

Prediksi Komposisi Bahan Per 100 Gram Makanan Pendamping ASI (MPASI) Menggunakan *Extreme Learning Machine* (ELM)

Wiranto Widotomo¹, Kartika Ananda Putri², Izhar Yusuf Siregar³, Dr. Didit Widyanto⁴, Mayanda Mega Santoni⁵
Informatika / Fakultas Ilmu Komputer
Universitas Pembangunan Nasional Veteran Jakarta
Jl. RS Fatmawati No.1 Pondok Labu Jakarta Selatan 12450
wirantow@upnvj.ac.id, kartika@upnvj.ac.id, izharys@upnvj.ac.id, didit.widyanto@upnvj.ac.id,
megasantoni@upnvj.ac.id

Abstrak. Makanan pendamping ASI (MPASI) adalah asupan gizi bagi balita yang berusia 6 - 24 bulan selain dari ASI sebagai penunjang kebutuhan balita. Kekurangan gizi bagi balita dapat menyebabkan gangguan pertumbuhan fisik maupun mental, menurunkan tingkat kecerdasan, bahkan dapat menyebabkan kematian. Solusi dari permasalahan tersebut dengan memprediksi jumlah komposisi bahan dalam suatu resep makanan pendamping (ASI) menggunakan *Extreme Learning Machine*, sehingga dapat menyesuaikan kebutuhan gizi balita pada resep bubur daging sapi. Penelitian ini juga mencari arsitektur terbaik seperti jumlah *hidden neuron*, fungsi aktivasi dan nilai bobot awal *random*. Sebelum melakukan *training* dan *testing* data dinormalisasi terlebih dahulu menggunakan *min-max normalization*. Arsitektur terbaik pada model ini menggunakan 900 *neuron* di *hidden layer*, fungsi aktivasi *Rectifier Linier Unit*, dan nilai bobot awal *random* dengan *range* -1 sampai 1. Berdasarkan rancangan penelitian ini, hasil pengujian pada proses *training* memiliki nilai MAPE terkecil, yaitu sebesar 0,000000163147. Sedangkan pada proses *testing* memiliki nilai MAPE sebesar 0,768392942.

Kata kunci: MPASI, Prediksi, *Extreme Learning Machine*

1. Pendahuluan

Anak ada usia 0-2 tahun merupakan usia yang rawan gizi buruk, karena masa pertumbuhan dan perkembangan usia ini masih sangat ideal atau masa keemasan, sehingga jika menemui masalah, maka mungkin berdampak negatif. [1]. Berdasarkan Profil Kesehatan Indonesia pada tahun 2017, di Jakarta bayi yang mengalami gizi buruk meningkat 2,07% dari tahun 2016 sebesar 12,07% menjadi 14,14% pada tahun 2017. Menurut *World Health Organization* (WHO), prevalensi anak usia di bawah 5 tahun mengalami *stunting* sebesar 22,2% dengan tiga perempat dari anak-anak tersebut tinggal di wilayah Asia Tenggara atau wilayah Afrika.

Penyebab seorang anak mengalami permasalahan gizi dibagi menjadi dua yaitu penyebab tidak langsung dan penyebab langsung. Penyebab tidak langsung yaitu faktor ekonomi keluarga, pendidikan keluarga, jumlah tanggungan. *World Health Organization* (WHO) menyatakan pada tahun 2014 memprediksi 2 dari 5 anak tidak tumbuh dengan baik pada negara-negara berpenghasilan rendah [2]. Sedangkan penyebab langsung adalah asupan balita seperti MPASI yang diberikan. Pengetahuan ibu tentang MP-ASI dan jenis MP-ASI yang ditambahkan akan sangat berpengaruh terhadap insufisiensi gizi. Menurut penelitian tahun 2010 oleh Muthmainnah di Puskesmas Pamulang, pengetahuan ibu tentang makanan pendamping ASI adalah 54,5%, ibu dengan pengetahuan makanan pendamping cukup sebanyak 31,2%, dan ibu yang memiliki pengetahuan makanan pendamping sedikit sebanyak 14,3%. Berdasarkan kajian tahun 2017 oleh Nababan et al di tiga desa binaan Puskesmas Sapta Bakti Bengkulu, dari bulan Maret hingga April 2017, terdapat sebanyak 59 ibu yang memiliki bayi berusia kurang dari enam bulan, yang sebagian besar memiliki tingkat pendidikan rendah sebanyak 32 ibu (54,2%), hampir 27 (45,8%) memiliki pengetahuan baik. Dari kedua penelitian tersebut menunjukkan bahwa pengetahuan ibu tentang makanan pendamping ASI ini masih cukup rendah, sehingga perlu dilakukan upaya untuk mengatasi masalah tersebut salah satu caranya adalah dengan memprediksi nilai komposisi bahan yang terdapat pada MPASI.

Dalam penelitian yang berjudul "Perbandingan Algoritma ELM Dan *Backpropagation* Terhadap Prestasi Akademik Mahasiswa" oleh Heny Pratiwi dan Kusno Harianto dari STMIK Widya Cipta Dharma. Berdasarkan penelitian tersebut menunjukkan bahwa algoritma tersebut memiliki 14,84% tingkat *error* lebih rendah dibandingkan dengan metode *Backpropagation* yang memiliki tingkat *error* sebesar 28,2%. Dari tahapan pengujian model didapatkan hasil yang paling akurat adalah algoritma *Extreme Learning Machine* karena memiliki tingkat *error* yang paling kecil [3].

Berdasarkan uraian diatas, maka diperlukan sistem prediksi energi dari setiap menu agar membantu ibu dalam menentukan energi MPASI untuk anaknya. Terdapat metode yang diterapkan salah satu metodenya adalah jaringan syaraf tiruan dengan algoritma *Extreme Learning Machine* (ELM). Metode *Extreme Learning Machine* (ELM) merupakan *Machine Learning Feedforward* dengan satu *hidden layer* yang membutuhkan data *input* dan *output* yang digunakan untuk melatih jaringan syaraf tiruan sehingga diperoleh bobot yang diinginkan.

2. Tinjauan Pustaka

2.1 MPASI

MP-ASI adalah makanan tambahan atau makanan kedua yang diberikan kepada bayi selain ASI, setelah bayi berusia 6 bulan sampai usia bayi mencapai 24 bulan karena untuk memenuhi gizi balita tidak cukup hanya dengan ASI [4]. Kebutuhan energi dan gizi tidak cukup hanya dari ASI atau bisa disingkat MPASI. Makanan Pendamping ASI atau bisa disingkat MPASI adalah makanan atau minuman yang mengandung zat gizi dan diberikan kepada bayi atau anak usia dari 6 bulan sampai 24 bulan guna memenuhi kebutuhan gizi selain dari ASI.

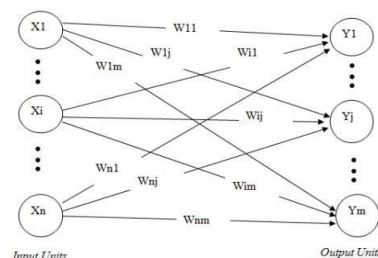
MPASI merupakan peralihan asupan yang berawal dari susu menuju ke makanan yang bersifat semi padat. Setelah enam bulan seiring bertambah usia bayi keperluan zat gizi baik makronutrien maupun mikronutrien tidak mampu diperoleh hanya dari ASI saja. Jenis MPASI juga perlu diperhatikan, sesuai dengan Permenkes No. 24 Tahun 2007 menyebutkan MPASI untuk usia bayi 6 – 12 bulan adalah makananlumut atau bubur, untuk usia bayi 12 – 24 bulan adalah makanan biskuit. Pergantian jenis makanan dari bubur menjadi biskuit membuat bayi dalam kemampuan makan (*oromotor skills*) terus bertumbuhkembang dan mulau menunjukkan minat makan selain berbentuk susu. Proses ini juga membuat si bayi memiliki kemampuan motorik mulut berkembang dari menghisap sampai menelan makanan berbentuk semi padat dengan mengangkat makanan dari bagian depan lidah menuju bagian belakang lidah.

2.2 Jaringan Syaraf Tiruan

Jaringan syaraf tiruan (JST) atau *Artificial Intelligence* merupakan sistem komputasi dimana arsitektur dan operasi yang diberikan petunjuk dari keilmuan tentang sel syaraf biologis.dalam otak [5]. Jaringan SyarafTiruan (JST) merupakan cabang ilmu dari *Machine Learning*. Metode yang digunakan untuk membangun program jaringan syaraf tiruan yang cerdas dengan memodelkan sistem bagaimana jaringan syaraf pada otak manusia bekerja. Jaringan saraf tiruan dapat dipahami sebagai otak buatan yang bekerja,dapat berpikir seperti manusia, serta dapat menyimpulkan sesuatu permasalahan dari informasi yang diterimanya, dan secerdas manusia. Istilah JST digunakan karena jaringan syaraf tiruan diimplementasikan menggunakan program komputer yang dapat melakukan sejumlah besar perhitungan selama proses pembelajaran. Jaringan syaraf tiruan (JST) biasanya dilakukan untuk penelitian kecerdasanbuatan. Terdapat tiga arsitektur jaringan syaraf tiruan yaitu:

1. Jaringan Lapis Tunggal (*single layer net*)

Jaringan jenis ini hanya memiliki lapisan *neuron* yang digunakan untuk nilai masukan (*input*) dan nilai keluaran (*output*) tanpa memiliki lapisan tersembunyi (*hidden layer*). Arsitektur jaringan lapis tunggal (*single layer net*) dapat diilustrasikan pada Gambar 1.

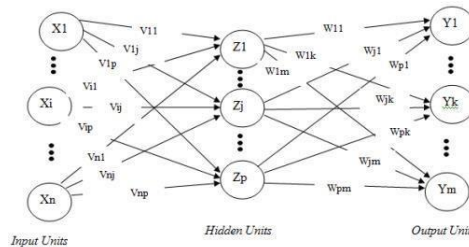


Gambar. 1. Jaringan Lapis Tunggal (*single layer net*)

Pada Gambar 1 diperlihatkan bahwa arsitektur tersebut dijelaskan bahwa n buah sebagai nilai masukan (X_1, X_2, \dots, X_n) dan m buah keluaran (Y_1, Y_2, \dots, Y_m). Dalam arsitektur ini semua nilai output menghubungkan semua nilai input.

2. Jaringan Multi Lapis (*multi layer net*)

Berbeda dengan jaringan lapis tunggal, jenis arsitektur ini memiliki satu atau beberapa lapisan tersembunyi (*hidden layer*) dan untuk memecahkan suatu masalah arsitektur ini cenderung lebih baik daripada arsitektur lapisan tunggal, namun proses pelatihannya rumit. Arsitektur jaringan multi lapis (*multi layer net*) dapat diilustrasikan pada Gambar 2.

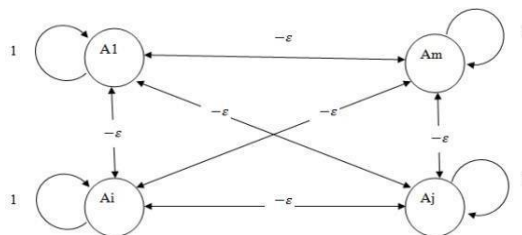


Gambar. 2. Jaringan Multi Lapis (*multi layer net*)

Pada Gambar 2 terlihat bahwa arsitektur jaringan lapisan tunggal dengan n buah sebagai unit *input* (X_1, X_2, \dots, X_n), p buah unit sebagai lapisan tersembunyi (Z_1, Z_2, \dots, Z_p) dan m buah sebagai unit *output* (Y_1, Y_2, \dots, Y_m). Arsitektur ini dapat menyelesaikan permasalahan yang lebih kompleks dibandingkan dengan arsitektur lapisan tunggal namun proses pelatihannya lebih kompleks dan lama.

3. Jaringan Kompetitif (*competitive net*)

Bentuk dari jaringan kompetitif mirip dengan jaringan lapisan tunggal ataupun lapisan ganda. Namun, pada unit input diberikan sinyal oleh neuron pada unit output (sering disebut *feedback loop*). Arsitektur jaringan kompetitif (*competitive net*) dapat diilustrasikan pada Gambar 3.



Gambar. 3. Jaringan Kompetitif (*competitive net*)

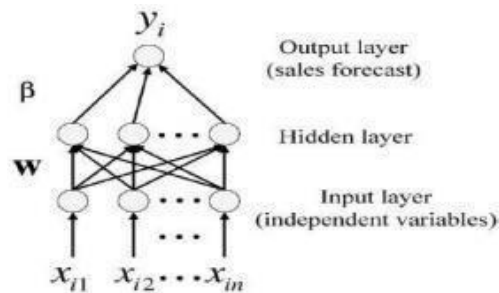
Pada Gambar 3 jaringan ini memperoleh nilai bobot untuk diri sendiri dari setiap neuron bernilai 1, dan neuron lainnya adalah bobot random negatif dengan bobot $-\epsilon$.

2.3 Extreme Learning Machine

Metode ELM (*Extreme Learning Machine*) pertama kali diperkenalkan pada tahun 2006 oleh Huang, G.B., Zhu, Q.Y. dan Siew, C.K. Huang dkk (2004). ELM (*Extreme Learning Machine*) merupakan salah satu model pembelajaran jaringan syaraf tiruan. ELM (*Extreme Learning Machines*) secara umum dapat disebut SLFN (*Single Hidden Layer Feedward Neural Networks*). Metode pembelajaran ELM dibentuk untuk mengatasi kekurangan dari metode feedforward terutama dalam hal kecepatan belajar.. Huang (2004) menyebutkan dua alasan mengapa JST *feedforward* memiliki *learning speed* yang rendah, yaitu:

1. Saat melakukan proses pelatihan (*training*) menggunakan *slow gradient based learning algorithm*.
2. Semua bobot pada jaringan ditentukan secara iteratif dengan memanfaatkan metode pembelajaran tersebut.

Semua barometer pada pembelajaran JST *feedforward* seperti *backpropagation* harus ditentukan secara manual. Barometer yang ditunjukkan adalah *input weight* dan *hidden bias*. Barometer ini saling berhubungan antara satu *layer* ke *layer* yang lain, sehingga memerlukan waktu yang lama untuk *learningspeed* dan sering terjebak *local minima*. Berikut adalah arsitektur dari *Extreme Learning Machine* yang dapat dilustrasikan pada Gambar 4.



Gambar. 4. Struktur ELM

Tahapan perhitungan dengan metode ELM yang pertama adalah normalisasi data kemudian data akan diproses pada langkah proses *training* dan proses *testing*.

1. Normalisasi Data

Proses normalisasi data merupakan bagian dari pra proses data yang bertujuan untuk mengubah kolom nilai numerik kedalam ke skala yang sama, tanpa mengganggu perbedaan dalam rentang nilai. Metode tersebut merupakan metode normalisasi dengan mengolah nilai minimum dan maksimum dari setiap fitur sehingga membuat keseimbangan nilai perbandingan antara nilai sebelum dan sesudah. Rentang yang dipakai dalam metode ini adalah 0-1. Metode *Min-Max Normalization* dapat dirumuskan dengan Persamaan 1 sebagai berikut [6].

$$X' = \frac{(X_t - X_{min})}{(X_{max} - X_{min})} \quad (1)$$

Keterangan :

X' = Nilai normalisasi
 X_t = Nilai data yang akan di normalisasi X_{min} = Nilai data minimal di field
 yang sama X_{max} = Nilai data maksimal di field yang sama

2. Pembagian Data

Untuk penelitian ini pembagian data menggunakan *hold-out validation*. Metode *hold-out validation* membagi data menjadi dua kelompok, yaitu data latih dan data uji. Data latih digunakan untuk membangun model, dan data uji digunakan untuk menguji kinerja model yang dibentuk oleh data latih. Namun, metode ini memiliki kelemahan, karena hasil evaluasi dapat sangat bervariasi tergantung pada distribusi data dalam set pelatihan dan tes [7]. Berdasarkan uraian diatas, kesimpulan *hold-out validation* merupakan suatu proses pembagian data sebanyak 2 bagian, untuk satu bagian akan dijadikan data latih (*training*) sebanyak 90 persen dan satu bagian lainnya akan dijadikan data uji (*testing*) sebanyak 10 persen. Dengan metode ini model akan mencoba semua variasi data yang ada. Oleh karena itu, *hold-out validation* ini memiliki kemampuan untuk menghasilkan evaluasi model yang baik, sebagaimana pembagian data menggunakan *hold-out validation* dapat dideskripsikan pada **Tabel 1**.

Tabel 1. Pembagian Data Latih (*Training*) dan Data Uji (*Testing*)

| Nomor | Data Latih (Data ke-) | Data Uji (Data ke-) |
|-------|-----------------------|---------------------|
| 1 | 21 s.d. 200 | 1 s.d. 20 |

3. Proses *Training*

Proses *training* harus dilakukan sebelum melakukan proses prediksi. Tujuan proses ini untuk memperoleh nilai *output weight*. Prosedur proses *training* dapat dijelaskan sebagai berikut:

- a. Tahapan pertama adalah menginisialisasi nilai *random*. Nilai ini dinisialisasi secara *random* dengan rentang nilai antara -1 hingga 1.

- b. Setiap *input* $X_i(i=1,2,\dots,n)$ memperoleh sinyal dan memberikan ke semua unit pada lapisan *hidden*. *Neuron* yang terdapat di *hidden layer* berjumlah 900 sehingga menghasilkan bobot *input*.
- c. Menghitung nilai *output* pada *hidden layer* ditunjukkan pada Persamaan 2.

$$\sum_{i=1}^N \beta_i (W_i \cdot x_j + b_i) \quad (2)$$

Dimana :

W_i = nilai bobot yang terhubung dengan *input layer* dan *hidden layer*. β = nilai bobot yang terhubung dengan *hidden layer* dan *output layer*
 b_i = nilai bias dari kode ke - i
 $w_i x_j$ = inner produk dari w_i dan x_j

- d. Menghitung nilai *output* pada *hidden layer* dengan menggunakan fungsi aktivasi *Rectifier Linier Unit* yang ditunjukkan pada persamaan 3.

$$H(x) = \max(0, x) \quad (3)$$

Dimana :

$H(x)$ = Matriks *output hidden layer*
 x = *Input*

- e. Menghitung matriks *Moore-Penrose Generalized Invers* yang ditunjukkan pada persamaan 4.

$$H^+ = (H(x)^T \cdot H(x))^{-1} H(x)^T \quad (4)$$

Dimana :

H^+ = Matriks *Moore-Penrose Generalized Invers*
 $H(x)$ = Matriks *output hidden layer*

- f. Menghitung *output weight*. Setelah didapatkan nilai dari matriks *Moore-Penrose Generalized Invers* kemudian dihitung *output weight* yang ditunjukkan pada Persamaan 5.

$$\beta = H^+ \cdot T \quad (5)$$

Dimana :

β = Matriks *output weight*
 H^+ = Matriks *Moore-Penrose Generalized Invers*
 T = Matriks target

4. Proses *Testing*

Pada proses *testing* bertujuan untuk mendapatkan nilai *output weight* dari hasil proses pelatihan (*training*) sebelumnya. Proses pengujian (*testing*) dilakukan dengan cara menggunakan nilai *input weight*, bias, dan *output weight* yang diperoleh dari proses pelatihan (*training*) sebelumnya. Berikut ini adalah tahapan proses pengujian (*testing*) :

- g. Tahapan pertama adalah mendefinisikan nilai *input weight* yang diperoleh dari proses pelatihan (*training*).
- h. Nilai keluaran di *hidden layer* dihitung dengan memanfaatkan fungsi aktivasi.
- i. Nilai *output weight* yang diperoleh dari proses pelatihan (*training*) digunakan untuk mengukur keluaran *output layer* yang merupakan hasil ramalan atau prediksi yang ditunjukkan pada Persamaan 6.

$$O = H(x) \cdot \beta \quad (6)$$

Keterangan :

O = *Output Peramalan*
 $H(x)$ = Matriks Keluaran *hidden layer*
 β = Matriks *output weight*

- j. Langkah terakhir adalah menghitung nilai galat atau *error* semua *output layer*. Nilai galat ini membuktikan

nilai kesalahan dari hasil ramalan yang didapatkan.

3 . Menghitung Kesalahan Prediksi

Mean Absolute Percentage Error (MAPE) merupakan pengukuran tingkat kesalahan dengan menghitung ukuran distorsi antara data aktual dengan data prediksi yang diperoleh. MAPE mengindikasikan seberapa besar galat antara nilai prediksi dengan nilai nyata. Pendekatan ini menguntungkan ketika ukuran variabel itu penting untuk mengevaluasi ketepatan prediksi. Rumus MAPE dapat dijelaskan pada Persamaan 7 sebagai berikut :

$$MAPE = 100\% \sum_{t=1}^n \frac{|X_t - Y_t|}{X_t} \quad (7)$$

Dimana,

MAPE = Mean Absolute Percentage Error
 n = Jumlah data
 Y_t = Nilai output pada periode ke- t
 X_t = Nilai data aktual pada periode ke- t

3. Hasil dan Pembahasan

3.1 Data

Penelitian ini menggunakan data primer yaitu berupa resep makanan utama MPASI. Di dalam resep tersebut terdapat bahan makanan serta energi yang dihasilkan dari makanan utama MPASI tersebut. Dataset ini memiliki 200 record dan bersifat regresi. Variabel pada dataset ini dibagi menjadi dua variabel utama yaitu, bahan makanan dan zat gizi yang terkandung. Dimana variabel zat gizi yang menjadi nilai *input* terdapat 4 fitur yaitu, kalori, karbohidrat, protein, lemak. Sementara itu variabel bahan makanan yang menjadi nilai *output* terdapat 6 fitur yaitu, nasi merah, daging sapi, daun bayam, minyak kelapa, bawang putih. Dataset Nilai Gizi Resep Bubur Daging Sapi dapat dideskripsikan pada Tabel 2.

Tabel 2. Nilai Gizi Resep Bubur Daging Sapi (5 Data Awal)

| No | Zat Gizi | | | |
|----|----------|-------------|---------|---------|
| | Kalori | Karbohidrat | Protein | Lemak |
| 1 | 935,913 | 75,334 | 46,299 | 51,37 |
| 2 | 2111,54 | 48,43 | 40,743 | 201,222 |
| 3 | 1376,23 | 140,179 | 50,969 | 69,064 |
| 4 | 1313,56 | 83,684 | 16,104 | 107,128 |
| 5 | 1317,85 | 91,684 | 32,403 | 95,2 |

Dataset Komposisi Bahan Resep Bubur Daging Sapi dapat dideskripsikan pada Tabel 3.

Tabel 2. Komposisi Bahan Resep Bubur Daging Sapi (5 Data Awal)

| No | Bahan Makanan | | | | | |
|----|-----------------|------------------|-----------------|--------------------|-------------------|-------------------|
| | Nasi Merah (gr) | Daging Sapi (gr) | Daun Bayam (gr) | Minyak Kelapa (ml) | Bawang Merah (gr) | Bawang Putih (gr) |
| 1 | 39 | 141 | 64 | 24 | 66 | 179 |
| 2 | 23 | 127 | 90 | 177 | 180 | 56 |
| 3 | 147 | 125 | 164 | 42 | 119 | 44 |
| 4 | 36 | 5 | 137 | 104 | 183 | 163 |
| 5 | 84 | 80 | 85 | 78 | 15 | 104 |

3.2 Normalisasi Data

Sebelum dilakukan proses *training* dataset ini dilakukan proses normalisasi, kumpulan data ini harus dikumpulkan dan diubah menjadi nilai dengan interval 0 sampai dengan 1 menggunakan Persamaan 1. Proses normalisasi hanya dilakukan pada variabel input pada Tabel 1 sedangkan pada Tabel 2 tidak dilakukan proses normalisasi. Hasil penghitungan normalisasi pada nilai gizi resep bubur daging sapi dideskripsikan pada **Tabel 3**.

Tabel 3. Hasil Normalisasi Nilai Gizi Resep Bubur Daging Sapi (5 Data Awal)

| No | Zat Gizi | | | |
|----|------------|-------------|------------|------------|
| | Kalori | Karbohidrat | Protein | Lemak |
| 1 | 0.20781952 | 0.31255744 | 0.57665624 | 0.17131755 |
| 2 | 0.65819731 | 0.17126457 | 0.49061542 | 0.83910126 |
| 3 | 0.37650198 | 0.65310667 | 0.64897637 | 0.25016711 |
| 4 | 0.35249413 | 0.35640949 | 0.10905318 | 0.41979127 |
| 5 | 0.35413722 | 0.39842343 | 0.36146127 | 0.36663666 |

3.3 Penentuan Parameter Jaringan

Dalam pemilihan parameter pada algoritme *Extreme Learning Machine* diperlukan uji parameter seperti unit masukan (*input neuron*), unit tersembunyi (*hidden neuron*), unit keluaran (*output neuron*), fungsi aktivasi, bobot *random*. Untuk unit masukan (*input neuron*) sebanyak 4 neuron dengan asumsi merupakan perwakilan dari jumlah zat gizi. Pada unit keluaran (*output neuron*) sebanyak 1 neuron yang merupakan hasil dari peramalan. Sementara itu jumlah neuron di hidden layer, fungsi aktivasi, bobot *random* dilakukan beberapa percobaan, sehingga diharapkan menghasilkan model *Extreme Learning Machine* yang optimal (nilai *error* terkecil). Pengujian parameter ini menggunakan data *training* dengan pembagian data dari baris ke-21 sampai baris ke-200 dengan target komposisi bahan Nasi Merah. Berikut merupakan inialisasi parameter yang akan digunakan pada **Tabel 4**.

Tabel 4. Inialisasi Parameter Jaringan Syaraf Tiruan

| Karakteristik | Spesifikasi (Nilai) |
|--|--|
| Jumlah unit masukan (<i>input neuron</i>) | 4 |
| Jumlah unit tersembunyi (<i>hidden neuron</i>) | 100, 200, 300, 400, 500, 600, 700, 800, 900, 1000 |
| Jumlah unit keluaran (<i>output neuron</i>) | 1 |
| Fungsi aktivasi | <i>Sigmoid Biner, Rectifier Linier Unit</i> |
| Inialisasi bobot random | Positif (0 s.d 1) Negatif (-1 s.d 1) |

Berdasarkan **Tabel 4** kemudian dibentuk beberapa kombinasi parameter yang akan diujikan pada penelitian ini. Parameter yang akan dikombinasikan yaitu jumlah *neuron* di *hidden layer*, fungsi aktivasi, dan iniliasasi bobot random awal. Terdapat dua kombinasi untuk dilakukan pengujian pada parameter, kombinasi pertama yang dapat dideskripsikan pada **Tabel 5**.

Tabel 5. Kombinasi Pertama

| Jumlah Neuron di Hidden Layer | Fungsi Aktivasi | Nilai Bobot Awal Random |
|-------------------------------|-----------------|-------------------------|
| 100 | | |
| 200 | | |
| 300 | | |
| 400 | | |

| | | |
|------|----------------------------------|--------------------------|
| 500 | <i>Rectifier Linear Unit</i> | Negatif (-1 sampai 1) |
| 600 | | |
| 700 | | |
| 800 | | |
| 900 | | |
| 1000 | | |

Kombinasi keempat pada pengujian parameter dapat dideskripsikan pada **Tabel 6**.

Tabel 6. Kombinasi Kedua

| Jumlah Neuron di Hidden Layer | Fungsi Aktivasi | Nilai Bobot Awal Random |
|--|----------------------------------|------------------------------------|
| 100 | <i>Rectifier Linear Unit</i> | Positif(0 sampai 1) |
| 200 | | |
| 300 | | |
| 400 | | |
| 500 | | |
| 600 | | |
| 700 | | |
| 800 | | |
| 900 | | |
| 1000 | | |

Setelah kombinasi parameter yang akan diujikan telah didefinisikan pada Tabel 5 sampai Tabel 6. Selanjutnya melihat hasil kesalahan prediksi menggunakan perhitungan MAPE yang tujuannya adalah melihat kombinasi terbaik untuk dataset resep bubur daging sapi. Hasil MAPE tersebut dapat dideskripsikan pada **Tabel 7**.

Tabel 7. Hasil MAPE Setiap Kombinasi

| Jumlah <i>Neuron</i> di <i>Hidden Layer</i> | Kombinasi Pertama | Kombinasi Kedua |
|---|-------------------|-----------------|
| 100 | 0,2540303 | 0,3416816 |
| 200 | 2,74175 | 0,3416816 |
| 300 | 0,1348342 | 0,3416816 |
| 400 | 0,146534 | 0,3416816 |
| 500 | 0,043984 | 0,3416816 |
| 600 | 0,01694052 | 0,3416816 |
| 700 | 0,007787535 | 0,3416816 |
| 800 | 0,004741567 | 0,3416816 |
| 900 | 0,00000008281 | 0,3416816 |
| 1000 | 0,00000012529 | 0,3416816 |

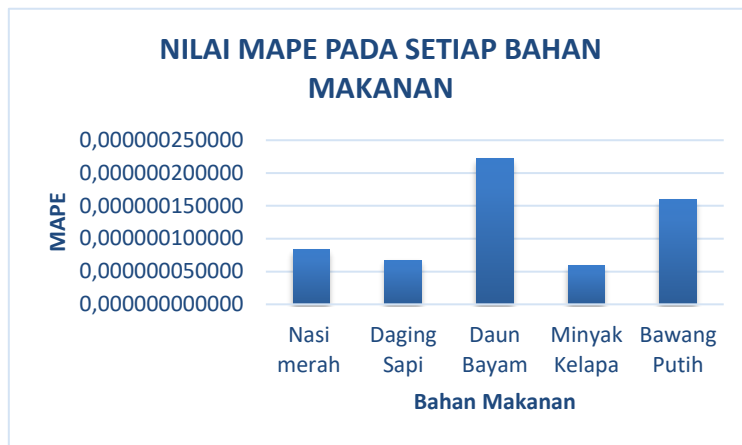
Pada Tabel 7 didapatkan nilai MAPE pada setiap kombinasi parameter. Nilai MAPE yang paling kecil adalah kombinasi ketiga dengan jumlah *neurin* di *hidden layer* sebanyak 900 *neuron* dengan nilai MAPE sebesar 0,00000008281. Inilasi parameter yang optimal dapat dideskripsikan pada **Tabel 8**.

Tabel 8. Inisialisasi Parameter Jaringan Syaraf Tiruan Terbaik

| Karakteristik | Spesifikasi (Nilai) |
|--|------------------------------|
| Jumlah unit masukan (<i>input neuron</i>) | 4 |
| Jumlah unit tersembunyi (<i>hidden neuron</i>) | 900 |
| Jumlah unit keluaran (<i>output neuron</i>) | 1 |
| Fungsi aktivasi | <i>Rectifier Linier Unit</i> |
| Inisialisasi bobot random | Negatif (-1 s.d 1) |

3.4 Hasil Training

Setelah melakukan proses menentukan parameter jaringan yang tujuannya untuk menemukan parameter terbaik dan optimal sehingga selanjutnya bisa dilakukan proses pelatihan (*training*) dan proses pengujian (*testing*). Jika akurasi hasil prediksi kurang memuaskan, maka parameter yang sudah dipilih akan dilatih ulang hingga mencapai hasil akurasi yang optimal. Pembagian data pada proses pelatihan (*training*) ini menggunakan data latih yang tertera pada Tabel 1. Berdasarkan proses pelatihan (*training*) tersebut didapatkan nilai prediksi yang selanjutnya membandingkan antara nilai asli dan nilai prediksi sehingga didapatkan nilai MAPE sebesar 0,000000163147. Menggunakan data latih berikut ini merupakan hasil nilai *output* komposisi bahan dari resep bubur daging sapi yaitu nasi merah, daging sapi, daun bayam, minyak kelapa, bawang putih, dan bawang merah. Selain itu didapatkan nilai MAPE di setiap target pada setiap bahan makanan yang diilustrasikan pada Gambar 5.

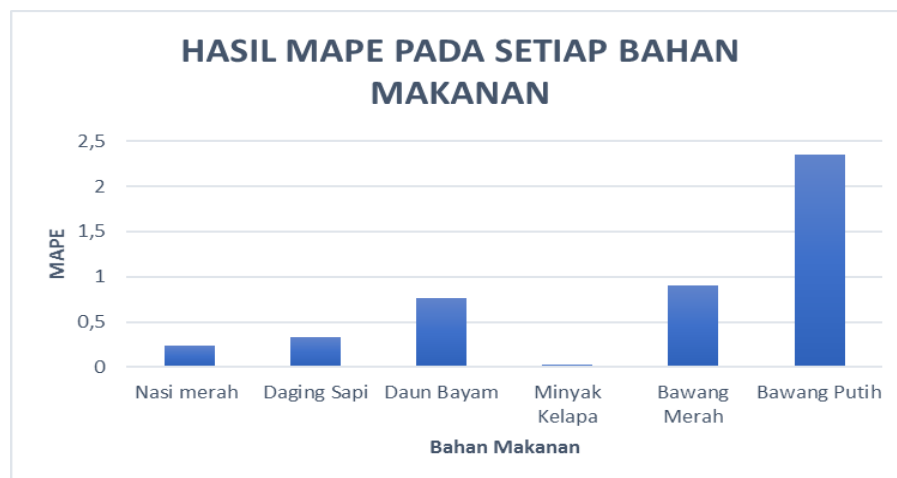


Gambar. 5. Hasil MAPE *Training* di Setiap Bahan

Berdasarkan Gambar 5 dengan menggunakan pembagian data latih tersebut, bahan makanan Minyak Kelapa memiliki nilai MAPE terkecil sebesar 0,000000059228. Sementara itu bahan makanan Daun Bayam memiliki nilai MAPE terbesar sebesar 0,00000022148.

3.5 Hasil *Testing*

Berikut merupakan hasil dari proses *testing* menggunakan arsitektur yang sudah ditentukan dan pembagian data *training* menggunakan data latih yang tertera pada Tabel 1. Setelah itu dilakukan proses pengujian menggunakan data uji yang tertera pada Tabel 1. Berdasarkan proses pengujian (*testing*) tersebut didapatkan nilai prediksi yang selanjutnya membandingkan antara nilai asli dan nilai prediksi sehingga didapatkan nilai MAPE sebesar 0,768392942. Menggunakan data uji berikut ini merupakan hasil nilai *output* komposisi bahan dari resep bubur daging sapi yaitu nasi merah, daging sapi, daun bayam, minyak kelapa, bawang putih, dan bawang merah. Selain itu didapatkan nilai MAPE di setiap target pada setiap bahan makanan yang diilustrasikan pada Gambar 6.



Gambar. 6. Hasil MAPE *Testing* di Setiap Bahan

Berdasarkan Gambar 6 dengan menggunakan pembagian data uji tersebut, bahan makanan Minyak Kelapa memiliki nilai MAPE terkecil sebesar 0,02984695. Sementara itu sama halnya dengan proses *training*, bahan makanan Bawang Putih memiliki nilai MAPE terbesar sebesar 2,346228. Pada bahan makanan Bawang Putih hasil prediksi yang didapatkan merupakan nilai negatif, yang dimana sangat jauh dengan nilai data aslinya. Sehingga ini merupakan alasan kenapa pada bahan makanan Bawang Putih memiliki nilai MAPE yang paling besar diantara yang lainnya.

4. Kesimpulan dan Saran

4.1 Kesimpulan

Berdasarkan hasil pengujian dan pembahasan dari prediksi komposisi bahan MPASI menggunakan metode *Extreme Learning Machine* (ELM) maka diperoleh kesimpulan sebagai berikut:

1. Pada peramalan nilai komposisi bahan pada MPASI dengan resep bubur daging sapi menggunakan tiga parameter yaitu jumlah *neuron* di *hidden layer*, fungsi aktivasi, dan nilai bobot random. Melalui hasil pengujian, kemudian didapatkan kombinasi parameter ketiga dengan jumlah *neuron* di *hidden layer* sebanyak 900 *neuron* dikarenakan memiliki nilai MAPE terkecil sebesar 0,00000008281. Kombinasi tersebut terdiri dari fungsi aktivasi *Rectifier Linear Unit* dan bobot awal random dari *range* -1 sampai dengan 1.
2. Metode evaluasi yang digunakan pada peramalan nilai komposisi bahan resep bubur daging sapi menggunakan metode *Extreme Learning Machine* (ELM) yaitu *Mean Absolute Percentage* (MAPE). Melalui hasil pengujian yang telah dilakukan didapatkan bahwa pada proses *training* memiliki nilai MAPE sebesar 0,000000163147. Sedangkan pada proses *testing* memiliki nilai MAPE yaitu sebesar 0,768392942.

4.2 Saran

Berdasarkan hasil penelitian ini, maka penulis memberikan beberapa saran agar penelitian selanjutnya dapat dijadikan bahan pertimbangan, yaitu sebagai berikut:

1. Penelitian ini belum mengimplementasikan menggunakan aplikasi, sehingga diharapkan penelitian ini dapat dikembangkan dalam bentuk aplikasi yang sederhana.
2. Pada penelitian ini hanya menggunakan satu resep MPASI saja yaitu resep bubur daging sapi, diharapkan untuk penelitian selanjutnya menggunakan berbagai macam resep MPASI.
3. Penelitian ini hanya berfokus kepada metode *Extreme Learning Machine* (ELM) untuk penelitian selanjutnya diharapkan membandingkan beberapa metode jaringan syaraf tiruan lainnya agar mengetahui apakah metode ini cocok untuk kasus ini atau tidak.
4. Penelitian ini diharapkan menggunakan pembagian data seperti *10-fold cross validation*.

Referensi

- [1] Pusat Data dan Informasi Kementerian Kesehatan RI, "Data & Informasi 2015 'Profil Kesehatan Indonesia,'" 2015.
- [2] H. Pratiwi and K. Harianto, "Perbandingan Algoritma ELM Dan Backpropagation Terhadap Prestasi Akademik Mahasiswa," *J-SAKTI (Jurnal Sains Komput. dan Inform.,* vol. 3, no. 2, p. 282, 2019, doi: 10.30645/j-sakti.v3i2.147.
- [3] E. Ridhawati and L. Julianti, "Pemilihan Makanan Pendamping ASI Pada Batita Dengan Menerapkan Metode TOPSIS," 2018.
- [4] E. B. Prasetya and N. Amri, "Sistem Informasi Untuk Menentukan Menu Makanan Pendamping Asi (Mpas) Bayi Berdasarkan Angka Kecukupan Gizi (Akg) Menggunakan Metode Forward Chaining," *Resist. (elektRONIKA kEndali Telekomun. tenaga List. kOmputeR)*, vol. 2, no. 1, p. 15, 2019, doi: 10.24853/resistor.2.1.15-22.
- [5] E. A. Dewi, "PERBANDINGAN METODE HOLT WINTER'S EXPONENTIAL SMOOTHING DAN EXTREME LEARNING MACHINE (ELM) PADA PERAMALAN PENJUALAN SEMEN," *Univ. Islam Indones.,* no. 21, pp. 1-9, 2018.
- [6] R. J. D. S, "PERAMALAN CURAH HUJAN MENGGUNAKAN METODE EXTREME LEARNING MACHINE," 2020.
- [7] Jamaludin, C. Rozikin, and A. S. Y. Irawan, "Klasifikasi Jenis Buah Mangga dengan Metode Backpropagation," *Techné J. Ilm. Elektrotek.,* vol. 20, no. 1, pp. 1-12, 2021, doi: 10.31358/techne.v20i1.231.