

Implementasi Deep Learning Menggunakan Arsitektur Long Short Term Memory Untuk Prediksi Curah Hujan Kota Malang

Muhammad Rizki^{*1}, Setio Basuki², Yufis Azhar³

^{1,2,3}Teknik Informatika/Universitas Muhammadiyah Malang

mrizkijarman@gmail.com^{*1}, setio_basuki@umm.ac.id², yufis.az@gmail.com³

Abstrak

Tidak selamanya cuaca di Indonesia berjalan dengan normal atau sesuai dengan musimnya, cuaca sering berubah secara tiba-tiba setiap saat karena ada faktor-faktor yang mempengaruhi penurunan dan peningkatan curah hujan. perkiraan cuaca sangatlah dibutuhkan dan sangat bermanfaat olah berbagai pihak karena bisa menjadi acuan bagi berbagai kalangan untuk menjalani kegiatan mereka sehari-hari. Penelitian dilakukan menggunakan metode Deep Learning karena dari beberapa penelitian sebelumnya yang menggunakan Deep Learning dalam kasus yang berbeda mampu menghasilkan akurasi diatas 85%. Deep learning adalah jaringan yang terdiri dari beberapa layer. Layer-layer tersebut berasal dari kumpulan node-node. Arsitektur yang digunakan yaitu Long ShortTerm Memory (LSTM) karena pada penelitian-penelitian sebelumnya menggunakan LSTM dalam kasus yang berbeda mendapat hasil yang baik yaitu RME yang dihasilkan kecil. LSTM memiliki struktur seperti rantai dan struktur pada tiap sel terdapat 3 gate yaitu forget gate, input gate, dan output gate. Oleh karena itu, perhitungan yang dilakukan lebih kompleks ditambah lagi dengan Deep Learning diharapkan mendapat hasil yang lebih akurat. Data yang digunakan yaitu data curah hujan kota Malang yang berasal dari BMKG.

Kata Kunci: Prediksi, Curah Hujan, Deep Learning, Long Shot Term Memory

Abstract

The weather in Indonesia does not always run normally or in accordance with the season, the weather often changes suddenly at any time because there are factors that affect the decrease and increase in rainfall. weather forecasts are needed and very useful if the various parties because it can be a reference for various circles to undergo their daily activities. The study was conducted using Deep Learning method because of some previous research using Deep Learning in different cases able to produce accuracy above 85%. Deep learning is a network consisting of several layers. The layers are derived from a collection of nodes. The architecture used is Long Short Term Memory (LSTM) because in previous studies using LSTM in different case got good result that is small generated RME. LSTM has a structure like chains and structures in each cell there are 3 gates of forget gate, input gate, and output gate. Therefore, the calculations performed more complex plus the Deep Learning is expected to get more accurate results. The data used is the rainfall data of Malang city that comes from BMKG.

Keywords: Prediction, Rainfall, Deep Learning, Long Shot Term Memory

1. Pendahuluan

Indonesia merupakan negara dengan iklim tropis karena berada di terletak di garis khatulistiwa. Seperti kita ketahui, iklim tropis mempunyai 2 musim yaitu musim kemarau dan musim hujan. Tiap daerah memiliki intensitas iklim yang berbeda-beda. Contohnya musim hujan, tiap daerah di Indonesia pasti memiliki intensitas hujan yang berbeda-beda tergantung beberapa faktor seperti garis lintang, ketinggian tempat, jarak dari sumber air, arah angin, suhu tanah, dan luas daratan. Tidak selamanya cuaca di Indonesia berjalan dengan normal atau sesuai dengan musimnya, cuaca sering berubah secara tiba-tiba setiap saat karena ada faktor-faktor yang mempengaruhi penurunan dan peningkatan curah hujan. Seperti di daerah Malang yang intensitas curah hujannya bisa dibilang cukup tinggi karena dalam sepekan pasti terjadi hujan dengan intensitas yang berbeda-beda tiap harinya. Oleh sebab itu, perkiraan cuaca sangatlah dibutuhkan dan sangat bermanfaat olah berbagai pihak karena bisa menjadi acuan bagi berbagai kalangan untuk menjalani kegiatan mereka sehari-hari.

Pemanfaatan model prakiraan cuaca dengan teknik modern baru memiliki ketepatan sekitar 70%, sedangkan untuk prakiraan iklim masih dibawah 50% (Adiningsih, 2000). Adapun Badan Meteorologi dan Geofisika menggunakan metode probabilitas, metode deret harmonis, metode analogi untuk meramalkan cuaca dan iklim di Indonesia (Gunawan, 2001). Pada penelitian sebelumnya telah dilakukan peramalan cuaca dengan menggunakan metode fuzzy clustering (Arifin, 2007) yang dapat menghasilkan ketepatan sebesar 69%. Selanjutnya prediksi cuaca maritim menggunakan jaringan saraf tiruan (Kresnawan, 2008) mampu menghasilkan ketepatan tingkat akurasi sebesar 60,7% untuk prediksi arus laut, 72,4% untuk prediksi ketinggian gelombang laut, dan 26,122% untuk prediksi curah hujan [1]. Nilai RMSE untuk proses prediksi lebih tinggi daripada pada proses peramalan. Namun akurasi memiliki nilai yang hampir sama yaitu rata-rata diatas 75% [1]. Pada Penerapan Algoritma Support Vector Machine (SVM's) Untuk Prediksi Curah Hujan Per Jam yang memiliki nilai eror prediksi terbaik yaitu menghasilkan nilai RMSE 5.40967105251817 [2] Hasil perhitungan menggunakan metode ANFIS untuk prediksi cuaca dengan pola pembelajaran sebanyak 365 data diperoleh tingkat akurasi sebesar 75.07% dengan nilai iterasi 16. Dan pembelajaran untuk 50 data memperoleh akurasi yang cukup tinggi yaitu 86%, kluster yang dihasilkan cukup optimal [3].

Penelitian ini dilakukan menggunakan metode Deep Learning karena dari beberapa penelitian yang telah dilakukan menggunakan Deep Learning [4], [5], [6], [7] dalam kasus – kasus yang berbeda mampu menghasilkan akurasi diatas 85%. Adapun penelitian-penelitian tersebut seperti Analisis Perbandingan Teknik Ekstraksi Informasi Web Menggunakan Deep Learning Vs. Naive Bayes Vs. Back Propagation Neural Networks Pada Ekstraksi Dokumen Web yang dilakukan oleh J. Sharmila dan A. Subramani. Hasilnya metode Deep Learning mempunyai hasil yang paling baik dari presisi, recall maupun f-measure. Rata-rata nilai presisi, recall dan f-measure menggunakan metode Deep Learning yaitu 94%, 74% dan 73% [4].

Deep Residual Learning for Image Recognition (ARXIV 2015) yang dilakukan oleh Kaiming He, Xiangyu Zhang, Shaoqing Ren dan Jian Sun mempunyai hasil akurasi 93.57% dan error sebesar 6.43% [7].

LSTM network: a deep learning approach for short-term traffic forecast yang dilakukan oleh Zheng Zhao, Weihai Chen, Xingming Wu, Peter C. Y. Chen², Jingmeng Liu yang memperkirakan lalu lintas jalur terpendek dengan perbandingan kinerja yang dilakukan menggunakan model LSTM, SAE, RBF, SVM dan ARIMA. hasilnya model LSTM lebih baik karena RMEnya lebih kecil daripada model lain yaitu dalam 15 menit RMEnya rata-rata 6,22, dalam waktu 30 menit RMEnya rata-rata 9,37, dalam waktu 45 menit RMEnya rata-rata 13,11, dan dalam waktu 60 menit RMEnya rata-rata 17,02[8],

Earthquake Prediction based on Spatio-Temporal Data Mining: An LSTM Network Approach yang dilakukan oleh Qianlong Wang, Yifan Guo, Lixing Yu, and Pan Li. Mereka melakukan 3 kali percobaan yaitu pertama, perbandingan antara tanpa dan dengan dekomposisi dengan 3x3 sub-wilayah dan slot waktu satu bulan rata-rata keakuratannya 85,12% Kedua, hasil perbandingan antara tanpa dan dengan dekomposisi dengan 3x3 sub-wilayah dan dua minggu slot waktu rata-rata keakuratannya menjadi 86%. Ketiga, hasil perbandingan antara tanpa dan dengan dekomposisi dengan 5x5 sub-wilayah dan slot waktu satu bulan rata-rata keakuratannya 87,59 [7].

Deep Learning menunjukkan hasil yang baik. Oleh sebab itu, saya menggunakan metode ini untuk mengetahui apakah prediksi curah hujan yang dihasilkan menggunakan metode Deep Learning lebih akurat dari algoritma-algoritma yang ada dan telah digunakan pada penelitian-penelitian sebelumnya untuk prediksi. Penelitian ini dilakukan menggunakan data curah hujan kota Malang dari BMKG dengan metode Deep Learning dengan arsitektur Long Short Term Memory (LSTM).

2. Metode Penelitian

2.1 Pengumpulan data

Data curah hujan yang digunakan untuk penelitian ini diperoleh dari situs BMKG dari tahun 1989-2017. Atribut yang digunakan yaitu curah hujan. Data tersebut dijadikan acuan sebagai informasi dari masa lalu yang mana akan digunakan pada proses pelatihan data (data training) menggunakan Deep Learning dengan arsitektur Long Short Term Memory (LSTM).

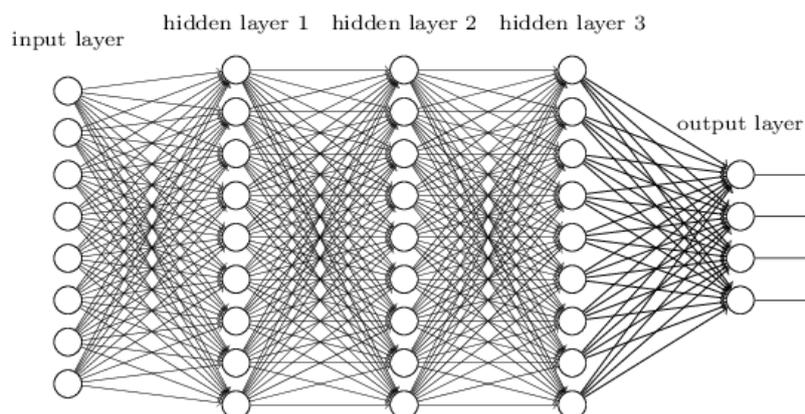
2.2 Preprocessing Data

Preprocessing data merupakan sebuah proses dimana data yang akan digunakan sebagai pelatihan disiapkan terlebih dahulu. Pada penelitian ini preprocessing data disiapkan dengan melalui sebuah proses untuk menangani data yang hilang atau kosong dengan berbagai cara seperti mencari rata-rata suatu atribut untuk kelas yang sama. Setelah itu data di normalisasi menggunakan MinMaxScaler dengan range (0, 1). Metode Min-Max merupakan metode normalisasi dengan melakukan transformasi linier terhadap data asli. Kita dapat dengan mudah menormalkan dataset menggunakan kelas preprocessing MinMaxScaler dari perpustakaan scikit-learn, dengan Persamaan 1 berikut.

$$x' = \frac{x - \min x}{\max x - \min x} \quad (1)$$

2.3 Deep Learning

Deep Learning adalah jaringan yang terdiri dari beberapa layer. Layer-layer tersebut merupakan kumpulan dari node-node. Sebuah node hanya sebagai tempat perhitungan terjadi. Sebuah node input digabung dengan weight (bobot), setelah input dan weight ini dijumlahkan dan jumlahnya melewati tahapan yang disebut fungsi aktivasi node, untuk menentukan seberapa jauh sinyal yang berlangsung lebih lanjut melalui jaringan, hal tersebut mempengaruhi hasil akhir. Perbedaan Deep Learning dengan neural network yaitu lebih banyaknya hidden layer pada Deep Learning. Jika lebih dari tiga layer (termasuk input dan output) maka memenuhi syarat sebagai "deep" learning. Jadi Deep Learning bisa didefinisikan secara teknis yaitu machine learning yang mempunyai lebih dari satu hidden layer. Ilustrasi Deep Learning dapat dilihat pada Gambar 2 terdapat 4 layer dan tiap layer mempunyai jumlah node yang berbeda-beda.



Gambar 2. Ilustrasi Deep Learning

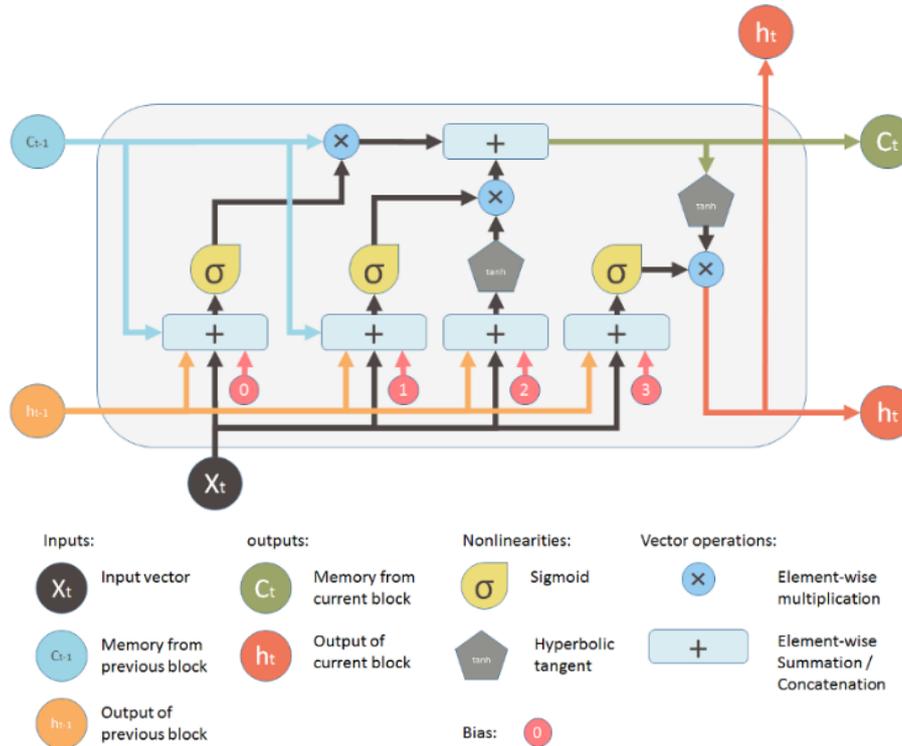
2.4 Long Short Term Memory (LSTM)

Saat menggunakan LSTM dalam letak deret waktu, satu parameter penting yang harus dipilih adalah panjang urutan yang merupakan data LSTM untuk dilihat kembali. Ini dapat dilihat sama seperti mengambil `window_size = 5` siklus untuk menghitung fitur rolling di Aplikasi Prediksi Curah hujan yang bergulir rata-rata dan standar deviasi. Dengan menggunakan LSTM dimana model mengekstrak fitur abstrak dari urutan nilai sensor di data daripada teknik secara manual. Harapannya adalah jika ada pola dalam nilai sensor ini di dalam data sebelum terjadi kegagalan, pola tersebut harus dikodekan oleh LSTM.

Salah satu keunggulan penting LSTM adalah kemampuan mengingat dari sekuens long term (ukuran data) yang sulit dicapai dengan teknik fitur tradisional. Sebagai contoh pada Gambar 3, komputasi rata-rata bergulir di atas ukuran data dan dapat menyebabkan hilangnya informasi karena mendeskripsikan nilai selama periode yang panjang, sebaliknya, dengan menggunakan semua nilai sebagai masukan dapat memberikan hasil yang lebih baik. Sementara rekayasa fitur dengan ukuran data yang besar, LSTM dapat menggunakan ukuran data yang lebih besar dan menggunakan semua informasi data sebagai masukan.

Kemudian membangun deep network. Layer pertama adalah layer LSTM dengan jumlah unit dan diikuti layer LSTM lain dengan jumlah unit. Dropout juga diterapkan setelah setiap layer

LSTM mengontrol overfitting. Layer akhir adalah layer keluaran dengan aktivasi single unit dan sigmoid karena ini adalah masalah klasifikasi biner.



Gambar 3. Ilustrasi LSTM

Rumus LSTM dibagi menjadi 4, seperti pada Persamaan 1, Persamaan 2, Persamaan 3, dan Persamaan 4.

- Forget Gate

$$f_t = \sigma(W_{fx}X_t + W_{fh}h_{t-1} + W_{fc}C_{t-1} + b_f) \tag{1}$$

- Input Gate

$$i_t = \sigma(W_{ix}X_t + W_{ih}h_{t-1} + W_{ic}C_{t-1} + b_i) \tag{2}$$

- Memory Update

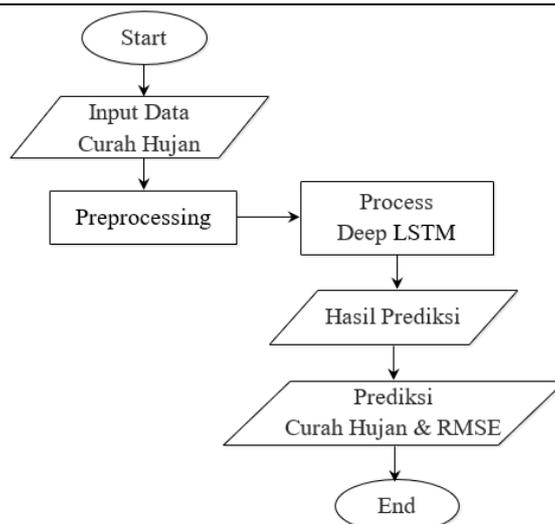
$$c_t = f_t \circ C_{t-1} + i_t \circ \phi(W_{cx}X_t + W_{ch}h_{t-1} + b_c) \tag{3}$$

- Output Gate

$$\begin{aligned} o_t &= \sigma(W_{ox}X_t + W_{oh}h_{t-1} + W_{oc}C_t + b_o) \\ h_t &= o_t \circ \phi(c_t) \end{aligned} \tag{4}$$

2.5 Flowchart Sistem

Flowchart sistem merupakan bagan yang menunjukkan alur kerja atau apa yang sedang dikerjakan di dalam sistem secara keseluruhan dan menjelaskan urutan dari prosedur-prosedur yang ada di dalam sistem. Dengan kata lain, flowchart pada Gambar 4 ini merupakan dekripsi secara grafik dari urutan prosedur-prosedur yang membentuk suatu sistem.



Gambar 4. Flowchart Sistem Secara Umum

3. Hasil Penelitian dan Pembahasan

Pengujian dilakukan dengan menghitung hasil dari prediksi periode pada waktu ke – i dengan mengukur nilai kesalahan prediksi antara hasil prediksi dengan data uji. Sebelum dilakukan pengujian diperlukan beberapa bahan dalam melakukan penelitian ini dari awal pembuatan program hingga melakukan pengujian.

3.1 Spesifikasi Kebutuhan Hardware dan Software

3.1.1 Kebutuhan Hardware

Untuk hardware menggunakan 1 buah komputer/laptop yang memiliki spesifikasi sebagai berikut: Memori atau RAM 8GB, Processor Intel®Core™ i7-7700HQ CPU @ 2.80 GHz dan Tipe Sistem 64 bit.

3.1.2 Kebutuhan Software

Sistem Operasi windows 10, bahasa pemrograman Python, Library Keras.

3.2 Pengujian Prediksi

Pengujian dilakukan menggunakan data dari BMKG Kota Malang untuk melihat prediksi curah hujan kota Malang dan untuk melihat nilai kesalahan dari peramalan digunakan rumus RMSE (Root Mean Squared Error) Persamaan 5 yang merupakan akar kuadrat dari MSE.

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (A_i - F_i)^2} \quad (5)$$

3.2.1. Hasil Pengujian Jumlah Neuron Hidden Layer

Uji prediksi wajib dilakukan untuk mengujur keberhasilan dalam melakukan suatu prediksi dengan mengukur kesalahan dari hasil prediksi. Dengan demikian telah diperoleh kesimpulan dalam melakukan proses uji prediksi lengkap beserta perbandingan nilai kesalahan RMSE.

Tabel 1. Kombinasi Parameter Neuron Hidden Layer

No.	Neuron Hidden	Epoch	RMSE Data Train	RMSE Data Test
1.	16	10	12.342	11.562
2.	32	10	12.271	11.488
3.	64	10	12.267	11.491
4.	128	10	12.275	11.503
5.	256	10	12.247	11.481

Berdasarkan Tabel 1, menunjukkan bahwa jumlah neuron hidden layer 256 menghasilkan hasil yang lebih optimal dibandingkan dengan jumlah neuron hidden layer lainnya. Neuron hidden layer mengolah nilai input dan menghubungkannya dengan neuron output, sehingga jumlah hidden neuron akan menentukan nilai output yang dihasilkan oleh LSTM. Tidak ada aturan pasti mengenai jumlah neuron hidden layer yang paling optimal dalam memprediksi time series. Sehingga untuk mendapatkan jumlah neuron hidden layer yang paling baik dapat dilakukan melalui percobaan.

3.2.2. Hasil Pengujian Jumlah Epoch

Tabel 2. Kombinasi Parameter Jumlah Epoch

No.	Neuron Hidden	Epoch	RMSE Data Train	RMSE Data Test
1.	256	10	12.247	11.481
2.	256	50	12.218	11.415
3.	256	100	12.108	11.314
4.	256	150	12.079	11.288
5.	256	200	12.090	11.300

Berdasarkan Tabel 2, menunjukkan bahwa epoch 150 menghasilkan hasil yang lebih optimal dibandingkan dengan epoch lainnya. Max epoch tidak boleh terlalu besar dan terlalu kecil untuk menemukan solusi optimum. Tidak ada aturan pasti mengenai besarnya max epoch yang paling optimal dalam memprediksi time series. Sehingga untuk mendapatkan jumlah epoch yang paling baik dapat dilakukan melalui percobaan.

3.2.3. Hasil Pengujian Parameter Komposisi Data

Tabel 3. Kombinasi Parameter Komposisi Data

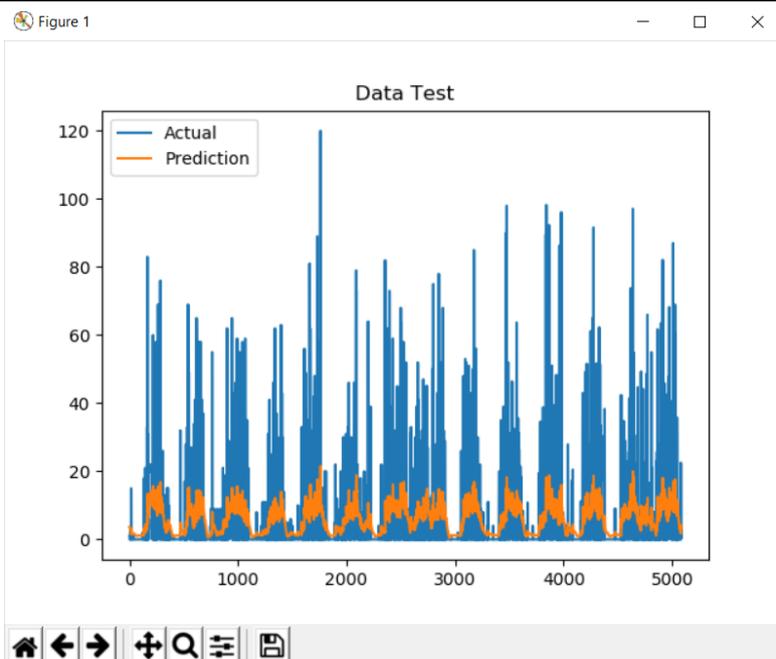
No.	Komposisi Data	Neuron Hidden	Epoch	RMSE Data Train	RMSE Data Test
1.	50% Data Train 50% Data Test	256	150	12.079	11.288
2.	60% Data Train 40% Data Test	256	150	11.786	11.510
3.	70% Data Train 30% Data Test	256	150	11.581	11.931
4.	80% Data Train 20% Data Test	256	150	11.530	12.210
5.	90% Data Train 10% Data Test	256	150	11.591	12.595

Berdasarkan Tabel 3, menunjukkan bahwa komposisi data yang optimal digunakan yaitu skema 1 dengan komposisi data train 50% dan data test 50%. Hal ini karena skema 1 memiliki tingkat error yang paling rendah pada data uji. Tidak ada aturan pasti mengenai komposisi data train dan data test yang paling optimal dalam memprediksi time series. Sehingga untuk mendapatkan komposisi data yang paling baik dapat dilakukan melalui percobaan.

Pada hasil pengujian Gambar 4, prediksi dengan arsitektur Long Short Term Memory (LSTM), maka diperoleh hasil sebagai berikut:

1. Jumlah neuron hidden layer dengan hasil paling optimal yaitu dengan 256 neuron hidden layer. Hal ini karena jumlah 256 neuron hidden layer memiliki tingkat error yang paling rendah yaitu pada data train sebesar 12.247 dan pada data test sebesar 11.481.
2. Jumlah epoch dengan hasil paling optimal yaitu 150 epoch. Hal ini karena jumlah 150 epoch memiliki tingkat error yang paling rendah yaitu pada data train sebesar 12.079 dan pada data test sebesar 11.288.

Komposisi Data Train dan Data Test dengan hasil yang paling optimal yaitu dengan komposisi data train 50% dan data test 50%. Hal ini karena komposisi data train 50% dan data test 50% memiliki tingkat error yang paling rendah yaitu pada data train sebesar 12.079 dan pada data test sebesar 11.288.



Gambar 4. Curva Perbandingan Hasil Prediksi Dan Data Aktual Pada Data Test

4. Kesimpulan

Setelah melakukan analisis, perancangan, implementasi dan hasil pengujian implementasi Deep Learning menggunakan arsitektur Long Short Term Memory (LSTM) untuk prediksi curah hujan kota Malang, menunjukkan bahwa hasil prediksi yang diperoleh tergolong cukup baik. Sehingga dapat disimpulkan bahwa pada penelitian tugas akhir Deep Learning dengan arsitektur Long Short Term Memory (LSTM) dapat bekerja cukup optimal. Berikut pembahasan terkait pada penelitian tugas akhir ini adalah sebagai berikut:

1. Aplikasi berhasil memproses prediksi curah hujan kota Malang dengan parameter curah hujan.
2. Jumlah neuron hidden layer dengan hasil paling optimal yaitu dengan 256 neuron hidden layer. Hal ini karena jumlah 256 neuron hidden layer memiliki tingkat error yang paling rendah yaitu pada data train sebesar 12.247 dan pada data test sebesar 11.481.
3. Jumlah epoch dengan hasil paling optimal yaitu 150 epoch. Hal ini karena jumlah 150 epoch memiliki tingkat error yang paling rendah yaitu pada data train sebesar 12.079 dan pada data test sebesar 11.288.
4. Komposisi Data Train dan Data Test dengan hasil yang paling optimal yaitu dengan komposisi data train 50% dan data test 50%. Hal ini karena komposisi data train 50% dan data test 50% memiliki tingkat error yang paling rendah yaitu pada data train sebesar 12.079 dan pada data test sebesar 11.288.

Berdasarkan hasil yang diperoleh, maka ada beberapa hal yang harus ditambahkan sebagai berikut:

1. Perlu dilakukan lagi pengkajian ulang terhadap penentuan jumlah parameter karena kemungkinan besar jumlah parameter berdampak pada hasil prediksi. Dengan demikian diperlukan eksperimen atau penelitian terkait lebih lanjut mengenai berapa jumlah dan apa saja parameter yang perlu ditambahkan untuk menghasilkan prediksi terbaik menggunakan Deep Learning dengan arsitektur LSTM pada prediksi curah hujan kota Malang.
2. Jumlah data yang digunakan harus lebih banyak agar hasilnya semakin baik.

Notasi

- f_t : Forget Gate
 i_t : Input Gate
 C_t : Memory Update
 O_t : Output Candidate
 h_t : Output Gate

X_t	: Nilai Input
h_{t-1}	: Nilai Output Blok sebelumnya
ct_{t-1}	: Memory Blok Sebelumnya
b	: Bias
W	: Weight
ϕ	: Tanh
σ	: Sigmoid

Referensi

- [1] Candra Dewi, et al. 2014. *Prediksi Cuaca Pada Data Time Series Menggunakan Adaptive Neuro Fuzzy Inference System (Anfis)*. Program Studi Informatika/Illmu Komputer, Universitas Brawijaya.
- [2] Reffi Avrilliani Nedyia Putri. 2016. *Penerapan Algoritma Support Vector Machine (Svm's) Untuk Prediksi Curah Hujan Per Jam Menggunakan Data Dari Radar*. Program Studi Teknik Informatika Universitas Muhammadiyah Malang.
- [3] Mas'ula. 2013. *Implementasi Adaptive Neuro Fuzzy Inference System Untuk Prediksi Cuaca (Studi Kasus Stasiun Klimatologi Karangploso Kab. Malang)*. Program Studi Teknik Informatika Universitas Muhammadiyah Malang.
- [4] Sharmila, J., & Subramani, A. 2016. *A Comparative Analysis Of Web Information Extraction Techniques Deep Learning Vs. Naive Bayes Vs. Back Propagation Neural Networks In Web Document Extraction*. Ictact Journal on Soft Computing, 6(2).
- [5] Kaiming He, et al. 2015. *Deep Residual Learning For Image Recognition* arXiv:1512.03385v1 [cs.CV].
- [6] Zheng Zhao, et al. 2017. *Lstm Network: A Deep Learning Approach For Short-Term Traffic Forecast*. The Institution of Engineering and Technology
- [7] Qianlong Wang, et al. 2016. *Earthquake Prediction Based On Spatio-Temporal Data Mining: An Lstm Network Approach*. DOI 10.1109/TETC.2017.2699169, IEEE