

BAB I

PENDAHULUAN

1.1 Latar Belakang

Indonesia termasuk ke dalam negara dengan iklim tropis dengan curah hujan yang tinggi. Tingginya curah hujan bisa berdampak baik kepada masyarakat, tetapi juga dapat membahayakan keselamatan masyarakat juga. Semakin sempitnya lahan terbuka yang tersedia dapat menyebabkan bencana banjir. Banjir adalah meluapnya permukaan air sungai atau aliran air lainnya, yang disebabkan ketidakmampuan menampung lagi aliran air. Jika bencana banjir terus berlanjut ketinggiannya, maka dapat memberikan kerugian yang cukup parah. Dampak dari terjadinya banjir bisa menyebabkan masalah pada kesehatan, ekonomi, hingga menimbulkan korban jiwa [1]. Selain banjir, gempa bumi juga merupakan bencana alam yang sering melanda Indonesia. Hal tersebut karena Indonesia terletak pada pertemuan lempeng tektonik dengan jajaran gunung api aktif. Dalam kurun waktu dari tahun 2008 hingga 2018 tercatat telah terjadi 134 bencana gempa bumi. Deretan bencana gempa bumi tersebut memiliki besaran magnitudo yang berbeda-beda, 11 April 2012 di Simeuleu memiliki kekuatan magnitudo 8,4 SR pada kedalaman 12,93 kilometer. Dengan besaran magnitudo dan kedalaman tersebut, gempa tersebut mengakibatkan tsunami yang menghantam Meulaboh, Sabang, dan Lahewa [2].

Mitigasi bencana adalah upaya yang dilakukan untuk mengurangi dampak dari bencana alam. Sistem mitigasi bencana termasuk ke dalam tahap awal penanggulangan bencana [3]. Tahapan penanggulangan bencana terdiri dari tiga tahapan [4], tahap awal bencana, tahap saat bencana, dan tahap setelah terjadinya bencana. Sistem mitigasi bencana sudah mulai memanfaatkan teknologi komunikasi seperti *Internet of Things* (IoT). Dengan menggunakan teknologi IoT, pengamat hanya cukup memasang sensor dan dapat dimonitoring dari jarak jauh. Dalam beberapa menit kemudian didapatkan data sensor yang dapat digunakan sebagai parameter dalam proses pengolahan data. Dalam metode pengolahan data sensor untuk peringatan bencana, penelitian sebelumnya ada yang menggunakan *fuzzy logic* dalam proses identifikasi atau klasifikasi data [5 -10]. Kelebihan dari metode *fuzzy logic* adalah *output* data yang linguistik dan dapat digunakan untuk

memodelkan fungsi-fungsi *non-linear* yang kompleks. Meskipun begitu, *fuzzy logic* tidak dapat untuk mengolah data sekuensial dengan jumlah yang banyak. Selain *fuzzy logic*, ada juga yang menggunakan metode *Artificial Neural Network* (ANN), yang bisa disebut juga dengan Jaringan Syaraf Tiruan (JST). Metode ANN, dalam variannya, mampu digunakan untuk sistem prediksi, karena dapat melakukan proses klasifikasi dan regresi [5]. Metode ANN memiliki berbagai varian dalam perkembangannya, seperti *Long Short-Term Memory* (LSTM), *Hopfield Network* (HN), *Convolutional Neural Network* (CNN), *Support Vector Machine* (SVM), dan lain-lain. Terdapat beberapa penelitian yang menggunakan ANN dengan variannya untuk membentuk sistem peringatan dini banjir atau gempa bumi. Putri dan Setiawan (2017) [6] membangun ANN dengan algoritma pembelajaran *Backpropagation* untuk mendeteksi banjir dan menghasilkan nilai *Mean Square Error* (MSE) 0,9. Azizulhaq (2021) [7] membangun ANN dengan varian *Radial Basis Function* (RBF) untuk memprediksi banjir dan dihasilkan nilai *Mean Absolute Error* (MAE) 0,048. Al-Ayubi (2020) [8] membuat sistem estimasi gempa bumi menggunakan ANN dengan algoritma pembelajaran *Backpropagation*, dalam sistemnya menggunakan sinyal gelombang P dan menghasilkan MSE 0,369. Pranesthi (2020) [9] membuat sistem peringatan dini gempa bumi berdasarkan sinyal geomagnetik dan analisa pola waktu musim kemarau, menggunakan ANN dengan algoritma pembelajaran *Backpropagation* dan menghasilkan tingkat akurasi 85%. Elrizki (2020) [10] membuat sistem peringatan dini gempa bumi berdasarkan sinyal geomagnetik dan analisa pola waktu musim kemarau, menggunakan ANN dengan varian RBF menghasilkan tingkat akurasi 85%.

Pada penelitian ini, pengolahan data sensor menggunakan model *neural network* dalam membangun program prediksi banjir, deteksi banjir, dan deteksi gempa bumi. Model *neural network* dipilih karena model *neural network* memiliki sifat *self-learning*. Sifat *self-learning* ini dibutuhkan karena model pengolahan data yang sudah dibangun bisa saja mengalami penurunan performa akurasi akibat perubahan data lapangan. Meskipun model *neural network* memiliki sifat *self-learning*, akurasi hasil dari model *neural network* tetap bagus. Untuk membangun program prediksi banjir menggunakan model *Recurrent Neural Network* (RNN). Selain model RNN, dalam penelitian ini juga menggunakan model *Long Short-*

Term Memory (LSTM) yang merupakan jenis dari RNN sebagai pembanding model RNN. Hal tersebut karena model RNN memiliki keterbatasan pada jumlah data optimum yang dapat diproses. Model LSTM dapat mengatasi hal tersebut, tetapi tidak ada jumlah pasti data yang tidak bisa dimasukkan pada model RNN. Sehingga model LSTM di sini sebagai pembanding model RNN untuk melihat performa model RNN dalam memprediksi. Dalam model RNN terdapat proses umpan balik untuk memasukkan nilai unit tersembunyi dari data sebelumnya. Dengan begitu, RNN sering digunakan pada pengolahan data *time-series* yang datanya mengikuti pola urutan waktu [11]. Sedangkan untuk membangun program deteksi banjir dan gempa bumi menggunakan model *Artificial Neural Network* (ANN). Model ANN adalah bentuk model paling sederhana pada model *neural network*. Program prediksi banjir dibentuk dengan regresi tinggi air dengan model RNN kemudian diklasifikasi bersamaan parameter lain dengan model ANN. Program deteksi banjir dan gempa bumi dibentuk dengan klasifikasi banjir dan gempa bumi dengan model ANN. Dalam penelitian ini juga menggunakan inisialisasi Nguyen-Widrow. Nguyen-Widrow merupakan algoritma inisialisasi untuk mencari nilai awal bobot dan bias di dalam model *neural network*, sehingga dapat mempercepat proses pelatihan model. Proses penerapan model ANN dan RNN ke aplikasi dengan menghubungkan model dan aplikasi menggunakan API. Dengan begitu, dapat dibangun aplikasi untuk memberikan informasi penting berkaitan bencana banjir dan gempa bumi.

1.2 Rumusan Masalah

Dari latar belakang yang sudah dikemukakan pada paragraf di atas, berikut rumusan masalah yang didapat:

1. Bagaimana rancangan dan evaluasi model RNN dan LSTM dalam regresi tinggi air dan model ANN dalam klasifikasi banjir dan gempa bumi dengan inisialisasi Nguyen-Widrow?
2. Bagaimana implementasi model ANN dan RNN yang sudah dirancang ke dalam aplikasi monitoring banjir dan gempa bumi?
3. Bagaimana analisis kelayakan pengimplementasian model ANN dan RNN ke dalam aplikasi monitoring banjir dan gempa bumi?

1.3 Tujuan Penelitian

1. Merancang dan mengevaluasi model RNN dan LSTM dalam regresi tinggi air dan model ANN dalam klasifikasi banjir dan gempa bumi dengan inisialisasi Nguyen-Widrow.
2. Mengimplementasikan model ANN dan RNN yang sudah dirancang ke dalam aplikasi monitoring banjir dan gempa bumi.
3. Menganalisis kelayakan pengimplementasian model ANN dan RNN ke dalam aplikasi monitoring banjir dan gempa bumi.

1.4 Batasan Masalah

Dalam penelitian ini terdapat beberapa batasan dari penelitian ini yaitu:

1. Fokus pada penelitian ini yaitu proses pengolahan data hasil monitoring parameter bencana banjir dan gempa bumi.
2. Model yang digunakan yaitu model *neural network*.
3. Model *neural network* yang sudah dilatih diimplementasikan ke dalam aplikasi.
4. Algoritma inisialisasi bobot dan bias yang diuji cobakan adalah algoritma Nguyen-Widrow.

1.5 Metode Penelitian

Model ANN, RNN, dan LSTM dibentuk dan diuji di dalam aplikasi Google Colab. Proses pembentukan model ANN, RNN, dan LSTM dibangun melalui *trial and error*. Proses pengujian pada model ANN dilakukan dengan melakukan beberapa kali percobaan dengan parameter yang sama. Proses pengujian pada model RNN dan LSTM dilakukan dengan membandingkan hasil prediksi tinggi air dari kedua model tersebut. Hasil dari proses pengujian untuk melihat kemampuan dari model *neural network* tersebut. Setelah model diuji, kemudian model diimplementasikan ke dalam aplikasi. Selanjutnya dianalisis kelayakan model dalam pengimplementasian ke dalam aplikasi.