

# **BAB I**

## **PENDAHULUAN**

### **1.1. Latar Belakang**

Perairan Danau Toba merupakan perairan dengan kualitas air tercemar, terutama di sekitar wilayah Kecamatan Haranggaol Horison, Kabupaten Simalungun, Provinsi Sumatera Utara (Haro *et al.*, 2013). Tingkat pencemaran di perairan Danau Toba bervariasi dari ringan hingga sedang.

Pencemaran di perairan Danau Toba tidak hanya bersumber dari aktivitas rumah tangga, namun juga aktivitas lainnya, seperti tambak ikan, peternakan, dan industri. Selain itu, populasi eceng gondok dan limbah dari sungai yang bermuara ke Danau Toba juga menjadi sumber pencemaran pada perairan Danau Toba.

Pengukuran tingkat pencemaran pada perairan Danau Toba selama ini dilakukan dengan mengambil sampel air pada titik tertentu di perairan Danau Toba. Sampel ini kemudian akan dikirim ke laboratorium untuk diuji, dan didapat hasil pengujian sampel tersebut. Proses pengukuran ini memerlukan waktu yang relatif lama, terutama dalam proses pengambilan sampel dan pengujian pada laboratorium. Karena itu, dibutuhkan sebuah cara yang sesuai untuk mengatasi kendala tersebut melalui kecanggihan teknologi informasi dan komputer.

Perkembangan teknologi informasi dan komputer memungkinkan pengukuran kualitas air melalui peralatan seperti Arduino dan Raspberry Pi, dengan memanfaatkan sensor-sensor seperti sensor *dissolved oxygen* (DO), sensor keasaman, dan sensor suhu. Hal ini memungkinkan peneliti untuk merancang sebuah alat yang dapat digunakan untuk melakukan pengukuran secara berkelanjutan.

Selain perangkat keras tersebut, dibutuhkan juga sebuah metode yang dapat digunakan untuk melakukan klasifikasi data secara berkelanjutan, salah satunya dengan menggunakan jaringan saraf tiruan, atau *artificial neural networks* (Kasabov, 2007). *Artificial neural network* telah digunakan dalam memantau proses pemasangan baut secara otomatis (Lara *et al.*, 1999), pengaturan keseimbangan tegangan sistem listrik (Popovic *et al.*, 1998), dan sistem pembangkit listrik tenaga angin (Ata, 2015).

Permasalahan yang dihadapi dalam penggunaan *artificial neural network* adalah waktu yang digunakan dalam pemrosesan data, terutama dalam proses *learning*. Penelitian untuk meningkatkan kecepatan pemrosesan data dalam *artificial neural network* dimulai oleh Werbos (1974) dan Rumelhart *et al.* (1986), dengan menggunakan algoritma *back-propagation*, di mana nilai *error* akan dihitung berdasarkan data pada setiap node dan fungsi aktivasi.

Walaupun terjadi perbaikan pada waktu komputasi, algoritma ini memiliki kelemahan dalam memproses data dengan jumlah besar (Deng *et al.*, 2015). Chandra & Sharma mengembangkan *multilayer perceptron* dengan parameter yang diatur dengan fungsi trigonometri (2014) dan menerapkan parameter fungsi trigonometri pada *deep neural network* (2016). Hinton dan Teh (2006) memperkenalkan algoritma *learning* yang dapat digunakan untuk mempercepat waktu komputasi dalam *deep belief nets*.

Salah satu algoritma yang dapat digunakan untuk mempercepat waktu komputasi pada *artificial neural network* adalah *extreme learning machine* (ELM). *Extreme learning machine* dikemukakan oleh Huang *et al.* (2006) pada *single hidden layer feedforward neural network*. *Extreme learning machine* memiliki kemampuan untuk mengurangi waktu komputasi pada *artificial neural networks* secara signifikan.

*Extreme learning machine* telah digunakan dalam beberapa penelitian. Huang *et al.* (2013) menggunakan *extreme learning machine* untuk mendeteksi tumor liver. Fu *et al.* (2015) menggunakan *extreme learning machine* untuk melakukan prediksi pergerakan rotasi kapal. Pangaribuan & Suharjito (2014) menggunakan *extreme learning machine* untuk diagnosis diabetes mellitus. Zhai & Du (2008) menggunakan *extreme learning machine* untuk identifikasi spesies tanaman. Huang *et al.* (2016) menggabungkan *extreme learning machine* dan histogram gradien warna terorientasi (*histogram of oriented gradients/HOG*) dalam pengenalan rambu lalu lintas.

Berdasarkan latar belakang di atas, penulis mengajukan proposal penelitian dengan judul “**Klasifikasi Kualitas Air Danau Toba Menggunakan *Extreme Learning Machine***”. Hasil yang diharapkan dari penelitian ini adalah perkiraan kualitas air yang diukur di perairan Danau Toba.

## **1.2. Rumusan Masalah**

Penurunan kualitas air di perairan Danau Toba, telah mempengaruhi industri pariwisata di Danau Toba. Untuk mengantisipasi hal tersebut, diperlukan sebuah metode untuk melakukan klasifikasi kualitas air di perairan Danau Toba, agar dapat dimanfaatkan oleh pihak-pihak yang terkait untuk mengetahui tingkat pencemaran air di perairan Danau Toba.

## **1.3. Batasan Penelitian**

Untuk menghindari penyimpangan dan perluasan yang tidak diperlukan, penulis membuat batasan sebagai berikut:

1. Klasifikasi dilakukan berdasarkan data pengukuran kualitas air di perairan Danau Toba, di mana pengukuran dilakukan di tiga titik, yaitu Parapat, Ajibata, dan Haranggaol;
2. Analisis dilakukan berdasarkan pengukuran kadar *dissolved oxygen*, temperatur air, dan tingkat keasaman (*pH*); dan
3. *Extreme learning machine* dijalankan dalam *single hidden layer feedforward neural networks*.

## **1.4. Tujuan Penelitian**

Penelitian ini bertujuan untuk menganalisis tingkat pencemaran yang terjadi di perairan Danau Toba melalui hasil pengukuran kadar kualitas air, sehingga dapat digunakan sebagai alat peringatan dini kualitas air di perairan Danau Toba.

### 1.5. Manfaat Penelitian

Manfaat yang diperoleh dari penelitian ini adalah sebagai berikut:

1. Melakukan analisis terhadap kualitas air di Danau Toba, dengan menggunakan *extreme Learning Machine* dan memaparkan hasilnya sesuai waktu pengukuran;
2. Melakukan klasifikasi terhadap tingkat kualitas air di Danau Toba, dengan menggunakan metodologi yang dilakukan, agar dapat memberikan notifikasi mengenai kualitas air Danau Toba kepada pemerintah setempat, agar dapat melakukan antisipasi sesuai tingkat kualitas air yang didapat; dan
3. Menambah wawasan mengenai penerapan *artificial neural networks*, khususnya *extreme learning machines*, dalam proses klasifikasi.

### 1.6. Metodologi Penelitian

Terdapat beberapa tahapan dalam penelitian ini untuk menghasilkan suatu sistem yang sesuai dengan yang diharapkan, yaitu:

#### 1. Studi Literatur

Dalam kegiatan ini, peneliti mempelajari dokumen yang berkaitan dengan literatur dan teori sebagai referensi dalam penelitian, di mana literatur tersebut berhubungan dengan penelitian yang dilakukan. Literatur tersebut dapat berupa buku, jurnal, tesis, makalah, serta sumber lainnya yang diperoleh dari internet. Referensi yang dikumpulkan berkaitan dengan *artificial neural network*, *extreme learning machine*, serta data kualitas air di Danau Toba.

#### 2. Pengambilan Data

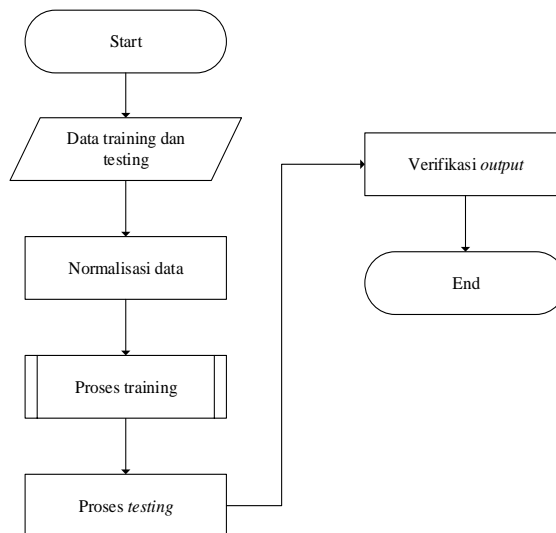
Setelah studi literatur dilakukan, penelitian akan dilanjutkan dengan pengambilan data. Data yang digunakan adalah data yang diperoleh dari penelitian yang dilakukan oleh Rahmat *et al.* (2016), yang dilakukan di perairan Danau Toba. Pengukuran dilakukan di tiga titik, yaitu di perairan Danau Toba sekitar Ajibata, Parapat, dan Haranggaol.

### 3. Analisis Permasalahan

Dalam tahap ini, dilakukan analisis terhadap data yang telah dikumpulkan. Hal ini bertujuan untuk mengetahui parameter-parameter yang dibutuhkan untuk kalkulasi kualitas air.

### 4. Pembangunan Program

Dalam tahap ini, dilakukan perancangan sistem yang dapat digunakan untuk melakukan pemantauan kualitas air di perairan Danau Toba. Proses dari program yang akan dirancang dapat dilihat pada Gambar 1.1.



**Gambar 1.1** Gambaran umum sistem

Masukan yang diterima pada sistem ini berupa data *training* dan data *testing*. Data yang diterima merupakan hasil pengukuran *dissolved oxygen* (oksigen terlarut), suhu permukaan, suhu air, tingkat keasaman, konduktivitas listrik, dan tingkat reduksi oksidasi air. Parameter biologis, seperti kandungan bakteri dan alga dalam air, tidak digunakan dalam penelitian ini. Hal ini dikarenakan pengujian parameter biologis harus menggunakan uji lab, sehingga tidak memungkinkan untuk dilakukan pengukuran secara langsung.

Setelah masukan diterima oleh sistem, data masukan akan diproses oleh *neural network*. Proses yang akan dilakukan adalah sebagai berikut:

a. Normalisasi data

Normalisasi akan dilakukan pada data *training* dan *testing* yang diterima oleh sistem. Hal ini bertujuan untuk mengurangi peluang terjadinya kesalahan penghitungan oleh jaringan.

b. Penentuan jumlah node pada *hidden layer*

Menurut Heaton (2008), penentuan jumlah node pada *artificial neural network*, khususnya pada *hidden layer*, merupakan hal yang penting sebelum menjalankan *training*, karena *hidden layer* berperan penting dalam penghitungan hasil akhir dari *artificial neural network*.

Jumlah node pada *hidden layer* yang tidak optimal dapat menyebabkan permasalahan tertentu pada proses *training*. *Hidden layer* yang memiliki jumlah node terlalu sedikit dapat menyebabkan kondisi *underfitting*, di mana node yang tersedia tidak dapat bekerja secara maksimal untuk mendeteksi sinyal yang diterima dari *input layer*. Sebaliknya, jumlah node yang terlalu banyak dapat berakibat pada bertambahnya waktu yang dibutuhkan *artificial neural network* untuk memproses data. Selain itu, jumlah node yang terlalu banyak juga dapat berakibat pada kondisi *overfitting*, di mana jumlah informasi yang diterima tidak cukup untuk diproses dalam *training* karena banyaknya kapasitas pemrosesan informasi yang dimiliki jaringan. Penentuan jumlah neuron pada *hidden layer* dapat memenuhi aturan-aturan berikut:

- a. Jumlah neuron pada *hidden layer* harus melebihi jumlah neuron di *input layer* dan tidak boleh melebihi jumlah neuron di *output layer*;
- b. Jumlah neuron pada *hidden layer* harus setara dengan  $\frac{2}{3}$  (dua per tiga) dari jumlah neuron pada *input layer* dan *output layer*; dan
- c. Jumlah neuron pada *hidden layer* tidak boleh melebihi dua kali jumlah neuron pada *input layer*.

Aturan mengenai jumlah neuron pada *hidden layer* ini dapat digunakan sebagai pertimbangan. Walaupun begitu, proses menentukan jumlah neuron pada

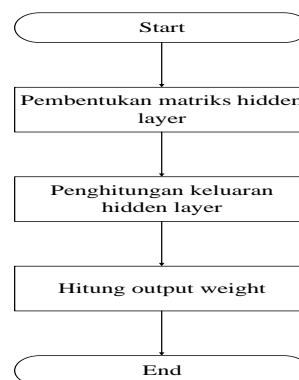
*hidden layer* merupakan proses *trial and error*. Hal ini dilakukan agar jaringan dapat menyesuaikan diri dengan masalah yang akan diselesaikan.

Dalam penelitian ini, *extreme learning machines* akan dijalankan dalam *single hidden layer feedforward neural network*, dengan *input layer* yang memiliki 7 neuron, dan *output layer* yang memiliki 2 neuron. Dengan mengacu pada ketentuan kedua, maka jumlah neuron pada *hidden layer* yang diperlukan dalam penelitian ini adalah 4 neuron.

c. Penentuan fungsi aktivasi

Setelah menentukan jumlah neuron pada *hidden layer*, langkah selanjutnya adalah menentukan fungsi aktivasi yang akan digunakan oleh neuron pada jaringan. Pada penelitian ini, fungsi aktivasi sigmoid, sine, dan hardlim akan digunakan sebagai perbandingan hasil kalkulasi.

d. Proses *training*



**Gambar 1.2.** Uraian proses *training*

Proses *training* merupakan tahap awal dari penggunaan *artificial neural networks*. Pada proses *training*, jaringan akan melakukan pembelajaran terhadap data *training* yang telah dinormalisasi pada tahap normalisasi data.

Proses *training*, sesuai dengan gambar 1.2, terdiri dari beberapa tahap, yaitu pembentukan matriks *hidden layer*, penghitungan keluaran *hidden layer*, dan penghitungan *output weight*. Penjelasan dari proses *training* dapat dijelaskan sebagai berikut:

1. Pembentukan matriks *hidden layer*

Matriks *hidden layer* pada sistem ini merupakan kumpulan dari nilai *weight* yang diterima tiap neuron pada *hidden layer* dari *input layer*, dan bias yang dimiliki oleh tiap *hidden neuron*. Matriks *hidden layer* ini akan disusun secara acak.

2. Penghitungan keluaran *hidden layer*

Setelah matriks *hidden layer* dibentuk, proses kalkulasi keluaran dari setiap *weight* dan bias yang tergabung dalam matriks *hidden layer* dilakukan menggunakan fungsi aktivasi. Dalam penelitian ini, fungsi aktivasi yang digunakan adalah fungsi sigmoid, sinus, dan *hardlim*. Hasil dari proses training yang dilakukan akan digunakan sebagai perbandingan mengenai akurasi kalkulasi oleh jaringan.

3. Hitung *output weight*

Setelah penghitungan keluaran *hidden layer* dilakukan, *output weight* dari jaringan dihitung menggunakan persamaan 1.1:

$$\beta = H^+T \quad (1.1)$$

di mana  $H^+$  merupakan hasil invers Moore-Penrose dari matriks  $H$ , dan  $T$  merupakan transpos dari himpunan  $[t_1, t_2, \dots, t_N]$ .

e. Proses *testing*

Proses *testing* merupakan tahap yang dilakukan setelah proses *training* selesai dilakukan. Proses *testing* dilakukan sebagai percobaan terhadap hasil *training* yang telah dihitung oleh jaringan.

f. Verifikasi

Verifikasi merupakan langkah yang dilakukan setelah hasil dari proses *testing* telah didapat. Verifikasi dilakukan untuk memastikan bahwa hasil yang diberikan sesuai dengan hasil pengukuran secara manual di lapangan.

Keluaran yang dihasilkan dari sistem ini berupa grafik pencemaran air yang telah dihitung melalui *extreme learning machine*, di mana hasil yang didapat merupakan nilai indeks kualitas air (*water quality index*, WQI).



5. Uji Coba Sistem

Dalam tahap ini, dilakukan uji coba terhadap sistem yang telah dibangun. Hal ini dilakukan untuk mengetahui apakah sistem sudah bekerja sesuai dengan fungsi yang diharapkan.

6. Dokumentasi dan Penyusunan Laporan

Dalam tahap ini, dilakukan dokumentasi dan menyusun laporan akhir dari analisis yang telah dilakukan mengenai penggunaan *extreme learning machine* dalam melakukan pemantauan kualitas air dan tingkat pencemaran air di perairan Danau Toba.