BAB 4  
IMPLEMENTASI DAN PENGUJIAN SISTEM

Bab ini membahas tentang hasil yang diperoleh dari implementasi metode yang digunakan, yaitu *extreme learning machine* (ELM), untuk melakukan prediksi kualitas air di Danau Toba. Bab ini akan menjabarkan hasil perancangan antarmuka yang digunakan dalam proses prediksi, prosedur operasional dari aplikasi yang dirancang, dan hasil prediksi yang didapat menggunakan ELM.

# Implementasi Sistem

Dalam penelitian ini, tahap *preprocessing* akan diimplementasikan ke dalam sistem menggunakan bahasa pemrograman Python, sedangkan tahap pelatihan hingga visualisasi hasil akhir akan diimplementasikan ke dalam sistem menggunakan bahasa pemrograman MATLAB.

## Spesifikasi perangkat keras dan perangkat lunak yang digunakan

Spesifikasi perangkat keras dan perangkat lunak yang digunakan untuk membangun sistem ini adalah sebagai berikut:

1. Prosesor Intel(R) Core(TM) i5-3317U CPU @ 1.70 GHz
2. Kapasitas memori RAM sebesar 4 GB
3. Sistem operasi Windows 10 Pro 64-bit
4. *Hard drive* yang memiliki kapasitas sebesar 500 GB
5. *Software* yang digunakan adalah Python versi 2.7 dan MATLAB versi R2015a (8.5.0.197613)
6. *Library* yang digunakan adalah *library* *elm\_train* dan *elm\_predict* yang diterbitkan oleh Zhu *et al.* (2004)

## Implementasi perancangan antarmuka

Perancangan antarmuka sistem dibuat berdasarkan rancangan yang telah dilakukan pada bab 3. Antarmuka aplikasi yang telah dirancang pada penelitian ini ditunjukkan o-leh Gambar 4.1. Antarmuka yang dirancang terdiri dari halaman utama, di mana dalam halaman utama terdapat menu-menu yang digunakan untuk proses prediksi kualitas air menggunakan *extreme learning machine*.



**Gambar 4.1.** Hasil rancangan antarmuka aplikasi

Setelah data latih, data uji, jumlah *hidden neuron*, dan fungsi aktivasi untuk proses prediksi telah diberikan kepada aplikasi, proses prediksi akan dimulai dan menghasilkan grafik prediksi seperti yang ditunjukkan oleh gambar 4.2. Pada grafik akan ditampilkan indeks kualitas air yang diperkirakan, indeks kualitas air yang terukur dari hasil pengukuran, batas kualitas air sangat baik, batas kualitas air baik, dan batas kualitas air cukup.



**Gambar 4.2.** Grafik hasil perkiraan kualitas air

## Implementasi data

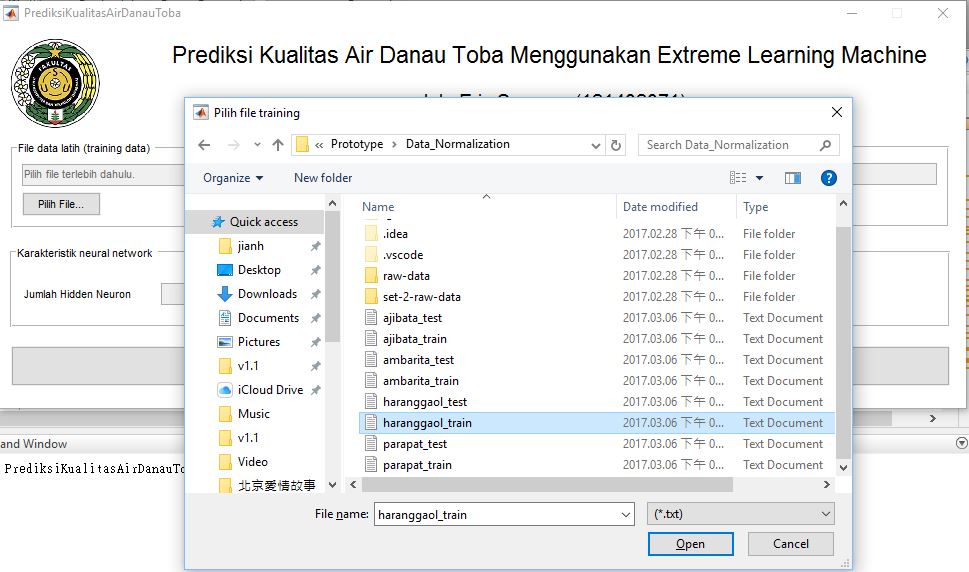
Data yang digunakan dalam penelitian ini bersumber dari penelitian yang dilakukan oleh Rahmat *et al.* (2016), di mana parameter-parameter kualitas air diukur dalam kurun waktu yang sedemikian rupa, dan disimpan dalam format dokumen teks, dengan hasil pengukuran setiap parameter dipisahkan oleh tanda titik koma. Pengukuran dilakukan pada beberapa lokasi, dan dikumpulkan dalam beberapa *file*. *File* tersebut akan diolah terlebih dahulu dalam tahap *preprocessing* sehingga dihasilkan data latih dan data uji yang dapat digunakan oleh *extreme learning machine*. Rincian dari *file* yang digunakan dalam penelitian ini ditunjukkan oleh Tabel 4.1.

**Tabel 4.1.** Rincian data yang digunakan dalam penelitian

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Nama *file*** | **Lokasi** | **Jumlah baris data** | | | |
| **Awal** | **Setelah penyaringan** | **Latih** | **Uji** |
| 1 | DATA ajibata.txt | Ajibata | 2203 | 2112 | 1268 | 844 |
| 2 | DATA Haranggaol.txt | Haranggaol | 6374 | 3532 | 2120 | 1412 |
| 3 | DATA parapat.txt | Parapat | 2446 | 1452 | 872 | 580 |
| 4 | DATA parapat resume.txt |
| 5 | DATA samosir.txt | Ambarita | 6129 | 3113 | 1869 | 1244 |
| 6 | DATA samosir resume.txt |

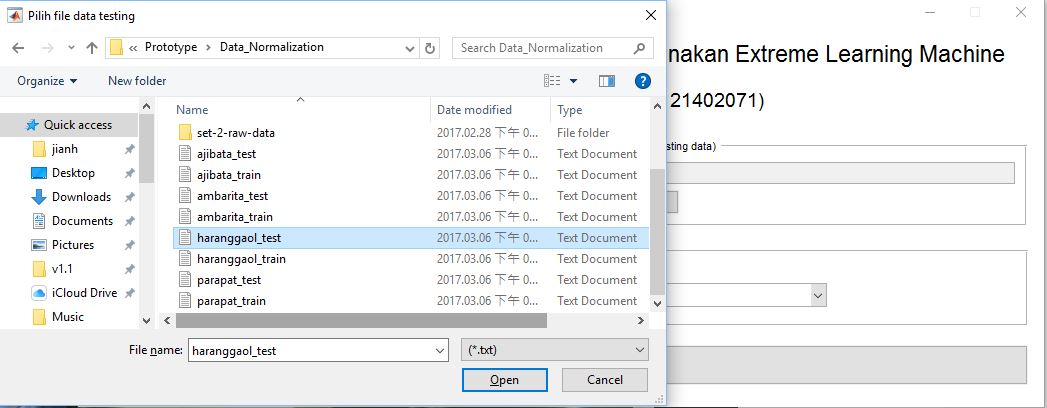
# Prosedur Operasional

Bagian ini akan memaparkan prosedur yang harus dilakukan dalam menggunakan aplikasi yang dikembangkan dalam penelitian ini. Setelah antarmuka tampil di layar, pengguna dapat memilih *file* data latih (*training data*) dengan memilih tombol “Pilih” pada bagian “File data latih (training data)”. Tombol “Pilih” akan mengarahkan pengguna pada dialog pemilihan *file* data latih, di mana *file* yang dapat dipilih adalah dokumen teks dengan ekstensi “txt”. Ilustrasi dari tahap pemilihan *file* data latih ditunjukkan oleh Gambar 4.3.



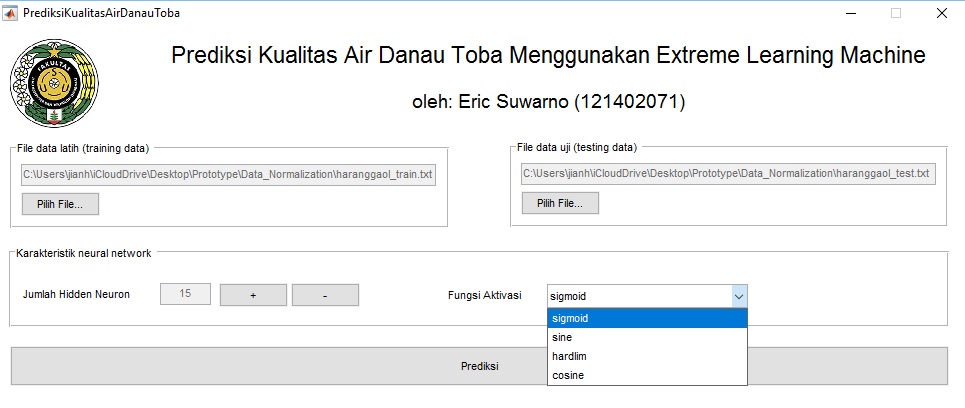
**Gambar 4.3.** Ilustrasi pemilihan *file* data latih (*training data*)

*File* data uji (*testing data*) juga dapat diberikan kepada aplikasi dengan menekan tombol “Pilih” pada bagian “File data uji (training data)”. Dialog pemilihan *file* data uji akan muncul setelah tombol “Pilih” ditekan, seperti yang ditunjukkan pada Gambar 4.4. Data uji yang dapat diterima dalam aplikasi ini berupa dokumen teks dengan ekstensi “txt”.



**Gambar 4.4.** Ilustrasi pemilihan *file* data uji (*testing data*)

Jumlah neuron pada *hidden layer* dapat diatur melalui tombol plus dan minus pada bagian karakteristik *neural network*. Sedangkan, fungsi aktivasi yang akan digunakan oleh *neural network* dapat dipilih melalui *dropdown box* pada bagian karakteristik neural network. Ilustrasi pengaturan fungsi aktivasi dan jumlah neuron pada *hidden layer* ditunjukkan oleh Gambar 4.5.

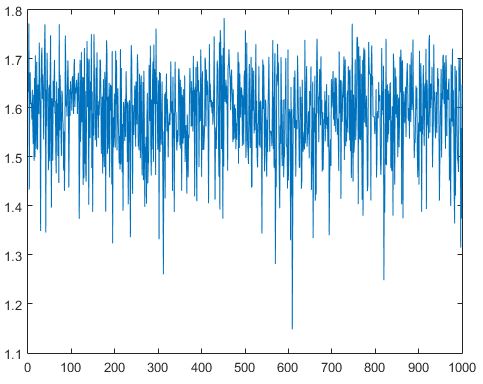


**Gambar 4.5.** Ilustrasi pengaturan karakteristik *neural network*

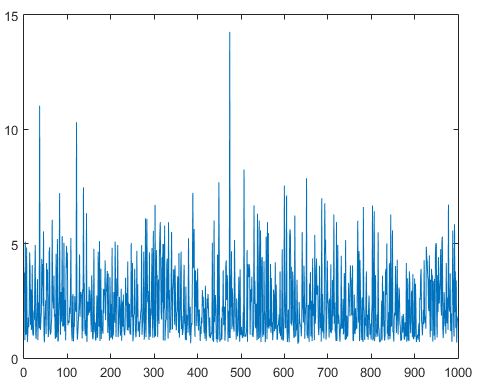
# Hasil Pengujian

Bagian ini akan memaparkan hasil yang didapatkan dari implementasi *Extreme Learning Machine* (ELM) dalam melakukan proses prediksi kualitas air Danau Toba. Proses pengujian akan dilakukan terhadap data uji setelah proses pelatihan telah selesai dilakukan melalui *extreme learning machine* menggunakan data latih yang disediakan. Akurasi dari hasil pengujian yang didapat akan direpresentasikan melalui nilai *root mean square error* (RMSE).

Hasil penelitian menunjukkan bahwa tingkat akurasi yang dihasilkan, yang ditunjukkan oleh nilai RMSE yang didapat, sangat dipengaruhi oleh besaran nilai *input weight* dan *bias* yang dihasilkan. Hal ini disebabkan karena *input weight* dan *bias* pada *extreme learning machine* ditentukan secara acak. Karena itu, tingkat akurasi yang dihasilkan dari proses pengujian juga bervariasi. Grafik tingkat akurasi yang dihasilkan oleh *extreme learning machine* dalam proses pelatihan dan pengujian ditunjukkan oleh Gambar 4.6 dan Gambar 4.7. Pada percobaan ini, *extreme learning machine* dijalankan dengan menggunakan 80 buah *hidden neuron* dan fungsi *hardlim* sebagai fungsi aktivasi. Data yang digunakan dalam percobaan ini adalah data pengukuran di Haranggaol. Hasil yang ditunjukkan oleh Gambar 4.6 dan Gambar 4.7 didapat dari perulangan proses pelatihan sebanyak 1.000 kali.



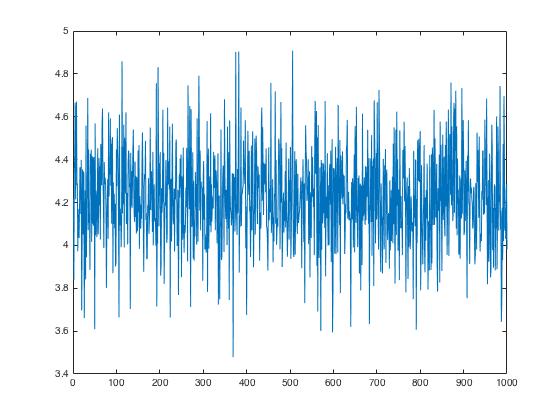
**Gambar 4.6.** Grafik akurasi pelatihan *extreme learning machine*



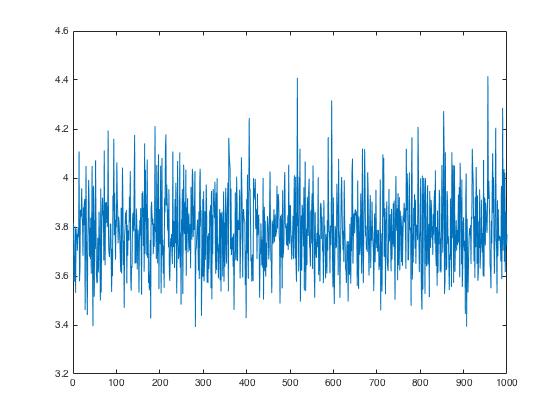
**Gambar 4.7.** Grafik akurasi pengujian *extreme learning machine*

Hasil prediksi juga sangat ditentukan oleh fungsi aktivasi yang digunakan oleh setiap neuron dalam *artificial neural network* pada proses pelatihan. Fungsi aktivasi yang digunakan pada *artificial neural network* mempengaruhi jumlah neuron yang terdapat pada *hidden layer*.

Dari penelitian yang dilakukan, diketahui bahwa *artificial neural network* yang menggunakan fungsi sigmoid dan *sine* sebagai fungsi aktivasi membutuhkan setidaknya 10 hingga 15 neuron pada *hidden layer* agar dapat memberikan hasil prediksi dengan akurasi yang lebih tinggi. Perbandingan dari akurasi prediksi menggunakan *extreme learning machine* pada *artificial neural network* dengan fungsi aktivasi sigmoid ditunjukkan oleh Gambar 4.8.



(a)



(b)

**Gambar 4.8.** Perbandingan grafik akurasi pelatihan *extreme learning machine* menggunakan fungsi aktivasi sigmoid dengan (a) 12 dan (b) 24 *hidden neuron*

Hasil dari penelitian yang dilakukan juga menunjukkan bahwa *artificial neural network* yang menggunakan fungsi *hardlim* sebagai fungsi aktivasi membutuhkan setidaknya 75 hingga 100 neuron pada *hidden layer* agar dapat memberikan hasil prediksi dengan akurasi yang lebih tinggi. Besar dari nilai akurasi yang didapat dari proses prediksi dapat diketahui dari nilai RMSE yang didapat.