

Pembuatan Aplikasi Peramalan Penjualan Susu Sapi Perah Menggunakan Extreme Learning Machine

Lubby Gennady¹⁾ Dyah Erny Herwindiati²⁾ Janson Hendryli³⁾

¹⁾²⁾³⁾ Teknik Informatika, Universitas Tarumanagara

Jl. Letjen S. Parman No. 1. Jakarta Barat 11440, Indonesia

email: lubby.535180126@stu.untar.ac.id¹⁾, dyahh@fti.untar.ac.id²⁾, jansonh@fti.untar.ac.id³⁾

ABSTRAK

UMKM XYZ merupakan salah satu usaha yang bergerak dalam bidang susu sapi perah. Usaha tersebut menjual susu sapi segar dengan varian rasa. Dalam penjualan tersebut menyalurkan susu sapi segar sesuai dengan permintaan konsumen. Setiap periode, usaha sering mengalami perubahan permintaan susu sapi segar. Akibatnya, apabila permintaan konsumen tidak sesuai dengan ketersediaan dapat mengalami kerugian karena susu sapi tidak memiliki waktu simpan yang cukup lama. Oleh sebab itu, dilakukan rancangan aplikasi peramalan penjualan berbasis Android dengan metode *Extreme Learning Machine* (ELM) untuk menangani kasus tersebut. Tujuan dari aplikasi tersebut adalah melakukan peramalan permintaan susu agar sesuai dengan jumlah permintaan yang dibutuhkan. yang telah dilakukan dengan pengujian *Black Box Testing* dan hasil evaluasi kesalahan pada metode Extreme Learning Machine dengan nilai MAPE. Hasil pengujian pada Black Box Testing menunjukkan bahwa aplikasi telah layak untuk melakukan pencatatan transaksi. Hasil tersebut didapatkan nilai terbaik menggunakan aktivasi sigmoid biner dengan 4 fitur, jumlah *hidden neuron* sebanyak 10 serta pembagian data training dan testing yaitu 90%:10%. Hasil nilai kesalahan pada model original didapatkan nilai kesalahan MAPE 6.6558%, model coklat dengan nilai kesalahan 5.624%, model stroberi dengan nilai kesalahan 6.2874%.

Kata Kunci

Android, Extreme Learning Machine, Time Series, Susu sapi

1. Pendahuluan

Pada era serba teknologi saat ini, bisnis harus dapat beradaptasi dengan teknologi yang berkembang pesat. Bisnis terlihat baik-baik saja juga dituntut untuk selalu berinovasi dalam hal proses bisnis. Salah satu langkah untuk proses bisnis yang efektif adalah mengidentifikasi proses yang memerlukan sebuah

perubahan dan menganalisa proses yang dianggap tidak efisien. Oleh karena itu, perlu pengembangan proses bisnis supaya dapat berjalan dengan efektif sesuai dengan perkembangan teknologi saat ini.

UMKM XYZ merupakan usaha yang bergerak pada bidang susu sapi perah. Bisnis yang dilakukan adalah menyalurkan susu sapi segar yang sudah diolah menjadi siap minum kepada konsumen. Pada setiap periode, permintaan konsumen sering mengalami perubahan permintaan. Akibatnya, apabila tidak dapat memenuhi permintaan konsumen dapat mengalami kerugian ataupun kerusakan produk karena susu tidak mempunyai waktu simpan yang cukup lama.

Pada kasus tersebut diperlukan sebuah aplikasi yang dapat membantu melakukan peramalan permintaan susu. Proses rancangan aplikasi peramalan penjualan menggunakan metode *Extreme Learning Machine* yang berjalan pada perangkat *Smartphone* Android yang cukup banyak dimiliki oleh banyak pengguna. Peramalan permintaan dapat membantu UMKM XYZ dalam menentukan permintaan susu sesuai dengan kebutuhan.

2. Dasar Teori

2.1 Jaringan Saraf Tiruan

Jaringan Saraf Tiruan merupakan implementasi yang meniru cara kerja sistem saraf otak pada manusia secara komputasi dalam mengelola informasi. Jaringan Saraf Tiruan memiliki dua tahap yaitu proses training yang berguna untuk pelatihan jaringan pada setiap input neuron dan proses testing yang berguna untuk melakukan pengujian data yang belum pernah dilatih sebelumnya dengan menggunakan bobot-bobot yang telah dihasilkan pada tahap pelatihan [1].

2.2 Fungsi aktivasi

Fungsi aktivasi merupakan bagian dari dalam jaringan saraf tiruan yang melakukan perhitungan antara nilai input yang menjadi nilai output. Salah satu jenis Fungsi aktivasi yang digunakan yaitu fungsi

sigmoid biner. Fungsi sigmoid biner menghasilkan *output* dengan rentang 0 hingga 1. Berikut persamaan pada rumus fungsi aktivasi sigmoid biner [2].

$$g(x) = \frac{1}{1+\exp(-x)} \quad (1)$$

Keterangan:

$g(x)$ = fungsi g dengan nilai *input* x
 x = nilai *input* fungsi aktivasi

2.3 Normalisasi & Denormalisasi

Normalisasi adalah salah satu bentuk *pre-processing* yang biasanya menyetarakan nilai pada data memiliki selisih yang lebih jauh dengan data nilai lain. Salah satu cara untuk melakukan proses normalisasi data menggunakan perhitungan Min-Max Normalization. Nilai hasil Min-Max Normalization memiliki rentang nilai antara 0 dan 1. Berikut persamaan pada normalisasi [3].

$$x_n = \frac{(x_0 - x_{min})}{(x_{max} - x_{min})} \quad (2)$$

Keterangan:

x_n = hasil normalisasi
 x_0 = nilai sebelum normalisasi
 x_{min} = nilai minimum *dataset*
 x_{max} = nilai maksimum *dataset*

Pada denormalisasi dilakukan dengan mengembalikan nilai dari data normalisasi menjadi nilai sebenarnya. Berikut persamaan pada denormalisasi.

$$x_0 = x_n \cdot (x_{max} - x_{min}) + x_{min} \quad (3)$$

Keterangan:

x_0 = nilai hasil denormalisasi
 x_n = nilai dinormalisasi
 x_{min} = nilai minimum *dataset*
 x_{max} = nilai maksimum *dataset*

2.4 Extreme Learning Machine (ELM)

Extreme Learning Machine merupakan algoritma bagian dari Jaringan Saraf Tiruan diperkenalkan oleh Huang Guang Bin [4]. Extreme Learning Machine termasuk ke dalam jenis *feedforward neural networks* dengan *single hidden layer* atau disebut dengan *Single Hidden Layer Feedforward Neural Networks* (SLFNs).

Pada arsitektur Extreme Learning Machine terdapat tiga bagian yaitu bagian *input layer*, *hidden layer*, dan *output layer*. Parameter pada Extreme

Learning Machine berupa *weight* dan *bias* yang didapat secara acak. Dalam melakukan perhitungan Extreme Learning Machine dibagi menjadi dua proses, yaitu proses *training* dan proses *testing*. Berikut langkah Extreme Learning Machine pada masing-masing proses *training* dan *testing*.

2.4.1 Proses Training:

Proses *Training* dilakukan untuk melakukan pelatihan data sebelum melakukan peramalan. Hasil yang akan didapatkan berupa nilai *output weight*. Berikut langkah-langkah dalam melakukan proses *Training*:

- Langkah pertama melakukan inialisasi input *weight* dan bias secara acak. Rentang inialisasi pada *weight* adalah antara -1 hingga 1 serta rentang bias adalah antara 0 hingga 1.
- Langkah kedua menghitung nilai H_{init} . Berikut persamaan pada H_{init} .

$$H_{init} = X.W^T + b \quad (4)$$

Keterangan:

H_{init} = hasil matriks
 X = matriks data *training*
 W^T = matriks transposisi *weight*
 b = *bias*

- Langkah ketiga menghitung keluaran *hidden layer* yang didapatkan dari hasil fungsi aktivasi dengan H_{init} .
- Langkah keempat melakukan transposisi pada hasil matriks *hidden layer* (H^T) kemudian mengalikan dengan matriks *hidden layer* (H).
- Langkah kelima melakukan perhitungan *Moore-Penrose Generalized Inverse* dengan cara inversi matriks setelah perkalian transposisi matriks *hidden layer* matriks *hidden layer* ($(H^T.H)^{-1}$) kemudian dikalikan transpose *hidden layer* (H^T) seperti pada persamaan berikut.

$$H^+ = (H^T.H)^{-1}.H^T \quad (5)$$

- Langkah ke tujuh menghitung *output weight*. Berikut persamaan pada *output weight*.

$$\hat{\beta} = H^+ . Y \quad (6)$$

Keterangan:

$\hat{\beta}$ = *output weight*
 H^+ = matriks *Moore-Penrose Generalized Inverse*
 Y = matriks *Target*

2.4.2 Proses *Testing*:

Proses *testing* dilakukan untuk mendapatkan hasil peramalan yang telah dilakukan pada proses *training*. Proses *testing* menggunakan nilai *weight*, dan bias yang didapatkan dari proses *training*. Berikut langkah-langkah dalam melakukan proses *testing*.

- Langkah Pertama mendapatkan nilai *weight*, *bias*, dan *output weight* yang dihasilkan dari proses *training*.
- Langkah kedua menghitung H_{init} yang perhitungannya sama seperti pada proses *training* dengan perbedaan pada nilai X merupakan matriks data *testing* yang kemudian menghitung hidden layer (H) dengan fungsi aktivasi sigmoid biner.
- Langkah ketiga menghitung *output layer* (\hat{Y}) yang berarti hasil peramalan. Berikut persamaan pada *output layer*.

$$\hat{Y} = H \cdot \hat{\beta} \tag{7}$$

Keterangan:

- \hat{Y} = *output layer*
- H = *hidden layer* dengan fungsi aktivasi
- $\hat{\beta}$ = *output weight*

2.5 Mean Absolute Percentage Error

Mean Absolute Percentage Error (MAPE) merupakan cara untuk mengukur tingkat akurasi pada perhitungan. Berikut persamaan pada MAPE [5].

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{A_i - F_i}{A_i} \right| \times 100 \tag{8}$$

Keterangan:

- A_i = nilai aktual
- F_i = nilai peramalan
- n = jumlah periode

2.6 Android

Android merupakan sistem operasi yang dikembangkan untuk perangkat *mobile* berbasis Linux [6].

3. Hasil Percobaan

Dalam Pengujian Metode *Extreme Learning Machine* terdapat beberapa hal yang diuji yaitu, pengujian jumlah fitur, jumlah *hidden neuron* dan jumlah pembagian *training* dan *testing*. Data yang digunakan berupa penjualan mingguan susu sapi dengan 3 model yaitu original, coklat, dan stroberi yang berisi 192 data masing-masing dengan periode 1

Januari 2018 – 31 Desember 2021 Dalam pengujian didapatkan nilai MAPE yang berfungsi untuk mengetahui nilai kesalahan pada hasil pengujian serta mencari nilai kesalahan yang minimal pada melakukan peramalan susu sapi perah. Berikut hasil pengujian yang telah dilakukan.

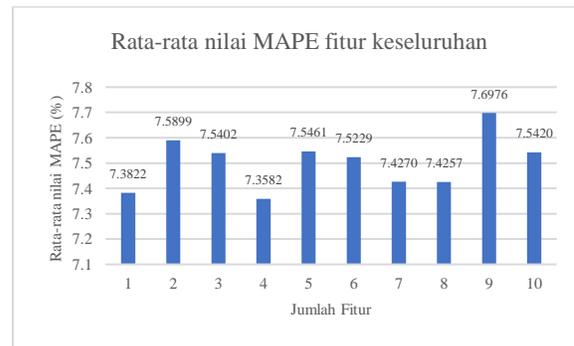
3.1 Hasil Pengujian Jumlah Fitur



Gambar 1 Grafik Hasil Pengujian Jumlah Fitur

Pengujian pada fitur data dilakukan untuk mengetahui pengaruh dari jumlah fitur yang disediakan. Penentuan fitur dilakukan dari penjualan dari jumlah minggu sebelumnya. Parameter yang ditetapkan pada pengujian ini adalah jumlah *hidden neuron* 3, input *weight* dengan range [-1,1], fungsi aktivasi sigmoid biner dengan perbandingan data *training* dan *testing* sebesar 80%:20%.

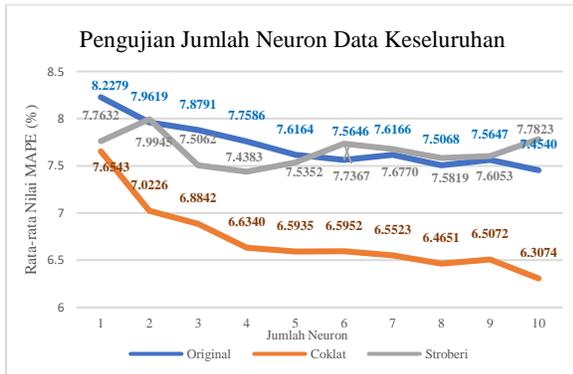
Berdasarkan grafik diatas menunjukkan nilai kesalahan terendah pada jumlah fitur 1 pada model original yaitu dengan nilai kesalahan 7.542% dan model Stroberi dengan nilai kesalahan 7.2302%. Pada model coklat nilai terendah pada jumlah fitur 4 yaitu dengan nilai kesalahan 6.7629%.



Gambar 2 Grafik Hasil Pengujian Jumlah Fitur Keseluruhan

Berdasarkan grafik diatas menunjukkan bahwa rata-rata secara keseluruhan (meliputi model original, coklat, dan stroberi) yang memiliki nilai kesalahan terendah adalah terdapat pada 4 jumlah fitur dengan nilai kesalahan yaitu 7.3582%

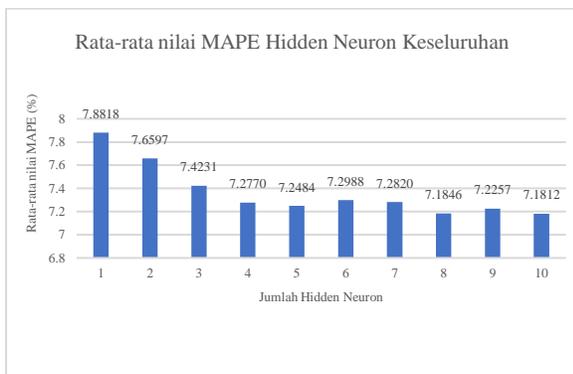
3.2 Hasil Pengujian Jumlah Neuron



Gambar 3 Grafik Hasil Pengujian Jumlah Hidden Neuron

Pengujian jumlah neuron digunakan untuk mengetahui hasil yang terbaik pada *neuron* dalam *Extreme Learning Machine* yang menghasilkan nilai kesalahan terendah. Parameter yang ditetapkan pada pengujian ini adalah hasil jumlah fitur dengan nilai rata-rata keseluruhan yaitu 4 fitur serta fungsi aktivasi sigmoid biner dengan perbandingan data training dan testing sebesar 80%:20%, dan input *weight* dengan range [-1,1]. Pengujian dilakukan sebanyak 10 kali dalam percobaan.

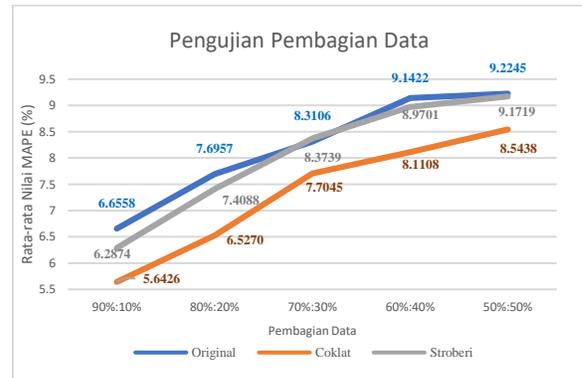
Berdasarkan grafik diatas menunjukkan nilai kesalahan terendah pada jumlah *hidden neuron* 10 pada model original yaitu dengan nilai kesalahan 7.454%, model coklat dengan nilai kesalahan 6.3074%, dan model stroberi pada jumlah *hidden neuron* 4 dengan nilai kesalahan 7.4383%.



Gambar 4 Grafik Hasil Pengujian Jumlah Hidden Neuron Keseluruhan

Berdasarkan grafik diatas menunjukkan bahwa nilai kesalahan terendah keseluruhan (meliputi model original, coklat, dan stroberi) adalah pada jumlah *hidden neuron* 10 dengan nilai kesalahan yaitu 7.1812%.

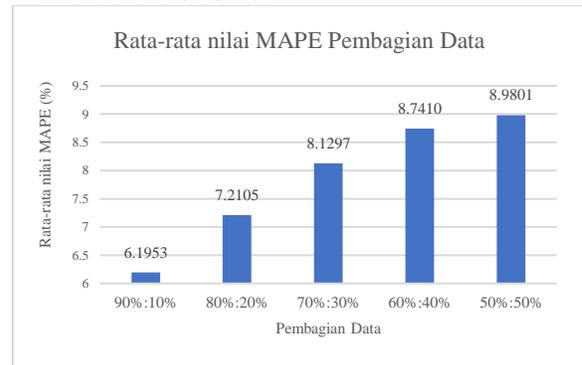
3.3 Hasil Pengujian Pembagian Data Training dan Testing



Gambar 5 Grafik Hasil Pengujian Pembagian Data Training dan Testing

Pengujian perbandingan untuk data training dan testing bertujuan untuk mengetahui pengaruh dalam perbandingan jumlah data *training* dan jumlah data *testing* dengan nilai kesalahan. Parameter yang ditetapkan pada pengujian ini adalah hasil jumlah fitur dengan nilai rata-rata keseluruhan yaitu 4 fitur serta *hidden neuron* terbaik yaitu 10 *hidden neuron*, fungsi aktivasi sigmoid biner, dan input *weight* dengan range [-1,1]. Pengujian dilakukan sebanyak 10 kali dalam percobaan

Berdasarkan grafik diatas menunjukkan nilai kesalahan terendah pada pembagian data yaitu pembagian data 90%:10% pada model original yaitu dengan nilai kesalahan 6.6558%, model coklat dengan nilai kesalahan 5.6426%, dan model stroberi dengan nilai kesalahan 6.2874%.



Gambar 6 Grafik Hasil Pengujian Pembagian Data Training dan Testing

Berdasarkan grafik diatas menunjukkan bahwa nilai kesalahan keseluruhan terendah (meliputi model original, coklat dan stroberi) yaitu pembagian 90%:10% dengan nilai 6.1953%. Pada setiap pengurangan data *training* mengalami kenaikan tingkat kesalahan sehingga jumlah pembagian data mempengaruhi hasil peramalan.

3.4 Hasil Rancangan

Pengujian metode *Extreme Learning Machine* telah dilakukan. Selanjutnya membuat rancangan aplikasi peramalan penjualan. Berikut hasil rancangan aplikasi pada gambar berikut.



Gambar 7 Tampilan Akhir Login



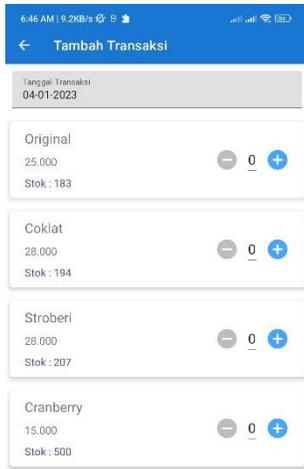
Gambar 8 Tampilan Akhir Register



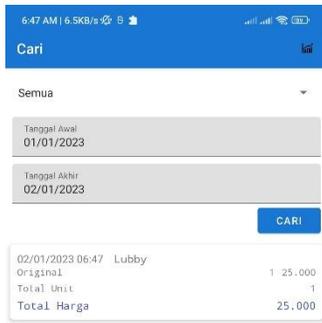
Gambar 9 Tampilan Akhir Lupa Password



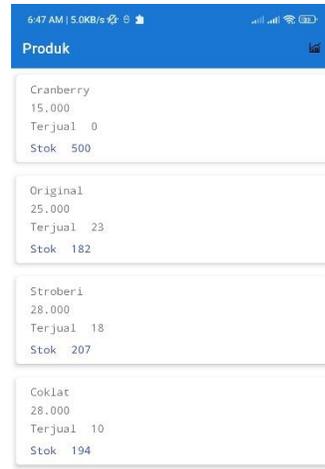
Gambar 10 Tampilan Akhir Home



Gambar 11 Tampilan Akhir Tambah Transaksi



Gambar 12 Tampilan Akhir Cari Transaksi



Gambar 13 Tampilan Akhir Daftar Produk



Gambar 14 Tampilan Akhir Tambah Produk



Gambar 15 Tampilan Akhir Profil Akun



Gambar 16 Tampilan Akhir Ubah Password



Gambar 17 Tampilan Akhir Kelola Pengguna



Gambar 18 Tampilan Akhir Peramalan

3.5 Hasil Pengujian *Black Box Testing*

Saat aplikasi sudah dirancang, dilakukan pengujian *Black Box Testing*. Berikut hasil skenario pengujian.

Tabel 1 Pengujian Skenario *Black Box Testing*

No	Aktivitas Pengujian	Kasus Pengujian	Hasil yang diharapkan	Hasil Pengujian
1	Masuk ke halaman home dengan login	Memasukkan email dan password lalu ketuk tombol "Masuk"	Berhasil login dan pindah ke halaman home	Sesuai
2	Melakukan registrasi akun	Memasukkan nama lengkap, email, dan password lalu ketuk tombol "Registrasi"	Registrasi berhasil dan pindah ke halaman home	Sesuai
3	Melakukan pemulihan password	Menekan tombol "Lupa Password" pada halaman login lalu memasukkan email yang terdaftar pada aplikasi	Permintaan pemulihan berhasil dan dapat dicek pada Email	Sesuai
4	Melakukan penambahan dan penghapusan Transaksi	Mengetuk navigasi "Home" lalu Mengetuk tombol + pada tampilan Home dan memasukan transaksi yang ingin didata kemudian klik simpan, kemudian menekan transaksi teratas yang sudah dibuat untuk melakukan perubahan pada daftar produk yang telah dipilih	Transaksi berhasil tambah dan muncul pada halaman Home dan berhasil memperbaiki transaksi	Sesuai
5	Melakukan pencarian transaksi	Mengetuk navigasi "cari" lalu Mengetuk kolom "Tanggal Awal" kemudian memilih tanggal awal dan akhir lalu mengetuk tombol "Pilih" dan mengetuk tombol "Cari"	Pencarian transaksi tampil sesuai dengan kurun waktu yang dipilih	Sesuai
6	Melakukan penambahan dan perubahan detail produk	Mengetuk navigasi "Produk" lalu mengetuk tombol + kemudian memasukan nama produk dan ketuk tombol "Simpan" lalu mengetuk pada produk yang dibuat sebelumnya untuk mengubah harga dan stok	Produk berhasil ditambahkan dan produk dapat memperbaiki dengan perubahan yang dilakukan	Sesuai

No	Aktivitas Pengujian	Kasus Pengujian	Hasil yang diharapkan	Hasil Pengujian
7	Melakukan perubahan password	Mengetuk navigasi "Profil" lalu mengetuk tombol "Ubah Password" lalu memasukkan kolom "Password Lama, "Password baru", dan "Ulang password baru" lalu mengetuk tombol "Ubah password"	Password berhasil terganti	Sesuai
8	Melakukan peramalan penjualan	Mengetuk tombol logo Chart pada Action bar tampilan Home lalu memilih produk yang ingin diramal dan memasukkan jumlah minggu yang ingin diprediksi dan mengetuk tombol "Peramalan"	Hasil prediksi tampil sesuai dengan jumlah minggu yang ingin diprediksi serta Chart untuk penjualan dan peramalan	Sesuai

4. Kesimpulan

1. Hasil metode *Extreme Learning Machine* yang telah dilalui dengan pengujian jumlah fitur, hidden neuron, dan pembagian data didapatkan nilai kesalahan sebesar 6.6558% pada model Original, 5.624% pada model Coklat, 6.2874% pada model Stroberi. Masing-masing model menggunakan 4 fitur, *hidden neuron* sebanyak 10 serta pembagian jumlah data training dan testing yaitu 90%:10%. Hasil tersebut menunjukkan bahwa metode *Extreme Learning Machine* dapat melakukan peramalan penjualan dengan baik.
2. Perancangan aplikasi yang telah dibuat telah berjalan dengan baik berdasarkan hasil Black Box Testing.
3. Pada pengembangan yang dapat dilakukan untuk penelitian selanjutnya adalah modifikasi ELM seperti OP-ELM (Optimally Pruned Extreme Learning Machine) agar meningkatkan akurasi hasil peramalan yang didapatkan.
4. Memperluas *platform* aplikasi yang tersedia untuk penggunaan aplikasi peramalan penjualan. Saat ini aplikasi hanya dapat berjalan pada Android saja.

REFERENSI

- [1] Afida Izati, Nisa; Warsito, Budi, dan Widihari, Tatik. “Prediksi Harga Emas Menggunakan Feed Forward Neural Network Dengan Metode Extreme Learning Machine,” *Jurnal Gaussian*, Vol. 8. 2019.
- [2] Ariwanda, Galih; Cholissodin, Imam, dan Tibyani. “Prediksi Harga Cabai Rawit di Kota Malang Menggunakan Algoritme Extreme Learning Machine (ELM),” *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, Vol. 3, Nomor 6, 2019.
- [3] Huang, Guang Bin, Qin Yu Zhu, dan Chee Kheong Siew. “Extreme learning machine: Theory and applications,”. *Neurocomputing*, Vol. 70, Nomor 1–3, 2006.
- [4] Prakoso, Bakhtiyar Hadi. “Implementasi Support Vector Regression pada Prediksi Inflasi Indeks Harga Konsumen,” *MATRIK: Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer*, Vol. 19, Nomor 1, 2019.
- [5] Maiyana, Efmi. “Pemanfaatan Android Dalam Perancangan Aplikasi Kumpulan Doa,” *Jurnal Sains dan Informatika*, Vol. 4, Nomor 1, 2018.

Lubby Gennady, mahasiswa S1, program studi Teknik Informatika, Fakultas Teknologi Informasi Universitas Tarumanagara.