah melakukan prediksi radiasi matahari dengan berbagai metode mulai dari menggunakan model fisik seperti *Numerical Weather Prediction (NWP)*, model statistik, hingga menggunakan model kecerdasan buatan. Pada penelitian ini, 3 algoritma akan dibandingkan untuk membandingkan algoritma terbaik dalam objek prediksi radiasi matahari. Algoritma tersebut yaitu *Gated Recurrent Unit (GRU)*, *Long Short-Term Memory (LSTM)* dan *Deep Q-Network (DQN)*. Data yang akan digunakan pada penelitian ini berasal dari hasil penelitian *Development of software for estimating clear sky solar radiation in Indonesia* oleh Himsar Ambarita (2017) dengan jangkauan data selama 6 tahun secara total. Nantinya sebagai evaluasi model, akan digunakan 4 metrik evaluasi untuk menghitung besaran kesalahan model. Metrik evaluasi tersebut diantaranya yaitu *Mean Squared Error (MSE)*, *Root Mean Squared Error (RMSE)*, *Mean Absolute Error (MAE)*, dan R^2 (*Coefficient of Determination*). Tujuan utama penelitian ini adalah membandingkan keakuratan algoritma regresi seperti *GRU* dan *LSTM* yang dibandingkan dengan algoritma *DQN* yang mempunyai basis perhitungan berdasarkan nilai *reward* dalam melakukan prediksi radiasi matahari.

Kata Kunci: Prediksi, Radiasi Matahari, Deep Q-Network, Gated Recurrent Unit, Long Short-Term Memory

**PREDICTION OF SOLAR RADIATION USING HYBRIDIZATION OF
DEEP Q-NETWORK (DQN) AND GATED RECURRENT UNIT
(GRU) ALGORITHMS**

ABSTRACT

Long-term prediction of solar radiation is very important in providing detailed information about solar resources in a region. For entrepreneurs, investors and governments, this is very useful for large-scale solar energy development. The information allows them to make better decisions regarding solar energy investments and policies in a region. Several previous studies have predicted solar radiation with various methods ranging from using physical models such as Numerical Weather Prediction (NWP), statistical models, to using artificial intelligence models. In this research, 3 algorithms will be compared to define the best algorithm in solar radiation prediction object. These algorithms are Gated Recurrent Unit (GRU), Long Short-Term Memory (LSTM) and Deep Q-Network (DQN). The data that will be used in this research comes from the results of the research Development of software for estimating clear sky solar radiation in Indonesia by Himsar Ambarita (2017) with data coverage for 7 years in total. Later as a model evaluation, 4 evaluation metrics will be used to calculate the amount of model error. These evaluation metrics include Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and R^2 (Coefficient of Determination). The main objective of this research is to compare the accuracy of regression algorithms such as GRU and LSTM compared to the DQN algorithm which has a calculation based on reward values in predicting solar radiation.

Keywords: *Prediction, Solar Radiation, Deep Q-Network, Gated Recurrent Unit, Long Short-Term Memory*

DAFTAR ISI

PERSETUJUAN	i
PERNYATAAN	ii
UCAPAN TERIMA KASIH	iii
ABSTRAK	v
<i>ABSTRACT</i>	vi
DAFTAR ISI	vii
DAFTAR TABEL	x
DAFTAR GAMBAR	xii
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	4
1.3 Tujuan Penelitian	4
1.4 Batasan Masalah	5
1.5 Manfaat Penelitian	5
1.6 Metodologi Penelitian	5
1.6.1 Studi Literatur	5
1.6.2 Analisis Permasalahan	6
1.6.3 Perancangan Sistem	6
1.6.4 Data Preparation	6
1.6.5 Implementasi	7
1.6.6 Penyusunan Laporan	7
1.7 Sistematika Penulisan	7
BAB 2 LANDASAN TEORI	9
2.1 Sumber Energi Terbarukan	9
2.3 Jangka Waktu Prediksi Radiasi Matahari	11

2.4	Prediksi Radiasi Matahari	11
2.5	<i>Deep Reinforcement Learning (DRL)</i>	11
2.6	<i>Long Short-Term Memory</i>	13
2.7	<i>Gated Recurrent Unit</i>	14
2.8	Penelitian Terdahulu	16
2.9	Perbedaan Penelitian	18
BAB 3	ANALISIS DAN PERANCANGAN SISTEM	20
3.1	Data yang digunakan	20
3.2	Arsitektur Umum Sistem	21
3.2.1	Data Collection	22
3.2.2	Data Pre-Processing	22
3.2.3	Data Splitting	28
3.2.4	Proses Training	28
3.2.5	Proses Testing	34
3.3	Perancangan Antarmuka Sistem	35
3.3.1	Rancangan Tampilan Halaman Utama	36
3.3.2	Rancangan Tampilan Halaman Training	36
3.3.3	Rancangan Tampilan Halaman Testing	38
3.3.4	Rancangan Tampilan Halaman Predict	38
BAB 4	IMPLEMENTASI DAN PENGUJIAN	40
4.1	Implementasi Sistem	40
4.1.1	Perangkat Keras dan Perangkat Lunak	40
4.1.2	Implementasi Rancangan Antarmuka	41
4.2	Prosedur Operasional	45
4.3	Pelatihan Sistem	46
4.3.1	Hasil Pelatihan Model Gated Recurrent Unit	49
4.3.2	Hasil Pelatihan Model Long Short-Term Memory	54

4.3.3 Hasil Pelatihan Model Deep Q-Network	59
4.4 Pengujian Sistem	64
4.4.1 Hasil Pengujian Model Gated Recurrent Unit	65
4.4.2 Hasil Pengujian Model Long-Short Term Memory	68
4.4.3 Hasil Pengujian Model Deep Q-Network	72
4.4.4 Perbandingan Model Gated Recurrent Unit, Long Short-Term Memory dan Deep Q-Network	77
BAB 5 PENUTUP	79
5.1 Kesimpulan	79
5.2 Saran	79

DAFTAR TABEL

Tabel 3.1 Rincian jumlah data yang digunakan	20
Tabel 3.2 <i>pseudocode</i> pelatihan model menggunakan algoritma <i>DQN</i>	31
Tabel 3.3 Training Hyperparameter untuk algoritma <i>DQN</i>	32
Tabel 3.4 <i>Training Hyperparameter</i> untuk algoritma <i>LSTM</i>	32
Tabel 3.5 Training Hyperparameter untuk algoritma <i>GRU</i>	33
Tabel 4.1 Nilai metrik evaluasi dari pengujian model <i>GRU</i> dengan jangka prediksi 7 hari	49
Tabel 4.2 Nilai metrik evaluasi setelah pelatihan model <i>GRU</i> dengan jangka prediksi 30 hari	51
Tabel 4.3 Nilai metrik evaluasi setelah pelatihan model <i>GRU</i> dengan jangka prediksi 60 hari	52
Tabel 4.4 Nilai metrik evaluasi setelah pelatihan model <i>GRU</i> dengan jangka prediksi 90 hari	53
Tabel 4.5 Nilai metrik evaluasi setelah pelatihan model <i>LSTM</i> dengan jangka prediksi 7 hari	55
Tabel 4.6 Nilai metrik evaluasi setelah pelatihan model <i>LSTM</i> dengan jangka prediksi 30 hari	56
Tabel 4.7 Nilai metrik evaluasi setelah pelatihan model <i>LSTM</i> dengan jangka prediksi 60 hari	57
Tabel 4.8 Nilai metrik evaluasi setelah pelatihan model <i>LSTM</i> dengan jangka prediksi 90 hari	58
Tabel 4.9 Nilai metrik evaluasi dari pengujian model <i>DQN</i> dengan jangka prediksi 7 hari	60
Tabel 4.10 Nilai metrik evaluasi dari pengujian model <i>DQN</i> dengan jangka prediksi 30 hari	61
Tabel 4.11 Nilai metrik evaluasi dari pengujian model <i>DQN</i> dengan jangka prediksi 60 hari	63
Tabel 4.12 Nilai metrik evaluasi dari pengujian model <i>DQN</i> dengan jangka prediksi 90 hari	64

Tabel 4.13 Perbandingan Nilai Metrik Evaluasi dari kombinasi terbaik pada masing-masing algoritma pada jangka waktu yang beragam

77

DAFTAR GAMBAR

Gambar 2.1 Skema opsi energi terbarukan (Pandey, C. K., et al., 2013)	10
Gambar 2.2 Arsitektur Umum <i>Reinforcement Learning</i>	12
Gambar 2.3 Arsitektur <i>Long Short-Term Memory (LSTM)</i> (Ozdemir & Ozkan Yildirim, 2023)	14
Gambar 2.4 Arsitektur Gated Recurrent Unit (GRU) (Ozdemir & Ozkan Yildirim, 2023)	15
Gambar 3.1 Alat pengukur radiasi matahari pada penelitian (Ambarita, 2017)	21
Gambar 3.2 Arsitektur Umum Sistem	22
Gambar 3.3 Distribusi data suhu dan radiasi matahari sebelum proses <i>data cleaning</i>	23
Gambar 3.4 Distribusi data suhu dan radiasi matahari setelah proses <i>handling outliers</i>	24
Gambar 3.5 Rata-rata besaran radiasi matahari untuk setiap jam setelah proses <i>data cleaning</i>	25
Gambar 3.6 Rata-rata besaran radiasi matahari untuk setiap tahun setelah proses <i>data cleaning</i>	26
Gambar 3.7 Rata-rata persentase suhu dan kelembapan udara untuk setiap jam setelah proses <i>data cleaning</i>	26
Gambar 3.8 Diagram <i>Correlation Heatmap</i> dari dataset	27
Gambar 3.9 Rancangan Tampilan Halaman Utama	35
Gambar 3.10 Rancangan Tampilan Halaman Training dengan tabel	36
Gambar 3.11 Rancangan Tampilan Halaman <i>Training</i> dengan grafik	37
Gambar 3.12 Rancangan Tampilan Halaman <i>Testing</i> dengan menampilkan hasil berbentuk tabel	37
Gambar 3.13 Rancangan Tampilan Halaman <i>Testing</i> dengan menampilkan hasil berbentuk grafik	38
Gambar 3.14 Rancangan Tampilan Halaman <i>Predict</i> dengan menampilkan hasil berbentuk tabel	39

Gambar 3.15 Rancangan Tampilan Halaman <i>Predict</i> dengan menampilkan hasil berbentuk grafik	39
Gambar 4.1 Tampilan Halaman Utama	41
Gambar 4.2 Tampilan Halaman Training dengan tabel	42
Gambar 4.3 Tampilan Halaman <i>Training</i> dengan grafik	42
Gambar 4.4 Tampilan Halaman <i>Training</i> dengan grafik dengan nilai loss <i>RMSE</i> dan R^2	43
Gambar 4.5 Tampilan Halaman Testing dengan Tabel	43
Gambar 4.6 Tampilan Halaman Testing dengan Grafik	44
Gambar 4.7 Tampilan Halaman Predict dengan Tabel	45
Gambar 4.8 Tampilan Halaman Predict dengan Grafik	45
Gambar 4.9 Struktur Neural Network pada algoritma GRU dan LSTM	47
Gambar 4.10 Struktur Neural Network pada algoritma DQN	48
Gambar 4.11 Pergerakan nilai metrik evaluasi selama pelatihan <i>model</i> dari empat kombinasi parameter terbaik pada algoritma <i>GRU</i> dengan jangka prediksi 7 hari	50
Gambar 4.12 Pergerakan nilai metrik evaluasi selama pelatihan <i>model</i> dari empat kombinasi parameter terbaik pada algoritma <i>GRU</i> dengan jangka prediksi 30 hari	51
Gambar 4.13 Pergerakan nilai metrik evaluasi selama pelatihan <i>model</i> dari empat kombinasi parameter terbaik pada algoritma <i>GRU</i> dengan jangka prediksi 60 hari	53
Gambar 4.14 Pergerakan nilai metrik evaluasi selama pelatihan <i>model</i> dari empat kombinasi parameter terbaik pada algoritma <i>GRU</i> dengan jangka prediksi 90 hari	54
Gambar 4.15 Pergerakan nilai metrik evaluasi selama pelatihan <i>model</i> dari empat kombinasi parameter terbaik pada algoritma <i>LSTM</i> dengan jangka prediksi 7 hari	55
Gambar 4.16 Pergerakan nilai metrik evaluasi selama pelatihan <i>model</i> dari empat kombinasi parameter terbaik pada algoritma <i>LSTM</i> dengan jangka prediksi 30 hari	57
Gambar 4.17 Pergerakan nilai metrik evaluasi selama pelatihan <i>model</i> dari empat kombinasi parameter terbaik pada algoritma <i>LSTM</i> dengan jangka prediksi 60 hari	58
Gambar 4.18 Pergerakan nilai metrik evaluasi selama pelatihan <i>model</i> dari empat kombinasi parameter terbaik pada algoritma <i>LSTM</i> dengan jangka prediksi 60 hari	59
Gambar 4.19 Pergerakan nilai <i>reward</i> selama pelatihan <i>model</i> dari empat kombinasi parameter terbaik pada algoritma <i>DQN</i> dengan jangka prediksi 7 hari	61
Gambar 4.20 Pergerakan nilai <i>reward</i> selama pelatihan <i>model</i> dari empat kombinasi parameter terbaik pada algoritma <i>DQN</i> dengan jangka prediksi 30 hari	62

Gambar 4.21 Pergerakan nilai *reward* selama pelatihan *model* dari empat kombinasi parameter terbaik pada algoritma *DQN* dengan jangka prediksi 60 hari 63

Gambar 4.22 Pergerakan nilai *reward* selama pelatihan *model* dari empat kombinasi parameter terbaik pada algoritma *DQN* dengan jangka prediksi 90 hari 64

Gambar 4.23 Perbandingan nilai prediksi dan nilai sebenarnya dari empat kombinasi parameter terbaik pada algoritma *GRU* dengan jangka prediksi 7 hari 65

Gambar 4.24 Perbandingan nilai prediksi dan nilai sebenarnya dari empat kombinasi parameter terbaik pada algoritma *GRU* dengan jangka prediksi 30 hari 66

Gambar 4.25 Perbandingan nilai prediksi dan nilai sebenarnya dari empat kombinasi parameter terbaik pada algoritma *GRU* dengan jangka prediksi 60 hari 67

Gambar 4.26 Perbandingan nilai prediksi dan nilai sebenarnya dari empat kombinasi parameter terbaik pada algoritma *GRU* dengan jangka prediksi 90 hari 68

Gambar 4.27 Perbandingan nilai prediksi dan nilai sebenarnya dari empat kombinasi parameter terbaik pada algoritma *LSTM* dengan jangka prediksi 7 hari 69

Gambar 4.28 Perbandingan nilai prediksi dan nilai sebenarnya dari empat kombinasi parameter terbaik pada algoritma *LSTM* dengan jangka prediksi 30 hari 70

Gambar 4.29 Perbandingan nilai prediksi dan nilai sebenarnya dari empat kombinasi parameter terbaik pada algoritma *LSTM* dengan jangka prediksi 60 hari 71

Gambar 4.30 Perbandingan nilai prediksi dan nilai sebenarnya dari empat kombinasi parameter terbaik pada algoritma *LSTM* dengan jangka prediksi 90 hari 72

Gambar 4.31 Perbandingan nilai prediksi dan nilai sebenarnya dari empat kombinasi parameter terbaik pada algoritma *DQN* dengan jangka prediksi 7 hari 73

Gambar 4.32 Perbandingan nilai prediksi dan nilai sebenarnya dari empat kombinasi parameter terbaik pada algoritma *DQN* dengan jangka prediksi 30 hari 74

Gambar 4.33 Perbandingan nilai prediksi dan nilai sebenarnya dari empat kombinasi parameter terbaik pada algoritma *DQN* dengan jangka prediksi 60 hari 75

Gambar 4.34 Perbandingan nilai prediksi dan nilai sebenarnya dari empat kombinasi parameter terbaik pada algoritma *DQN* dengan jangka prediksi 90 hari 76

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Pemanasan global dan krisis energi selama beberapa dekade terakhir akibat penggunaan bahan bakar fosil dalam jumlah besar sangat berdampak pada berbagai bidang kehidupan. Berbagai masalah mulai dari kondisi suhu bumi yang meningkat, tingkat polusi yang semakin tinggi, hingga peningkatan jumlah emisi CO₂ di udara yang semakin banyak. Hal ini mendorong pemerintah untuk mulai mengembangkan sumber energi terbarukan yang ramah lingkungan.

Sumber energi terbarukan sudah banyak dikembangkan, diantaranya seperti Sumber energi tenaga air, tenaga panas bumi, tenaga surya, tenaga angin, hingga tenaga pasang surut air laut. Diantara beberapa sumber energi terbarukan yang telah banyak dikembangkan, sumber energi tenaga surya memiliki perkembangan yang sangat pesat dalam pengembangan dan pemanfaatannya. Berdasarkan data dari *International Renewable Energy Agency*, Energi surya menyumbang lebih dari setengah dari penambahan kapasitas energi terbarukan dengan rekor 133 GW pada tahun 2021, diikuti oleh energi angin dengan total 93 GW, dan kapasitas energi angin lepas pantai mencapai rekor 21 GW.

Radiasi surya yang mencapai permukaan bumi setiap tahunnya mencapai sekitar 3.400.000 eksajoule, sementara konsumsi energi manusia mencapai sekitar 450 eksajoule. Hal ini mengungkapkan bahwa hanya dalam waktu 1 jam 10 menit radiasi surya sudah dapat memenuhi konsumsi energi global selama satu tahun (Himsar Ambarita, 2017). Menurut Kementerian Energi dan Sumber Daya Mineral Pemerintah Indonesia, potensi energi surya di pulau-pulau Indonesia mencapai rata-rata 4,8 kWh/m²/hari dengan durasi radiasi 10 sampai 12 jam perhari (Nematchoua et al., 2022).

Sumber utama dari pembangkit listrik surya diambil melalui panel surya yang terdiri dari kumpulan sel surya (*photovoltaic cells*). Mengingat sifat cuaca yang sulit diprediksi, pengelola layanan pembangkit listrik tenaga surya diwajibkan untuk menjaga tingkat pasokan listrik yang stabil. Operasi sistem tenaga surya dengan tingkat penetrasi tinggi menimbulkan beberapa tantangan karena sifatnya yang tidak teratur dan fluktuatif, yang

tergantung pada parameter meteorologis dan terutama dinamika awan (Solano et al., 2022). Produksi listrik tenaga surya yang fluktuatif dapat menyebabkan terjadinya inversi arus daya dengan variasi tegangan dan frekuensi, serta ketidakseimbangan antara permintaan dan pasokan energi (Solano et al., 2022). Oleh karena itu peramalan yang akurat tentang radiasi matahari sangat penting untuk menjaga operasi yang aman pada sistem pembangkit listrik tenaga surya.

Banyak model prediksi yang telah diusulkan dalam literatur yang ditargetkan pada jangka waktu prediksi yang beragam. Prediksi sangat pendek (hitungan jam), pendek (1 hari), menengah (1 bulan) hingga jangka Panjang (1 tahun) masing-masing digunakan untuk aplikasi yang berbeda. Prediksi jangka pendek dan sangat pendek dapat digunakan untuk operasi sistem tenaga secara langsung (*real-time*). Prediksi jangka pendek dapat digunakan untuk tujuan perencanaan penugasan, sedangkan prediksi jangka menengah dan panjang dapat digunakan untuk tujuan pemeliharaan dan pasar energi (Solano et al., 2022)

Algoritma prediksi dapat diklasifikasikan menjadi model fisik, seperti *Numerical Weather Prediction (NWP)*; model statistik, seperti *Autoregressive Moving Average (ARMA)* dan *Autoregressive Integrated Moving Average (ARIMA)*; dan model data-driven, seperti algoritma kecerdasan buatan.

Di dalam literatur, beberapa model telah diusulkan untuk memprediksi radiasi matahari. Model-model empiris diperoleh dari formula matematika (Fan et al., 2018). Model-model tersebut mudah digunakan, namun mereka tidak mampu memprediksi dengan tepat berbagai data radiasi matahari dalam jangka pendek karena adanya perubahan kondisi meteorologi yang teratur (Jahani & Mohammadi, 2019). Saat ini, dengan perkembangan teknologi, kecerdasan buatan semakin populer. Kecerdasan Buatan dapat diaplikasikan di berbagai bidang (Long et al., 2014).

Banyaknya algoritma kecerdasan buatan yang ada tentunya memungkinkan eksperimen yang lebih luas untuk menentukan metode terbaik dalam menyelesaikan berbagai permasalahan. Berdasarkan penelitian terdahulu, Penggunaan metode kecerdasan buatan seperti *Support Vector Machine (SVM)*, *Artificial Neural Network (ANN)*, *Kernel and Nearest-Neighbor (k-NN)*, dan *Extreme Gradient Boosting (XGBT)* untuk memprediksi radiasi matahari sudah banyak di teliti dan di uji coba dengan berbagai kelebihan dan kekurangan masing-masing.

Ağbulut, Ü., et al., pada tahun 2021 melakukan penelitian untuk membandingkan algoritma-algoritma kecerdasan buatan yang digunakan untuk memprediksi radiasi matahari harian menggunakan data dari *The Turkish State Meteorological Service (TSMS)* menghasilkan hasil yang cukup baik dengan nilai *Root Mean Squared Error (RMSE)* berkisar antara 2.157 MJ/m² hingga 3.092 MJ/m² diantara keempat stasiun cuaca. Namun dari beberapa algoritma yang diuji belum ada algoritma yang menghasilkan hasil terbaik di semua tempat (Ağbulut et al., 2021).

Edna S. Solano, et al., pada tahun 2022 juga melakukan penelitian untuk memprediksi radiasi matahari jangka sangat pendek menggunakan basis data asli dari *The Brazilian National Institute of Meteorology website (INMET)*. Dengan menggunakan algoritma kecerdasan buatan *Support Vector Machine (SVM)*, *Extreme Gradient Boosting (XGBT)*, *Categorical Boosting (CatBoost)*, serta *Voting-Average (VOA)*, menghasilkan nilai *error* terendah pada algoritma *VOA* dengan nilai *Root Mean Squared Error (RMSE)* berkisar antara 0,43 MJ/m² hingga 0,58 MJ/m² pada musim panas. Namun, distribusi *error* akan menjadi semakin besar seiring waktu prediksi yang meningkat (Solano et al., 2022).

Berdasarkan beberapa penelitian terdahulu yang telah dilakukan, suhu dan kelembapan udara menjadi parameter penting dalam prediksi radiasi matahari. Dalam penelitian “*Development of software for Estimating Clear Sky Solar Radiation in Indonesia*” oleh Himsar Ambarita pada tahun 2017, alat pengukur radiasi matahari telah dibuat dan dioperasikan di Medan, Sumatera Utara. Melalui alat pengukur radiasi matahari tersebut, diambil data kecepatan angin, suhu udara, kelembapan udara, titik embun dan jumlah radiasi matahari yang mencapai permukaan bumi.

Pada penelitian *Water quality prediction method based on LSTM neural network*, model *long short-term memory (LSTM)* digunakan sebagai algoritma prediksi. Penelitian tersebut melakukan prediksi dengan objek kualitas air danau. Hasil penelitian menunjukkan algoritma *LSTM* yang selalu unggul dalam semua situasi dibandingkan dengan algoritma *back propagation neural network (BP NN)* dan *online sequential extreme learning machine (OS-ELM)* (Wen & Li, 2023).

Dalam artikel *Deep Reinforcement Learning for Stock Prediction*, model *DQN* digunakan sebagai algoritma prediksi. *DQN* dinilai lebih baik dibandingkan algoritma seperti *Feedforward Neural Network (FFNN)* dan *Recurrent Neural Network (RNN)* dalam memprediksi data yang tidak stabil seperti harga saham (Zhang & Lei, 2022).

Dilain artikel yaitu *Prediction of Water Level in Lakes by RNN-Based Deep Learning Algorithms to Preserve Sustainability in Changing Climate and Relationship to Microcystin*, model-model *RNN* digunakan sebagai algoritma prediksi. Objek dari penelitian pada artikel ini adalah ketinggian permukaan air pada danau yang disebut sebagai *Lake Water Level (LWL)*. Hasil penelitian menunjukkan algoritma *GRU* yang masih superior dibandingkan algoritma lain yang dilakukan perbandingan seperti *Naïve Method*, *ANN*, *LSTM*, *Stacked LSTM*, serta *Bidirectional LSTM* (Zhang & Lei, 2022).

Dengan kelebihan dari beberapa algoritma tersebut, akan dilakukan perbandingan algoritma yang paling baik dalam memprediksi besaran radiasi matahari jangka pendek hingga menengah menggunakan dataset dari penelitian *Development of software for Estimating Clear Sky Solar Radiation in Indonesia* (Ambarita, 2017). Dengan demikian, penulis melakukan penelitian dengan judul “Perbandingan Prediksi Radiasi Sinar Matahari Menggunakan Algoritma *Gated Recurrent Unit (GRU)*, *Long Short-Term Memory (LSTM)* dan *Deep Q-Network (DQN)*”.

1.2 Rumusan Masalah

Dalam rangka penyelesaian masalah prediksi, algoritma kecerdasan buatan kerap kali menjadi pilihan utama karena kemampuannya yang unggul dalam mengidentifikasi pola kompleks dari sebuah data. Dalam ranah kecerdasan buatan, salah satu algoritma yang terbukti sangat baik dalam menyelesaikan masalah prediksi deret waktu adalah *recurrent neural network (RNN)*. Dilain sisi, dalam artikel *Deep Reinforcement Learning for Stock Prediction*, *DQN* dapat memprediksi dengan baik pada data yang tidak stabil. Pada penelitian tersebut hasil prediksi dari *DQN* dibandingkan dengan algoritma *RNN* seperti *LSTM* dan *GRU*. Hasilnya, *DQN* dapat menghasilkan model yang lebih baik dalam memprediksi harga saham.

Berdasarkan penelitian tersebut, muncul gagasan untuk mengeksplorasi algoritma *DQN* untuk melakukan prediksi pada data yang lebih stabil. Algoritma lainnya yang akan dilakukan perbandingan yaitu algoritma *LSTM* dan *GRU*. Algoritma-algoritma tersebut akan dibandingkan dalam menyelesaikan masalah prediksi menggunakan data meteorologi

1.3 Tujuan Penelitian

Penelitian ini bertujuan dalam melakukan perbandingan pada beberapa algoritma untuk melakukan eksplorasi terkait algoritma *recurrent neural network (RNN)* dan *deep q-*

network (*DQN*) dalam melakukan prediksi data meteorologi. Objek yang akan diprediksi adalah besaran radiasi matahari dengan jangkauan prediksi mulai dari 7 hari, 30 hari, 60 hari hingga 90 hari.

1.4 Batasan Masalah

Batasan Masalah ditentukan untuk menghindari pembahasan yang terlalu meluas dalam penelitian. Berikut merupakan Batasan masalah untuk penelitian ini:

1. Penelitian ini memprediksi besaran radiasi matahari dengan batasan jangka prediksi mulai dari 7 hari, 30 hari, 60 hari hingga 90 hari.
2. Algoritma yang akan dilakukan perbandingan yaitu algoritma *long short-term memory* (*LSTM*), *gated recurrent unit* (*GRU*) dan *deep q-network* (*DQN*) tanpa kombinasi algoritma.
3. Data yang digunakan untuk melatih data maupun pengujian untuk prediksi menggunakan ekstensi *csv*

1.5 Manfaat Penelitian

Manfaat yang bisa didapatkan dari penelitian ini adalah sebagai berikut:

1. Menguji keefektifan algoritma *DQN* dalam melakukan prediksi dengan data yang lebih stabil.
2. Menguji keefektifan algoritma *DQN*, *LSTM* dan *GRU* dalam melakukan prediksi menggunakan data meteorologis.
3. Menjadi rujukan bagi penelitian berikutnya dalam menggunakan algoritma kombinasi algoritma *DQN*, *LSTM* dan *GRU* sebagai algoritma prediksi.
4. Berdasarkan hasil penelitian nantinya, algoritma dapat dimanfaatkan pada berbagai permasalahan di berbagai bidang seperti meteorologi dan pembangkit listrik tenaga surya.

1.6 Metodologi Penelitian

Tahapan-tahapan penelitian yang dilakukan yaitu sebagai berikut.

1.6.1 Studi Literatur

Studi Literatur dilakukan dengan cara mengumpulkan rujukan-rujukan mengenai penelitian terdahulu mengenai prediksi radiasi matahari serta penggunaan algoritma *DQN*, *LSTM* serta *GRU*. Rujukan penelitian dapat diambil dari jurnal, artikel, skripsi, serta sumber-sumber lainnya yang dapat digunakan sebagai referensi penelitian.

1.6.2 Analisis Permasalahan

Setelah melakukan studi literatur, disimpulkan masalah-masalah yang terdapat baik pada topik maupun masalah yang dihadapi pada saat pelaksanaan penelitian terdahulu. Dari hasil tersebut dapat dilakukan analisis masalah yang perlu diselesaikan melalui penelitian ini baik dari topik penelitian yaitu radiasi matahari, maupun algoritma yang digunakan dalam penelitian ini.

1.6.3 Perancangan Sistem

Berdasarkan hasil studi literatur dan analisis permasalahan yang telah dilakukan, dibuatlah rancangan sistem berupa arsitektur umum untuk model prediksi yang akan dibuat dalam penelitian ini.

1.6.4 Data Preparation

Data collection merupakan tahap awal *Data Preparation* yaitu tahap mengumpulkan data cuaca dan radiasi matahari. Data cuaca dan radiasi matahari tersebut diambil dari hasil *dataset* penelitian *Development of software for estimating clear sky solar radiation in Indonesia* (Ambarita, 2017). Data tersebut berisi data kecepatan angin, suhu udara, kelembapan udara, titik embun, dan radiasi matahari yang diambil selama 6 tahun, dimulai dari tahun 2017 hingga 2022 di Kota Medan, Indonesia yang diambil melalui alat pengukur radiasi matahari dengan sensor angin, sensor radiasi (*Pyranometer*), serta sensor suhu dan kelembapan.

Tahapan selanjutnya yaitu *data pre-processing*. Pada tahap ini, data yang didapat dari alat pengukur radiasi matahari memerlukan tahapan *data integration* sebelum menyajikan data untuk dilakukan *model training*. Hal ini dilakukan dikarenakan data yang diterima melalui alat pengukur radiasi matahari dikelompokkan menjadi 1 tabel untuk setiap harinya, dan 1 dokumen untuk setiap bulannya. Sebagian dokumen belum memiliki nama kolom standar yang sama untuk setiap parameternya. Oleh karena itu, pada tahapan ini, data perlu disatukan menjadi satu tabel dengan nama kolom yang seragam.

Data yang diterima melalui alat pengukur radiasi matahari yang beroperasi selama 6 tahun tidak selalu beroperasi dengan baik, terdapat beberapa hari dimana sensor tidak mencatat data dengan baik maupun tidak beroperasi. Oleh karena itu, diperlukan proses *data cleaning*, untuk memastikan data dalam keadaan baik sebelum melanjutkan ke tahapan *model training*.

Proses selanjutnya adalah memilih fitur yang akan digunakan saat proses pelatihan *model*. Proses ini dilakukan pada proses *feature selection* dengan melihat hubungan fitur dengan target prediksi. Tahap terakhir adalah *Data Transformation* dimana seluruh data angka yang dimiliki diubah menjadi skala 0-1. Hal ini dilakukan untuk memastikan semua fitur berkontribusi secara merata serta juga untuk mempercepat waktu pelatihan *model*.

1.6.5 Implementasi

Model kecerdasan buatan yang akan digunakan pada penelitian ini menggunakan 3 jenis algoritma yaitu *long short-term memory (LSTM)*, *gated recurrent unit (GRU)* dan *deep q-network (DQN)*. Seluruh algoritma yang digunakan pada penelitian ini dibuat menggunakan basis *library tensorflow* dan *keras*. Untuk algoritma LSTM dan GRU bisa langsung dilakukan dengan mendesain komposisi *neural network* sesuai dengan data.

Sementara terkhusus pada algoritma *DQN* diperlukan tahapan tambahan yaitu pembuatan *environment* sebagai tempat aliran data secara langsung. Pada algoritma *DQN*, *output* dari *neural network* bukan merupakan hasil prediksi, tetapi hasil perhitungan *q-value* yang merupakan hasil perhitungan skor pada *neural network*. Nantinya nilai *q-value* tertinggi pada *output neural network* tersebut yang akan menjadi hasil prediksi.

Setelah proses pelatihan data, dihasilkan model kecerdasan buatan yang dapat dilakukan untuk memprediksi radiasi matahari. Dari model tersebut dilakukan pengukuran tingkat *error* melalui perhitungan metrik evaluasi. Beberapa metrik evaluasi yang akan dihitung yaitu *mean squared error (MSE)*, *root mean squared error (RMSE)*, *mean absolute error (MAE)* dan *coefficient of determination (R^2)*.

Setelah seluruh model dapat bekerja dengan baik, akan dibuat antarmuka pengguna berbasis website agar pengguna dapat melakukan proses training, testing, serta prediksi tanpa perlu menjalankan melalui basis kode.

1.6.6 Penyusunan Laporan

Pada tahap ini dilakukan kegiatan penyusunan laporan penelitian berdasarkan proses penelitian dan hasil yang diperoleh dari penelitian.

1.7 Sistematika Penulisan

Sistematika penulisan penelitian ini terdiri dari lima bab, yaitu:

Bab 1: Pendahuluan

Bab satu berisi latar belakang, rumusan masalah, tujuan penelitian, batasan masalah, manfaat penelitian, metodologi penelitian, dan sistematika penulisan dari penelitian ini.

Bab 2: Landasan Teori

Bab dua berisi teori yang berhubungan dengan penelitian serta berkaitan dengan masalah dalam penelitian. Beberapa teori diantaranya adalah teori terkait algoritma yang digunakan, masalah yang ingin diselesaikan, serta keterkaitan dengan data yang digunakan.

Bab 3: Analisis dan Perancangan

Bab tiga berisi analisis data, perancangan arsitektur umum beserta penjelasan dari tahapan–tahapan yang terdapat di dalam arsitektur umum penelitian serta perancangan antarmuka pada sistem yang dibangun.

Bab 4: Implementasi dan Pengujian Sistem

Bab empat berisi pembahasan mengenai implementasi dari perancangan yang telah dibahas pada bab sebelumnya. Bab ini juga berisi hasil pengujian dan analisis dari sistem yang telah dibuat.

Bab 5: Kesimpulan dan Saran

Bab lima berisi kesimpulan dan saran setelah melakukan penelitian ini yang berguna untuk mengembangkan penelitian selanjutnya.

BAB 2

LANDASAN TEORI

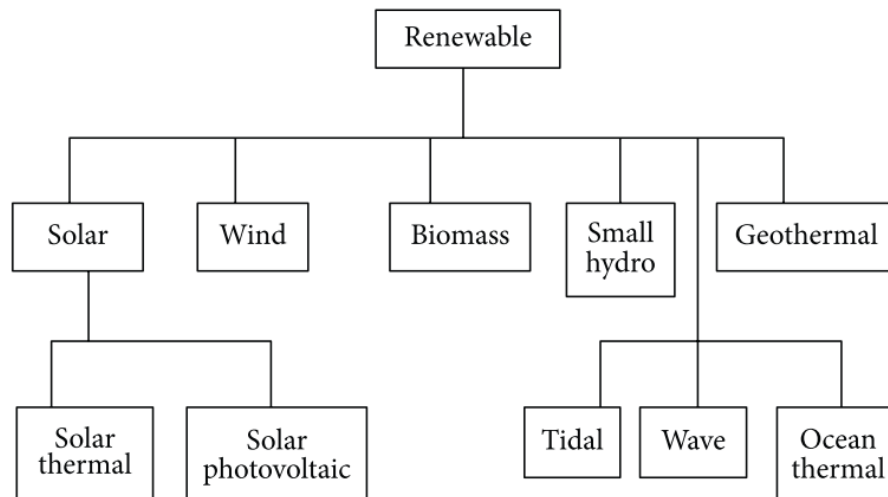
2.1 Sumber Energi Terbarukan

Komponen utama dari gas rumah kaca adalah Karbondioksida (CO₂). Hal ini menyebabkan adanya kekhawatiran global untuk mengurangi emisi karbon. Dalam hal ini, berbagai kebijakan dapat diterapkan, seperti meningkatkan penerapan energi terbarukan dan mendorong inovasi teknologi (Abolhosseini et al., 2021).

Dalam tujuan pengembangan energi terbarukan, PBB meresmikan program *Sustainable Development Goals (SDGs)* pada 25 September 2015. Salah satu tujuan *SDGs* adalah menyediakan sumber energi yang bersih dan terjangkau. Banyak negara telah memprioritaskan program dan strategi pertumbuhan nasional menggunakan *SDGs*. Hal ini dikarenakan *SDGs* menekankan lebih dari tujuan pembangunan teknologi maupun masyarakat, tetapi juga persiapan untuk menghadapi perubahan iklim, masalah pangan, air, kesehatan, serta pengembangan energi terbarukan (Neha & Rambeer, 2021).

Sejak awal abad ke-19, penggunaan bahan bakar fosil terus meningkat demi memajukan industrialisasi dan gaya hidup modern. Bahan bakar fosil digunakan dengan mengorbankan pencemaran, bahaya kesehatan dan ekologi bumi (Pandey & Katiyar, 2013). Bahan bakar fosil semakin habis setiap harinya karena sifatnya yang tidak dapat diperbaharui. Oleh karena itu, hal tersebut menimbulkan tantangan-tantangan mengenai permintaan bahan bakar, perubahan iklim, dan pertumbuhan ekonomi (Behera et al., 2018).

Energi terbarukan merupakan sumber energi tak terbatas karena mereka bertambah dengan pemanfaatan maksimum (Seydioğulları, 2013). Selain itu, energi terbarukan seperti turbin air, angin, surya, biomassa dan hidrogen lebih murah dan ramah lingkungan. Sumber-sumber energi ini juga dapat menjaga kebutuhan energi yang aman (Usman et al., 2021). Tidak seperti minyak bumi, sumber energi terbarukan memberi perlindungan terhadap lingkungan, mengurangi polusi udara, menyediakan energi yang aman, dan manfaat ekonomi lainnya (Bhowmik et al., 2017).



Gambar 2.1 Skema opsi energi terbarukan (Pandey, C. K., et al., 2013)

Untuk memenuhi kebutuhan energi yang semakin banyak, saat ini terdapat kecenderungan global untuk mengganti bahan bakar konvensional dengan sumber energi terbarukan (Borovik & Albers, 2018). Berdasarkan hasil studi oleh *International Renewable Energy Agency* berjudul *Renewables 2022*, sumber energi terbarukan menyumbang hampir 30% dari total sumber energi dunia pada tahun 2021, dan diprediksi menjadi hampir 40% pada tahun 2027 (IEA, 2022). Hal ini tentunya akan berdampak baik bagi lingkungan, maupun penanganan pasokan energi yang terus meningkat.

2.1.1.1 Radiasi Matahari

Matahari merupakan salah satu sumber energi terbarukan yang paling banyak digunakan di dunia saat ini. Berdasarkan data dari *International Renewable Energy Agency*, Energi surya menyumbang lebih dari setengah dari penambahan kapasitas energi terbarukan dengan rekor 133 GW pada tahun 2021, diikuti oleh energi angin dengan total 93 GW, dan kapasitas energi angin lepas pantai mencapai rekor 21 GW (IEA, 2022).

Energi Matahari adalah energi yang dihasilkan dari pancaran sinar matahari. Pancaran energi tersebut diukur sebagai radiasi matahari. Radiasi matahari merupakan tingkat laju saat daya radiasi matahari pada suatu permukaan dibagi dengan luas permukaan. Listrik yang didapat melalui penggunaan radiasi matahari disebut daya matahari (Burhan Uddin Shahin et al., 2020).

Energi matahari merupakan sumber energi terbarukan bersih yang muncul sebagai solusi untuk mengurangi penggunaan minyak bumi dan emisi CO₂ (Solano et al., 2022). Keuntungan lainnya dari penggunaan energi matahari adalah kekebalannya terhadap

situasi tiruan seperti fluktuasi harga minyak, mengurangi impor, serta ketergantungan pada sumber daya eksternal (Jawaid & Nazirjunejo, 2017).

Secara umum, mengukur radiasi matahari global seringkali tidak memungkinkan, karena memerlukan proses yang mahal, memakan waktu dan memerlukan proses yang presisi. Radiasi matahari dipelajari menggunakan peralatan canggih dan instrumen yang mahal, yang tentunya menuntut pemeliharaan yang baik juga. Oleh karena itu, sangat sedikit stasiun cuaca di dunia yang mengukurnya.

2.3 Jangka Waktu Prediksi Radiasi Matahari

Banyak model prediksi radiasi matahari yang telah diusulkan dalam literatur dengan jangka waktu prediksi yang berbeda-beda. Jangka waktu prediksi mulai dari jangka sangat pendek (hitungan jam), jangka pendek (1 hari kedepan), menengah (1 bulan kedepan) dan jangka panjang (1 tahun kedepan). Berbagai macam jangka waktu prediksi radiasi matahari masing-masing digunakan untuk aplikasi yang berbeda-beda. Prediksi jangka pendek dan sangat pendek dapat digunakan untuk operasi sistem tenaga secara langsung (*real-time*). Prediksi jangka pendek dapat digunakan untuk tujuan perencanaan penugasan, sedangkan prediksi jangka menengah dan panjang dapat digunakan untuk tujuan pemeliharaan dan pasar energi. (Solano et al., 2022).

2.4 Prediksi Radiasi Matahari

Sumber energi matahari tidak teratur secara alamiah, sehingga daya listrik tenaga matahari tidak teratur dan sangat bergantung pada tingkat radiasi matahari, suhu udara, dan parameter-parameter atmosfer lainnya (Behera et al., 2018). Oleh karena itu, penting untuk memprediksi data radiasi matahari dengan menggunakan parameter iklim yang mudah diukur seperti kelembapan udara, suhu udara, kecepatan angin, tutupan awan, dan sebagainya (Ağbulut et al., 2021).

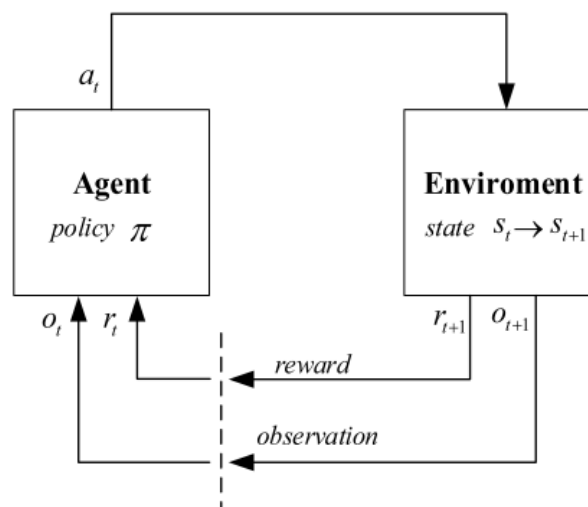
2.5 Deep Reinforcement Learning (DRL)

Ketika kita berpikir tentang alamiah manusia dalam belajar, ide awal yang muncul di benak kita adalah dengan berinteraksi langsung dengan lingkungan. Ketika kita ingin melakukan interaksi terhadap sesuatu, kita memikirkan bagaimana lingkungan sekitar menanggapi terhadap apa yang kita lakukan. Belajar melalui interaksi merupakan fondasi utama yang mendasari teori tentang kecerdasan (Barto, 2014).

Pembelajaran mesin bisa dikategorikan menjadi *supervised learning*, *unsupervised learning*, dan *reinforcement learning*. Pada *supervised learning*, terdapat data berlabel;

dalam *supervised learning* tidak terdapat data berlabel; dan dalam *reinforcement learning*, terdapat evaluasi umpan balik tanpa label (Mousavi et al., 2018).

Reinforcement Learning adalah sebuah metode dimana agen berinteraksi dengan lingkungan dan mempelajari kebijakan yang optimal dengan cara percobaan dan kesalahan untuk memecahkan masalah pengambilan keputusan (Mousavi et al., 2018). Teori *Reinforcement Learning* memberikan gambaran pada perspektif psikologis dan neurosains tentang perilaku hewan, tentang bagaimana agen dapat mengoptimalkan kontrol mereka terhadap lingkungan (Barto, 2014).



Gambar 2.2 Arsitektur Umum *Reinforcement Learning*

Dalam *Reinforcement Learning*, terdapat empat istilah penting yaitu *policy*, *reward signal*, *value function*, dan *model*. *Policy* merupakan cara bagaimana agen bertindak pada waktu tertentu. *Policy* memetakan kondisi lingkungan yang dirasakan serta tindakan yang harus diambil pada kondisi tersebut. *Reward Signal* sesuai artinya merupakan hadiah dari keputusan yang diambil. Tujuan utama agen adalah memaksimalkan total *reward* secara keseluruhan. Jika *reward* yang didapatkan tergolong rendah, *policy* akan diperbaiki agar dapat menyesuaikan dengan keadaan yang sama di masa depan (Barto, 2014).

Istilah dalam *reinforcement learning* selanjutnya adalah *value function*. *Value function* menentukan apa yang terbaik untuk dilakukan dalam jangka Panjang. Dengan kata lain, *value function* merupakan total *reward* yang diakumulasikan dalam satu jangka waktu (*state*) (Barto, 2014).

Istilah keempat dan terakhir dari istilah penting dalam *reinforcement learning* yaitu *model*. *Model* merupakan sesuatu yang berusaha meniru kebiasaan lingkungan dan

menyimpulkan bagaimana lingkungan akan bereaksi. Dalam penerapannya di *reinforcement learning*, *model* digunakan saat melakukan prediksi untuk keadaan selanjutnya yang akan terjadi (Barto, 2014).

Istilah *Deep Reinforcement Learning* dihasilkan dari penggunaan algoritma *Deep Learning* didalam *Reinforcement Learning*. Penggunaan *Deep Learning* memungkinkan algoritma *Reinforcement Learning* untuk menyelesaikan masalah pengambilan keputusan yang sebelumnya sulit diatasi seperti pengaturan dengan tindakan dengan dimensi yang cukup tinggi (Arulkumaran et al., 2017).

Salah satu alasan diciptakannya *Deep Reinforcement Learning* adalah menghasilkan sistem yang mampu belajar bagaimana beradaptasi pada lingkungan dunia nyata seperti pengaturan konsumsi daya listrik. Algoritma *Deep Reinforcement Learning* juga telah diaplikasikan untuk menyelesaikan berbagai jenis permasalahan, seperti robotika, dimana aturan mengenai tindakan kini dapat dipelajari langsung dari kamera (Arulkumaran et al., 2017).

2.6 Long Short-Term Memory

Jaringan Long Short-Term Memory (LSTM) terdiri dari sejumlah blok memori yang tersambung melalui lapisan yang terbuat dari kumpulan sel memori dengan koneksi berulang (*recurrent connections*) (Ozdemir & Ozkan Yildirim, 2023). Struktur LSTM terdiri dari tiga gerbang utama yaitu *input gate*, *output gate* serta *forget gate*. *Forget gate* (f_t) dapat dinyatakan pada rumus 2.1

$$f_t = \delta(W_f[h_{t-1}, X_t] + b_f) \quad (2.1)$$

Dimana δ merupakan sigmoid activation function, X_t sebagai vektor dari masukan, h_{t-1} menyatakan vektor *hidden* dari layer sebelumnya, serta W_f dan b_f menyatakan *weight* dan *bias* dari masing-masing *input*. *Input gate* (i_t) dapat dinyatakan pada persamaan 2.2

$$i_t = \delta(W_i[h_{t-1}, X_t] + b_i) \quad (2.2)$$

Output gate (o_t) mengendalikan *output* dari nilai *cell state* (C_t) dan dapat dinyatakan pada persamaan 2.3.

$$O_t = \delta(W_o[h_{t-1}, X_t] + b_o) \quad (2.3)$$

Sebagai tambahan, kandidat memory cell dapat dinyatakan pada persamaan 2.4

$$L_t = \tanh(W_c[h_{t-1}, X_t] + b_c) \quad (2.4)$$

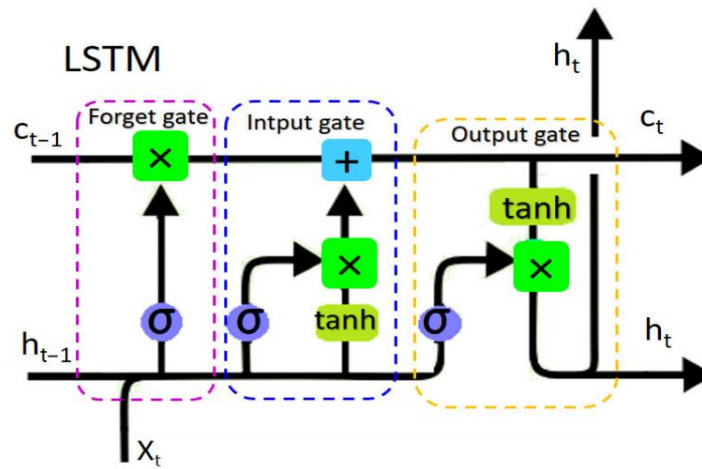
Representasi terkini dari *cell state* (C_t) saat ini adalah sebagai persamaan 2.5

$$C_t = f_t \times C_{t-1} + i_t \times L_t \quad (2.5)$$

Output akhir dari hidden state terkini dan penggunaannya sebagai input untuk *layer LSTM* berikutnya dapat dinyatakan pada persamaan 2.6

$$h_t = O_t \times \tanh(C_t) \quad (2.6)$$

Struktur dasar dari LSTM dapat dilihat pada **Gambar 2.3**.



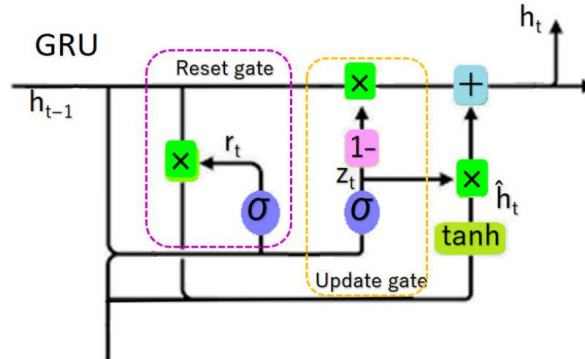
Gambar 2.3 Arsitektur *Long Short-Term Memory (LSTM)* (Ozdemir & Ozkan Yildirim, 2023)

2.7 Gated Recurrent Unit

Gated Recurrent Unit (GRU) adalah jenis arsitektur *Recurrent Neural Network (RNN)* yang diperkenalkan sebagai perbaikan dari *RNN* tradisional untuk mengatasi masalah *gradient* yang menghilang dan menangkap ketergantungan jangka panjang secara lebih efektif (Shen et al., 2018). *GRU* merupakan modifikasi dari arsitektur *Long Short-Term Memory (LSTM)* standar. Dibandingkan dengan *LSTM*, metode ini memiliki jumlah parameter yang lebih sedikit, sehingga memungkinkan waktu komputasi yang lebih cepat (Zarzycki & Ławryńczuk, 2022).

GRU terdiri dari serangkaian gerbang yang mengatur aliran informasi dalam jaringan. Gerbang-gerbang ini dirancang untuk secara selektif memperbarui dan mereset keadaan tersembunyi, memungkinkan model untuk mempertahankan informasi penting dan

menghapus informasi yang tidak relevan atau redundan. Dua gerbang utama dalam *GRU* adalah *update gate* dan *reset gate*.



Gambar 2.4 Arsitektur Gated Recurrent Unit (GRU) (Ozdemir & Ozkan Yildirim, 2023)

Langkah pertama dalam pembuatan model *GRU* adalah menghitung *update gate* (z_t) menggunakan rumus (2.7) yang digunakan untuk mengukur seberapa banyak informasi sebelumnya yang harus dipertahankan.

$$z_t = \sigma(w^{(z)}x_t + u^{(z)}h_{t-1} + b) \quad (2.7)$$

Dimana w dan u sebagai *weight*, x_t sebagai masukan, h_{t-1} sebagai *hidden state*, dan b sebagai *bias*.

Langkah selanjutnya adalah menghitung *reset gate* (r_t) menggunakan rumus (2.8) yang digunakan untuk mengukur seberapa banyak informasi yang harus dibuang dan bagaimana cara menggabungkan masukan baru dengan informasi sebelumnya.

$$r_t = \sigma(w^{(r)}x_t + u^{(r)}h_{t-1} + b) \quad (2.8)$$

Selanjutnya adalah menghitung calon *hidden state* (h'_t) yang akan digunakan *reset gate* untuk menyimpan informasi yang bersangkutan dari informasi sebelumnya. Calon *hidden state* dapat dilihat pada persamaan 2.9.

$$h'_t = \tanh(Wx_t + r_t \odot Uh_{t-1}) \quad (2.9)$$

Langkah terakhir adalah menghitung *hidden state* (h_t) menggunakan persamaan 2.10.

$$h_t = z_t \odot h_{t-1} + (1 - z_t) \odot h'_t \quad (2.10)$$

2.8 Penelitian Terdahulu

Beberapa penelitian mengenai prediksi radiasi matahari telah dilakukan dengan hasil yang beragam. Ağbulut, Ü., et al. (2021) melakukan penelitian berjudul “*Prediction of daily global solar radiation using different machine learning algorithms: Evaluation and comparison*” untuk membandingkan algoritma-algoritma kecerdasan buatan yang digunakan untuk memprediksi radiasi matahari harian. Penelitian tersebut menggunakan data dari *The Turkish State Meteorological Service (TSMS)* yang berisi parameter seperti, kelembapan udara, suhu, kecepatan angin, panjang hari, tutupan awan, serta radiasi matahari dari 4 stasiun cuaca di Turki yaitu Kırklareli, Tokat, Nevsehir and Karaman. Dengan menggunakan algoritma *Support Vector Machine (SVM)*, *Artificial Neural Network (ANN)*, *Kernel and Nearest-Neighbor (k-NN)*, dan *Deep Learning (DL)* dan dalam waktu prediksi 1 hari kedepan menghasilkan hasil dengan nilai *Mean Root Squared Error (MRSE)* berkisar antara 2,157 MJ/m² hingga 3,092 MJ/m² diantara keempat stasiun cuaca. Namun dari beberapa algoritma yang diuji belum ada algoritma yang menghasilkan hasil terbaik di semua tempat (Ağbulut et al., 2021).

Modeste Kameni Nematchoua, et al. (2022) melakukan penelitian untuk memprediksi radiasi matahari harian pada 27 negara di eropa berjudul “*Prediction of daily global solar radiation and air temperature using six machine learning algorithms; a case of 27 European countries*”. Penelitian tersebut menggunakan beberapa algoritma kecerdasan buatan yaitu *Linear Model (LM)*, *Decision Tree (DT)*, *Support Vector Machine (SVM)*, *Deep Learning (DL)*, *Random Forest (RF)* dan *Gradient Boosted Tree (GBT)*. Parameter yang digunakan untuk melatih data prediksi yaitu suhu udara, kecepatan angin, kelembapan udara dan besaran radiasi matahari. Penelitian ini menggunakan dataset dari 27 stasiun metereologi dengan ketinggian 100 m yang tersebar di masing-masing negara yang diteliti pada penelitian ini. Diantara beberapa algoritma yang digunakan, *SVM* dan *DL* menghasilkan hasil paling baik dengan nilai *RMSE* 0,189 MJ/m² untuk algoritma *SVM* dan 0,235 MJ/m² untuk algoritma *DL*. Sebaliknya, algoritma *Decision Tree* menghasilkan hasil terburuk dengan nilai *RMSE* mencapai 0,405.

Edna S. Solano et al. (2022) melakukan penelitian mengenai prediksi radiasi matahari berjudul “*Solar Radiation Forecasting Using Machine Learning and Ensemble Feature Selection*”. Penelitian tersebut menggunakan algoritma kecerdasan buatan yaitu *Support Vector Machine (SVM)*, *Extreme Gradient Boosting (XGBT)*, *Categorical Boosting*

(*CatBoost*) dan *Voting-Average (VOA)*. Dengan menggunakan database dari *The Brazillian National Institute of Meteorology Website (INMET)* dan jangka waktu prediksi 1-3 jam kedepan, menghasilkan hasil dengan nilai error terendah pada algoritma *VOA* dengan nilai *Mean Root Squared Error (MRSE)* sebesar 0.4326 MJ/m^2 pada musim panas dan 0.3431 MJ/m^2 pada musim dingin. Namun, distribusi error akan menjadi semakin besar seiring waktu prediksi yang meningkat (Solano et al., 2022).

Jawaid, F., et al., (2017) melakukan penelitian mengenai prediksi radiasi matahari harian berjudul “*Predicting Daily Mean Solar Power Using Machine Learning Regression Techniques*”. Penelitian ini menggunakan data informasi suhu di Los Angeles dan California yang dikelola oleh *National Oceanic and Atmospheric Administration (NOAA)*. Data yang dikumpulkan berupa suhu udara, kecepatan angin, titik embun, jarak pandang, sudut azimuth matahari dan sudut zenit matahari yang dikumpulkan selama 10 tahun. Penelitian ini menggunakan empat algoritma kecerdasan buatan yaitu *K-Nearest Neighbor (K-NN)*, *Linear Regression (LR)*, *Support Vector Machines (SVM)* dan *Artificial Neural Network (ANN)*. Beberapa algoritma yang digunakan menghasilkan hasil yang beragam dengan nilai *Mean Root Squared Error (MRSE)* yang cukup rendah untuk algoritma *ANN* dan pada algoritma *SVM*, nilai error berkurang secara signifikan seiring jangkauan data yang ditambah.

Pada penelitian “*Deep Reinforcement Learning for Stock Prediction*”, algoritma DQN digunakan dalam pembangunan model kecerdasan buatan untuk memprediksi harga saham. Ide utama dari penelitian ini didasari dengan ketidakstabilan nilai harga saham. Beberapa model dasar pembelajaran mesin, seperti feedforward neural networks (FFNN) dan model-model deep learning lain seperti recurrent neural network (RNN) dengan struktur memori, tidak mampuu memprediksi data dengan deret waktu yang tidak stabil dan autoregresi jangka panjang. Hasilnya hasil tes menunjukkan akurasi yang lebih tinggi dan lebih mampu beradaptasi dengan fluktuasi pasar (Zhang & Lei, 2022).

Penelitian sebelumnya seperti (Ağbulut et al., 2021) menggunakan jumlah data yang tergolong sedikit hanya selama 2 tahun. Tentunya dengan meningkatkan jumlah data diharapkan dapat menghasilkan hasil prediksi yang lebih akurat. Pada penelitian lainnya juga seperti (Edna S. Solano et al., 2022), data diambil di negara yang memiliki musim dingin yaitu Brazil. Pada penelitian ini, pengumpulan data dilakukan di Medan, Sumatera Utara, Indonesia yang memiliki curah matahari yang lebih stabil tanpa musim dingin.

Tentunya dengan adanya hal ini, diharapkan dapat menyelesaikan masalah penelitian sebelumnya yaitu masalah pemisahan data yang sulit dan tidak berkelanjutan.

Penelitian lainnya seperti (Bendali et al., 2022) juga mencoba melakukan prediksi radiasi matahari menggunakan beberapa algoritma prediksi seperti *Recurrent Neural Network (RNN)*, *Long Short-Term Memory (LSTM)*, *Multilayer Perceptron (MLP)*, simple *GRU* dan *Gated Recurrent Unit-Principal Component Analysis (GRU-PCA)*. Hasil penelitian menunjukkan algoritma *GRU* menghasilkan hasil prediksi terbaik dengan tingkat *error* terendah. Hal tersebut juga yang mendasari penulis memilih menggunakan algoritma *GRU* sebagai algoritma prediksi dalam penelitian ini.

2.9 Perbedaan Penelitian

Penelitian ini tentunya memiliki perbedaan dan kesamaan dengan beberapa penelitian yang telah diuraikan diatas. Perbedaan yang terdapat dalam penelitian ini dengan penelitian sebelumnya yang pernah dilakukan mencakup berbagai hal. Perbedaan yang pertama yaitu mengenai data yang digunakan dalam penelitian ini. Pada penelitian (Ağbulut et al., 2021) menggunakan parameter meteorologi yaitu suhu, tutupan awan, dan besaran radiasi matahari sebagai parameter meteorologi. Pada penelitian ini parameter suhu dan radiasi matahari tetap digunakan, dengan tambahan parameter seperti kelembapan udara dan titik embun.

Parameter seperti kelembapan udara digunakan karena dinilai dapat memengaruhi jumlah radiasi yang mencapai permukaan bumi. Hal ini dikarenakan kelembapan yang tinggi dapat meningkatkan penyerapan radiasi matahari oleh uap air di atmosfer, sehingga dapat mengurangi jumlah radiasi yang mencapai permukaan bumi. Begitu juga dengan titik embun, Peningkatan titik embun dapat mengindikasikan peningkatan kelembapan udara, yang dapat meningkatkan absorpsi dan penyebaran radiasi oleh uap air.

Perbedaan juga berbeda dari segi tempat penelitian. Penelitian ini menggunakan data dari penelitian *Development of software for Estimating Clear Sky Solar Radiation in Indonesia* (Ambarita, 2017). Penelitian tersebut menghasilkan data meteorologi yang digunakan untuk membangun model kecerdasan buatan pada penelitian ini. Berbeda dengan penelitian sebelumnya seperti (Jawaid & Nazirjunejo, 2017) dan (Tingting Zhu 1, 2,* & Zhenye Li 1 and Cong Wang 1, 2021) yang menggunakan data meteorologi yang diambil di negara Amerika Serikat, ataupun penelitian seperti (Ağbulut et al., 2021) yang menggunakan data dari negara turki yang memiliki 4 musim, data yang digunakan di

penelitian ini diambil di Kota Medan, Indonesia yang hanya memiliki 2 musim dengan suhu udara yang lebih stabil. Di negara tropis seperti Indonesia, variasi suhu tergolong lebih stabil, serta memiliki intensitas cahaya matahari yang tinggi sepanjang tahun. Dalam ruang lingkup penggunaan energi, hal ini tentunya dapat memberikan dasar yang lebih baik untuk mengidentifikasi tren, risiko, dan peluang yang relevan dengan kondisi setempat.

Dari segi algoritma, penelitian ini akan membandingkan 3 jenis algoritma yaitu *LSTM*, *GRU* dan *DQN*. Nantinya, penelitian akan menghasilkan *output* berupa hasil prediksi radiasi matahari dengan jangka mulai dari 7 hari, 30 hari, 60 hari hingga 90 hari.

BAB 3

ANALISIS DAN PERANCANGAN SISTEM

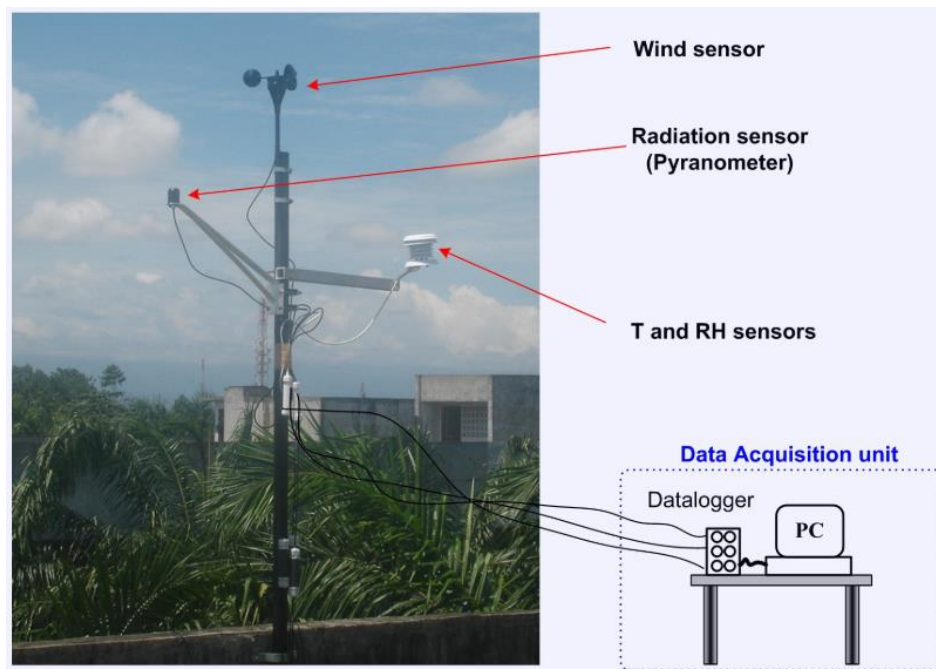
3.1 Data yang digunakan

Penelitian ini menggunakan data numerik yang berasal dari dataset penelitian bernama *Development of software for Estimating Clear Sky Solar Radiation in Indonesia* oleh *Himsar Ambarita* pada tahun 2017. Data yang didapat dari penelitian tersebut berupa data berformat *csv* berjumlah 63 dokumen yang tersebar pada tahun 2017 hingga 2022. Rincian jumlah data dapat dilihat pada **Tabel 3.1**.

Tabel 3.1 Rincian jumlah data yang digunakan

Tahun	Bulan	Jumlah
2017	Oktober - Desember	3 bulan
2018	Januari – Desember	12 bulan
2019	Januari - Desember	12 bulan
2020	Januari - Desember	12 bulan
2021	Januari - Desember	12 bulan
2022	Januari - Desember	12 bulan
Total		63 bulan

Masing-masing dokumen berisi data yang didapat dari hasil sensor penelitian (Ambarita, 2017), dengan jumlah 3 sensor yaitu, sensor kecepatan angin, sensor radiasi matahari, serta sensor suhu dan kelembapan udara dengan rentang pengukuran masing-masing 0-76 m/s, 0-1280 W/m², dan -40°C to 75°C dengan jangka pengukuran dilakukan tiap 2 menit.



Gambar 3.1 Alat pengukur radiasi matahari pada penelitian (Ambarita, 2017)

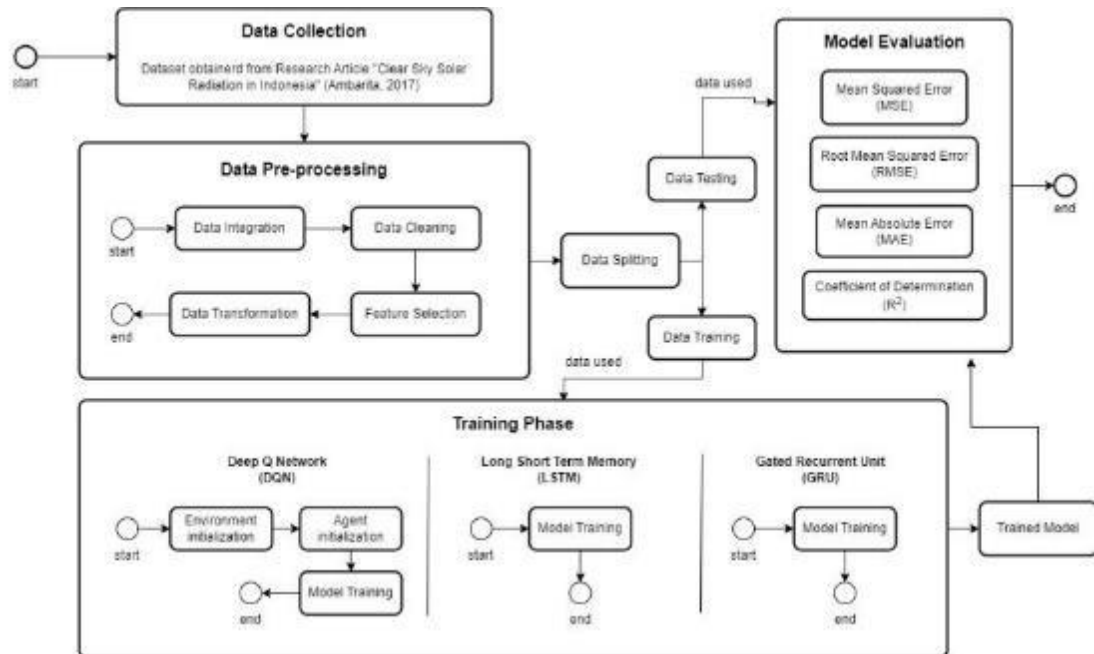
3.2 Arsitektur Umum Sistem

Arsitektur umum penelitian ini dapat dilihat pada **Gambar 3.2**. Untuk melakukan proses prediksi radiasi matahari dilakukan melalui beberapa tahapan. Data yang sebelumnya berbentuk banyak file, disatukan menjadi satu file agar mudah untuk melakukan pelatihan serta pengujian model nantinya. Untuk tujuan pengujian, data yang didapat akan dibagi menjadi 2 bagian yaitu *data training* dan *data testing*. Ukuran masing-masing data training dan data testing akan bervariasi berdasarkan jangka waktu mulai dari 7 hari, 30 hari, 60 hari hingga 90 hari.

Selanjutnya, akan dilakukan *pre-processing* pada data agar data yang digunakan untuk melatih model nantinya dapat bersih dan memudahkan proses pelatihan data. Setelah proses *pre-processing* selesai, akan dimulai proses pelatihan *model* dengan algoritma Hibridisasi *DQN* dan *GRU*. Output dari proses *Model Training* adalah *model* kecerdasan buatan yang dapat digunakan untuk memprediksi besaran radiasi matahari selama mulai dari 7, 30, 60 hingga 90 hari kedepan.

Setelah semua proses pelatihan *model* selesai, *model* yang telah didapat sebelumnya akan digunakan dalam proses pengujian data. Dalam Proses ini, *model* akan mencoba melakukan prediksi selama jangka waktu prediksi dan akan dibandingkan dengan *data*

testing untuk melihat seberapa banyak kesalahan prediksi yang nantinya akan menentukan kualitas *model* kecerdasan buatan.



Gambar 3.2 Arsitektur Umum Sistem

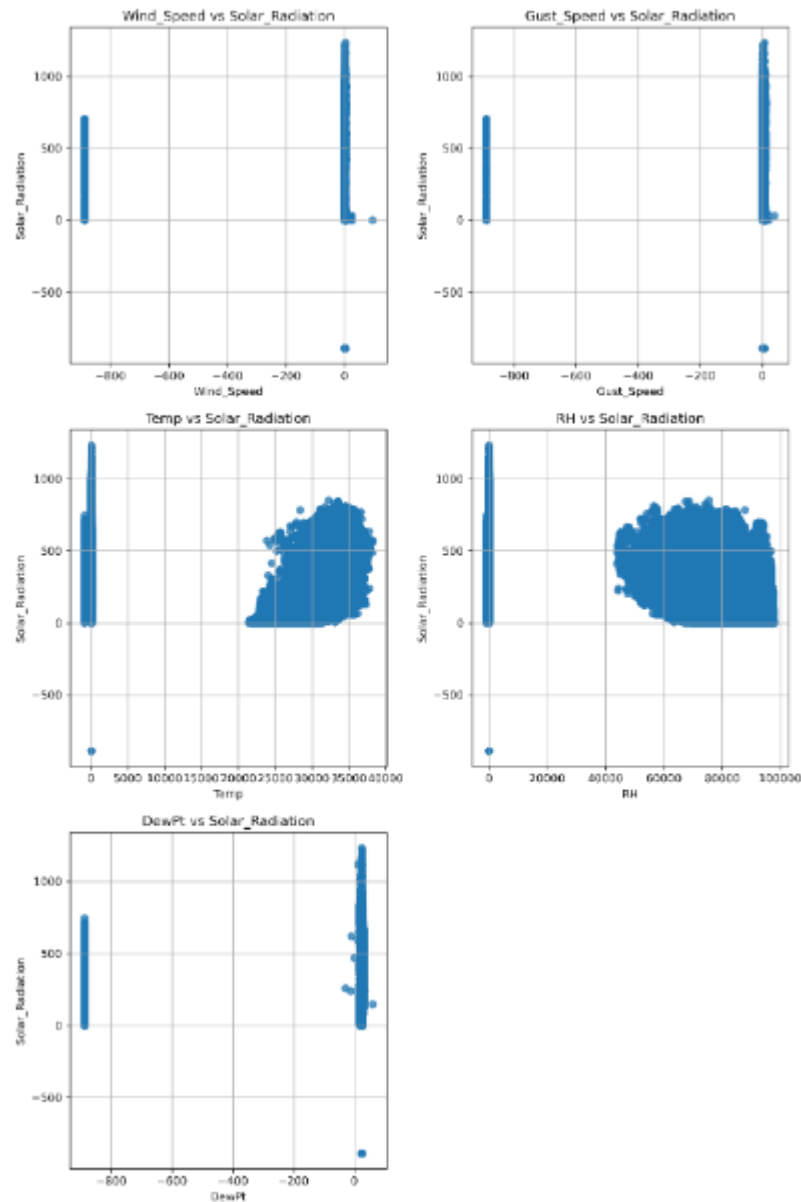
3.2.1 Data Collection

Data yang digunakan pada penelitian ini menggunakan data pada penelitian (Ambarita, 2017). Data yang didapat dari penelitian ini berupa dokumen berformat .csv dengan keterangan seperti pada **Tabel 3.1**. Setiap dokumen berisi data dengan parameter tanggal dan waktu tercatat, kecepatan angin dan kecepatan angin maksimum, suhu dan kelembapan udara, titik embun, serta besaran radiasi matahari. Sayangnya untuk data kecepatan angin dan kecepatan angin maksimum tidak dapat digunakan dalam penelitian ini dikarenakan data yang didapat kurang relevan karena pada mayoritas waktu data kecepatan angin dan kecepatan angin maksimum tidak tercatat pada dokumen.

3.2.2 Data Pre-Processing

Data yang didapat dari penelitian (Ambarita, 2017) masih memerlukan tahapan pemrosesan lanjutan sebelum dapat digunakan secara efektif. Dalam konteks ini, diperlukan upaya lebih lanjut untuk membersihkan dan menyaring data agar dapat memenuhi standar kualitas yang diperlukan melalui proses *data pre-processing*. *Data pre-processing* merupakan langkah awal sebelum melakukan *model training*. Pada tahap ini data yang sudah disatukan menjadi satu dokumen dengan format csv akan dibersihkan

dan di normalisasi sebelum dilanjutkan ke proses selanjutnya. Pada penelitian ini, dilakukan 4 proses *data pre-processing* yaitu *data integration*, *data cleaning*, *feature selection* dan *data transformation*.



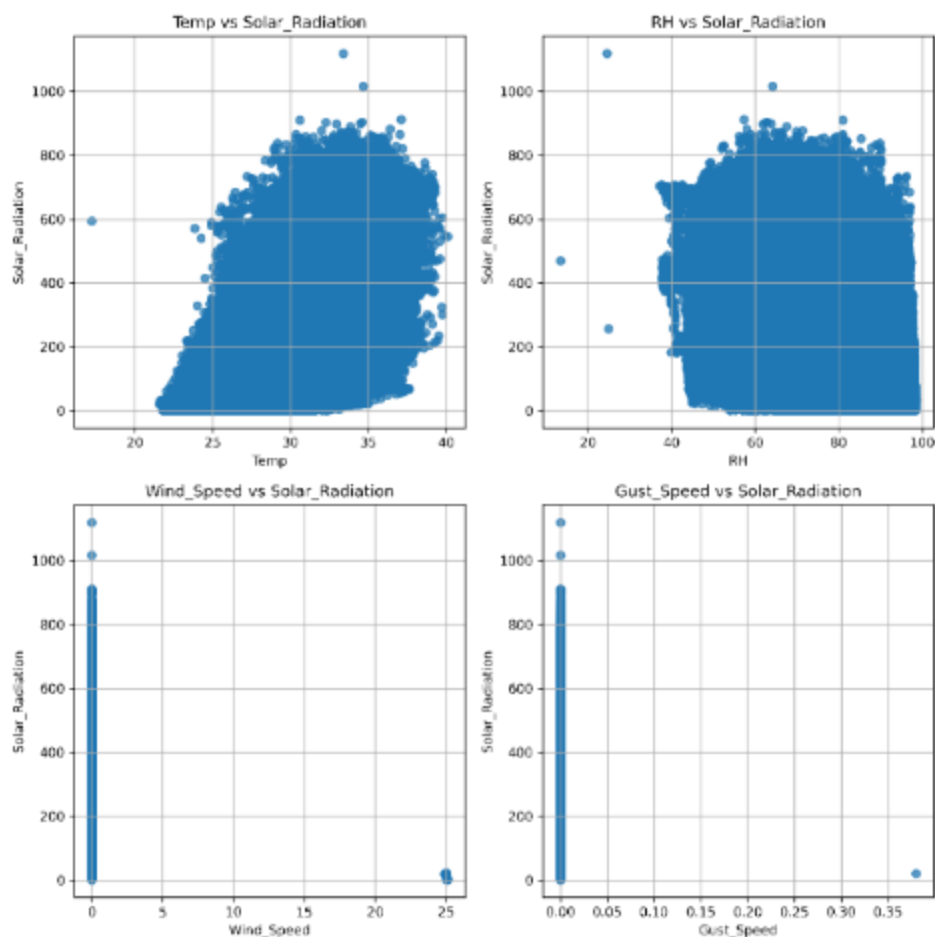
Gambar 3.3 Distribusi data suhu dan radiasi matahari sebelum proses *data cleaning*

Data yang didapat dari dataset penelitian ini masih memiliki beberapa masalah, yaitu judul kolom yang tidak konsisten dikarenakan jumlah data yang sangat besar, data yang di dapat dari sensor dibagi-bagi menjadi beberapa dokumen dan dalam setiap dokumen dibagi dalam beberapa halaman berdasarkan hari. Sebelum dapat melanjutkan ke tahapan selanjutnya, tentunya harus dilakukan integrasi data agar data dapat diolah secara holistik dan menyeluruh. Integrasi data menjadi langkah krusial sebelum melanjutkan ke tahapan selanjutnya dalam penelitian atau analisis data. Proses ini bertujuan untuk menyatukan

informasi dari berbagai sumber atau file data dengan tujuan menciptakan dataset yang lebih lengkap, konsisten, dan representatif.

Terdapat beberapa langkah yang dilakukan dalam tahapan *data integration* dalam penelitian ini. Langkah pertama adalah menyatukan seluruh dokumen menjadi satu dokumen yang terintegrasi. Nantinya, dokumen akan berisi satu table dengan parameter *Date_Time*, *Wind_Speed*, *Gust_Speed*, *Temp*, *RH*, dan *Solar_Radiation*.

Setelah tahapan *data integration*, masih banyak data salah yang kemungkinan diakibatkan adanya masalah yang terjadi saat pengambilan data oleh sensor. Data sebelum dilakukan tahapan *data cleaning* dapat dilihat pada **Gambar 3.3**.



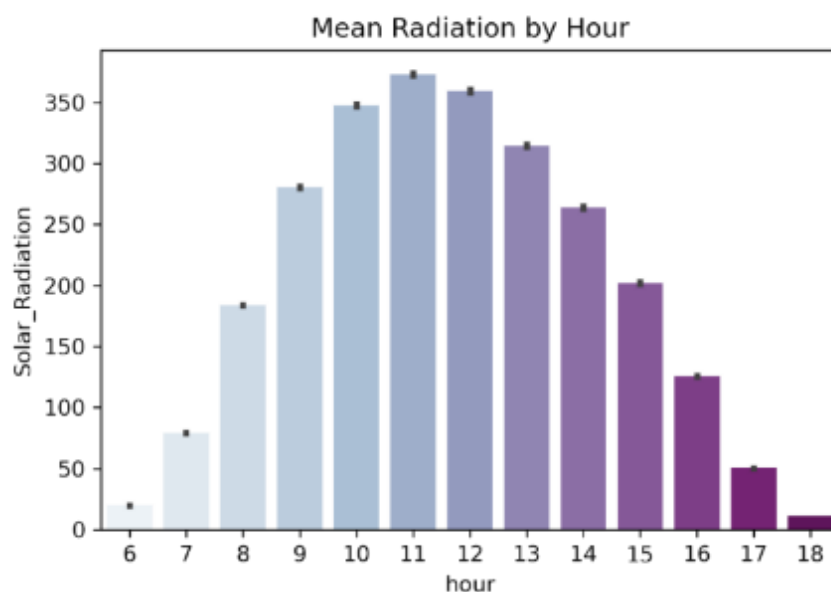
Gambar 3.4 Distribusi data suhu dan radiasi matahari setelah proses *handling outliers*

Untuk menghilangkan *outliers* pada data, ditentukan beberapa aturan yang disesuaikan dengan penelitian yang menghasilkan dataset yaitu sebagai berikut:

1. Besaran radiasi matahari harus berada dalam jangka 0-1280 W/m² dikarenakan nantinya data yang akan digunakan hanyalah saat matahari berada dalam siklus penyinaran.
2. Besaran suhu udara harus berada di rentang 0 hingga 75°C. Hal ini ditentukan berdasarkan standar suhu di Indonesia serta menyesuaikan dengan distribusi data sebelumnya.
3. Kelembapan udara dan titik embun dan kelembapan udara harus lebih besar dari 0% menyesuaikan dengan distribusi data dan jangkauan sensor.

Saat menghilangkan *outliers*, perlu dilakukan penghapusan fitur untuk parameter *DewPt* dikarenakan 16,1% data dari keseluruhan dataset tidak tercatat. Hasil dari proses ini dapat dilihat pada **Gambar 3.4**.

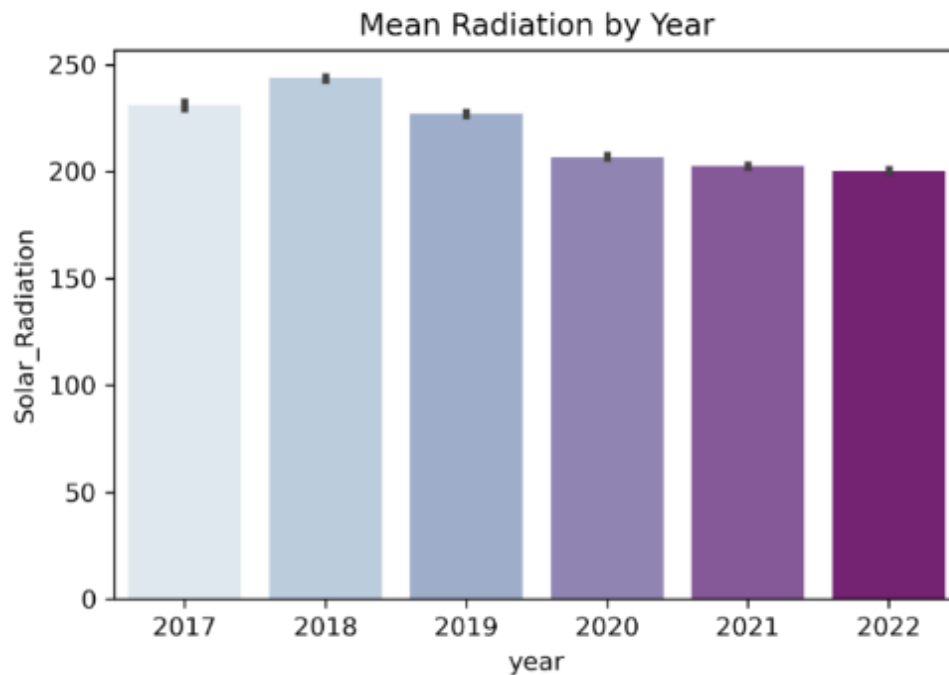
Sebelum memulai tahapan selanjutnya, seperti yang dapat dilihat pada **Gambar 3.4**, perlu dilakukan penghapusan fitur kembali untuk parameter *Wind Speed* dan *Gust Speed* dikarenakan data yang tidak tercatat dan berkumpul pada kecepatan 0 m/s.



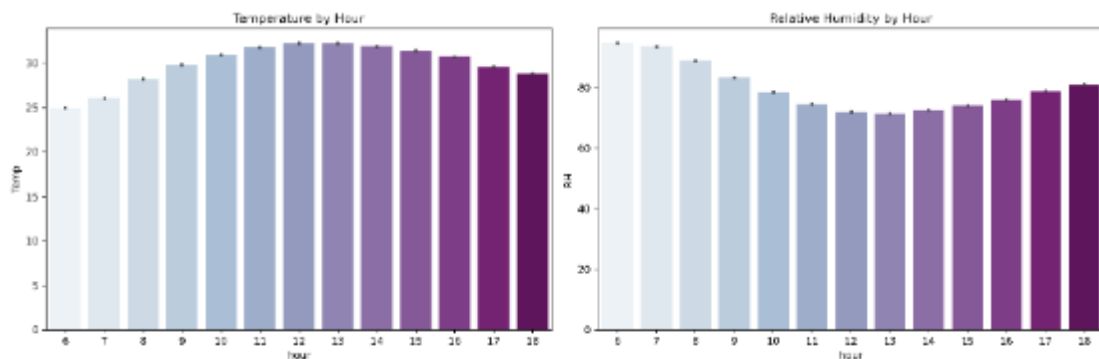
Gambar 3.5 Rata-rata besaran radiasi matahari untuk setiap jam setelah proses *data cleaning*

Setelah menghilangkan *outliers*, data akan dibersihkan lagi untuk memastikan data yang akan digunakan di proses selanjutnya hanyalah data yang diperlukan. Terdapat beberapa tahapan yang dilakukan dalam proses ini. Tahapan pertama adalah menyaring waktu penyinaran. Karena pada data yang didapat tidak ada label mengenai kapan data

penyinaran matahari dimulai dan berakhir, berdasarkan analisis dari data yang ada ditetapkan waktu penyinaran untuk setiap harinya dimulai dari saat pertama kali terjadi peningkatan besar radiasi matahari diatas 0 W/m^2 . Data selain pada masa penyinaran akan dihapus.



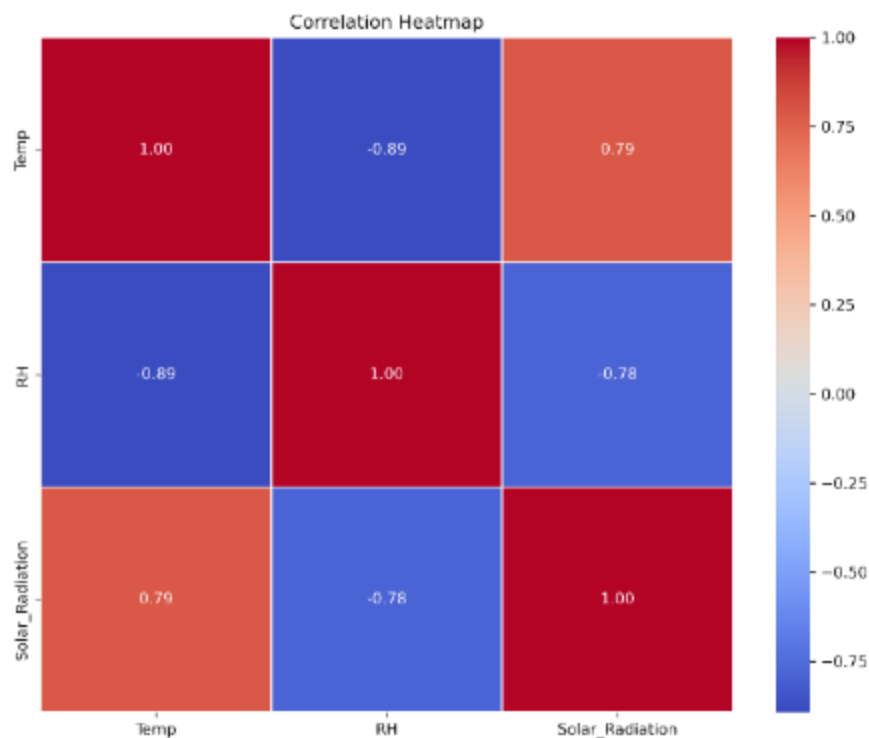
Gambar 3.6 Rata-rata besaran radiasi matahari untuk setiap tahun setelah proses *data cleaning*



Gambar 3.7 Rata-rata persentase suhu dan kelembapan udara untuk setiap jam setelah proses *data cleaning*

Selanjutnya data-data kosong dan *outliers* akan dilakukan pengisian menggunakan rata-rata data pada dataset. Hal ini dilakukan untuk menjaga konsistensi struktur data, terutama dikarenakan data yang digunakan adalah data *time-series*.

Setelah seluruh proses *data cleaning* dilakukan grafik rata-rata besaran radiasi matahari dapat dilihat pada **Gambar 3.5** dan **Gambar 3.6**, begitu juga dengan rata-rata persentase kelembapan udara yang dapat dilihat pada **Gambar 3.7**. Proses terakhir adalah menyederhanakan data dengan menghitung rata-rata nilai tiap kolomnya dengan pengelompokan tiap harinya. Hal ini dilakukan untuk menyesuaikan tujuan jangka prediksi yang selama 1 minggu hingga 4 bulan. Selain itu, hal ini juga dilakukan untuk mengurangi waktu komputasi dan kinerja *model* yang membutuhkan waktu cukup lama dalam proses pelatihan.



Gambar 3.8 Diagram *Correlation Heatmap* dari dataset

Pada tahapan selanjutnya yaitu *feature selection*, akan dilakukan pemilihan fitur yang akan digunakan saat pelatihan model. Selain dapat meningkatkan efisiensi komputasi, tahapan *feature selection* juga dapat mengurangi *noise* dan memperbaiki akurasi model dengan menghilangkan fitur yang tidak relevan atau redundan. Dengan mengurangi jumlah fitur, resiko *overfitting* pada *training data* juga berkurang karena *model* menjadi lebih sederhana. Gambaran relevansi fitur dapat digambarkan melalui diagram *correlation heatmap* yang dapat dilihat pada **Gambar 3.8**.

Dilihat dari diagram *correlation heatmap* pada **Gambar 3.8**, fitur suhu udara (*Temp*) dan kelembapan udara (*RH*) dapat digunakan pada proses selanjutnya dikarenakan korelasi yang cukup tinggi masing-masing 0.79 dan -0.78.

Pada penelitian ini, proses selanjutnya yaitu *data transformation* akan menggunakan metode *min-max scaling*, dengan menyesuaikan data dalam rentang mulai dari nilai minimum hingga nilai maksimum. Penelitian ini menggunakan rentang nilai mulai dari 0 hingga 1. Uraian matematis dari *min-max scaling* dapat dilihat pada persamaan 3.1 dan 3.2.

$$X_{std} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (3.1)$$

$$X_{sc} = X_{std} \times (max - min) + min \quad (3.2)$$

Dengan X_{min} dan X_{max} masing-masing sebagai nilai minimum dan maksimum dataset, min dan max masing-masing sebagai batas bawah dan batas atas, serta X sebagai nilai yang ingin diubah.

3.2.3 Data Splitting

Setelah fase *data preprocessing* selesai dilakukan, data perlu dibagi menjadi beberapa bagian. Sebagian data akan digunakan pada proses *training*, dan sebagian lain akan digunakan untuk kebutuhan validasi data pada proses *testing*. Pada penelitian ini, data akan dibagi menjadi dua bagian yaitu *data training* dan *data testing*. Jumlah data testing yang akan dilakukan pengujian adalah 7 hari, 30 hari, 60 hari serta 90 hari. Untuk jumlah *data training* akan disesuaikan sebanyak jumlah data secara keseluruhan dikurangi jumlah *data testing*.

3.2.4 Proses Training

Dalam pelatihan model kecerdasan buatan menggunakan algoritma *DQN*, terdapat dua parameter yang akan digunakan model untuk proses pelatihan yaitu *state space* (s_t) dan *action space* (a_t). *State space* merujuk pada semua kemungkinan keadaan atau situasi yang dapat diamati oleh agen pada suatu waktu tertentu. Keadaan dapat berupa representasi numerik atau vektor yang mencerminkan informasi penting tentang lingkungan saat ini. Pada penelitian ini, untuk melakukan prediksi, *state space* yang digunakan merupakan parameter prediksi tanpa termasuk dengan parameter hasil. Pada

penelitian ini, terdapat 7 parameter prediksi yang digunakan sebagai *state space* yaitu bulan (0-11), hari dalam bulan (0-30), suhu udara, serta tingkat kelembapan udara.

Pada penelitian ini, proses prediksi menggunakan algoritma *DQN* tidak dilakukan dengan regresi dikarenakan *DQN* menggunakan sistem perhitungan *q-value*. Metode *DQN* berfokus pada pengambilan keputusan tindakan. Oleh karena itu, harus didefinisikan *action space* (a_t) sebagai acuan keputusan. *Action space* merupakan daftar keputusan yang dapat diambil dalam proses training *DQN*. Dalam penelitian ini, *action space* didefinisikan sebagai rentang hasil prediksi yang ditentukan.

$$S_t = \{s_1, s_2, \dots, s_n\} \quad (3.3)$$

$$a_t = \{a_i\} \text{ dimana } a_i = 0 + 2 \cdot i, \text{ untuk } i = 0, 1, 2, \dots, n - 1 \quad (3.4)$$

Pada algoritma *DQN* juga dikenal istilah *agent*. *Agent* mengacu pada entitas yang belajar dan berinteraksi dengan lingkungan untuk mencapai tujuan tertentu. *Agent* merupakan komponen utama yang menggunakan algoritma pembelajaran mesin untuk memahami lingkungan, membuat keputusan berdasarkan *action space*, dan memaksimalkan total *reward* (r_t) yang diperoleh dari interaksi dengan lingkungan.

Pada penelitian ini, terdapat 3 fungsi utama yang terdapat dalam *agent*, yaitu inisialisasi, *update*, dan *act*. Inisialisasi digunakan untuk mendefinisikan *agent* melalui beberapa parameter seperti model kecerdasan buatan, *learning rate* (α), *discount factor* (γ), serta *optimizer*. Fungsi *update* bertanggung jawab untuk melakukan pembaruan terhadap model berdasarkan pengalaman yang diperoleh saat melakukan interaksi dengan lingkungan. Sedangkan fungsi *act* digunakan untuk melakukan pengambilan keputusan berdasarkan *state* saat itu. Dalam fungsi ini juga dilakukan perhitungan nilai target *q-value* ($Q(s_t, a_t)$) yang spesifik terhadap kondisi *state* (s_i) disaat itu. Fungsi *act* akan memilih *action* (a_i) dengan nilai *q-value* tertinggi. Berdasarkan algoritma dasar *q-learning*, nilai *q-value* akan diperbarui menggunakan persamaan 3.5.

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha \left[r_t + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t) \right] \quad (3.5)$$

Pada algoritma *DQN*, dibutuhkan lingkungan yang merupakan tempat *agent* melakukan pembelajaran. Dalam penelitian ini, terdapat 3 fungsi utama yang terdapat dalam *environment* yaitu inisialisasi, *reset*, dan *step*. Dalam fungsi inisialisasi akan didefinisikan beberapa parameter berupa *dataset* penelitian (*data*), *action space*, serta

state space. Dalam fungsi *reset* akan dilakukan pemilihan secara acak terhadap index (i), lalu akan disesuaikan dengan baris yang sesuai dalam *dataset* yang digunakan (*data_length*).

$$i = U[0, data_length - 1] \text{ dengan } U \text{ adalah variable acak} \quad (3.6)$$

$$subset = data[i, 0 : 7] \quad (3.7)$$

Nantinya untuk setiap perulangan episode akan dilakukan proses reset untuk memulai proses pelatihan kembali dengan data yang dimulai dengan index yang berbeda. Fungsi *step* digunakan saat *agent* selesai mengambil keputusan. Berdasarkan keputusan yang diambil *agent*, akan dihitung *reward* yang merupakan perbedaan mutlak antara hasil prediksi dan data sebenarnya. Dalam fungsi ini juga didefinisikan *state* selanjutnya untuk melanjutkan proses pelatihan.

Pada metode *DQN* perhitungan *q-value* digunakan menggunakan model pembelajaran mesin. Dalam penelitian ini, digunakan algoritma *feedforward neural network (FFNN)* sebagai model dasar. Nantinya *model* ini akan disempurnakan selama proses training *DQN* dan menghasilkan *model* baru setelah seluruh proses training selesai.

Pada proses pelatihan model, ditentukan jumlah episode yang merupakan jumlah perulangan untuk model dilatih. Dalam setiap perulangan, *agent* akan melakukan proses pelatihan di dalam *environment* dan hasil akhir *reward* akan di plot sampai seluruh episode berakhir.

Untuk mencari kinerja dan performa yang paling optimal untuk model algoritma *DQN*, akan dilakukan tahapan *hyperparameter tuning*. Beberapa parameter yang ditetapkan untuk dilakukan *hyperparameter tuning* dapat dilihat pada **Tabel 3.3**.

Perhitungan jumlah *action* dilakukan karena jumlah *output* pada algoritma *DQN* tidak bisa ditetapkan hanya dengan 1 output. Nilai prediksi yang bisa dipilih oleh *DQN Agent* harus ditetapkan terlebih dahulu sebelum proses *model training* dimulai. Nilai prediksi yang dapat dipilih *agent* ditetapkan sebagai $\{x \in \mathbb{R} \mid x = 100 + \alpha n, \text{ dengan } n = 0, 1, 2, \dots, N - 1\}$ dan $100 + \alpha N < 400$. Nilai α yang akan disesuaikan untuk menentukan banyaknya *action* yang dapat dipilih *agent*.

Tabel 3.2 *pseudocode* pelatihan model menggunakan algoritma *DQN*

Algoritma I: Pelatihan menggunakan algoritma <i>DQN</i>	
	Inisialisasi Environment:
1	Definisi <i>state space</i> (s_t) yaitu daftar parameter prediksi
2	Definisi <i>action space</i> (a_t) = $\{a_i\}$ dimana $a_i = 0 + 2 \cdot i$, untuk $i = 0, 1, 2, \dots, n - 1$
3	Definisi fungsi <i>reset</i> untuk memulai ulang fase training data dengan index awal berada pada posisi acak
4	Definisi fungsi <i>step</i> yang berfungsi untuk menghitung nilai reward pada <i>state</i> (r_i) yang dihitung dengan formula $-\text{[actual - predicted]}$, serta mengganti state ke tahap selanjutnya (s_{i+1}).
	Inisialisasi Agent (<i>DQN Agent</i>)
5	Definisi model menggunakan model dasar <i>feedforward neural network</i> (<i>FFNN</i>)
6	Penentuan <i>policy</i> menggunakan <i>BoltzmannQPolicy</i> serta penentuan <i>hyperparameter</i> untuk <i>DQN Agent</i>
	Fase Training:
8	foreach <i>training epochs</i> do
9	Jalankan fungsi reset untuk melanjutkan ke state selanjutnya (s_{i+1}).
10	while $s_{i+1} < s_n$ do
11	<i>Agent</i> melakukan prediksi berdasarkan state saat ini (s_i)
12	Hasil prediksi digunakan untuk menghitung reward pada <i>state</i> ini (r_i) melalui fungsi <i>step</i> pada <i>environment</i> . Fungsi <i>step</i> juga akan mengganti state (s_i) menjadi <i>state</i> selanjutnya (s_{i+1}).
	end
13	Simpan model untuk dilakukan fase testing
	end

Pada proses pelatihan model *LSTM* dan *GRU*, pelatihan dapat dilakukan secara langsung dari dataset tanpa harus melalui *environment* seperti pada algoritma *DQN*. Untuk mencari kinerja dan performa yang paling optimal pada model, akan dilakukan tahapan *hyperparameter tuning*. Pada Algoritma *LSTM* beberapa parameter yang ditetapkan untuk di tuning pada **Tabel 3.4**.

Tabel 3.3 Training Hyperparameter untuk algoritma *DQN*

Konfigurasi	Jenis parameter	Parameter	Jumlah
Model	actions	$x \in \mathbb{R} \mid x = 100 + n;$	5
		$x \in \mathbb{R} \mid x = 100 + 0.5n;$	
		$x \in \mathbb{R} \mid x = 100 + 0.25n;$	
		$x \in \mathbb{R} \mid x = 100 + 0.2n;$	
		$x \in \mathbb{R} \mid x = 100 + 0.125n;$	
Training	Jumlah Step	40.000	5
		50.000	
		60.000	
		70.000	
		100.000	

Tabel 3.4 Training Hyperparameter untuk algoritma *LSTM*

Konfigurasi	Jenis parameter	Parameter	Jumlah
LSTM Layer	Jumlah units	32	3
		64	
		128	
	Activation function	Tanh	3
		ReLu	
		Sigmoid	
	Recurrent activation function	Sigmoid	2
		ReLu	
		Tanh	
Dense Layer	Activation function	Linear	3
Konfigurasi model	Optimizer	Adam	1
	Learning Rate	0.01	3
Konfigurasi Training	Batch Size	16	3
		32	
		64	
		128	

Pada algoritma *GRU*, parameter yang akan dilakukan tuning hampir sama dengan algoritma *LSTM*. Perbedaan yang dilakukan hanya terletak pada layer yang menggunakan *LSTM Layer* sebagai *hidden layer*. Berikut ditentukan beberapa hyperparameter yang akan dilakukan tuning pada algoritma *GRU* yang dapat dilihat pada **Tabel 3.5**.

Pada algoritma *LSTM* dan *GRU* jumlah perulangan pelatihan (*epochs*) akan ditentukan dengan metode *early stopping*. Metode *early stopping* adalah bentuk regularisasi dengan tujuan menghindari *overfitting* melalui penghentian pelatihan pada *model* dengan metode perulangan seperti *LSTM* dan *GRU*. Dengan *early stopping*, fase pelatihan *model* akan dihentikan saat metrik yang dipantau telah berhenti meningkat.

Metode *early stopping* ini nantinya akan ditetapkan 2 parameter penting yaitu metrik yang dipantau serta jumlah toleransi iterasi tanpa peningkatan (*patience*). Pada algoritma *LSTM* dan *GRU*, *early stopping* akan diterapkan dengan *Mean Squared Error (MSE)* sebagai metrik yang dipantau dan rentang *patience* mulai dari 10 hingga 40.

Tabel 3.5 Training Hyperparameter untuk algoritma *GRU*

Konfigurasi	Jenis parameter	Parameter	Jumlah
GRU Layer	Jumlah units	32	3
		64	
		128	
	Activation function	Tanh	3
		ReLu	
		Sigmoid	
	Recurrent activation function	Sigmoid	2
		ReLu	
		Tanh	
Dense Layer	Activation function	Linear	3
Konfigurasi model	Optimizer	Adam	1
	Learning Rate	0.001	3
Konfigurasi Training	Batch Size	16	3
		32	
		64	
		128	

Model yang telah selesai dilatih akan menghasilkan *model* yang nantinya bisa digunakan untuk melakukan prediksi radiasi matahari. Model ini dapat disimpan dalam bentuk *file* dengan ekstensi *h5* agar dapat digunakan kembali tanpa melalui proses training ulang.

3.2.5 Proses Testing

Pada proses ini, model yang telah dihasilkan akan dilakukan pengetesan melalui prediksi untuk menghitung tingkat keakuratan model dengan menghitung seberapa jauh perbedaan hasil antara nilai prediksi model dan nilai sebenarnya. Dalam perhitungan ini, digunakan empat metrik evaluasi kinerja model yaitu *Mean Squared Error (MSE)*, *Root Mean Squared Error (RMSE)*, *Mean Absolute Error (MAE)*, dan R^2 (*Coefficient of Determination*).

Metrik evaluasi *Mean Squared Error (MSE)* menghitung kuadrat rata-rata kesalahan dari nilai prediksi yang dibandingkan dengan nilai asli. *MSE* dapat memberi penekanan yang lebih kuat pada jarak kesalahan karena nilai yang dikuadratkan. Formulasi penghitungan nilai *MSE* didefinisikan pada persamaan 3.8.

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - f_i)^2 \quad (3.8)$$

Dengan y_i sebagai nilai benar dari data pada posisi i sedangkan f_i sebagai nilai dari hasil prediksi pada posisi i .

Metrik evaluasi *Root Mean Squared Error (RMSE)* merupakan hasil perhitungan akar kuadrat dari nilai *MSE*. Nilai *RMSE* lebih sensitif dengan kesalahan yang lebih besar. Formulasi perhitungan nilai *RMSE* didefinisikan pada persamaan 3.9.

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - f_i)^2} \quad (3.9)$$

Metrik evaluasi *Mean Absolute Error (MAE)* menghitung nilai absolut dari nilai hasil prediksi yang dibandingkan nilai sebenarnya dari data. *MAE* memberikan ukuran yang lebih intuitif dikarenakan nilai perbandingan hasil prediksi dan nilai sebenarnya tidak dilakukan kalkulasi matematika lain sehingga tidak memberikan pinalti yang terlalu berat pada jarak kesalahan. Formulasi perhitungan nilai *MAE* didefinisikan pada persamaan 3.10.

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - f_i| \quad (3.10)$$

Metrik evaluasi *Coefficient of Determination* (R^2) mengukur seberapa baik nilai prediksi sesuai dengan data sebenarnya dengan skala 0 hingga 1. Semakin tinggi nilai R^2 , menunjukkan semakin baiknya kemampuan model dalam melakukan prediksi. Formulasi perhitungan nilai R^2 didefinisikan pada persamaan 3.11.

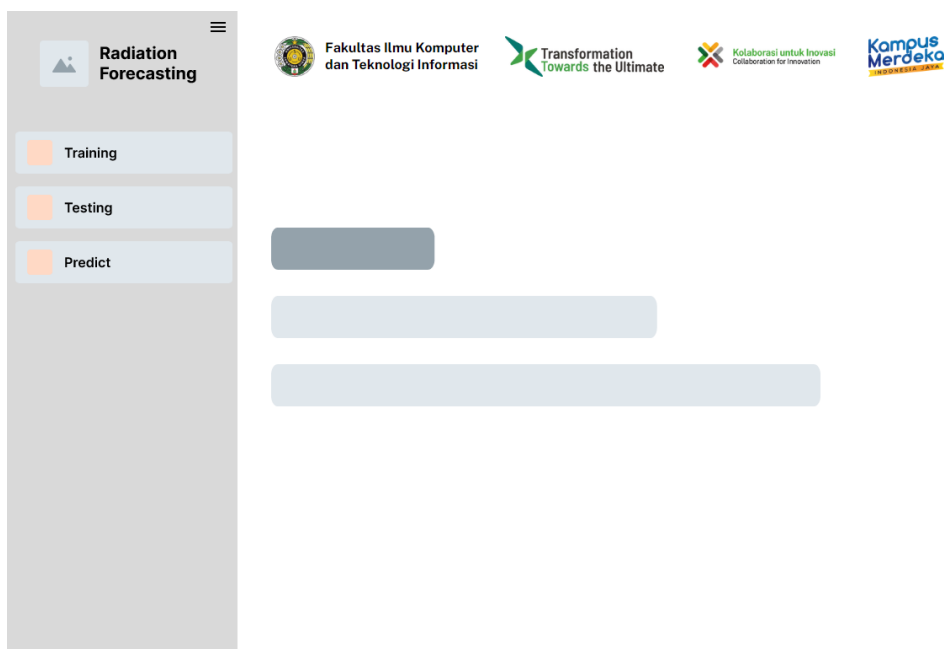
$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - f_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (3.11)$$

Dengan \bar{y} sebagai rata-rata nilai sebenarnya dari data secara keseluruhan.

3.3 Perancangan Antarmuka Sistem

Pada tahapan ini dilakukan pembuatan antarmuka sistem berbasis website agar nantinya proses pelatihan, pengujian, serta prediksi data dapat dilakukan melalui antarmuka sistem. Pada tahap perancangan antarmuka sistem terdapat 4 halaman yang akan dibuat yaitu halaman utama, halaman pelatihan, halaman pengujian, serta halaman prediksi.

Struktur utama dalam website antarmuka sistem ini akan terbagi menjadi dua bagian yaitu bagian navigasi dan bagian konten. Pada bagian navigasi menggunakan *sidebar* yang akan berisi tombol-tombol untuk melakukan navigasi antar fitur dan bagian konten akan berisi fitur yang bisa digunakan pada halaman tersebut.



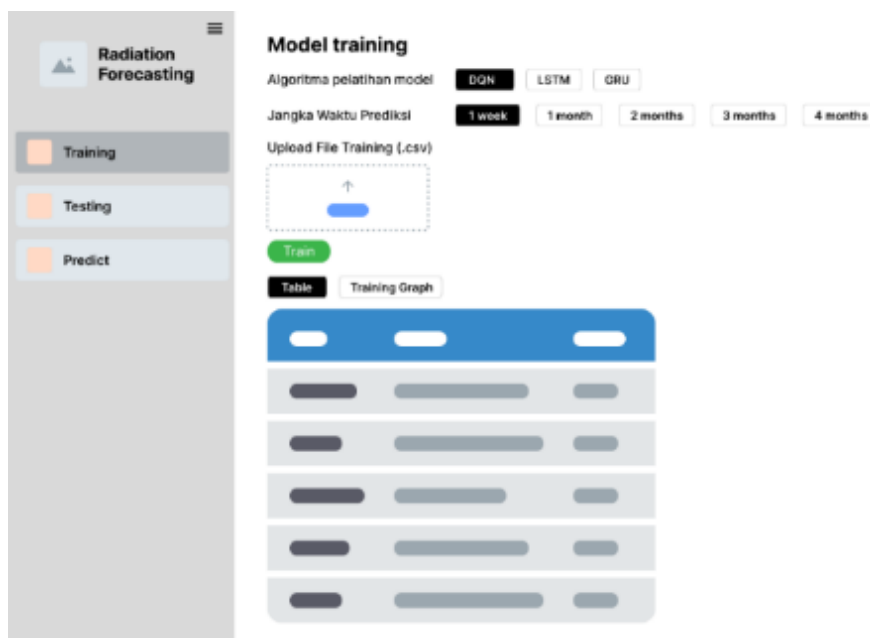
Gambar 3.9 Rancangan Tampilan Halaman Utama

3.3.1 Rancangan Tampilan Halaman Utama

Struktur konten halaman utama pada bagian atas akan berisi logo-logo yang berkaitan dengan universitas dan pada bagian Tengah akan diisi dengan informasi yang berkaitan dengan penelitian seperti penulis dan judul penelitian. Rancangan tampilan halaman utama dapat dilihat pada **Gambar 3.9**.

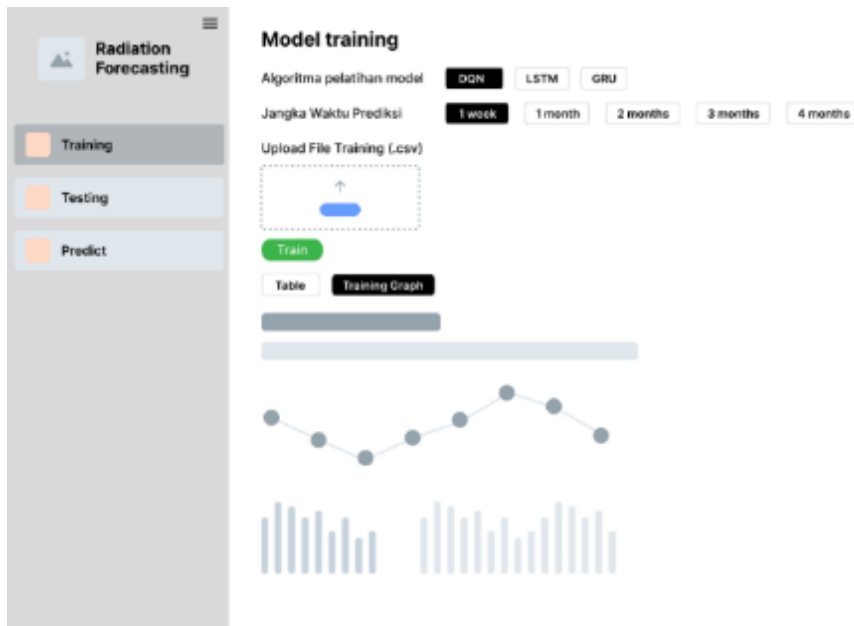
3.3.2 Rancangan Tampilan Halaman Training

Pada halaman *training*, halaman akan diisi dengan fitur pelatihan *model* yang dilakukan dengan memilih algoritma yang ingin dilatih serta mengunggah dokumen *dataset training* berformat *csv*. Fase pelatihan *model* akan dimulai saat pengguna menekan tombol *train* pada layar. Rancangan tampilan halaman training dapat dilihat pada **Gambar 3.10** dan **Gambar 3.11**.



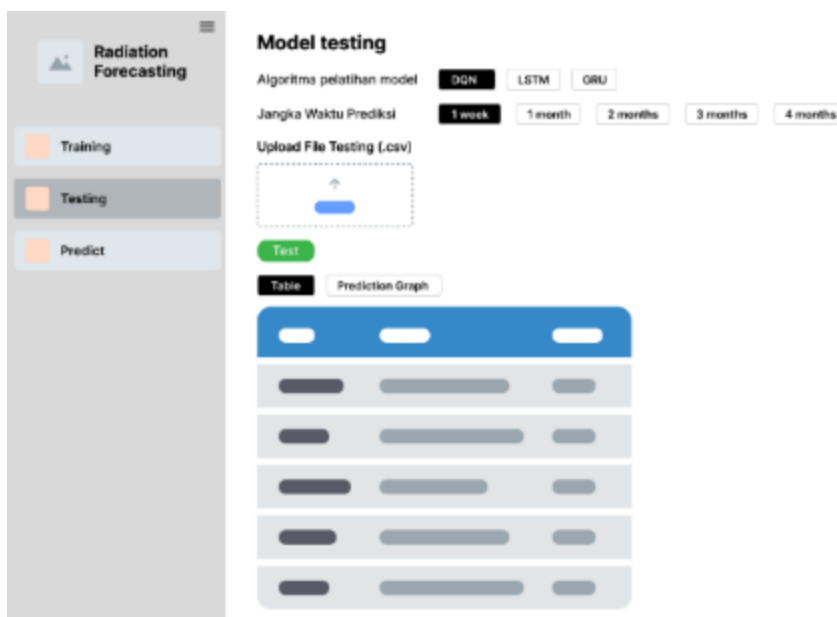
Gambar 3.10 Rancangan Tampilan Halaman Training dengan tabel

Setelah fase training selesai dilakukan, pengguna akan memiliki dua menu yang dapat diakses yaitu *table* dan *training graph*. Pada menu *table* pengguna dapat melihat perjalanan pelatihan model pada setiap perulangan dalam bentuk tabel. Pada menu *training graph*, pengguna dapat melihat pergerakan pelatihan model pada setiap perulangan dari algoritma yang dipilih.



Gambar 3.11 Rancangan Tampilan Halaman *Training* dengan grafik

Setelah seluruh proses pelatihan *model* selesai dilakukan, pengguna akan mendapatkan tombol *test model* yang akan mengarahkan pengguna ke halaman *testing* dan tombol *predict* yang akan mengarahkan pengguna ke halaman *predict*.

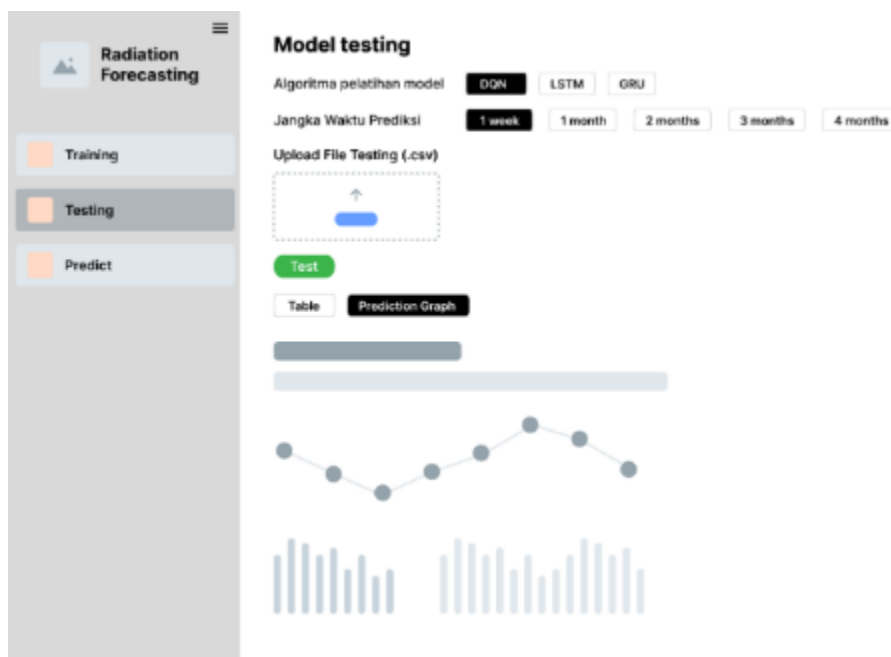


Gambar 3.12 Rancangan Tampilan Halaman *Testing* dengan menampilkan hasil berbentuk tabel

3.3.3 Rancangan Tampilan Halaman Testing

Pada halaman *testing*, pengguna dapat melakukan pengujian terhadap *model* yang telah dilakukan pelatihan pada menu *training*. Pengguna akan diarahkan untuk memilih algoritma yang diinginkan serta jangka waktu prediksi. Setelah itu, pengguna diharuskan mengunggah file berformat *csv* yang akan digunakan sebagai perbandingan dengan hasil prediksi *model*. Rancangan halaman *testing* dapat dilihat pada **Gambar 3.12** dan **Gambar 3.13**.

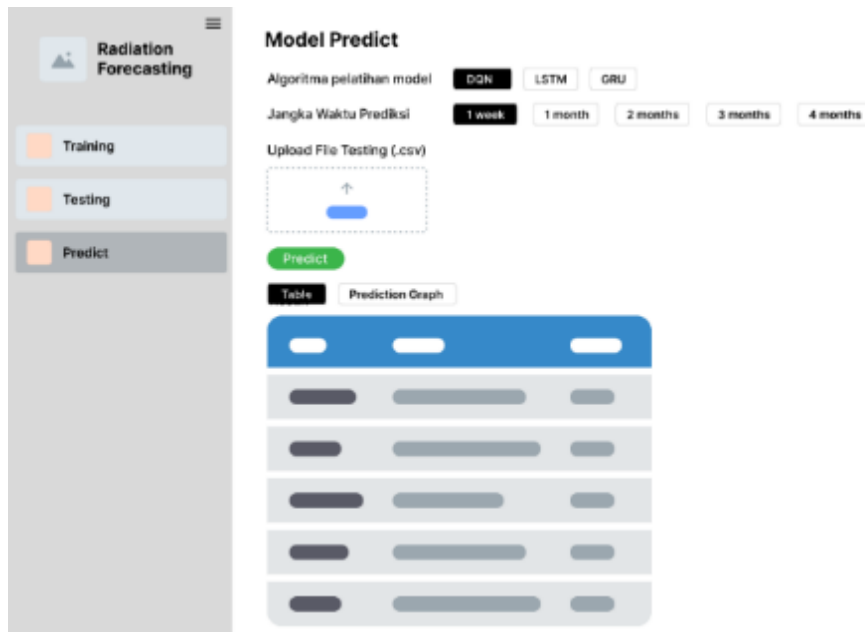
Setelah semua masukan telah terisi dengan baik, pengguna dapat menekan tombol *test* untuk menjalankan proses pengujian *model*. Setelah proses pengujian *model* selesai dilakukan, akan muncul dua menu yaitu *table* dan *prediction graph*. Menu *table* akan menampilkan perbandingan hasil pengujian dalam bentuk tabel, sedangkan menu *prediction graph* akan menampilkan perbandingan hasil pengujian dalam bentuk diagram garis.



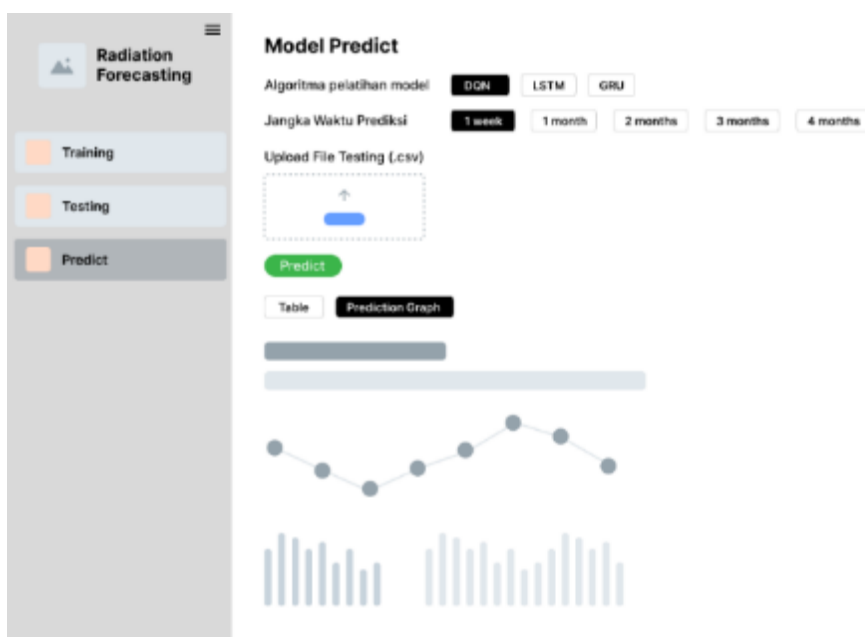
Gambar 3.13 Rancangan Tampilan Halaman *Testing* dengan menampilkan hasil berbentuk grafik

3.3.4 Rancangan Tampilan Halaman Predict

Pada halaman *predict*, pengguna dapat melakukan prediksi besaran radiasi matahari diluar dari *dataset*. Untuk mengakses fitur ini, pengguna akan diarahkan untuk memilih algoritma prediksi, jangka waktu prediksi serta mengunggah dokumen pengujian dengan ekstensi *csv*.



Gambar 3.14 Rancangan Tampilan Halaman *Predict* dengan menampilkan hasil berbentuk tabel



Gambar 3.15 Rancangan Tampilan Halaman *Predict* dengan menampilkan hasil berbentuk grafik

Dokumen pengujian diperlukan untuk mendapatkan fitur-fitur yang belum dimiliki seperti suhu dan kelembapan udara. Fitur-fitur ini akan diambil dan disesuaikan dengan jangka waktu prediksi yang diinginkan. Rancangan halaman *predict* dapat dilihat pada **Gambar 3.14** dan **Gambar 3.15**.

BAB 4

IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi Sistem

Dalam tahap implementasi sistem, dilakukan prediksi besaran radiasi matahari dari data meteorologi menggunakan metode Hibridisasi *Deep Reinforcement Learning (DRL)* dan *Gated Recurrent Unit (GRU)* kedalam sebuah sistem menggunakan bahasa pemrograman Python. Sistem akan berbasis *website* yang akan menggunakan teknologi pengembangan *web* yaitu HTML, CSS, dan Javascript untuk bagian *frontend* serta *framework* untuk *backend* yaitu Flask.

4.1.1 Perangkat Keras dan Perangkat Lunak

Adapun pembangunan sistem prediksi radiasi matahari memanfaatkan perangkat keras dan lunak sebagai berikut:

1. CPU: Processor AMD Ryzen 5 5600 @ 3.30GHz
2. GPU: NVIDIA GeForce RTX 3060 Laptop
3. RAM: 24 GB, DDR4 2400 MHz
4. Kapasitas SSD sebesar 500GB
5. Sistem Operasi: Windows 11 *Home Single Language* 64-bit

Sedangkan implementasi untuk membangun sistem ini menggunakan bahasa pemrograman *Python* 3.9.12 dan beberapa pendukung lainnya seperti:

1. Anaconda
2. Jupyter Notebook
3. Tensorflow v.2.11.0
4. Nvidia Cuda v.11.7
5. Scikit Learn v.1.0
6. Numpy v.1.22.4
7. Pandas v.1.3.3
8. Matplotlib v.3.5.1
9. Seaborn v.0.11.2
10. Flask v.3.0.3

4.1.2 Implementasi Rancangan Antarmuka

Setelah melakukan perancangan antarmuka pada BAB 3 maka akan dibuat implementasi dari rancangan tersebut yang berisikan tampilan utama, halaman *training*, halaman *testing* serta halaman *predict*.

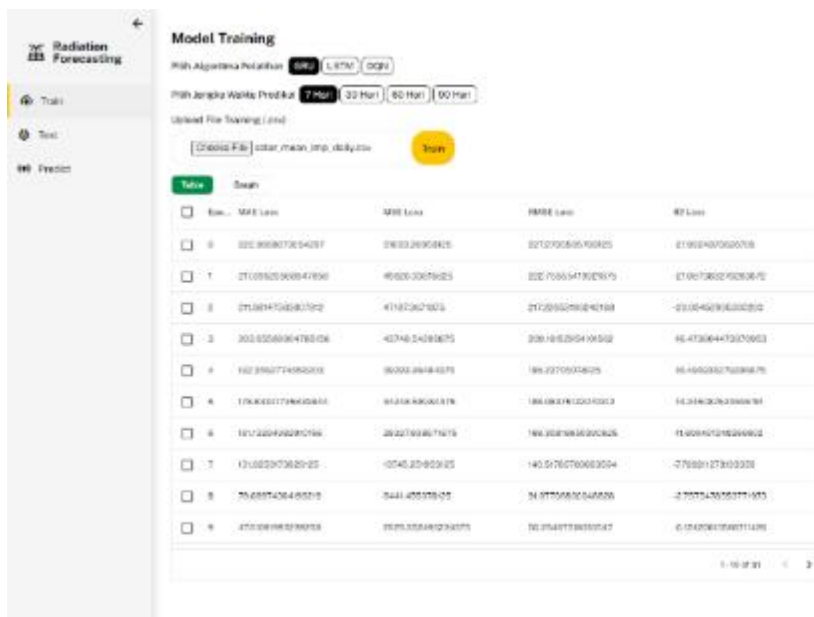
Tampilan halaman utama adalah tampilan yang akan muncul pertama kali saat sistem dijalankan. Tampilan halaman utama ini akan berisi judul penelitian serta data penulis. Pada bagian kanan layar, terdapat beberapa tombol yang dapat diakses untuk berpindah halaman untuk mengakses beberapa menu pada sistem prediksi radiasi matahari ini.



Gambar 4.1 Tampilan Halaman Utama

Halaman *training* merupakan halaman tempat pengguna akan melakukan pelatihan model. Pada menu ini, sebelum dapat melakukan proses pelatihan model, pengguna terlebih dahulu harus mengunggah dataset yang akan digunakan untuk melatih model. Setelah itu, pengguna juga harus mengisi algoritma yang ingin digunakan serta jangka waktu prediksi. Setelah semua data telah lengkap pengguna dapat menekan tombol *train* untuk memulai proses pelatihan model.

Setelah proses pelatihan model dimulai pengguna dapat melihat isi daftar metrik evaluasi pada setiap perulangan *training* pada menu table dengan menekan tombol table yang terletak di bawah masukan seperti dapat dilihat pada **Gambar 4.2**.



Model Training

Pilih Algoritma Pelatihan: **GRU** | LSTM | DNN

Pilih Jangka Waktu Prediksi: **7 Hari** | 30 Hari | 60 Hari | 90 Hari

Upload File Training (.csv): **Train**

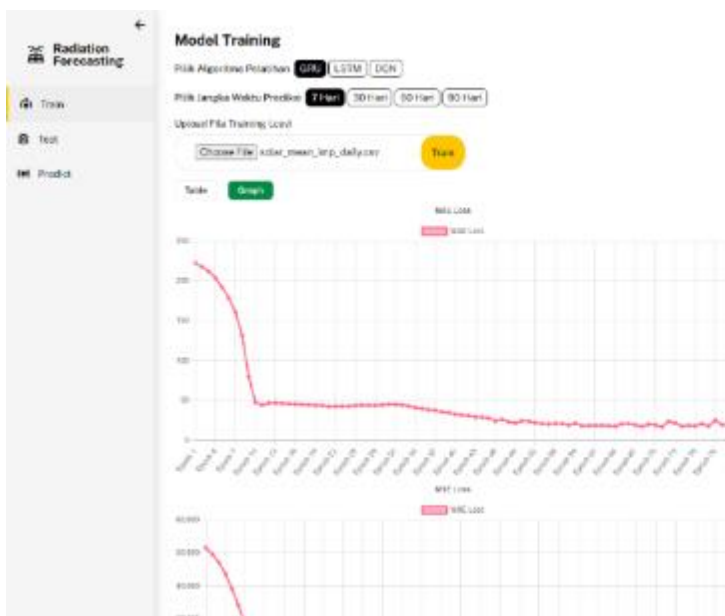
Tabel **Search**

	Epoch	MAE Loss	MSE Loss	RMSE Loss	R2 Loss
<input type="checkbox"/>	0	222.866807354037	29633.26358425	221.270555436825	21.9524687620758
<input type="checkbox"/>	1	271.09525908947690	46525.33079025	222.7586547302675	27.06736524528342
<input type="checkbox"/>	2	271.3814758581522	471823621075	2172955258246188	-23.05458396333282
<input type="checkbox"/>	3	303.52568364785106	43748.54283675	208.1815295419532	45.473064473270603
<input type="checkbox"/>	4	192.3962774850018	36002.89484375	189.23709034825	65.45603327638876
<input type="checkbox"/>	5	176.86321779405851	34234.86060376	186.0831912333323	54.24636263366876
<input type="checkbox"/>	6	157.2224922810766	2822739467675	169.3587683200826	11.60816721626682
<input type="checkbox"/>	7	131.022373829225	10745.25402925	143.5476076063204	-77939127310338
<input type="checkbox"/>	8	75.6557426403218	3541.455375125	59.67755632046506	-2.7577547075771575
<input type="checkbox"/>	9	47.0381883236078	2525.1553481234075	50.25437738035747	0.12435817588711439

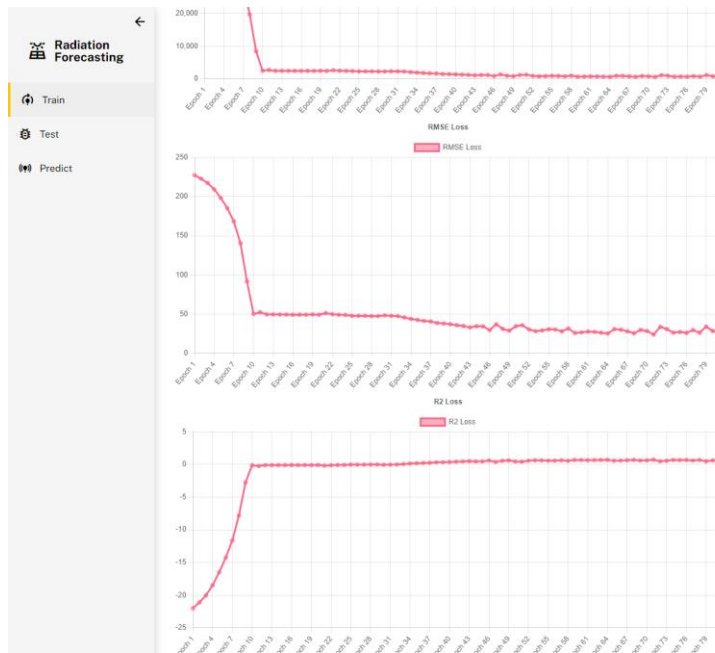
1 - 10 of 31

Gambar 4.2 Tampilan Halaman Training dengan tabel

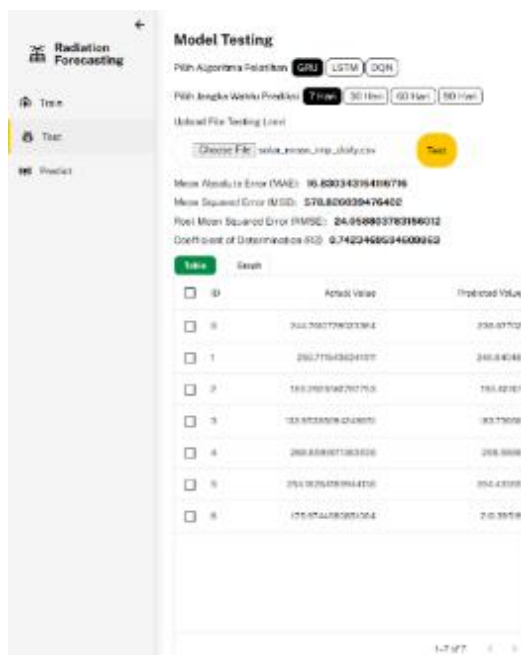
Pengguna juga dapat melihat pergerakan metrik evaluasi selama proses *training model* dalam bentuk grafik. Terdapat empat metrik evaluasi yang dapat dilihat pada grafik yaitu *Mean Squared Error (MSE)*, *Root Mean Squared Error (RMSE)*, *Mean Absolute Error (MAE)*, dan R^2 (*Coefficient of Determination*). Grafik proses *training model* tersebut dapat dilihat pada **Gambar 4.3**.



Gambar 4.3 Tampilan Halaman *Training* dengan grafik



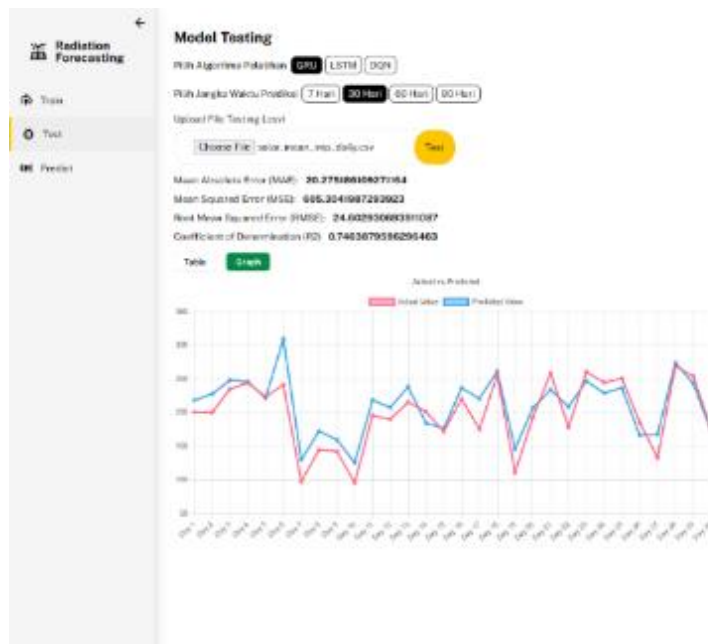
Gambar 4.4 Tampilan Halaman *Training* dengan grafik dengan nilai loss *RMSE* dan R^2



Gambar 4.5 Tampilan Halaman *Testing* dengan Tabel

Halaman selanjutnya yaitu halaman testing yang berfungsi untuk menampilkan nilai metrik evaluasi yang dites menggunakan dataset testing yang diunggah oleh pengguna. Pengguna akan diarahkan untuk memilih algoritma yang ingin digunakan serta memilih jangka waktu prediksi. Pada saat pemilihan masukan, pengujian *model* akan dapat dilakukan saat *model* dengan algoritma dan jangka waktu terpilih telah dilakukan *training* sebelumnya.

Selanjutnya pengguna akan diarahkan untuk mengunggah dokumen yang berisi *dataset testing* yang akan digunakan sebagai pembanding dengan hasil prediksi dari model yang telah dilatih sebelumnya. Untuk memulai proses pengujian *model*, pengguna diarahkan untuk menekan tombol *Test*. Setelah proses pengujian selesai, akan ditampilkan keempat nilai metrik evaluasi yang didapat dari perbandingan nilai prediksi dan nilai sebenarnya yaitu *MAE*, *MSE*, *RMSE* serta R^2 . Selain itu juga terdapat tombol tabel yang dapat digunakan untuk melihat nilai prediksi dari *model* serta nilai sebenarnya pada *dataset* seperti yang terlihat pada **Gambar 4.6**. Pengguna juga dapat melihat perbandingan nilai prediksi *model* dengan nilai sebenarnya dalam bentuk grafik seperti pada **Gambar 4.7**.

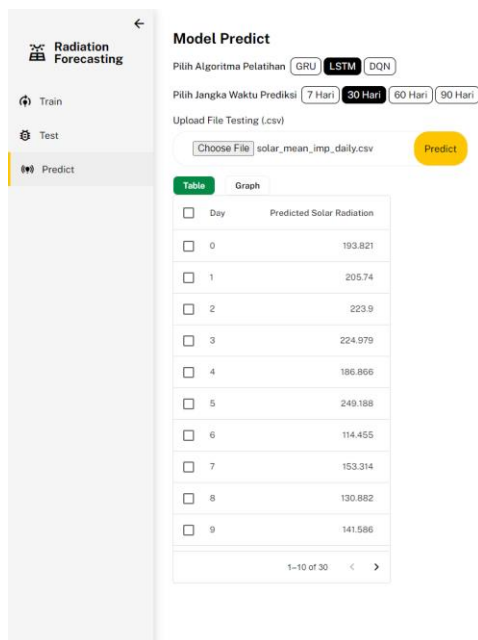


Gambar 4.6 Tampilan Halaman Testing dengan Grafik

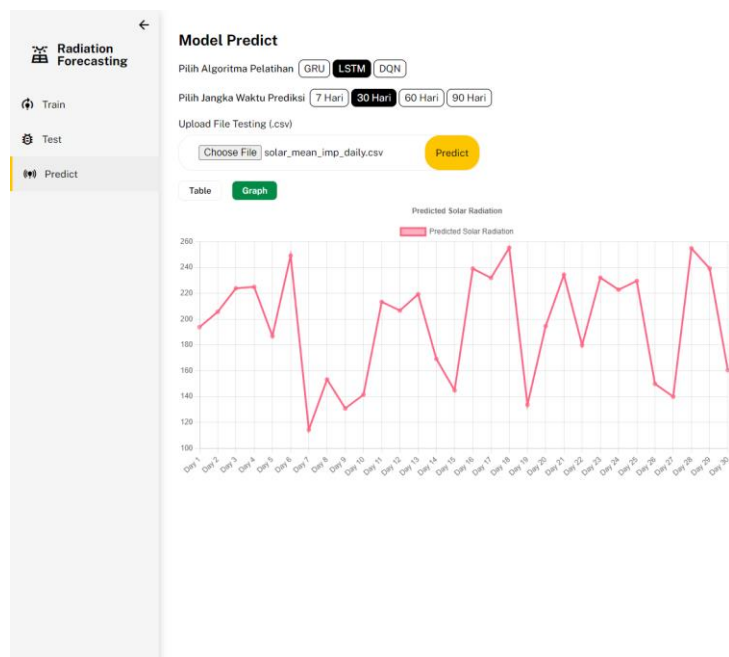
Fitur terakhir pada aplikasi tampilan aplikasi pada penelitian ini yaitu fitur *predict*. Fitur ini berguna dalam melakukan prediksi nilai besaran radiasi matahari diluar dataset. Pengguna akan diarahkan untuk memilih algoritma yang diinginkan serta jangka waktu prediksi yang diinginkan. Setelah itu, pengguna diharuskan untuk mengunggah *dataset testing* yang akan digunakan untuk mendapatkan nilai suhu serta kelembapan udara pada beberapa waktu terakhir yang disesuaikan dengan jangka waktu prediksi.

Untuk memulai proses prediksi, pengguna dapat menekan tombol *predict* yang terletak disamping tombol untuk mengunggah *dataset testing*. Setelah proses prediksi

selesai, nantinya pengguna dapat melihat hasil prediksi dalam bentuk tabel seperti pada **Gambar 4.7** ataupun dalam bentuk grafik seperti dapat dilihat pada **Gambar 4.8**.



Gambar 4.7 Tampilan Halaman Predict dengan Tabel



Gambar 4.8 Tampilan Halaman Predict dengan Grafik

4.2 Prosedur Operasional

Untuk menjalankan sistem prediksi radiasi matahari ini, diperlukan beberapa tahapan yang perlu dilakukan sebelum sistem dapat dijalankan pada perangkat. Sistem ini dibuat menggunakan 2 server yang berbeda masing-masing untuk bagian *frontend* dan *backend*. Pada *server backend* menggunakan framework flask yang merupakan tempat untuk

memanggil *Application Programming Interface (API)* yang sekaligus akan dilakukan proses *training*, *testing* serta fitur-fitur lainnya yang ada.

Server frontend menggunakan framework *NextJS* yang merupakan tempat pengguna nantinya berinteraksi dengan sistem melalui interface untuk menjalankan beberapa fitur seperti *training*, *testing*, serta *predict*.

Ketika pengguna pertama kali mengakses sistem, pengguna akan dihadapkan dengan halaman utama sistem. Berikutnya pengguna dapat melakukan navigasi antar halaman menggunakan tombol-tombol navigasi yang berada pada bagian kiri layar. Untuk menjalankan fitur *training* pengguna dapat menekan tombol training disebelah kiri layar lalu melakukan unggah *dataset training* serta memilih algoritma yang ingin digunakan serta waktu prediksi yang diinginkan, lalu menekan tombol *train*.

Untuk menggunakan fitur *testing model*, pengguna dapat melakukannya dengan menekan tombol testing yang terletak pada bagian kiri navigasi. Setelah itu, pengguna dapat mengunggah *dataset testing* yang akan dibandingkan dengan hasil prediksi model, lalu memilih algoritma serta waktu prediksi yang diinginkan lalu melanjutkan dengan menekan tombol *test*.

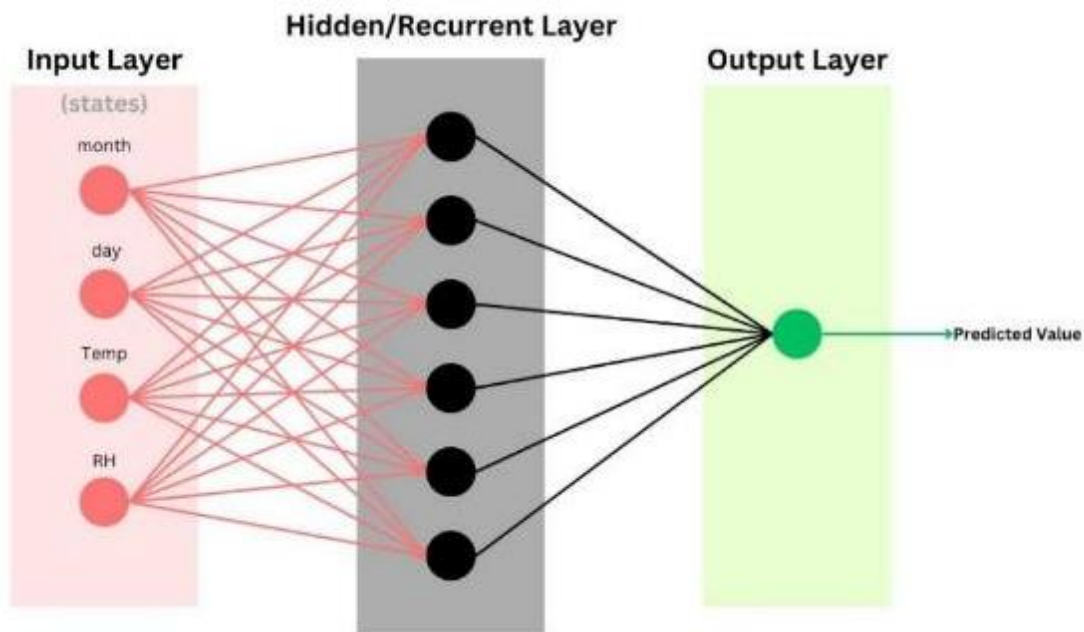
Untuk fitur *predict*, pengguna juga dapat melakukannya setelah selesai melakukan *training*. Tombol *predict* akan muncul saat proses training selesai dilakukan yang terletak di bagian kiri navigasi. Pengguna dapat memilih algoritma serta jangka waktu prediksi yang diinginkan lalu mengunggah *dataset testing* yang akan digunakan sebagai parameter suhu dan kelembapan udara. Setelahnya pengguna dapat menekan tombol *predict* untuk memulai prediksi.

4.3 Pelatihan Sistem

Pada pelatihan sistem, terdapat tiga algoritma yang akan diuji yaitu *Gated Recurrent Unit (GRU)*, *Long Short-Term Memory (LSTM)* dan *Deep Q-Network (DQN)*. Ketiga algoritma yang digunakan pada penelitian ini akan dilatih menggunakan *framework open-source* yang dikembangkan oleh Google yaitu *Tensorflow*. Versi *Tensorflow* yang digunakan adalah versi 2.11.0 dengan versi *python* 3.9.12.

Pada pelatihan sistem menggunakan algoritma *GRU* dan *LSTM*, pelatihan dapat langsung dilakukan melalui pendefinisian *Neural Network*, lalu dapat langsung dilakukan proses pelatihan model menggunakan dataset. Untuk mendapatkan *model* yang paling

optimal dalam melakukan prediksi, perlu dilakukan proses *hyperparameter tuning* untuk mencari konfigurasi terbaik untuk masing-masing algoritma. Untuk algoritma *GRU* akan dilakukan *hyperparameter tuning* beberapa parameter yang dapat dilihat pada **Tabel 3.5**. Sedangkan detail parameter yang akan dilakukan *hyperparameter tuning* untuk algoritma *LSTM* dapat dilihat pada **Tabel 3.4**.



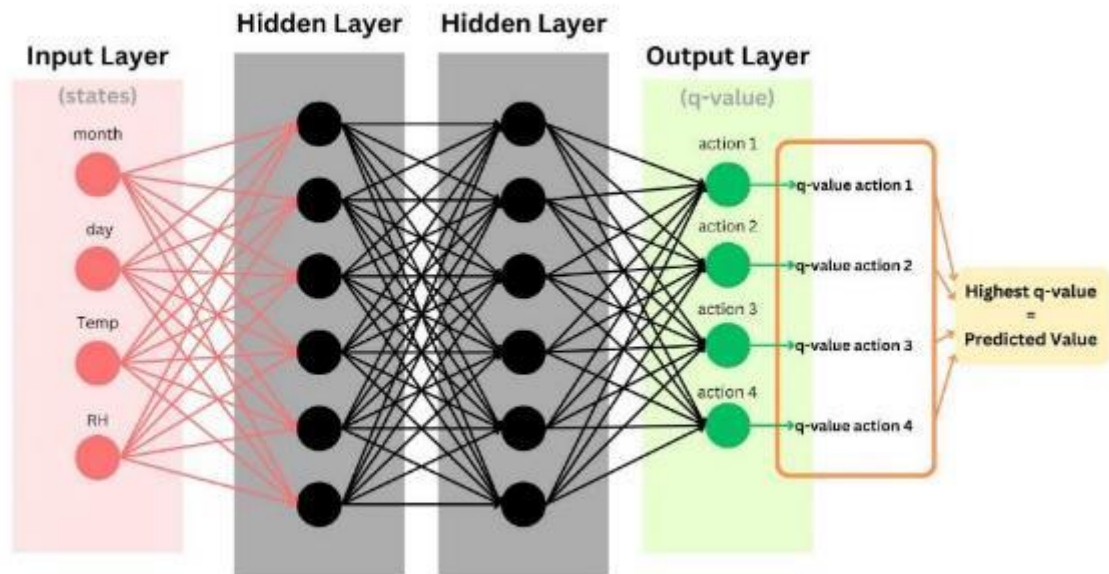
Gambar 4.9 Struktur *Neural Network* pada algoritma *GRU* dan *LSTM*

Pada algoritma *DQN*, proses *training* akan sedikit berbeda dengan algoritma *GRU* maupun *LSTM*. Seperti terlihat pada **Gambar 4.9**, pada algoritma *LSTM* dan *GRU* hanya memiliki satu buah node pada output layer yang sekaligus merupakan nilai prediksi dari model. Pada algoritma *DQN*, hal tersebut tidak dapat dilakukan dikarenakan pada algoritma *DQN*, penetapan output/action dilakukan melalui perhitungan nilai *q-value*. Sehingga untuk dapat melakukan prediksi, pada output layer akan dipetakan nilai-nilai yang dapat dipilih agent sebagai nilai prediksi. Nantinya, pada masing-masing action akan dilakukan perhitungan nilai *q-value*, dan nilai *q-value* tertinggi akan menjadi nilai prediksi model.

Pada algoritma *DQN* pendefinisian Neural Network akan menggunakan model *Feed Forward Neural Network (FFNN)* yang akan digunakan sebagai basis model untuk nantinya dilakukan *reinforce* oleh *DQN Agent*.

Pada proses pelatihan algoritma *DQN*, terdapat sedikit perbedaan cara pelatihan dibandingkan dengan algoritma *deep learning* lain. Pada algoritma *DQN*, model dilatih

pada suatu lingkungan khusus yang disebut *environment*. *Environment* akan menjadi tempat seluruh tahapan pelatihan dilakukan.



Gambar 4.10 Struktur *Neural Network* pada algoritma *DQN*

Pada penelitian ini, dibuat *environment* yang secara khusus digunakan pada kasus prediksi radiasi matahari menggunakan dataset penelitian. Dalam *environment* ini, akan didefinisikan beberapa konfigurasi pelatihan sebagai berikut.

1. **State space**, akan digunakan pada *input layer neural network*. *State space* akan berisi empat parameter yaitu bulan (*month*), hari (*day*), suhu udara (*Temp*) dan kelembapan udara (*RH*). Konfigurasi *state space* pada *neural network* dapat dilihat pada **Gambar 4.10**.
2. **Action space**, akan digunakan pada *output layer neural network*. Action space akan berisi pemetaan nilai yang dapat diprediksi model. Nilai action space yang akan digunakan dapat dilihat pada **Tabel 3.3**.
3. **Reward**, akan digunakan DQN Agent sebagai tolak ukur penilaian model saat melakukan pembaruan model. Perhitungan reward dilakukan dengan membandingkan nilai prediksi (f) dan nilai sebenarnya (y). Formula perhitungan reward dapat didefinisikan pada rumus 4.1.

$$Reward = -|y - f| \quad (4.1)$$

4.3.1 Hasil Pelatihan Model Gated Recurrent Unit

Pada model *Gated Recurrent Unit (GRU)*, daftar parameter yang dikombinasikan dapat dilihat pada **Tabel 3.5**. Selama proses *hyperparameter tuning*, terdapat beberapa kombinasi parameter dengan hasil terbaik pada keempat metrik evaluasi pada masing-masing jangka prediksi.

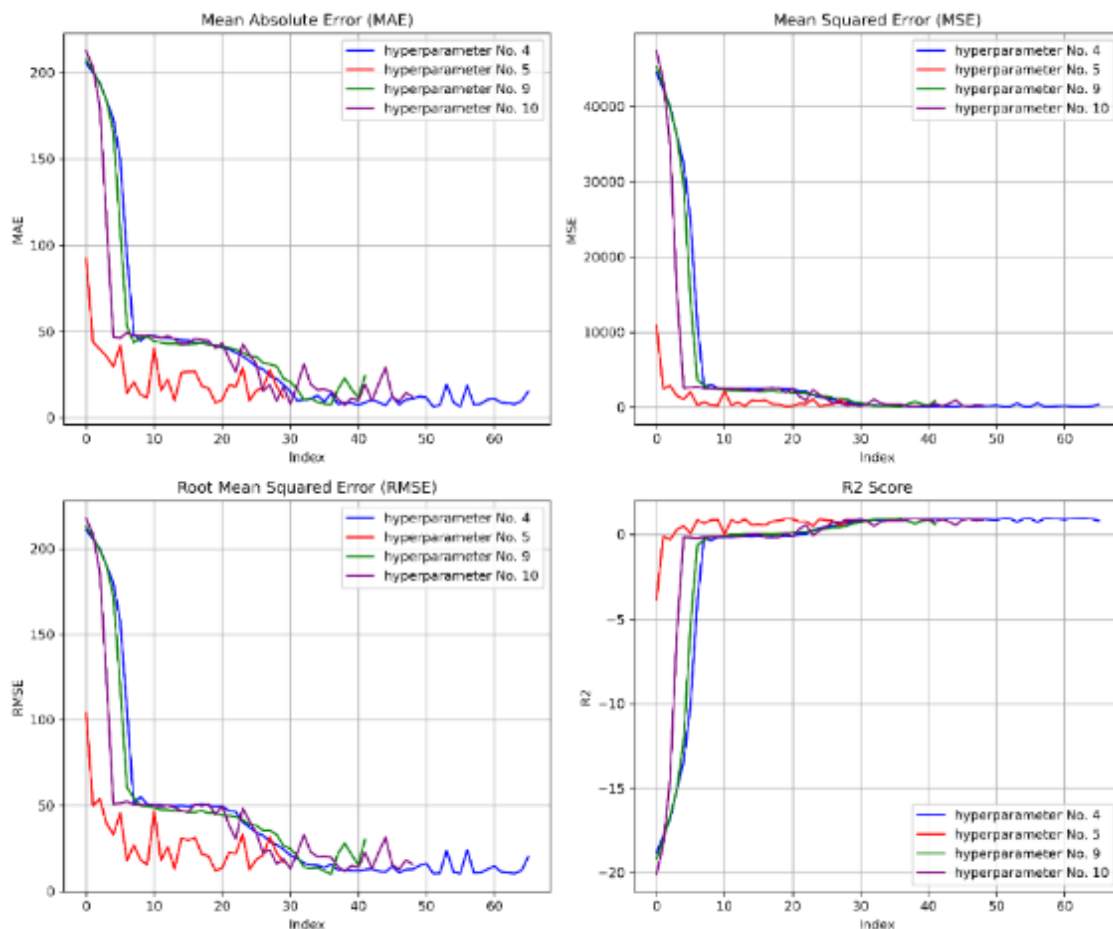
Pada pengujian model *GRU* dengan jangka waktu prediksi 7 hari, dapat dilihat pada **Tabel 4.1**, nilai metrik evaluasi cukup baik dengan nilai *RMSE* terbaik pada angka 9.980484. Begitu juga pada nilai metrik evaluasi seperti *MAE*, *MSE* dan R^2 dengan nilai masing-masing 7.677913, 99.61006 dan 0.955661 pada kombinasi parameter yang sama yaitu jumlah unit pada layer *GRU* berjumlah 32, *activation function* pada layer *GRU* menggunakan *rectified linear unit (ReLU)*, *recurrent activation function* menggunakan sigmoid serta jumlah *batch size* 128.

Pada **Gambar 4.11** juga dapat dilihat pergerakan hasil evaluasi *model* pada setiap *epoch* dengan peningkatan yang signifikan diawal diikuti dengan peningkatan perlahan menuju titik terbaik berdasarkan keempat metrik evaluasi.

Tabel 4.1 Nilai metrik evaluasi dari pengujian model *GRU* dengan jangka prediksi 7 hari

No	Jumlah Unit GRU Layer	Activation Function GRU Layer	Recurrent Activation Function GRU Layer	Batch Size	MAE	RMSE	MSE	R^2
1	32	tanh	sigmoid	32	9.553227	14.27722	203.8389	0.909265
2	64	tanh	sigmoid	32	10.12664	15.3571	235.8404	0.89502
3	32	relu	sigmoid	64	7.856225	12.88305	165.9729	0.92612
4	32	relu	sigmoid	128	6.605886	10.40297	108.2218	0.951827
5	32	relu	sigmoid	16	8.778643	11.90822	141.8056	0.936878
6	32	relu	tanh	32	11.16048	15.16195	229.8847	0.897671
7	32	relu	tanh	64	12.0211	14.51802	210.7728	0.906179
8	32	tanh	tanh	16	10.18414	13.9578	194.8202	0.91328
9	32	relu	sigmoid	128	7.677913	9.980484	99.61006	0.955661
10	64	relu	sigmoid	128	7.259145	11.62742	135.1969	0.93982

Pada pengujian model *GRU* dengan jangka waktu prediksi 30 hari seperti yang dapat dilihat pada **Tabel 4.2**, nilai metrik evaluasi sedikit mengalami kenaikan dikarenakan jumlah data yang semakin banyak. Pada jangka waktu prediksi 30 hari, kombinasi parameter yang memiliki hasil terbaik berdasarkan nilai metrik evaluasi sedikit berbeda dengan jangka waktu 30 hari.



Gambar 4.11 Pergerakan nilai metrik evaluasi selama pelatihan *model* dari empat kombinasi parameter terbaik pada algoritma *GRU* dengan jangka prediksi 7 hari

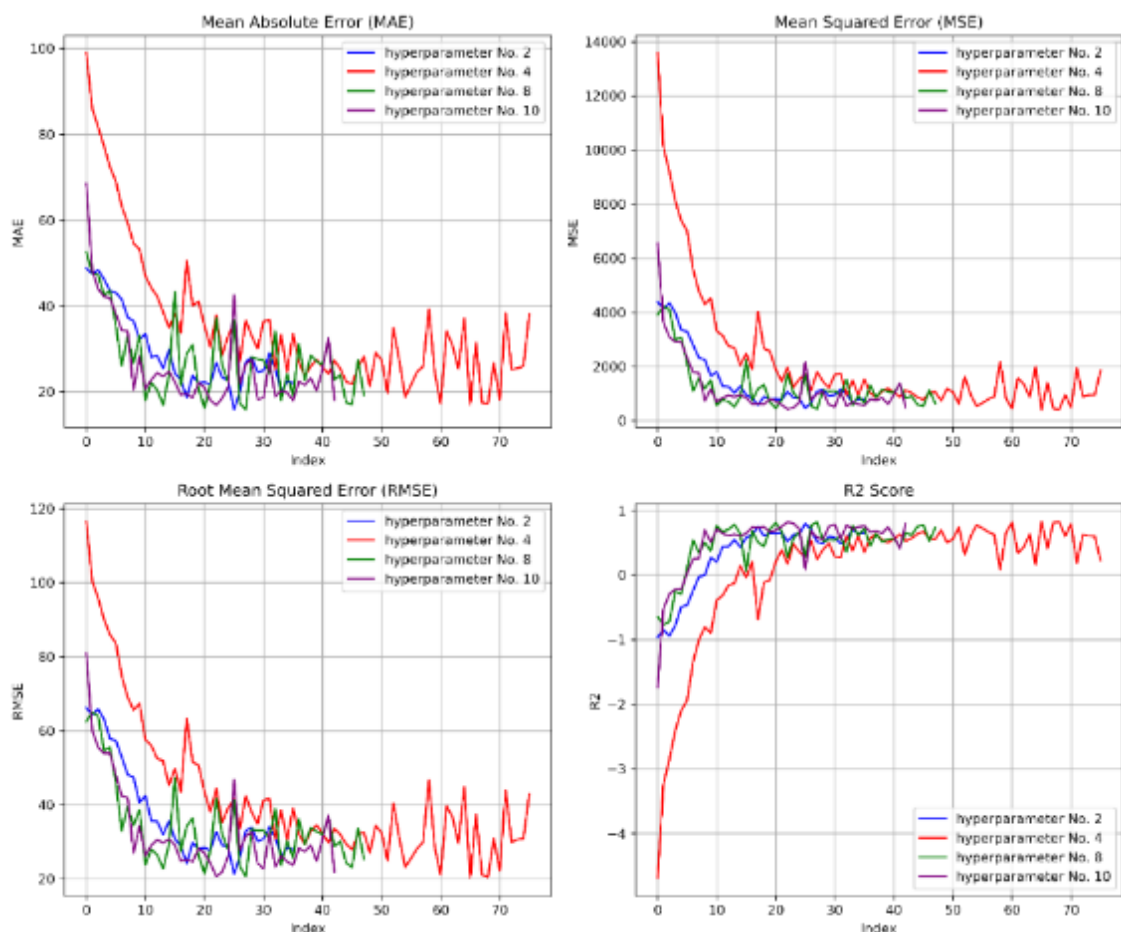
Pada pengujian model *GRU* dengan jangka waktu prediksi 30 hari seperti yang dapat dilihat pada **Tabel 4.2**, nilai metrik evaluasi sedikit mengalami kenaikan dikarenakan jumlah data yang semakin banyak. Pada jangka waktu prediksi 30 hari, kombinasi parameter yang memiliki hasil terbaik berdasarkan nilai metrik evaluasi sedikit berbeda dengan jangka waktu 30 hari.

Pada jangka waktu 30 hari, kombinasi terbaik yaitu *model* dengan jumlah unit pada layer *GRU* sebanyak 32 dengan menggunakan *activation function sigmoid* dan *recurrent activation ReLu* serta jumlah *batch size* 32. Nilai metrik evaluasi *MAE*, *RMSE*, *MSE* dan R^2 pada kombinasi parameter ini masing-masing yaitu 20.25074, 24.83073, 616.5652 dan 0.74167.

Pada **Gambar 4.12**, dapat dilihat pergerakan nilai metrik evaluasi model pada *hyperparameter* no. 4 mengalami peningkatan perlahan menuju nilai terbaik dengan jumlah *epoch* yang tergolong banyak dibandingkan kombinasi parameter lainnya.

Tabel 4.2 Nilai metrik evaluasi setelah pelatihan model *GRU* dengan jangka prediksi 30 hari

No	Jumlah Unit GRU Layer	Activation Function GRU Layer	Recurrent Activation Function GRU Layer	Batch Size	MAE	RMSE	MSE	R ²
1	32	relu	sigmoid	32	19.67317	23.43489	549.1939	0.769897
2	32	relu	sigmoid	16	16.00944	21.46797	460.8739	0.806902
3	32	tanh	relu	32	18.06866	22.25092	495.1034	0.79256
4	32	sigmoid	relu	32	17.0211	20.3052	412.3012	0.827253
5	32	tanh	tanh	32	20.25074	24.83073	616.5652	0.74167
6	32	relu	tanh	32	22.64362	27.78877	772.2156	0.676455
7	32	sigmoid	tanh	32	23.51504	27.87004	776.7389	0.67456
8	64	relu	sigmoid	32	15.88685	20.68554	427.8914	0.820721
9	128	relu	sigmoid	32	17.42758	22.58934	510.2783	0.786202
10	64	sigmoid	relu	32	16.91812	20.74947	430.5403	0.819611



Gambar 4.12 Pergerakan nilai metrik evaluasi selama pelatihan *model* dari empat kombinasi parameter terbaik pada algoritma *GRU* dengan jangka prediksi 30 hari

Pada **Tabel 4.3**, dapat dilihat daftar nilai metrik evaluasi yang diuji pada *model GRU* dengan jangka prediksi 60 hari. Dapat dilihat hasil terbaik berdasarkan nilai metrik

evaluasi memiliki kombinasi parameter dengan jumlah unit pada *layer GRU* berjumlah 64 dengan *activation function sigmoid*, *recurrent activation function ReLu* serta jumlah *batch size* 64. Nilai metrik evaluasi *MAE*, *RMSE*, *MSE* dan R^2 pada kombinasi parameter ini masing-masing yaitu 19.96351, 24.05158, 578.4784 dan 0.761528.

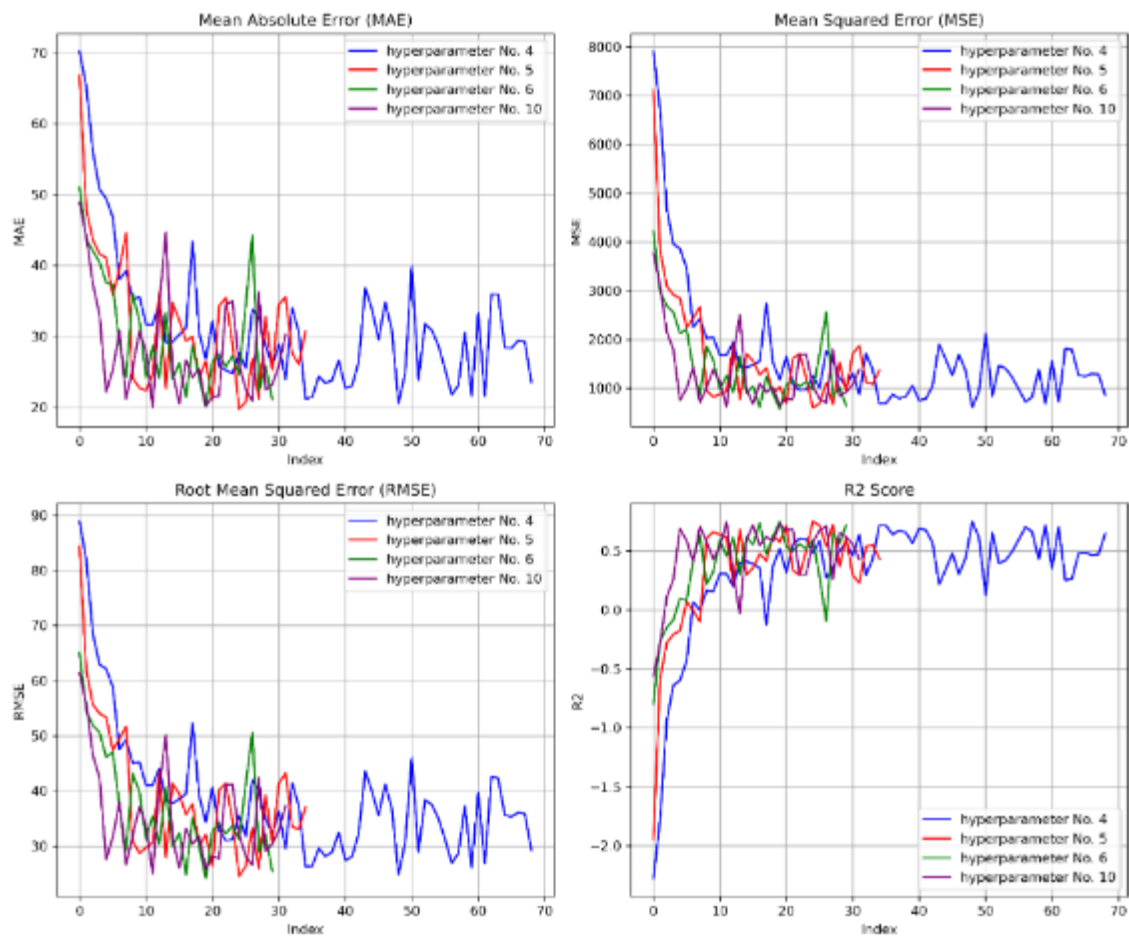
Pada grafik metrik evaluasi selama proses *model training* yang dapat dilihat pada **Gambar 4.13**, terlihat terjadi fluktuasi nilai metrik evaluasi yang cukup signifikan walaupun tetap terjadi peningkatan perlahan menuju posisi terbaik.

Tabel 4.3 Nilai metrik evaluasi setelah pelatihan model *GRU* dengan jangka prediksi 60 hari

No	Jumlah Unit GRU Layer	Activation Function GRU Layer	Recurrent Activation Function GRU Layer	Batch Size	MAE	RMSE	MSE	R^2
1	32	relu	sigmoid	32	27.24529	33.06943	1093.587	0.549179
2	32	tanh	relu	32	26.19533	33.2864	1107.984	0.543244
3	32	relu	relu	32	20.99422	25.95526	673.6754	0.722284
4	32	sigmoid	relu	32	20.5511	24.79541	614.8125	0.74655
5	64	sigmoid	relu	32	19.74867	24.51265	600.8699	0.752297
6	64	sigmoid	relu	64	19.96351	24.05158	578.4784	0.761528
7	32	tanh	tanh	32	22.63205	26.36299	695.0071	0.71349
8	64	tanh	tanh	32	22.02069	25.84772	668.1047	0.72458
9	64	relu	tanh	16	21.28124	25.1009	630.0551	0.740266
10	128	sigmoid	relu	32	19.90808	24.95245	622.6249	0.743329

Pada pelatihan model *GRU* dengan jangka prediksi 90 hari, terlihat pada **Tabel 4.4** model *GRU* mampu mempertahankan nilai metrik evaluasi jika dibandingkan dengan jangka waktu 60 hari. Terjadi sedikit peningkatan jika dibandingkan dengan jangka waktu 60 hari, tetapi hal ini tergolong baik dikarenakan perbedaan yang tidak terlalu signifikan.

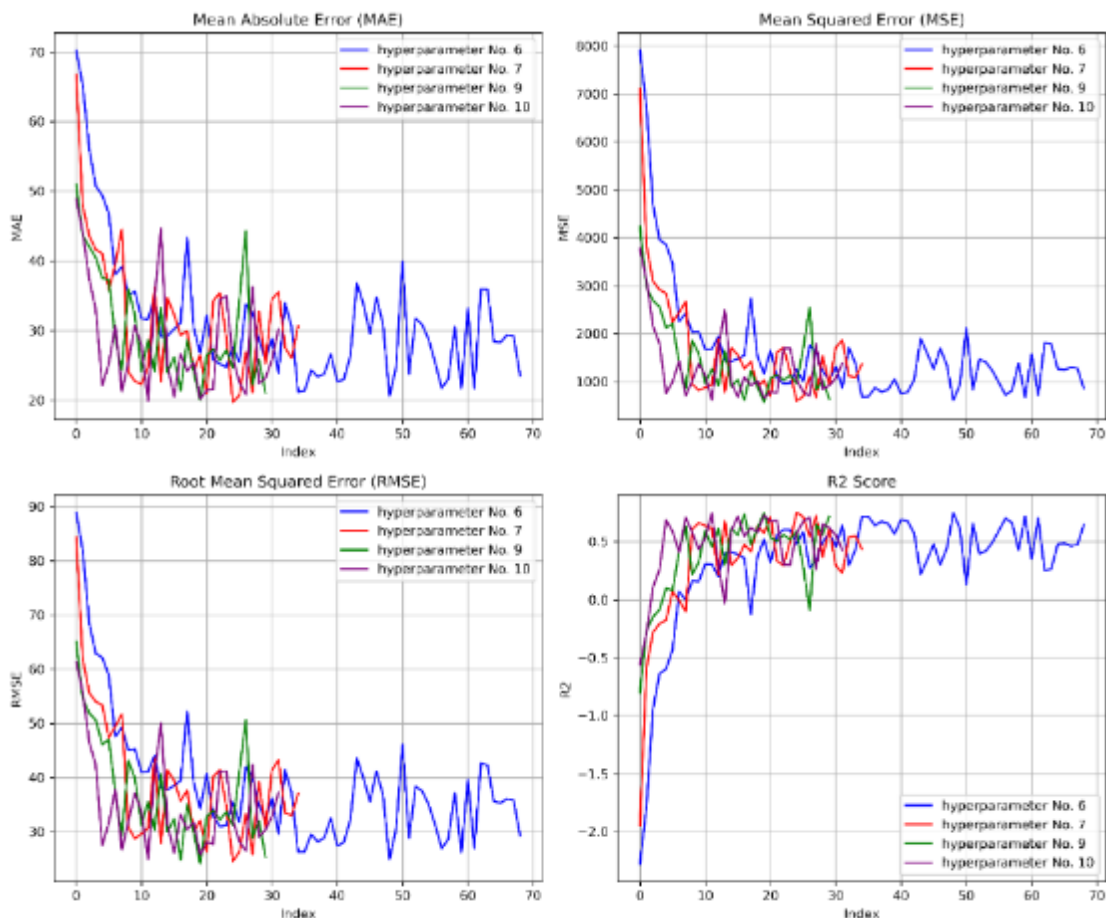
Pada **Gambar 4.14** juga dapat dilihat fluktuasi yang terjadi saat pelatihan model seperti pada pelatihan *GRU* dengan jangka prediksi 60 hari. Tetapi pada akhirnya pelatihan model tetap mengalami peningkatan perlahan menuju nilai metrik evaluasi terbaik.



Gambar 4.13 Pergerakan nilai metrik evaluasi selama pelatihan *model* dari empat kombinasi parameter terbaik pada algoritma *GRU* dengan jangka prediksi 60 hari

Tabel 4.4 Nilai metrik evaluasi setelah pelatihan model *GRU* dengan jangka prediksi 90 hari

No	Jumlah Unit GRU Layer	Activation Function GRU Layer	Recurrent Activation Function GRU Layer	Batch Size	MAE	RMSE	MSE	R ²
1	32	relu	sigmoid	16	21.36056	26.88745	722.9347	0.64733
2	64	tanh	relu	32	21.41106	26.38054	695.9328	0.660502
3	128	tanh	relu	32	21.75858	25.4055	645.4392	0.685134
4	128	tanh	relu	16	21.18083	25.5245	651.5003	0.682177
5	64	relu	relu	32	21.02331	26.3121	692.3267	0.662261
6	128	relu	relu	32	19.7836	24.44932	597.7691	0.708389
7	128	sigmoid	relu	32	20.52638	25.34016	642.1237	0.686752
8	32	tanh	tanh	32	20.94288	26.67071	711.327	0.652992
9	64	tanh	tanh	64	19.99927	24.881	619.0644	0.698001
10	128	relu	tanh	16	21.90147	24.95422	622.7131	0.696221



Gambar 4.14 Pergerakan nilai metrik evaluasi selama pelatihan *model* dari empat kombinasi parameter terbaik pada algoritma *GRU* dengan jangka prediksi 90 hari

4.3.2 Hasil Pelatihan Model Long Short-Term Memory

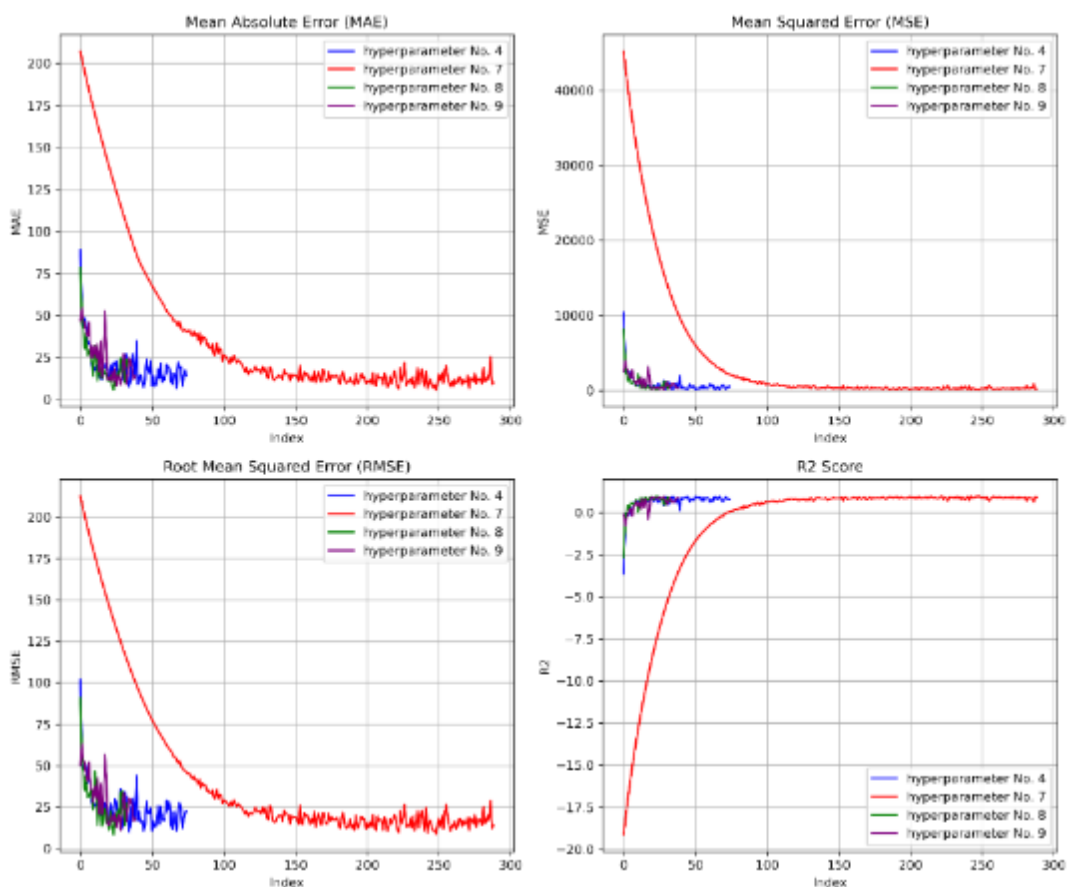
Untuk model *Long Short-Term Memory (LSTM)* parameter yang dilakukan *tuning* hampir mirip seperti pada *GRU*. Perbedaannya hanya terletak pada jumlah *unit* yang menggunakan *LSTM layer*. Keterangan parameter yang dibandingkan pada hyperparameter tuning menggunakan algoritma *LSTM* dapat dilihat pada **Tabel 3.4**. Setiap kombinasi parameter pada proses dicatat untuk menentukan kombinasi parameter terbaik berdasarkan keempat nilai metrik evaluasi yang telah ditentukan.

Pada pelatihan model *LSTM* dengan jangka waktu prediksi 7 hari, terlihat nilai metrik evaluasi yang lebih baik dibandingkan dengan algoritma *GRU* sebelumnya pada jangka waktu yang sama. Dengan kombinasi parameter yaitu jumlah *unit* pada *layer LSTM* sebanyak 128 dengan *activation function ReLu* dan *recurrent activation function hyperbolic tangent (Tanh)* serta jumlah *batch size 32*, model *LSTM* dapat memperoleh

nilai MAE , $RMSE$, MSE dan R^2 masing-masing 5.79068, 8.397713, 70.52159 dan 0.968609.

Tabel 4.5 Nilai metrik evaluasi setelah pelatihan model *LSTM* dengan jangka prediksi 7 hari

No	Jumlah Unit LSTM Layer	Activation Function LSTM Layer	Recurrent Activation Function LSTM Layer	Batch Size	MAE	RMSE	MSE	R^2
1	32	relu	sigmoid	32	9.852469	14.11917	199.351	0.911263
2	64	relu	sigmoid	32	7.273964	11.37264	129.337	0.942428
3	64	tanh	relu	32	9.029528	13.17418	173.5591	0.922744
4	64	tanh	relu	16	7.475403	9.928094	98.56704	0.956125
5	32	sigmoid	relu	32	9.059092	13.41086	179.8513	0.919943
6	128	sigmoid	relu	32	9.1883	11.51272	132.5428	0.941001
7	64	tanh	tanh	32	5.722169	8.918053	79.53166	0.964598
8	128	relu	tanh	32	5.79068	8.397713	70.52159	0.968609
9	128	relu	tanh	16	8.715781	11.27197	127.0574	0.943443
10	128	sigmoid	tanh	16	8.376715	12.95971	167.9541	0.925239



Gambar 4.15 Pergerakan nilai metrik evaluasi selama pelatihan *model* dari empat kombinasi parameter terbaik pada algoritma *LSTM* dengan jangka prediksi 7 hari

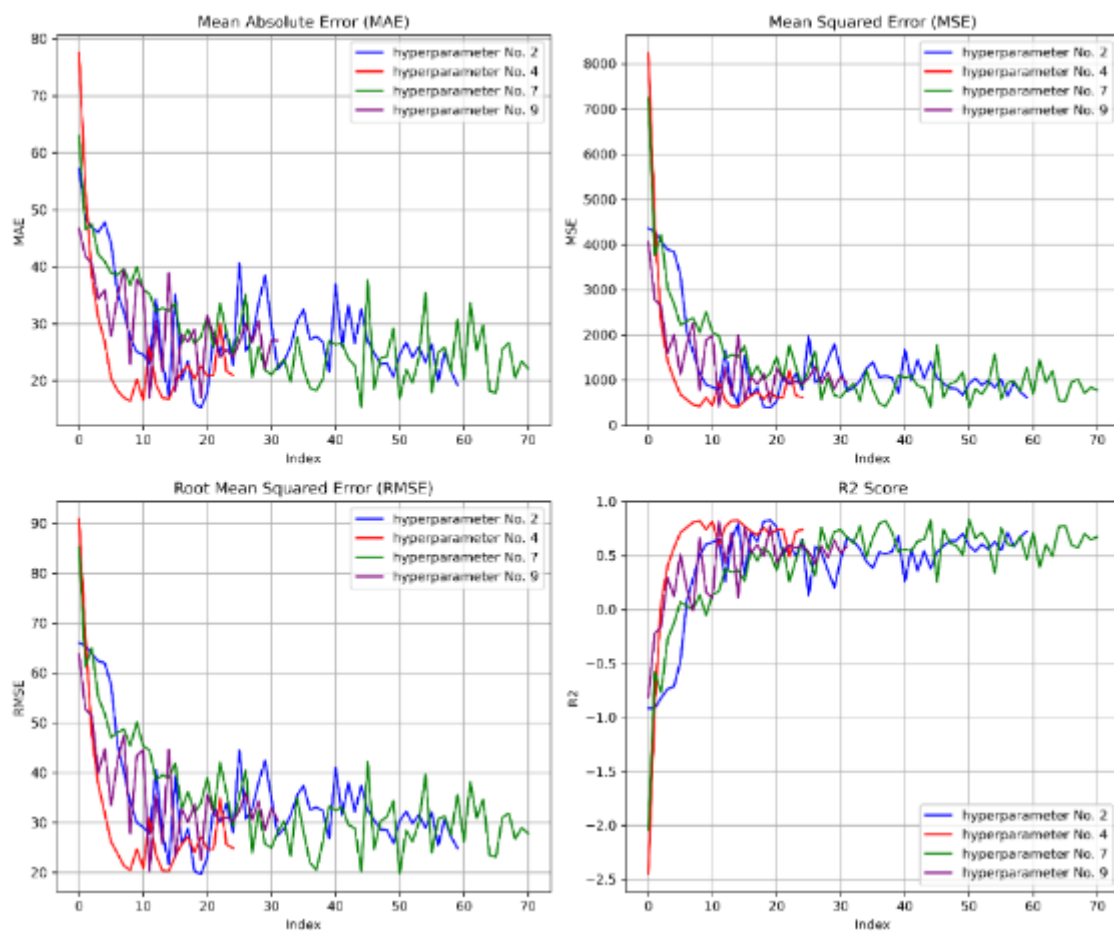
Pada jangka waktu prediksi 30 hari, mulai terlihat sedikit penurunan performa berdasarkan metrik evaluasi dibandingkan dengan jangka waktu 7 hari. Dengan hasil terbaik pada kombinasi jumlah *unit* pada *layer LSTM* berjumlah 128 dengan *activation function sigmoid*, *recurrent activation ReLu* dan ukuran *batch size* 32 menghasilkan nilai metrik evaluasi *MAE*, *RMSE*, *MSE* dan R^2 masing-masing 16.88909, 19.76373, 390.6052 dan 0.836343.

Pada **Gambar 4.15**, terlihat satu hyperparameter pada no. 7 memiliki garis yang paling panjang dikarenakan pembelajaran yang sedikit lambat dibandingkan tiga hyperparameter lain sehingga membutuhkan epoch yang lebih lama untuk mencapai metrik evaluasi terbaik.

Tabel 4.6 Nilai metrik evaluasi setelah pelatihan model *LSTM* dengan jangka prediksi 30 hari

No	Jumlah Unit LSTM Layer	Activation Function LSTM Layer	Recurrent Activation Function LSTM Layer	Batch Size	MAE	RMSE	MSE	R^2
1	32	relu	sigmoid	32	16.32841	21.21903	450.2471	0.811354
2	32	relu	sigmoid	16	15.88924	19.85069	394.05	0.8349
3	32	tanh	relu	32	17.89623	22.39885	501.7083	0.789793
4	32	relu	relu	32	16.77598	20.29747	411.9872	0.827384
5	32	relu	relu	64	15.8797	20.66057	426.8592	0.821153
6	64	sigmoid	relu	32	18.84417	23.92216	572.2695	0.760229
7	128	sigmoid	relu	32	16.88909	19.76373	390.6052	0.836343
8	128	relu	tanh	32	18.38759	21.44238	459.7755	0.807362
9	128	relu	tanh	16	16.97399	20.25494	410.2624	0.828107
10	128	sigmoid	tanh	32	18.45203	24.71684	610.922	0.744034

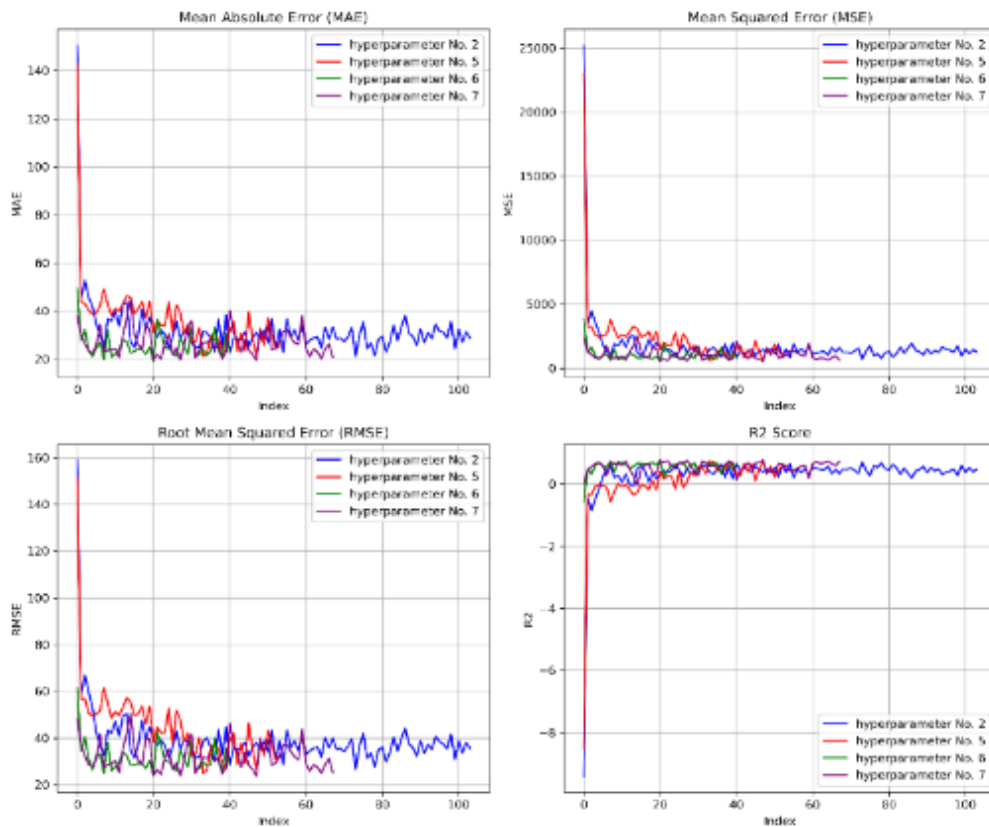
Pada jangka waktu prediksi 60 hari, terjadi kembali penurunan performa *model LSTM* berdasarkan empat metrik evaluasi. Dengan kombinasi parameter dengan jumlah *unit* pada *layer LSTM* berjumlah 128 dengan *activation function ReLu*, *recurrent activation function ReLu* serta jumlah *batch size* 32, memperoleh nilai metrik evaluasi *MAE*, *RMSE*, *MSE* dan R^2 masing-masing 19.47861, 23.66118, 559.8512 dan 0.769207.



Gambar 4.16 Pergerakan nilai metrik evaluasi selama pelatihan *model* dari empat kombinasi parameter terbaik pada algoritma *LSTM* dengan jangka prediksi 30 hari

Tabel 4.7 Nilai metrik evaluasi setelah pelatihan model *LSTM* dengan jangka prediksi 60 hari

No	Jumlah Unit LSTM Layer	Activation Function LSTM Layer	Recurrent Activation Function LSTM Layer	Batch Size	MAE	RMSE	MSE	R ²
1	32	relu	sigmoid	32	23.0757	27.58055	760.6867	0.686414
2	64	relu	sigmoid	32	21.03248	26.45211	699.7143	0.71155
3	32	tanh	relu	32	24.1987	31.54762	995.2523	0.589717
4	64	tanh	relu	32	24.40137	28.60188	818.0675	0.66276
5	128	tanh	relu	32	21.20276	24.90934	620.4751	0.744215
6	64	relu	relu	32	19.16015	23.91599	571.9748	0.764209
7	128	relu	relu	32	19.47861	23.66118	559.8512	0.769207
8	32	tanh	tanh	32	24.5149	28.50108	812.3117	0.665132
9	64	relu	tanh	32	22.37859	28.60103	818.0186	0.66278
10	128	relu	tanh	16	22.28446	26.87438	722.2323	0.702267



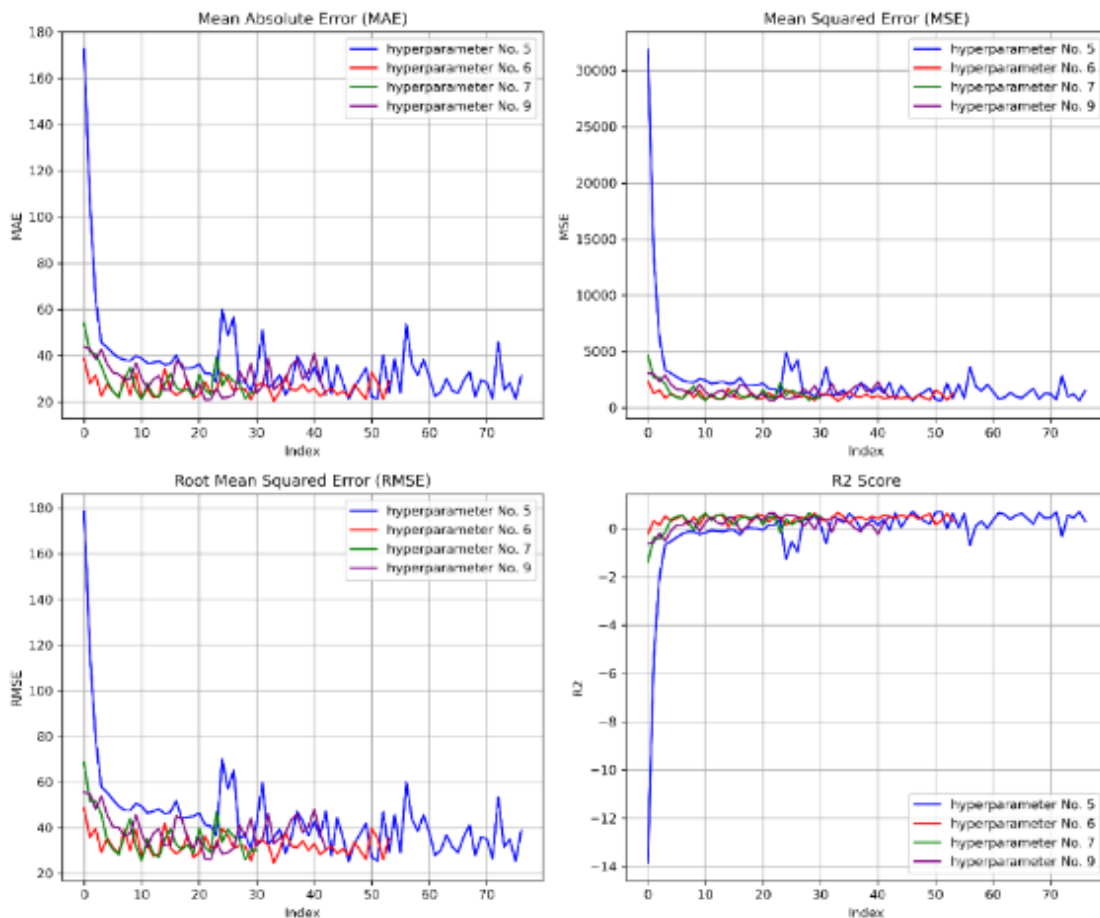
Gambar 4.17 Pergerakan nilai metrik evaluasi selama pelatihan *model* dari empat kombinasi parameter terbaik pada algoritma *LSTM* dengan jangka prediksi 60 hari

Tabel 4.8 Nilai metrik evaluasi setelah pelatihan model *LSTM* dengan jangka prediksi 90 hari

No	Jumlah Unit LSTM Layer	Activation Function LSTM Layer	Recurrent Activation Function LSTM Layer	Batch Size	MAE	RMSE	MSE	R ²
1	64	relu	sigmoid	16	22.47697	29.23956	854.9516	0.582927
2	64	relu	sigmoid	64	21.97785	27.15866	737.5929	0.640179
3	64	relu	sigmoid	128	20.90439	27.33289	747.0869	0.635547
4	128	tanh	relu	16	22.62523	29.08881	846.1586	0.587217
5	128	tanh	relu	64	21.2172	25.29805	639.9914	0.687792
6	64	relu	relu	32	20.42038	24.62833	606.5549	0.704103
7	128	relu	relu	32	21.118	25.70691	660.8454	0.677619
8	128	sigmoid	relu	32	22.68835	26.59027	707.0424	0.655082
9	128	sigmoid	relu	16	21.00832	25.98494	675.2173	0.670608
10	64	relu	tanh	32	21.80488	27.1314	736.1127	0.640901

Pada pelatihan model LSTM dengan jangka waktu prediksi 90 hari, terjadi kembali sedikit penurunan performa dibandingkan jangka waktu prediksi 60 hari. Pada **Tabel 4.8** dapat dilihat kombinasi terbaik berdasarkan keempat nilai metrik evaluasi didapat dengan kombinasi jumlah *unit* pada *layer LSTM* sebanyak 64 dengan *activation function ReLu* dan *recurrent activation ReLu* pada ukuran *batch size* 32. Nilai metrik evaluasi *MAE*,

$RMSE$, MSE dan R^2 pada kombinasi terbaik masing-masing 20.42038, 24.62833, 606.5549 dan 0.704103.



Gambar 4.18 Pergerakan nilai metrik evaluasi selama pelatihan *model* dari empat kombinasi parameter terbaik pada algoritma *LSTM* dengan jangka prediksi 60 hari

4.3.3 Hasil Pelatihan Model Deep Q-Network

Pada model *Deep Q-Network* (*DQN*) parameter yang dilakukan *tuning* berbeda dari algoritma *LSTM* maupun *GRU*. Terdapat dua parameter yang akan dilakukan *tuning* pada algoritma *DQN* yaitu *actions* dan *steps*. *Actions* akan digunakan sebagai jumlah *node* pada *output layer neural network*, sedangkan *steps* adalah jumlah langkah yang diambil oleh *agent* selama pelatihan model. Detail *hyperparameter tuning* untuk algoritma *DQN* dapat dilihat pada **Tabel 3.3**.

Sama halnya seperti pada algoritma *LSTM* dan *GRU*, kombinasi parameter akan dibandingkan dan dinilai menggunakan empat metrik evaluasi yaitu MSE , $RMSE$, MAE dan R^2 . Semua hasil dari model *DQN* merupakan hasil terbaik dari tiga kali pelatihan untuk masing-masing kombinasi parameter.

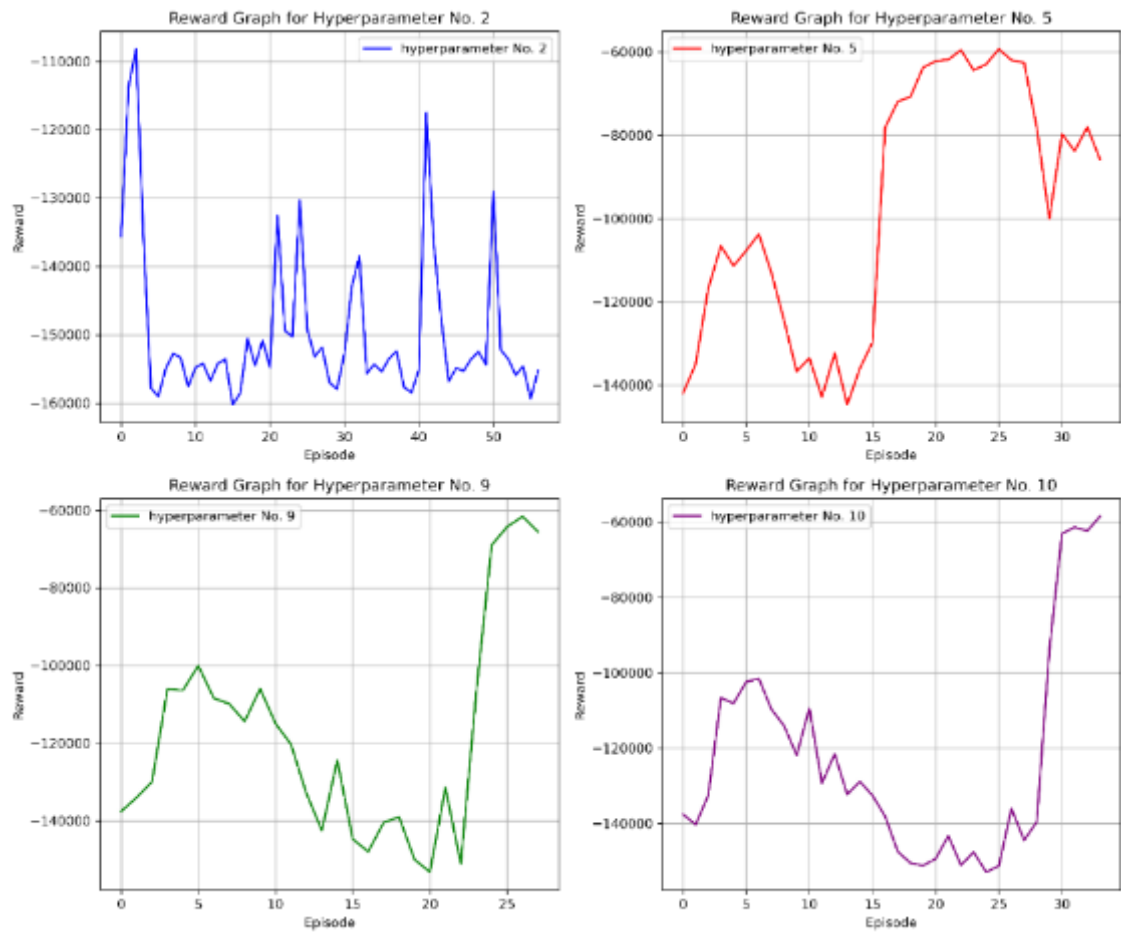
Pada **Tabel 4.9** terlihat dari seluruh kombinasi parameter model *DQN* yang tercatat, hanya satu kombinasi parameter yang dapat menyentuh nilai metrik evaluasi yang cukup baik dengan nilai *RMSE* 9.808426836.

Pada **Gambar 4.19**, pada jumlah *actions* 1200 keatas terlihat pelatihan *model* yang lebih stabil menuju puncak dikarenakan pilihan prediksi yang lebih banyak yang dapat dipilih oleh *agent*. Jumlah *actions* yang terlalu banyak seperti pada *hyperparameter* no. 8, juga dapat meningkatkan tingkat kompleksitas *neural network* yang membuat *agent* lebih sulit dalam melakukan prediksi.

Tabel 4.9 Nilai metrik evaluasi dari pengujian model *DQN* dengan jangka prediksi 7 hari

No	actions	Jumlah actions	step	MAE	RMSE	MSE	R ²
1	$x \in \mathbb{R} \mid x = 100 + 0.5n;$	600	70000	19.82345132	22.84619908	521.9488122	0.76766474
2	$x \in \mathbb{R} \mid x = 100 + 0.5n;$	600	100000	15.07397286	19.13102279	365.9960329	0.837084056
3	$x \in \mathbb{R} \mid x = 100 + 0.25n;$	1200	50000	16.72840647	21.1406808	446.9283845	0.801058609
4	$x \in \mathbb{R} \mid x = 100 + 0.125n;$	2400	50000	18.80305963	21.26440227	452.174804	0.798723269
5	$x \in \mathbb{R} \mid x = 100 + 0.25n;$	1200	60000	8.877353627	9.808426836	96.205237	0.957176129
6	$x \in \mathbb{R} \mid x = 100 + 0.25n;$	1200	70000	19.0333945	23.36272365	545.8168565	0.757040349
7	$x \in \mathbb{R} \mid x = 100 + 0.25n;$	1200	40000	17.33258955	19.32388568	373.4125578	0.833782736
8	$x \in \mathbb{R} \mid x = 100 + 0.125n;$	2400	60000	21.2441989	24.61950712	606.120131	0.730197531
9	$x \in \mathbb{R} \mid x = 100 + 0.2n;$	1500	50000	16.36698198	18.61833455	346.6423813	0.845698954
10	$x \in \mathbb{R} \mid x = 100 + 0.2n;$	1500	60000	14.47433622	17.2263119	296.7458217	0.867909427

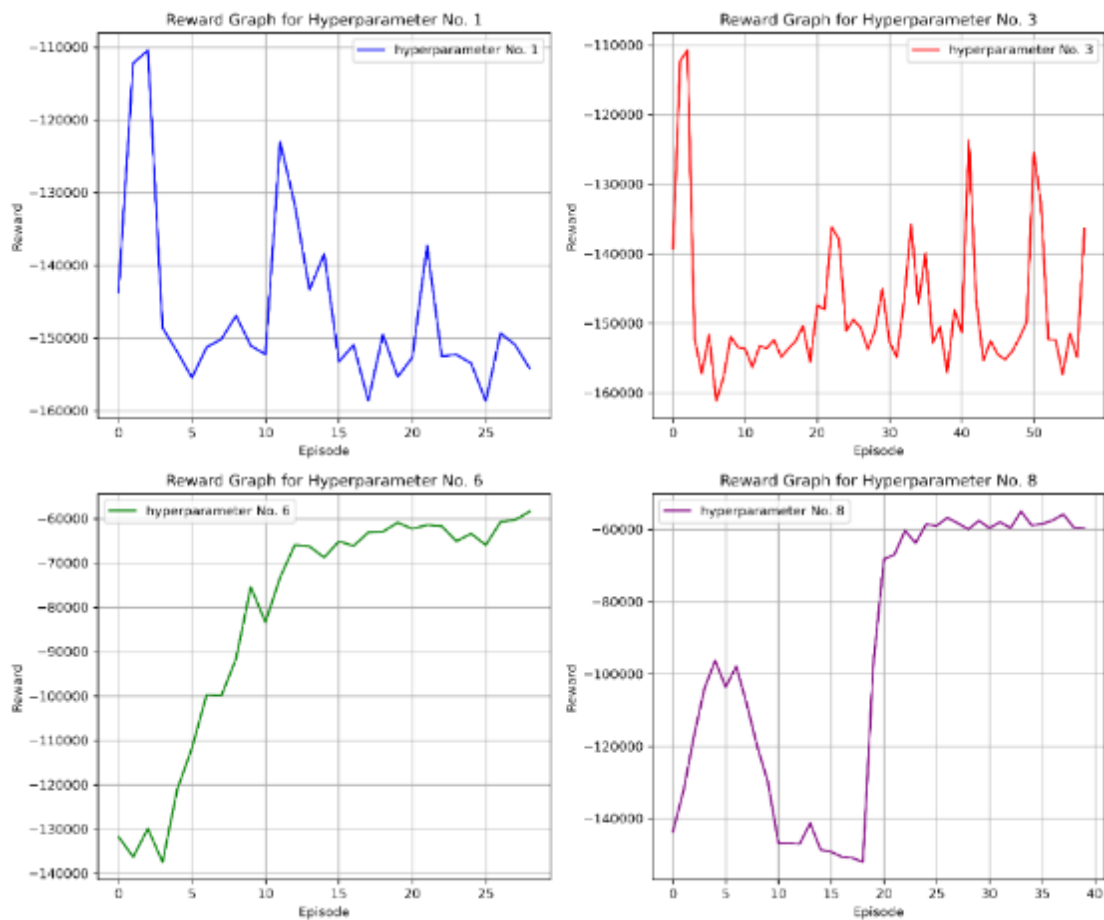
Pada pelatihan model *DQN* dengan jangka waktu prediksi 30 hari dapat terlihat satu parameter dengan pembelajaran yang cukup stabil pada *hyperparameter* no. 6 yang dapat dilihat pada **Gambar 4.20**. Pada pelatihan model *DQN*, fluktuasi pada saat pelatihan model sebenarnya sangat wajar dikarenakan adanya perhitungan acak diawal pelatihan agar *agent* dapat mempelajari *environment* dengan lebih baik. Seiring dengan bertambahnya *episode*, kemungkinan terjadinya perhitungan acak akan dikurangi agar *model* lebih mementingkan hasil dibandingkan mendapatkan informasi sebagai pembelajaran.



Gambar 4.19 Pergerakan nilai *reward* selama pelatihan *model* dari empat kombinasi parameter terbaik pada algoritma *DQN* dengan jangka prediksi 7 hari

Tabel 4.10 Nilai metrik evaluasi dari pengujian model *DQN* dengan jangka prediksi 30 hari

No	actions	Jumlah actions	step	MAE	RMSE	MSE	R ²
1	$x \in \mathbb{R} \mid x = 100 + 0.5n;$	600	50000	28.38938392	34.28200934	1175.256164	0.507587897
2	$x \in \mathbb{R} \mid x = 100 + 0.5n;$	600	70000	28.47947459	36.56431454	1336.949098	0.439841341
3	$x \in \mathbb{R} \mid x = 100 + 0.5n;$	600	100000	24.6339488	31.47403041	990.6145904	0.584949538
4	$x \in \mathbb{R} \mid x = 100 + n;$	400	50000	32.76248475	39.38385694	1551.088187	0.350120748
5	$x \in \mathbb{R} \mid x = 100 + 0.25n;$	1200	50000	28.89052368	38.84874864	1509.225271	0.367660589
6	$x \in \mathbb{R} \mid x = 100 + 0.125n;$	2400	50000	24.81868263	31.91717672	1018.70617	0.573179649
7	$x \in \mathbb{R} \mid x = 100 + 0.25n;$	1200	60000	29.64952973	36.48891224	1331.440716	0.442149257
8	$x \in \mathbb{R} \mid x = 100 + 0.25n;$	1200	70000	27.15950469	35.20753446	1239.570483	0.480641304
9	$x \in \mathbb{R} \mid x = 100 + 0.25n;$	1200	40000	29.64358551	36.52256166	1333.89751	0.441119903
10	$x \in \mathbb{R} \mid x = 100 + 0.125n;$	2400	60000	28.31441074	36.88052992	1360.173487	0.430110721



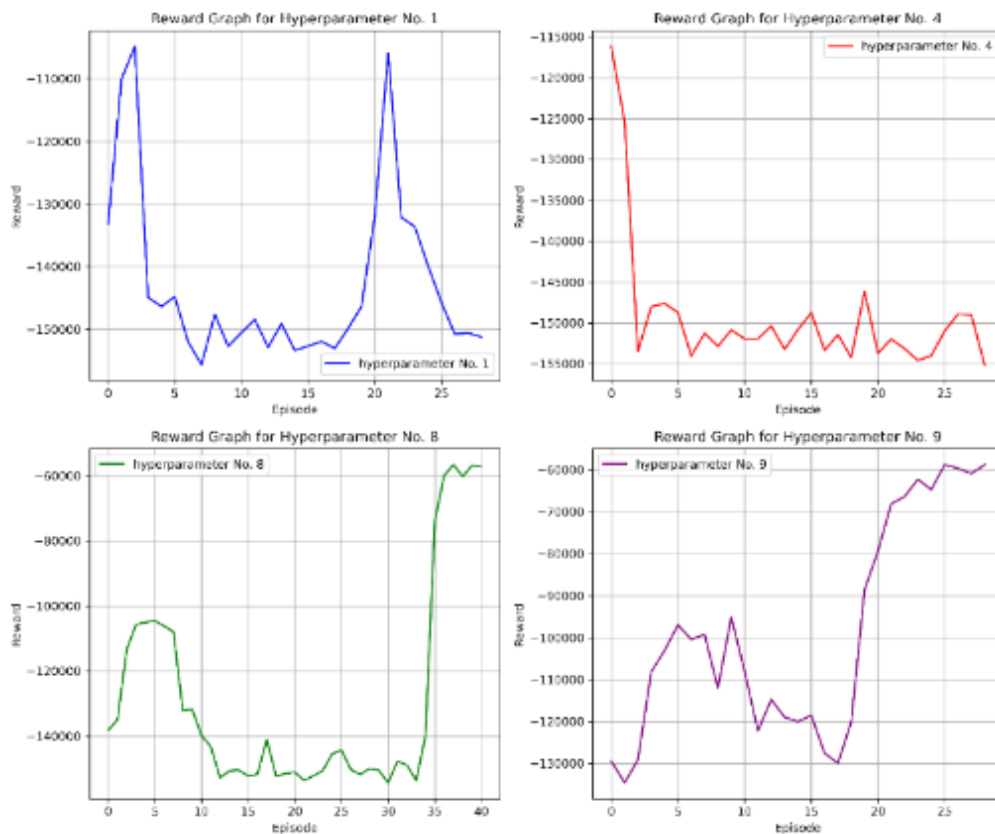
Gambar 4.20 Pergerakan nilai *reward* selama pelatihan *model* dari empat kombinasi parameter terbaik pada algoritma *DQN* dengan jangka prediksi 30 hari

Pada **Tabel 4.10**, terlihat model *DQN* masih kesulitan untuk melakukan prediksi dengan tingkat keakuratan yang tinggi. Kombinasi parameter terbaik dapat dilihat pada *hyperparameter* no. 3 dengan jumlah *actions* 600 dan jumlah *step* 100.000. Jika dilihat berdasarkan nilai metrik evaluasi, hasil ini masih tergolong kurang akurat jika dibandingkan dengan hasil pada algoritma *GRU* maupun *LSTM* sebelumnya.

Pada model *DQN* dengan jangka waktu prediksi 60 hari tidak terlihat peningkatan yang terlalu signifikan berdasarkan metrik evaluasi pada **Tabel 4.11**. Tetapi hasil ini masih belum dapat menyaingi algoritma *GRU* maupun *LSTM* berdasarkan keempat nilai metrik evaluasi.

Tabel 4.11 Nilai metrik evaluasi dari pengujian model *DQN* dengan jangka prediksi 60 hari

No	actions	Jumlah actions	step	MAE	RMSE	MSE	R ²
1	$x \in \mathbb{R} \mid x = 100 + 0.5n;$	600	50000	28.61743275	35.34983032	1249.610504	0.484860311
2	$x \in \mathbb{R} \mid x = 100 + 0.5n;$	600	70000	31.2609583	38.59913882	1489.893518	0.385805992
3	$x \in \mathbb{R} \mid x = 100 + n;$	600	100000	30.39641922	38.88740949	1512.230617	0.376597742
4	$x \in \mathbb{R} \mid x = 100 + 0.25n;$	400	50000	25.42871319	32.44452183	1052.646997	0.566056587
5	$x \in \mathbb{R} \mid x = 100 + 0.125n;$	1200	50000	29.02209431	36.36542386	1322.444052	0.454835394
6	$x \in \mathbb{R} \mid x = 100 + 0.25n;$	2400	50000	32.52821542	40.51219321	1641.237799	0.323415795
7	$x \in \mathbb{R} \mid x = 100 + 0.25n;$	1200	60000	26.57437451	35.53368864	1262.643029	0.47948778
8	$x \in \mathbb{R} \mid x = 100 + 0.125n;$	1200	70000	24.57290462	32.82820107	1077.690786	0.555732531
9	$x \in \mathbb{R} \mid x = 100 + 0.2n;$	1500	50000	27.22327526	34.03992463	1158.716469	0.522330487
10	$x \in \mathbb{R} \mid x = 100 + 0.2n;$	1500	60000	30.7178269	37.79998156	1428.838606	0.410975282

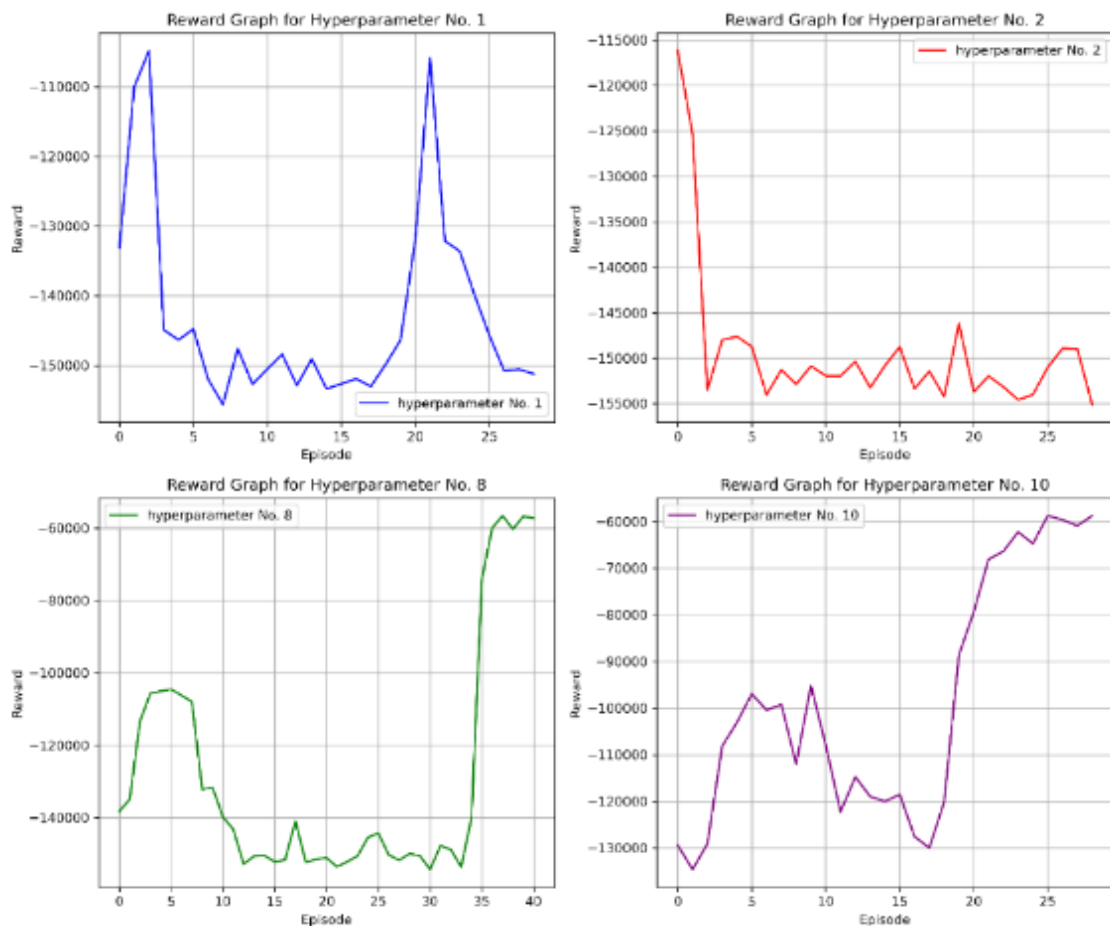


Gambar 4.21 Pergerakan nilai *reward* selama pelatihan *model* dari empat kombinasi parameter terbaik pada algoritma *DQN* dengan jangka prediksi 60 hari

Pada model *DQN* dengan jangka prediksi 90 hari terlihat kembali terjadi penurunan nilai metrik evaluasi dibandingkan dengan jangka waktu 60 hari. Seperi dapat dilihat pada **Tabel 4.12**, hasil metrik evaluasi terbaik didapat dengan nilai *MAE*, *RMSE*, *MSE* dan *R²* masing-masing 27.30856249, 34.39064066, 1182.716165 dan 0.423033524.

Tabel 4.12 Nilai metrik evaluasi dari pengujian model *DQN* dengan jangka prediksi 90 hari

No	actions	Jumlah actions	step	MAE	RMSE	MSE	R ²
1	$x \in \mathbb{R} \mid x = 100 + 0.5n;$	600	70000	27.84390387	34.88236843	1216.779628	0.406416287
2	$x \in \mathbb{R} \mid x = 100 + 0.5n;$	600	100000	28.20321542	36.1314888	1305.484483	0.363143243
3	$x \in \mathbb{R} \mid x = 100 + n;$	400	50000	30.90206678	38.28073238	1465.414471	0.28512432
4	$x \in \mathbb{R} \mid x = 100 + 0.25n;$	1200	50000	31.87980484	38.72391383	1499.541502	0.268476071
5	$x \in \mathbb{R} \mid x = 100 + 0.125n;$	2400	50000	29.592295	36.9308494	1363.887637	0.334652332
6	$x \in \mathbb{R} \mid x = 100 + 0.25n;$	1200	60000	30.03722385	38.50800574	1482.866506	0.276610664
7	$x \in \mathbb{R} \mid x = 100 + 0.25n;$	1200	70000	29.65892281	38.57462531	1488.001718	0.274105545
8	$x \in \mathbb{R} \mid x = 100 + 0.125n;$	2400	60000	27.30856249	34.39064066	1182.716165	0.423033524
9	$x \in \mathbb{R} \mid x = 100 + 0.2n;$	1500	50000	30.13792461	38.47296232	1480.16883	0.277926676
10	$x \in \mathbb{R} \mid x = 100 + 0.2n;$	1500	60000	29.03811918	35.79323537	1281.155698	0.375011597



Gambar 4.22 Pergerakan nilai *reward* selama pelatihan *model* dari empat kombinasi parameter terbaik pada algoritma *DQN* dengan jangka prediksi 90 hari

4.4 Pengujian Sistem

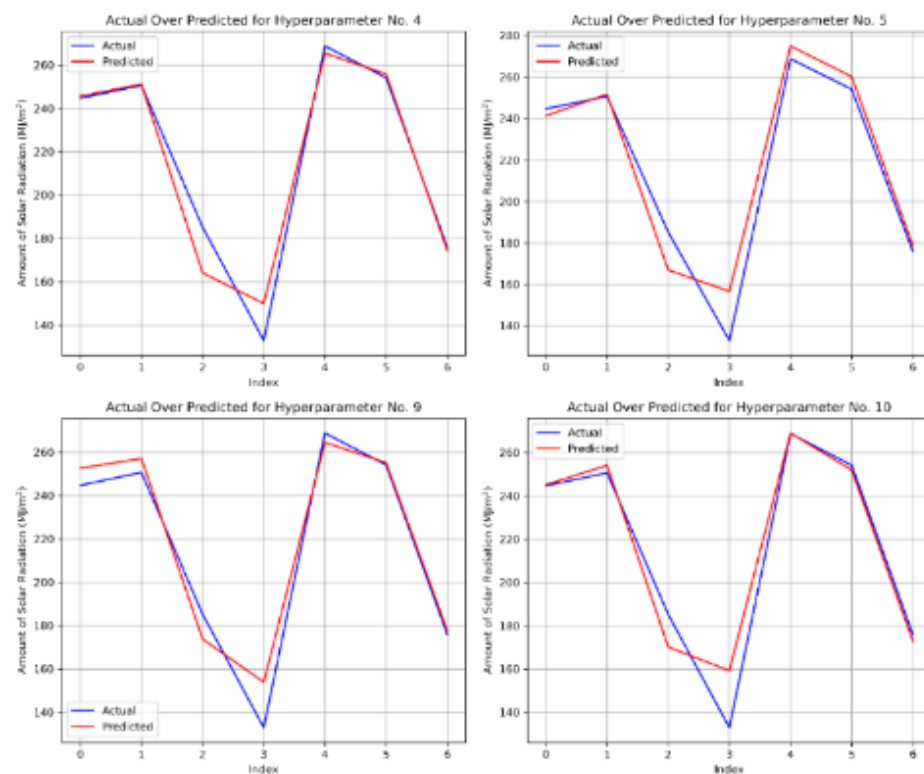
Setelah pelatihan sistem telah selesai dilakukan, untuk mengetahui kemampuan model lebih lanjut dalam melakukan prediksi, perlu dilakukan pengujian terhadap model yang

telah dibuat. Pengujian model akan dilakukan menggunakan empat metrik evaluasi yaitu *Mean Squared Error (MSE)*, *Root Mean Squared Error (RMSE)*, *Mean Absolute Error (MAE)*, dan R^2 (*Coefficient of Determination*). Semakin rendah nilai *MSE*, *RMSE* dan *MAE* berarti semakin baik model dalam melakukan prediksi. Sebaliknya semakin tinggi nilai R^2 menuju angka satu menunjukkan semakin baiknya model dalam melakukan prediksi.

Hasil pengujian nantinya akan terdiri dari 4 jenis jangka prediksi mulai dari 7 hari, 30 hari, 60 hari hingga 90 hari. Nantinya keseluruhan *model* akan dibandingkan untuk melihat kemampuan masing-masing algoritma dalam melakukan prediksi baik dalam jangka waktu yang pendek maupun jangka waktu Panjang.

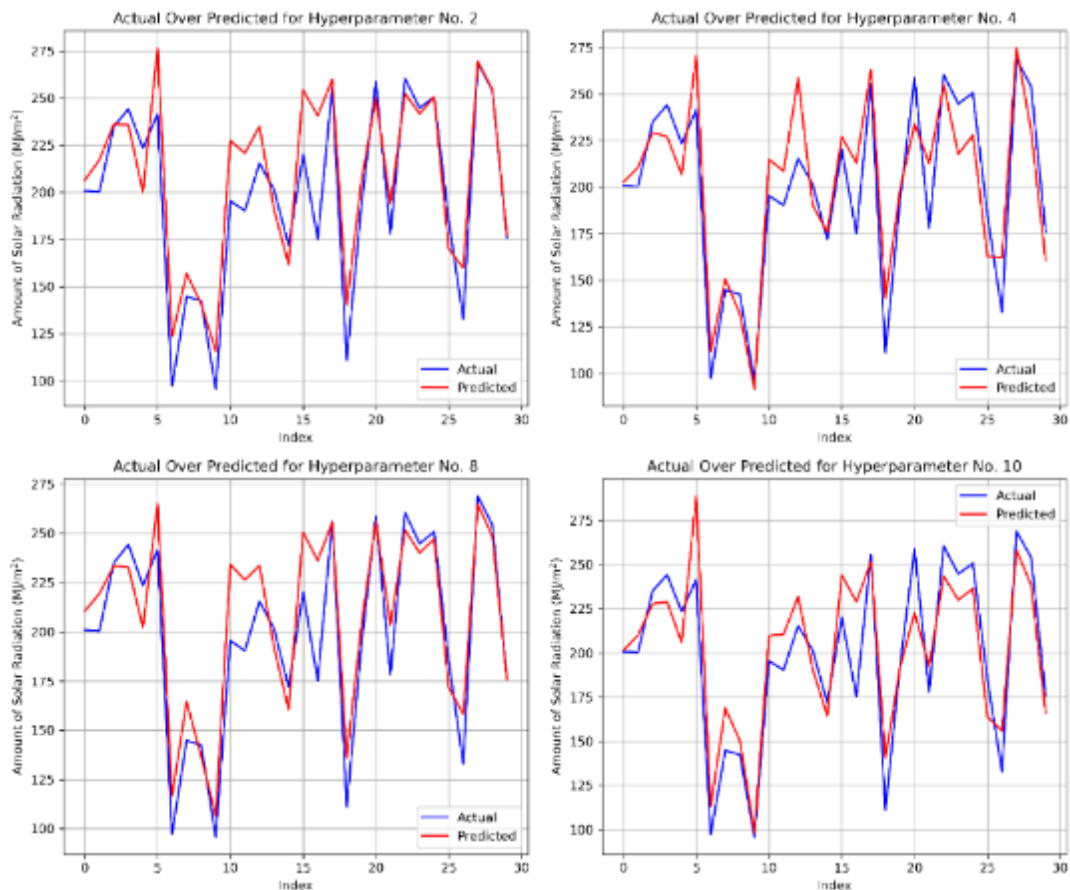
4.4.1 Hasil Pengujian Model Gated Recurrent Unit

Perbandingan hasil pengujian pada *model Gated Recurrent Unit (GRU)* untuk jangka prediksi 7 hari dapat dilihat pada **Tabel 4.1**. Pada tabel pengujian prediksi terlihat hyperparameter no. 4, 5, 9 serta 10 memiliki nilai metrik evaluasi terbaik dibandingkan hyperparameter lainnya. Dari keempat kombinasi parameter terbaik tersebut akan dibandingkan hasil prediksi oleh model dengan nilai prediksi sebenarnya pada *dataset*.



Gambar 4.23 Perbandingan nilai prediksi dan nilai sebenarnya dari empat kombinasi parameter terbaik pada algoritma *GRU* dengan jangka prediksi 7 hari

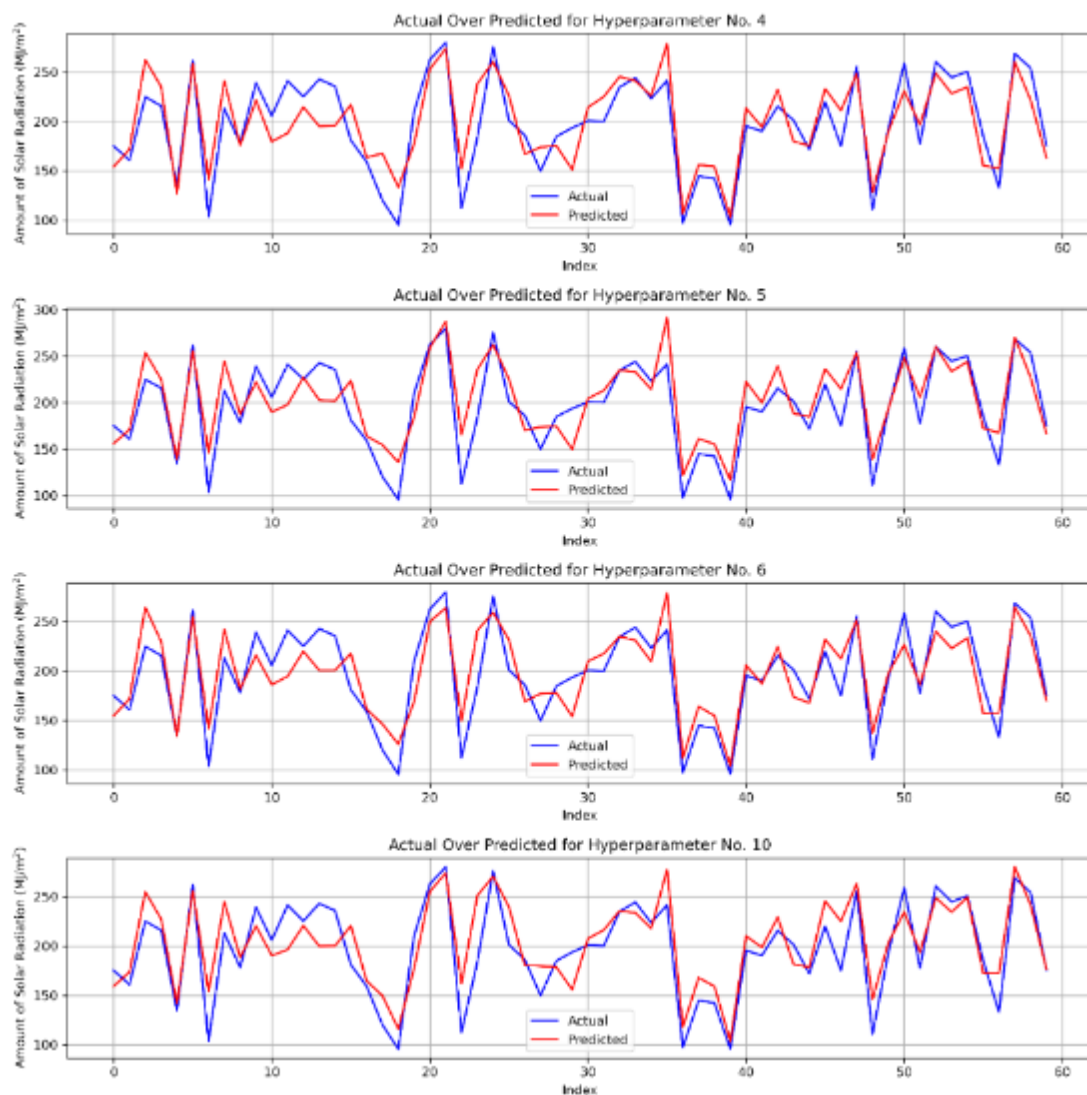
Untuk perbandingan data prediksi pada empat hasil terbaik dengan jangka waktu prediksi 7 hari akan dijabarkan pada **Gambar 4.23**. Terlihat pada jangka waktu prediksi 7 hari *GRU* cukup baik dalam melakukan prediksi. Walaupun pada prediksi hari ketiga dan keempat terjadi peningkatan *error* yang tidak terlalu signifikan tetapi pada hari-hari lain model *GRU* dapat memprediksi dengan sangat baik dengan jumlah *error* yang sangat sedikit.



Gambar 4.24 Perbandingan nilai prediksi dan nilai sebenarnya dari empat kombinasi parameter terbaik pada algoritma *GRU* dengan jangka prediksi 30 hari

Pada perbandingan hasil pengujian pada *model GRU* untuk jangka prediksi 30 hari yang dapat dilihat pada **Tabel 4.2**, *hyperparameter* no. 4 memiliki nilai metrik evaluasi terbaik dibandingkan kombinasi parameter lain. Dapat dilihat pada perbandingan nilai prediksi model dan nilai pada data sebenarnya pada **Gambar 4.24**, di *hyperparameter* no. 4 *model GRU* mampu memprediksi naik turunnya besaran radiasi matahari pada setiap harinya dengan sangat baik, walaupun masih terdapat satu kesalahan pada hari ke 4.

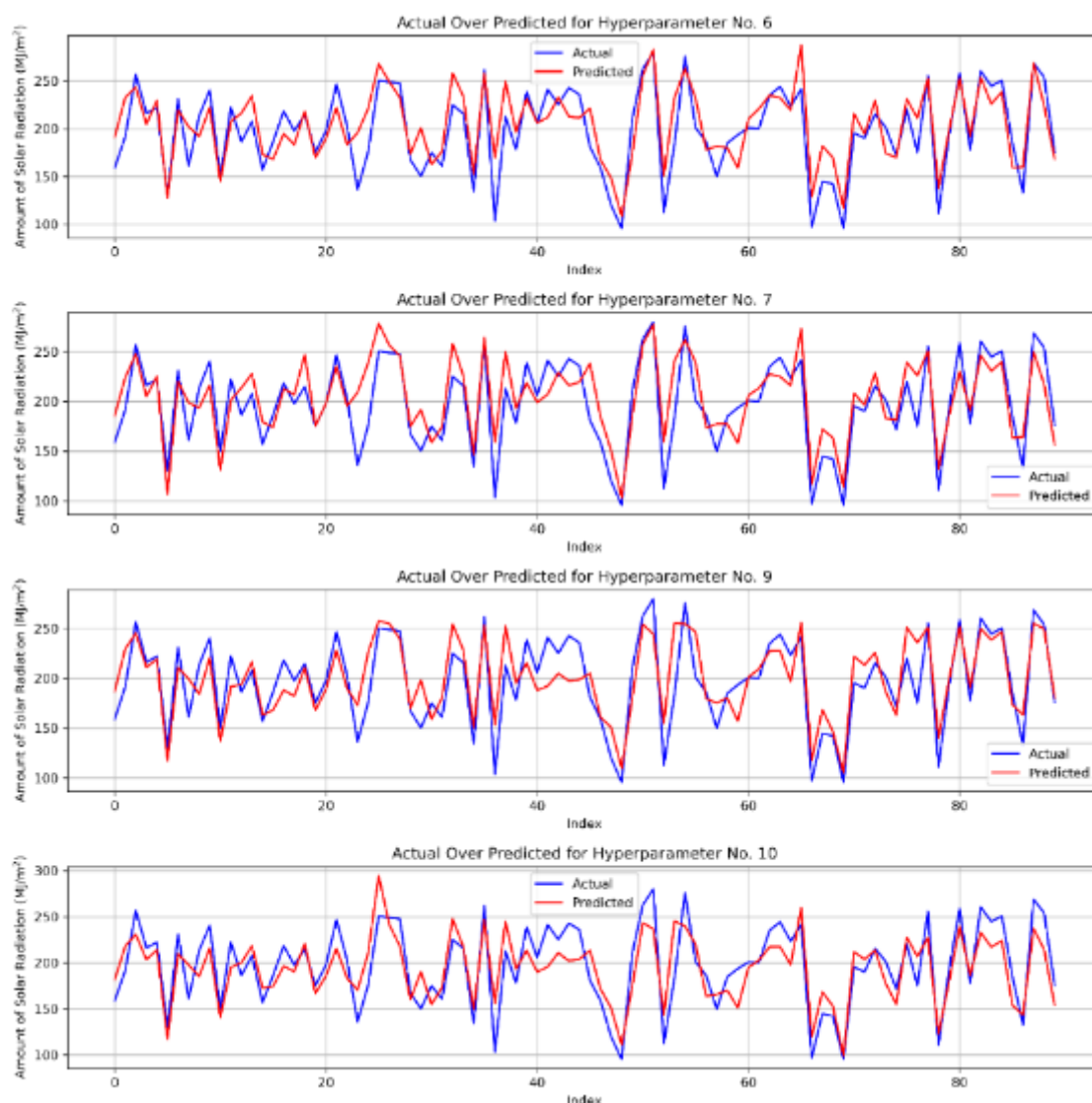
Pada perbandingan hasil pengujian pada *model GRU* untuk jangka prediksi 60 hari yang dapat dilihat pada **Tabel 4.3**, hyperparameter no. 6 memiliki hasil terbaik berdasarkan nilai metrik evaluasi. Pada perbandingan nilai prediksi model dan nilai sebenarnya yang dapat dilihat pada **Gambar 4.25**, terlihat mulai terdapat banyak kesalahan prediksi model dari segi naik turunnya nilai radiasi matahari per hari. Terjadi total 10 kesalahan prediksi naik turunnya nilai radiasi matahari per hari yang terdapat pada nilai prediksi *index* ke 2, 13, 14, 15, 16, 27, 29, 31, 33 serta 56.



Gambar 4.25 Perbandingan nilai prediksi dan nilai sebenarnya dari empat kombinasi parameter terbaik pada algoritma *GRU* dengan jangka prediksi 60 hari

Perbandingan hasil pengujian pada *model GRU* untuk jangka prediksi 90 hari dapat dilihat pada **Tabel 4.4**. Terlihat berdasarkan nilai metrik evaluasi hyperparameter no. 6 memiliki nilai terbaik. Untuk perbandingan data prediksi pada empat hasil terbaik akan

dijabarkan pada **Gambar 4.26**. Pada jangka waktu prediksi 90 hari, *model GRU* pada *hyperparameter* no. 6 memprediksi salah pada naik turunnya nilai besaran radiasi matahari sebanyak 14 kali yaitu pada *index* ke 9, 13, 16, 29, 30, 31, 42, 43, 45, 57, 58, 59, 61 serta 63.

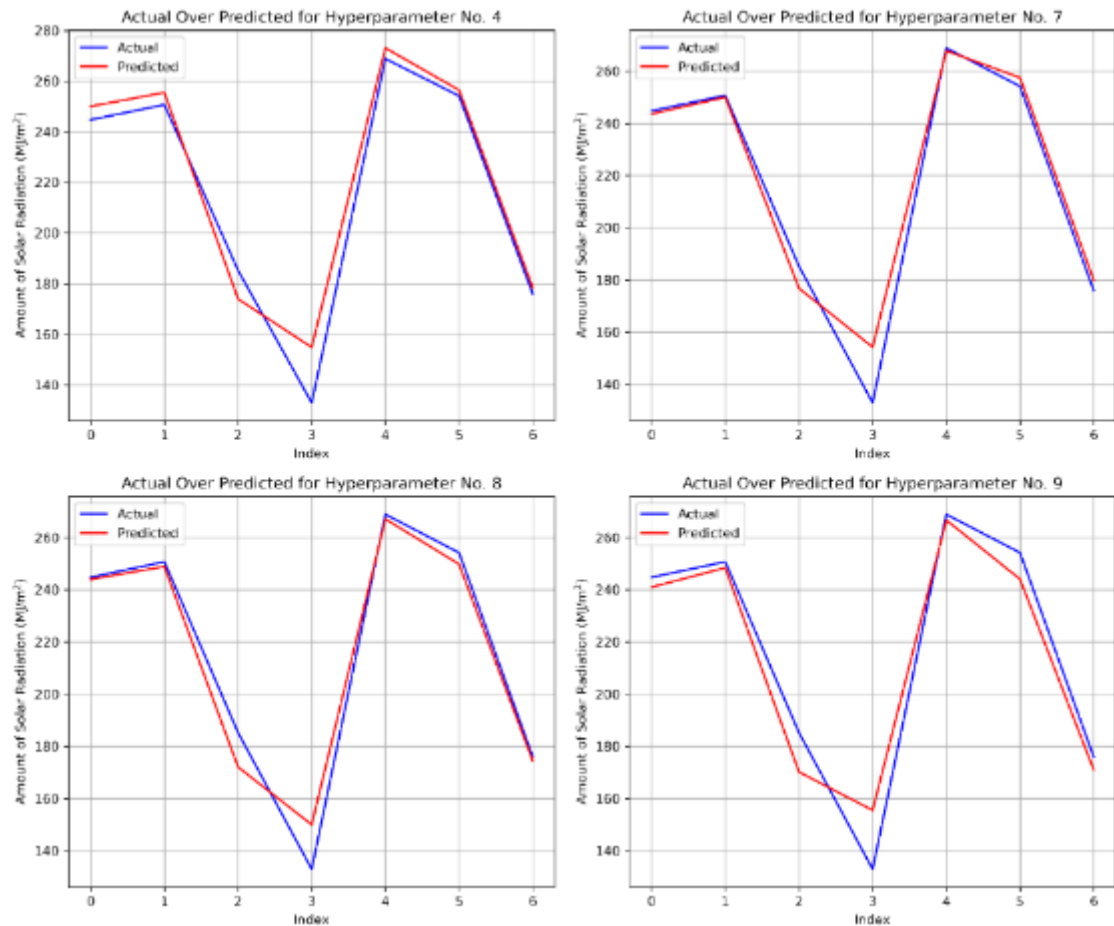


Gambar 4.26 Perbandingan nilai prediksi dan nilai sebenarnya dari empat kombinasi parameter terbaik pada algoritma *GRU* dengan jangka prediksi 90 hari

4.4.2 Hasil Pengujian Model Long-Short Term Memory

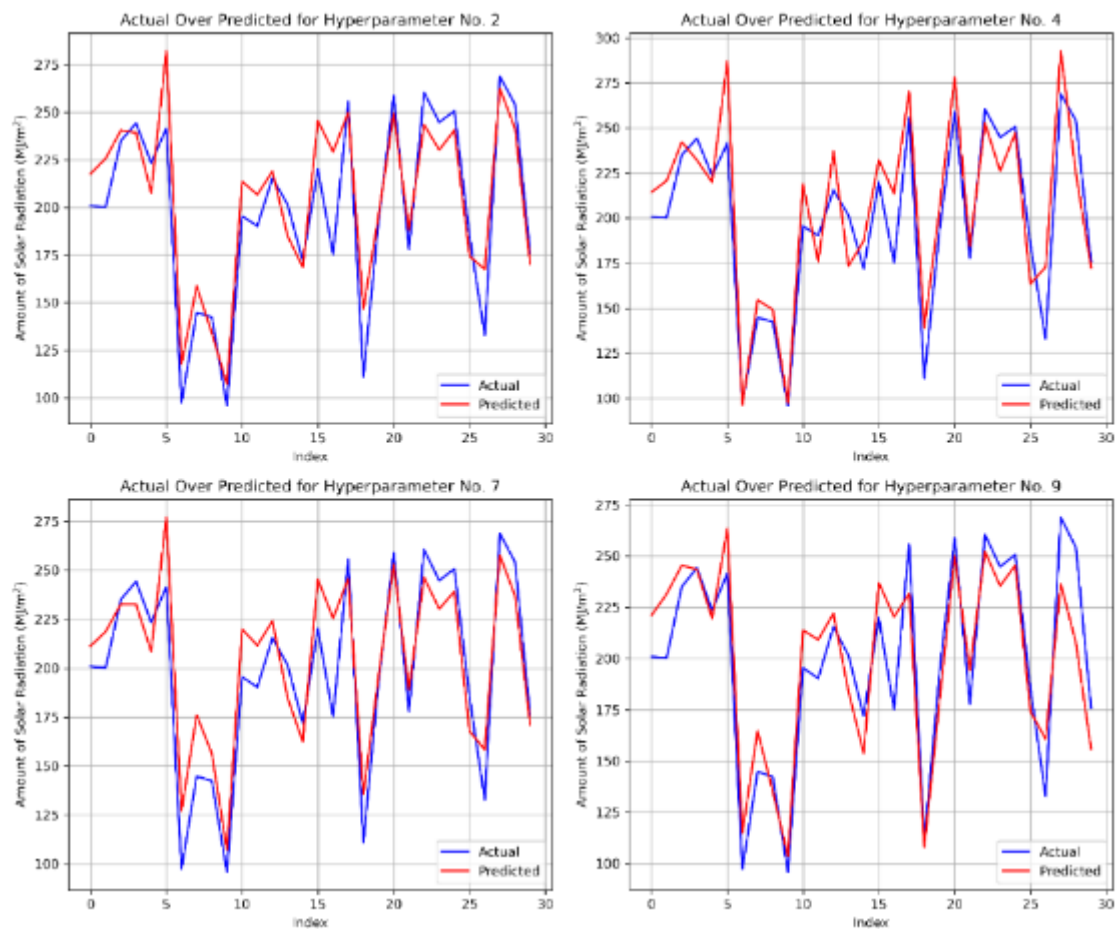
Perbandingan hasil pengujian pada *model Long-Short Term Memory (LSTM)* untuk jangka prediksi 7 hari dapat dilihat pada **Tabel 4.5**. Untuk kombinasi parameter terbaik berdasarkan nilai metrik evaluasi dimiliki oleh model dengan *hyperparameter* no. 8. Terlihat pada keempat kombinasi parameter terbaik pada **Gambar 4.27**, model *LSTM* dapat memprediksi data dengan cukup baik dengan nilai error yang minim. Seperti halnya

pada algoritma *GRU*, pada *index* ke 3 terdapat *error* yang cukup jauh antara nilai prediksi *model* dengan nilai sebenarnya pada *dataset* jika dibandingkan dengan hasil prediksi pada *index* lainnya.



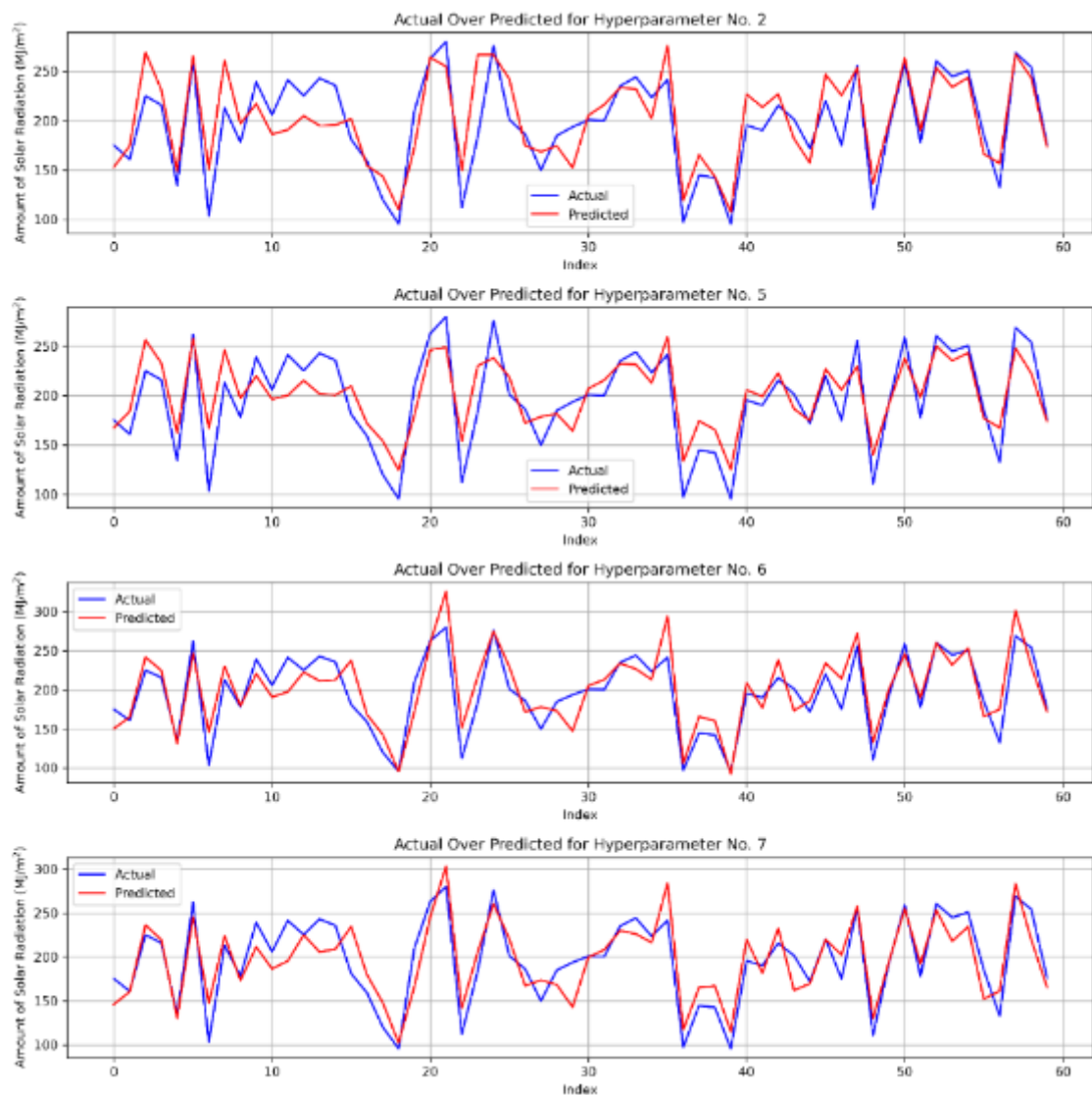
Gambar 4.27 Perbandingan nilai prediksi dan nilai sebenarnya dari empat kombinasi parameter terbaik pada algoritma *LSTM* dengan jangka prediksi 7 hari

Pada perbandingan hasil pengujian *model LSTM* untuk jangka prediksi 30 hari seperti yang dapat dilihat pada **Tabel 4.6**, nilai metrik evaluasi *model LSTM* masih mengungguli *GRU*. Tetapi untuk jumlah kesalahan prediksi naik dan turun pada tiap *index* prediksi, pada algoritma *LSTM* ditemukan dua kesalahan pada *index* ke 2 dan 4 dimana pada algoritma *GRU* hanya ditemukan satu kesalahan pada jangka waktu prediksi yang sama. Untuk perbandingan data prediksi pada empat hasil terbaik akan dijabarkan pada **Gambar 4.28**.



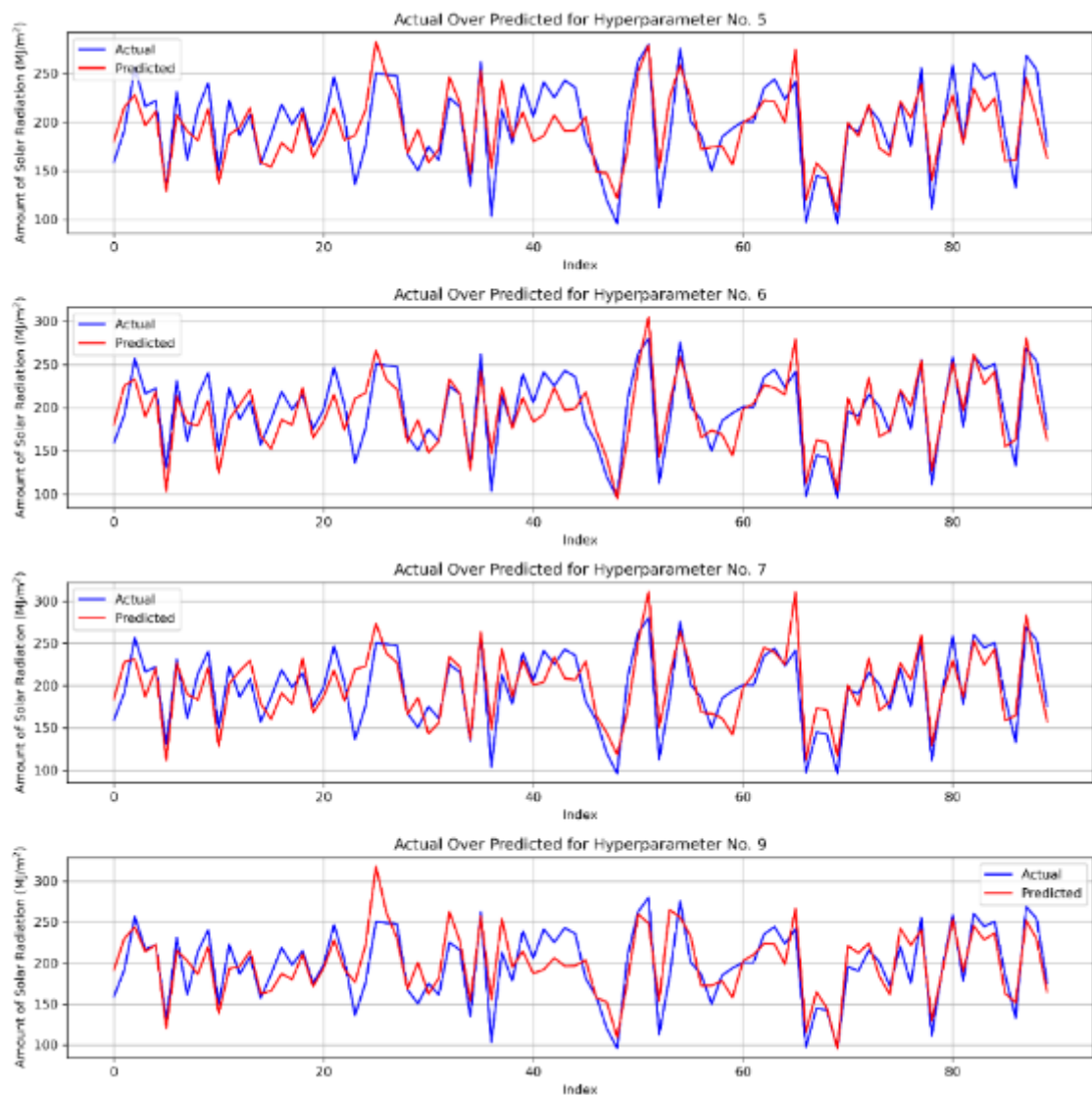
Gambar 4.28 Perbandingan nilai prediksi dan nilai sebenarnya dari empat kombinasi parameter terbaik pada algoritma *LSTM* dengan jangka prediksi 30 hari

Pada perbandingan hasil pengujian pada *model LSTM* untuk jangka prediksi 60 hari yang dapat dilihat pada **Tabel 4.7**, nilai metrik evaluasi terbaik dimiliki oleh *model LSTM* dengan *hyperparameter* no. 7. Untuk perbandingan data prediksi pada empat hasil terbaik akan dijabarkan pada **Gambar 4.29**. Pada *hyperparameter* no. 7, *LSTM* kembali memiliki kesalahan prediksi naik dan turunnya nilai besaran radiasi matahari yang lebih banyak dengan jumlah kesalahan pada 13 titik yaitu *index* ke 1, 12, 13, 14, 15, 27, 28, 29, 31, 33, 38, 44 serta 56.



Gambar 4.29 Perbandingan nilai prediksi dan nilai sebenarnya dari empat kombinasi parameter terbaik pada algoritma *LSTM* dengan jangka prediksi 60 hari

Perbandingan hasil pengujian pada *model Long-Short Term Memory (LSTM)* untuk jangka prediksi 90 hari dapat dilihat pada **Tabel 4.8**. Kombinasi parameter dengan nilai terbaik berdasarkan nilai metrik evaluasi dimiliki oleh model *LSTM* dengan *hyperparameter* no. 6. Pada **Gambar 4.30** dapat dilihat perbandingan data prediksi *model* dengan data sebenarnya. Pada *hyperparameter* no. 6, ditemukan jumlah kesalahan prediksi naik turunnya besaran radiasi matahari sebanyak 18 kali yang terdapat pada *index* ke 8, 12, 15, 23, 29, 30, 31, 42, 43, 44, 45, 57, 58, 59, 61, 63, 74 serta 86.



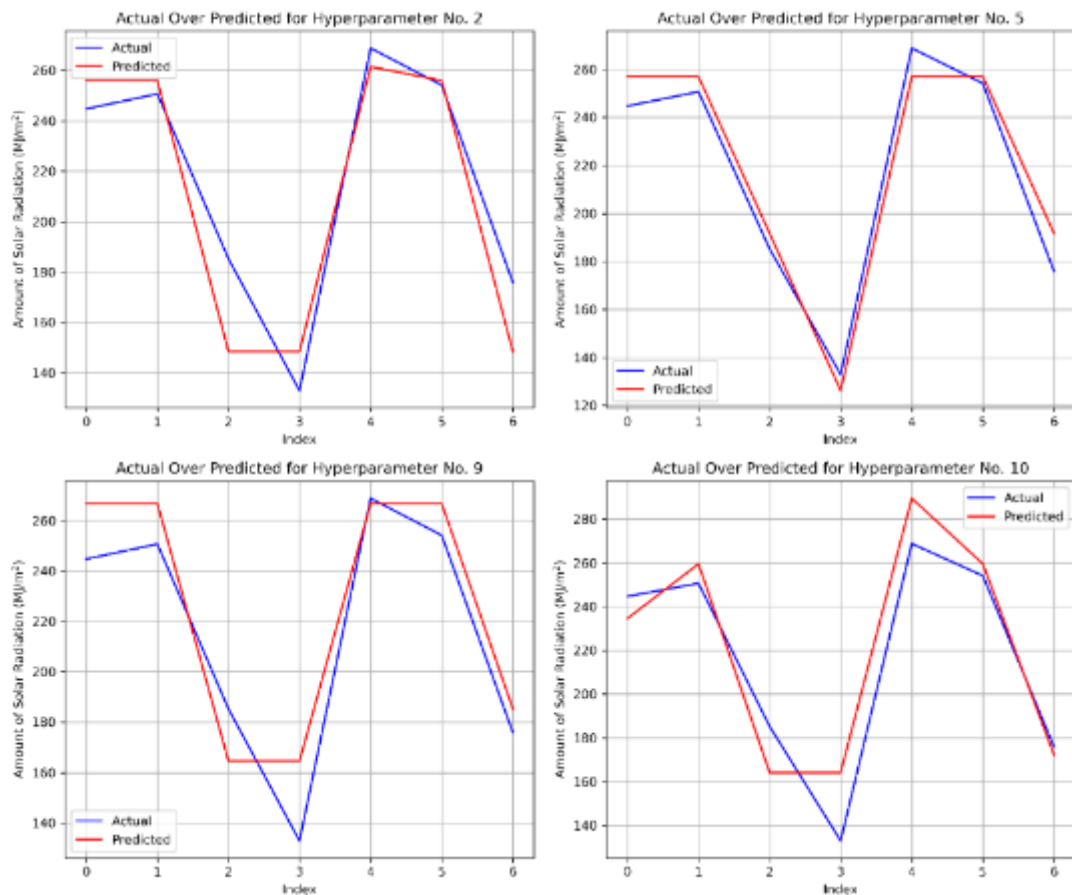
Gambar 4.30 Perbandingan nilai prediksi dan nilai sebenarnya dari empat kombinasi parameter terbaik pada algoritma *LSTM* dengan jangka prediksi 90 hari

Dari keseluruhan jangka waktu prediksi terlihat algoritma *LSTM* selalu memiliki kesalahan prediksi naik turunnya besaran radiasi matahari yang lebih banyak walaupun pada jangka waktu prediksi pendek seperti 7 hari, 30 hari, hingga 60 hari *LSTM* lebih unggul dari segi nilai keseluruhan metrik evaluasi.

4.4.3 Hasil Pengujian Model Deep Q-Network

Perbandingan hasil pengujian pada model *Deep Q-Network (DQN)* untuk jangka prediksi 7 hari dapat dilihat pada **Tabel 4.9**. Pada **Gambar 4.31** yang menjelaskan perbandingan data prediksi pada empat hasil terbaik, terlihat banyak garis lurus yang terjadi seperti 2 garis lurus pada *hyperparameter* no. 2 dan 5, 3 garis lurus pada *hyperparameter* no. 9

serta 1 garis lurus pada *hyperparameter* no. 10. Pada algoritma *DQN*, hal ini wajar terjadi dikarenakan penentuan *output* yang berdasarkan reward bukan berdasarkan regresi.



Gambar 4.31 Perbandingan nilai prediksi dan nilai sebenarnya dari empat kombinasi parameter terbaik pada algoritma *DQN* dengan jangka prediksi 7 hari

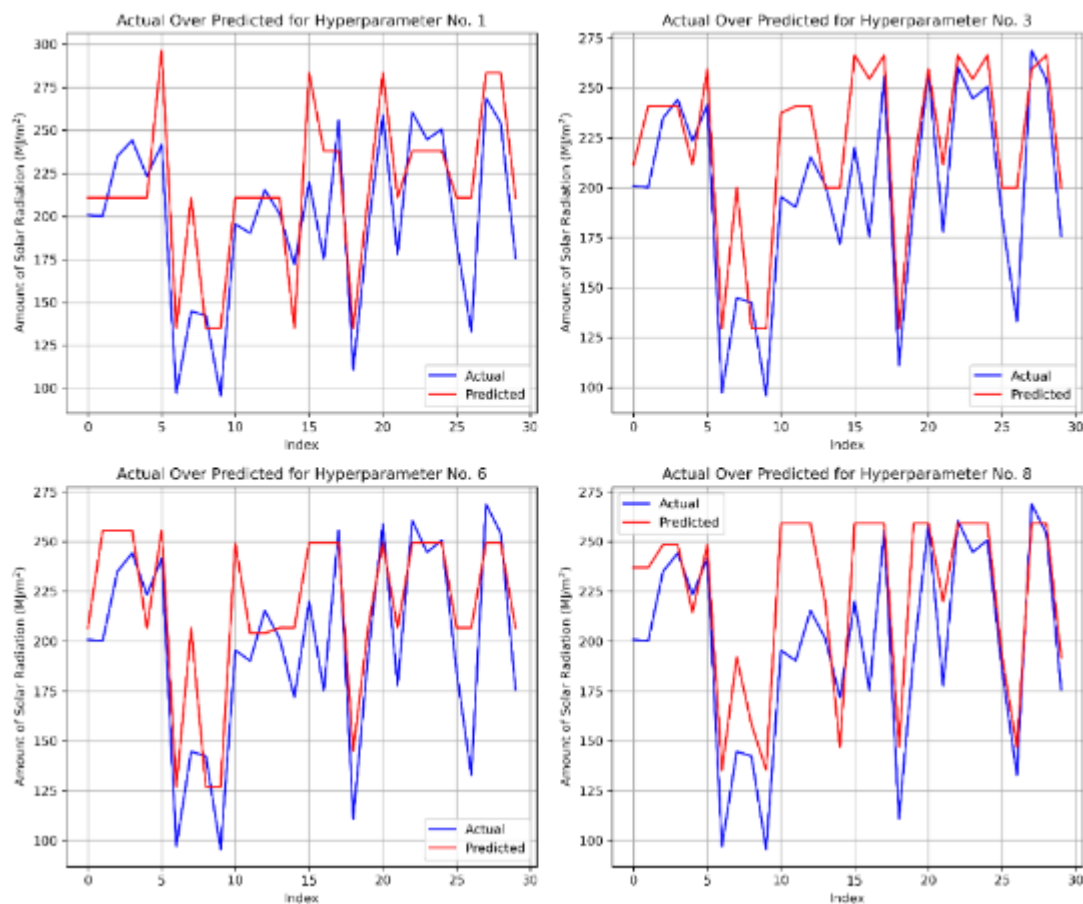
Banyaknya garis lurus yang terjadi saat prediksi model juga disebabkan karena struktur neural network *DQN* yang memiliki output yang banyak seperti pada **Gambar 4.10**, berbeda dengan algoritma *GRU* dan *LSTM* yang hanya memiliki 1 output.

Pada jangka waktu prediksi 7 hari, terutama pada model *DQN* di *hyperparameter* no. 5 terlihat *DQN* mampu memprediksi data dengan cukup baik bahkan mengungguli *GRU* berdasarkan nilai metrik evaluasi. Terlihat juga pada *hyperparameter* yang sama, algoritma *DQN* dapat memprediksi dengan sangat baik dengan tingkat *error* yang rendah pada masing-masing *index* prediksi.

Pada pengujian model *DQN* pada jangka waktu prediksi 30 hari sudah terlihat kemunduran yang sangat signifikan baik dari segi nilai metrik evaluasi maupun grafik perbandingan nilai prediksi model dengan data sebenarnya. Pada **Tabel 4.10** nilai metrik

evaluasi terbaik yang dapat diperoleh model yaitu nilai MAE , $RMSE$, MSE dan R^2 masing-masing 24.6339488, 31.47403041, 990.6145904 dan 0.584949538. Hasil ini sangat rendah jika dibandingkan dengan algoritma GRU maupun $LSTM$ dengan jangka waktu yang sama.

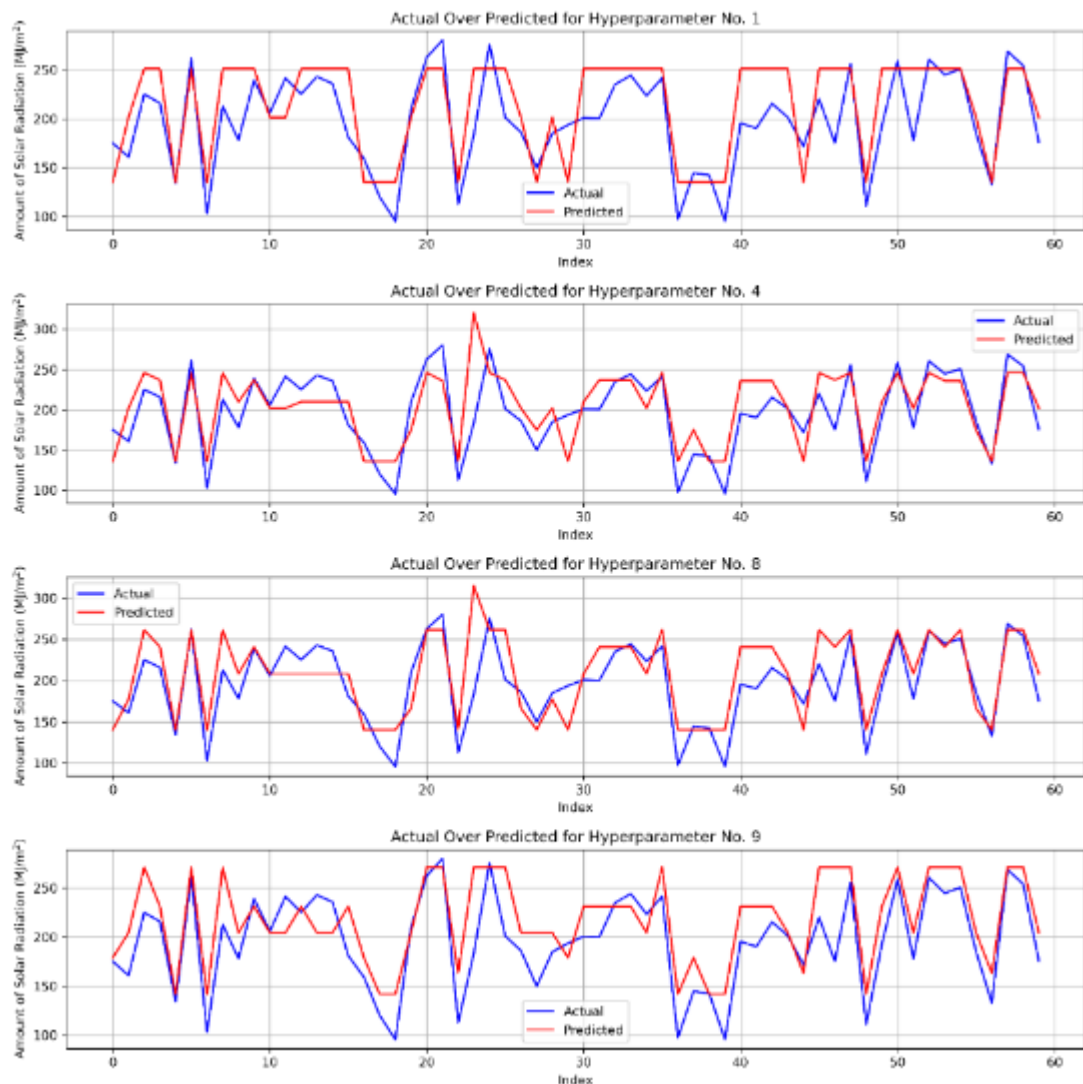
Dari segi prediksi naik turunnya nilai prediksi besaran radiasi matahari, terlihat pada **Gambar 4.32** *model* masih mampu memprediksi kenaikan serta penurunan besaran radiasi matahari tetapi memiliki nilai error yang sangat banyak dengan kesalahan pada 8 *index* prediksi pada model terbaik berdasarkan nilai metrik evaluasi pada *hyperparameter* no. 3.



Gambar 4.32 Perbandingan nilai prediksi dan nilai sebenarnya dari empat kombinasi parameter terbaik pada algoritma DQN dengan jangka prediksi 30 hari

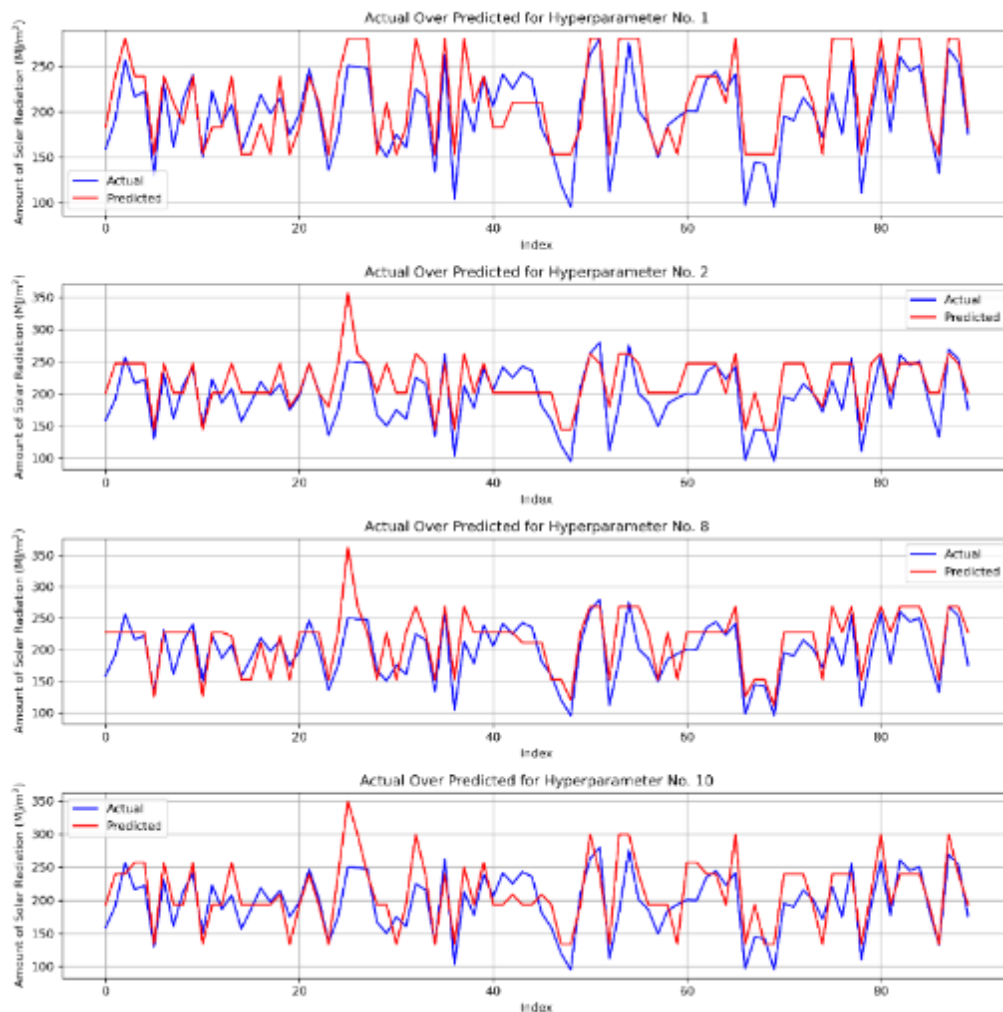
Tabel hasil pengujian *model DQN* untuk jangka waktu prediksi 60 hari dapat dilihat pada **Tabel 4.11**. Terlihat pada jangka waktu prediksi 60 hari algoritma DQN masih tertinggal dibandingkan algoritma GRU maupun $LSTM$. Pada **Gambar 4.33** juga dapat dilihat pada *hyperparameter* no. 4 yang merupakan *model* terbaik berdasarkan nilai

metrik evaluasi, *model* masih bisa melakukan prediksi dengan baik tetapi masih memiliki nilai *error* yang terlalu jauh. Dari segi prediksi naik turunnya besaran radiasi matahari, *model* juga tidak lebih baik dari algoritma *GRU* dan *LSTM* dengan jumlah kesalahan pada 19 *index* prediksi.



Gambar 4.33 Perbandingan nilai prediksi dan nilai sebenarnya dari empat kombinasi parameter terbaik pada algoritma *DQN* dengan jangka prediksi 60 hari

Pada pengujian *model DQN* dengan jangka waktu prediksi 90 hari, algoritma *DQN* masih belum mampu menyaingi baik algoritma *GRU* maupun *LSTM* dalam melakukan prediksi. Terlihat pada **Tabel 4.12** model terbaik berdasarkan metrik evaluasi masih memiliki nilai *MAE*, *RMSE*, *MSE* dan R^2 masing-masing 27.30856249, 34.39064066, 1182.716165 dan 0.423033524. Nilai ini masih belum mampu mendekati nilai metrik evaluasi pada algoritma *GRU* maupun *LSTM* dengan jangka waktu prediksi yang sama.



Gambar 4.34 Perbandingan nilai prediksi dan nilai sebenarnya dari empat kombinasi parameter terbaik pada algoritma *DQN* dengan jangka prediksi 90 hari

Pada **Gambar 4.34** juga terlihat algoritma masih dapat memprediksi dengan nilai yang cukup dekat dengan nilai sebenarnya, tetapi masih sangat banyak garis lurus yang terjadi yang mengakibatkan algoritma dinilai kurang mampu memprediksi naik turunnya besaran radiasi matahari. Dalam *model* dengan *hyperparameter* terbaik berdasarkan nilai metrik evaluasi yaitu *model* dengan *hyperparameter* no. 8, ditemukan kesalahan prediksi naik turunnya besaran radiasi matahari pada 38 *index* prediksi. Hal ini tergolong sangat banyak jika dibandingkan dengan algoritma *GRU* dan *LSTM* pada jangka waktu prediksi yang sama.

4.4.4 Perbandingan Model Gated Recurrent Unit, Long Short-Term Memory dan Deep Q-Network

Pada penelitian ini, *model* dari algoritma *GRU*, *LSTM* dan *DQN* yang telah dilakukan pelatihan hingga pengujian akan digunakan untuk dilakukan perbandingan antar ketiga algoritma. Untuk menunjukkan perbandingan antar algoritma, hasil prediksi *model* akan dibandingkan berdasarkan empat metrik evaluasi yang telah dipakai sebelumnya mulai dari fase pelatihan hingga pengujian sistem.

Untuk melakukan perbandingan, akan ditampilkan nilai metrik evaluasi pada masing-masing algoritma dengan jangka waktu 7 hari, 30 hari, 60 hari serta 90 hari. Masing-masing algoritma akan menggunakan nilai metrik evaluasi terbaik dari fase *hyperparameter tuning* sebelumnya.

Tabel 4.13 Perbandingan Nilai Metrik Evaluasi dari kombinasi terbaik pada masing-masing algoritma pada jangka waktu yang beragam

Algoritma / Jangka Waktu Prediksi	GRU				LSTM				DQN			
	MAE	RMSE	MSE	R ²	MAE	RMSE	MSE	R ²	MAE	RMSE	MSE	R ²
7 hari	7.6779	9.9804	99.61	0.9556	5.7906	8.3977	70.5215	0.9686	8.8773	9.8084	96.2052	0.9571
30 hari	17.0211	20.3052	412.3012	0.8272	16.889	19.7637	390.6052	0.8363	24.6339	31.474	990.6145	0.5849
60 hari	19.9635	24.0515	578.4784	0.7615	19.4786	23.6611	559.8512	0.7692	25.4287	32.4445	1052.6469	0.566
90 hari	19.7836	24.4493	597.7691	0.7083	20.4203	24.6283	606.5549	0.7041	27.30856	34.3906	1182.7161	0.423

Dari hasil pada **Tabel 4.13** dapat dilihat algoritma *GRU* dan *LSTM* memiliki nilai metrik evaluasi yang lebih baik dibandingkan algoritma *DQN*. Pada jangka waktu 7 dan 30 hari, algoritma *LSTM* mampu mengungguli *GRU* dengan cukup jauh. Tetapi, seiring meningkatnya jangka waktu prediksi, *GRU* terlihat mengungguli *LSTM* secara perlahan. Pada jangka waktu 60 hari, terlihat nilai metrik evaluasi pada algoritma *GRU* dan *LSTM* hanya memiliki jarak yang tidak signifikan meski masih diungguli oleh *LSTM*. Tetap pada jangka waktu 90 hari terlihat *GRU* lebih mampu melakukan prediksi dengan jangka yang lebih Panjang.

Pada jangka waktu prediksi 7 hari, *DQN* sempat sedikit mengungguli *GRU* berdasarkan nilai metrik evaluasi, tetapi pada jangka waktu yang lebih banyak *DQN* tampak belum mampu mengimbangi algoritma *GRU* dan *LSTM* dalam melakukan prediksi.

DQN masih dapat memprediksi dengan baik dengan nilai metrik evaluasi yang tidak terlalu tinggi walau masih belum bisa sebaik *GRU* maupun *LSTM*. Hal ini bisa disebabkan banyak hal, terutama *DQN* yang bukan merupakan algoritma regresi seperti layaknya

LSTM maupun *GRU*. *DQN* bekerja berdasarkan penilaian nilai *q-value* yang dihitung berdasarkan pengalaman atau interaksi *agent* dengan *environment*. Nilai *q-value* merepresentasikan seberapa baik atau buruknya suatu tindakan tertentu dalam keadaan tertentu berdasarkan nilai *reward* yang telah didefinisikan sebelumnya.

BAB 5

PENUTUP

5.1 Kesimpulan

Setelah seluruh fase pelatihan hingga pengujian model menggunakan algoritma *Gated Recurrent Unit (GRU)*, *Long Short-Term Memory (LSTM)* serta *Deep Q-Network (DQN)* menggunakan dataset dari penelitian *Development of software for estimating clear sky solar radiation in Indonesia* dapat diperoleh kesimpulan sebagai berikut.

1. Untuk jangka waktu prediksi singkat, algoritma *LSTM* unggul dibandingkan algoritma *GRU* dan *DQN* berdasarkan keempat nilai metrik evaluasi yaitu *Mean Squared Error (MSE)*, *Root Mean Squared Error (RMSE)*, *Mean Absolute Error (MAE)*, dan R^2 (*Coefficient of Determination*).
2. Menggunakan metrik evaluasi yang sama, algoritma *GRU* dapat memprediksi lebih baik dalam jangka waktu prediksi yang lebih panjang seperti jangka waktu prediksi 90 hari.
3. Algoritma *DQN* belum mampu menyaingi keakuratan prediksi *LSTM* dan *GRU* berdasarkan keempat metrik evaluasi. Terlihat ketimpangan yang cukup jauh baik dari nilai metrik evaluasi maupun grafik prediksi yang dibandingkan dengan data sebenarnya.

5.2 Saran

Terdapat juga beberapa saran penelitian sebagai berikut.

1. Data yang digunakan dalam penelitian memiliki data yang banyak tetapi terlalu banyak data yang tidak tercatat sehingga banyak parameter yang tidak dapat digunakan.
2. Jangka waktu prediksi pada penelitian ini terbatas pada jangka waktu 90 hari. Kedepannya dengan dataset yang lebih besar diharapkan jangka waktu prediksi dapat ditingkatkan untuk menunjukkan kemampuan masing-masing algoritma dengan lebih baik lagi.
3. Dalam penelitian berikutnya, parameter dataset yang digunakan bisa lebih beragam lagi agar prediksi dapat lebih kaya akan informasi dan lebih objektif dalam memperkirakan berbagai faktor meningkatnya besaran radiasi matahari.

4. Algoritma *DQN* yang digunakan pada penelitian ini hanya memiliki satu nilai *reward*. Kedepannya nilai *reward* dapat diganti seperti nilai metrik evaluasi ataupun kalkulasi matematika lainnya

DAFTAR PUSTAKA

- Abolhosseini, S., Heshmati, A., & Altmann, J. (2021). A Review of Renewable Energy Supply and Energy Efficiency Technologies. *SSRN Electronic Journal*, 8145. <https://doi.org/10.2139/ssrn.2432429>
- Ağbulut, Ü., Gürel, A. E., & Biçen, Y. (2021). Prediction of daily global solar radiation using different machine learning algorithms: Evaluation and comparison. *Renewable and Sustainable Energy Reviews*, 135(July 2020). <https://doi.org/10.1016/j.rser.2020.110114>
- Ambarita, H. (2017). Development of software for estimating clear sky solar radiation in Indonesia. *Journal of Physics: Conference Series*, 755(1). <https://doi.org/10.1088/1742-6596/755/1/011001>
- Arulkumaran, K., Deisenroth, M. P., Brundage, M., & Bharath, A. A. (2017). *A Brief Survey of Deep Reinforcement Learning*. <https://doi.org/10.1109/MSP.2017.2743240>
- Barto, R. S. S. and A. G. (2014). Reinforcement Learning: An Introduction, Second edition, in progress. *Decision Theory Models for Applications in Artificial Intelligence: Concepts and Solutions*, 63–80. <https://doi.org/10.4018/978-1-60960-165-2.ch004>
- Behera, M. K., Majumder, I., & Nayak, N. (2018). Solar photovoltaic power forecasting using optimized modified extreme learning machine technique. *Engineering Science and Technology, an International Journal*, 21(3), 428–438. <https://doi.org/10.1016/j.jestch.2018.04.013>
- Bendali, W., Saber, I., Bourachdi, B., Amri, O., Boussetta, M., & Mourad, Y. (2022). Multi Time Horizon Ahead Solar Irradiation Prediction Using GRU, PCA, and GRID SEARCH Based on Multivariate Datasets. *Journal Europeen Des Systemes Automatises*, 55(1), 11–23. <https://doi.org/10.18280/jesa.550102>
- Bhowmik, C., Bhowmik, S., Ray, A., & Pandey, K. M. (2017). Optimal green energy planning for sustainable development: A review. *Renewable and Sustainable Energy Reviews*, 71(December), 796–813. <https://doi.org/10.1016/j.rser.2016.12.105>
- Borovik, M. R., & Albers, J. D. (2018). Participation in the Illinois solar renewable energy

- market. *Electricity Journal*, 31(2), 33–39. <https://doi.org/10.1016/j.tej.2018.02.008>
- Burhan Uddin Shahin, M., Sarkar, A., Sabrina, T., & Roy, S. (2020). Forecasting Solar Irradiance Using Machine Learning. *2020 2nd International Conference on Sustainable Technologies for Industry 4.0, STI 2020*, 0, 19–20. <https://doi.org/10.1109/STI50764.2020.9350400>
- Fan, J., Wang, X., Wu, L., Zhang, F., Bai, H., Lu, X., & Xiang, Y. (2018). New combined models for estimating daily global solar radiation based on sunshine duration in humid regions: A case study in South China. *Energy Conversion and Management*, 156(November 2017), 618–625. <https://doi.org/10.1016/j.enconman.2017.11.085>
- IEA. (2022). *Renewables 2022, IEA, Paris*. 158. <https://www.iea.org/reports/renewables-2022>
- Jahani, B., & Mohammadi, B. (2019). A comparison between the application of empirical and ANN methods for estimation of daily global solar radiation in Iran. *Theoretical and Applied Climatology*, 137(1–2), 1257–1269. <https://doi.org/10.1007/s00704-018-2666-3>
- Jawaid, F., & Nazirjunejo, K. (2017). Predicting daily mean solar power using machine learning regression techniques. *2016 6th International Conference on Innovative Computing Technology, INTECH 2016*, 355–360. <https://doi.org/10.1109/INTECH.2016.7845051>
- Long, H., Zhang, Z., & Su, Y. (2014). Analysis of daily solar power prediction with data-driven approaches. *Applied Energy*, 126, 29–37. <https://doi.org/10.1016/j.apenergy.2014.03.084>
- Mousavi, S. S., Schukat, M., & Howley, E. (2018). Deep Reinforcement Learning: An Overview. *Lecture Notes in Networks and Systems*, 16, 426–440. https://doi.org/10.1007/978-3-319-56991-8_32
- Neha, & Rambeer, J. (2021). Renewable Energy Sources: A Review. *Journal of Physics: Conference Series*, 1979(1), 0–6. <https://doi.org/10.1088/1742-6596/1979/1/012023>
- Nematchoua, M. K., Orosa, J. A., & Afaifia, M. (2022). Prediction of daily global solar radiation and air temperature using six machine learning algorithms; a case of 27 European countries. *Ecological Informatics*, 69(March). <https://doi.org/10.1016/j.ecoinf.2022.101643>
- Ozdemir, S., & Ozkan Yildirim, S. (2023). Prediction of Water Level in Lakes by RNN-Based Deep Learning Algorithms to Preserve Sustainability in Changing Climate

- and Relationship to Microcystin. *Sustainability*, 15(22), 16008. <https://doi.org/10.3390/su152216008>
- Pandey, C. K., & Katiyar, A. K. (2013). Solar Radiation: Models and Measurement Techniques. *Journal of Energy*, 2013, 1–8. <https://doi.org/10.1155/2013/305207>
- Seydioğulları, H. S. (2013). Sürdürülebilir Kalkınma için Yenilenebilir Enerji Renewable Energy for Sustainable Development. *Planlama*, 23(1), 19–25. <https://doi.org/10.5505/planlama.2013.14633>
- Shen, G., Tan, Q., Zhang, H., Zeng, P., & Xu, J. (2018). Deep learning with gated recurrent unit networks for financial sequence predictions. *Procedia Computer Science*, 131, 895–903. <https://doi.org/10.1016/j.procs.2018.04.298>
- Solano, E. S., Dehghanian, P., & Affonso, C. M. (2022). Solar Radiation Forecasting Using Machine Learning and Ensemble Feature Selection. *Energies*, 15(19). <https://doi.org/10.3390/en15197049>
- Tingting Zhu 1, 2,*, Y. G. 1, & Zhenye Li 1 and Cong Wang 1. (2021). *Solar Radiation Prediction Based on Convolution Neural Network and Long Short-Term Memory*.
- Usman, M., Khalid, K., & Mehdi, M. A. (2021). What determines environmental deficit in Asia? Embossing the role of renewable and non-renewable energy utilization. *Renewable Energy*, 168, 1165–1176. <https://doi.org/10.1016/j.renene.2021.01.012>
- Wen, X., & Li, W. (2023). Time Series Prediction Based on LSTM-Attention-LSTM Model. *IEEE Access*, 11(April), 48322–48331. <https://doi.org/10.1109/ACCESS.2023.3276628>
- Zarzycki, K., & Ławryńczuk, M. (2022). Advanced predictive control for GRU and LSTM networks. *Information Sciences*, 616, 229–254. <https://doi.org/10.1016/j.ins.2022.10.078>
- Zhang, J., & Lei, Y. (2022). Deep Reinforcement Learning for Stock Prediction. *Scientific Programming*, 2022(M1). <https://doi.org/10.1155/2022/5812546>