

Prediksi Penderita Tuberkulosis Dengan Algoritma *Long Short-Term Memory*

Prediction Of Tuberculosis Using Long Short-Term Memory (LSTM) Algorithm

1st Diah Aisyah
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

diahaisyah@student.telkomuniversity.ac.id

2nd Tito Waluyo Purboyo
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

titowaluyo@telkomuniversity.ac.id

3rd Meta Kallista
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

metakallista@telkomuniversity.ac.id

Abstrak— Tuberkulosis (TB) merupakan penyakit menular yang disebabkan oleh kuman *Mycobacterium tuberculosis*, meskipun dapat menyerang organ apapun didalam tubuh. Ketika bakteri masuk melalui droplet di udara. Pada tahun 2015 terdapat 2.617 kasus tuberkulosis di Kabupaten Karawang. Kurangnya pengetahuan terhadap gejala penyakit dan sosialisasi terhadap pemeriksaan tuberkulosis secara dini merupakan faktor utama penyebab tingginya jumlah kasus penderita TB. Oleh karena itu perlu adanya sebuah sistem prediksi jumlah penderita tuberkulosis untuk memprediksi jumlah penderita dimasa yang akan datang. Dalam Penelitian ini menggunakan metode *Long Short-Term Memory*. (LSTM). LSTM merupakan evolusi dari algoritma *Recurrent Neural Network* (RNN) untuk mengatasi permasalahan pada RNN dalam mengelola data untuk periode yang lama. LSTM dianggap lebih unggul dibandingkan algoritma lainnya dalam mengelola data yang bersifat *time series*. Data yang penulis gunakan berasal dari Dinas Kesehatan Kabupaten Karawang dari tanggal 1 Januari 2020 hingga 31 Desember 2021. Berdasarkan Hasil Pengujian diketahui bahwa algoritma LSTM dengan partisi data 70%:30%, *epoch* sebesar 900, jumlah *hidden layer* sebanyak 1 layer LSTM dengan 64 neuron serta tipe optimasi Adam dengan *learning rate* sebesar 0.001. selain itu, hasil pengujian parameter terbaik menghasilkan nilai RMSE = 0.12019341, MAE = 0.0819 Dan R2 = 0.53508423.

Kata kunci— *tuberculosis, prediksi, long short-term memory*

Abstract— Tuberculosis (TB) is an infectious disease caused by the bacteria *Mycobacterium tuberculosis*, although it can attack any organ in the body. When bacteria enter through droplets in the air. In 2015 there were 2,617 cases of tuberculosis in Karawang Regency. Lack of knowledge of the symptoms of the disease and socialization of early tuberculosis screening are the main factors causing the high number of cases of TB patients. Therefore, it is necessary to have a prediction system for the number of tuberculosis patients to predict the number of patients in the future. In this study using the *Long Short-Term Memory* method. (LSTM). LSTM is an evolution of the *Recurrent Neural Network* (RNN) algorithm to overcome problems in the RNN in managing data for a long period. LSTM is considered superior to other algorithms in managing *time series* data. The data that the author uses comes from the Karawang District Health Office from January 1, 2020 to December 31, 2021. Based on the test results, it is known that the LSTM algorithm with 70%:30% data partition, 900 epochs, the number of hidden layers is 1 LSTM layer with 64 neurons and the Adam optimization type with a learning rate of 0.001. In addition, the results of the best

parameter testing resulted in the value of RMSE = 0.12019341, MAE = 0.0819 and R2 = 0.53508423.

Keywords— *tuberculosis, prediction, long short-term memory*

I. PENDAHULUAN

Tuberkulosis (TB) adalah penyakit menular yang disebabkan oleh kuman *Mycobacterium tuberculosis*, meskipun dapat menyerang organ apapun didalam tubuh. Ketika bakteri masuk melalui droplet di udara. Terjadinya infeksi Tuberkulosis bisa berakibat fatal, tetapi banyak kasus dapat dicegah dan diobati[1]. Indonesia berada pada peringkat ke-2 dengan penderita TB tertinggi di Dunia setelah India. Secara global, diperkirakan 10 juta orang menderita TB pada tahun 2019. Meskipun terjadi penurunan kasus baru TB, tetapi tidak cukup cepat untuk mencapai target Strategi END TB tahun 2020, yaitu pengurangan kasus TB sebesar 20% antara tahun 2015 – 2020. Pada tahun 2015 – 2019 penurunan kumulatif kasus TB hanya sebesar 9% [2]

Kasus Tuberkulosis di Indonesia hampir merata di semua daerah, salah satunya di Kabupaten Karawang. Pada tahun 2015 terdapat 2.617 kasus tuberkulosis di Kabupaten Karawang. Dua tahun kemudian, jumlah kasus turun menjadi 1.821 kasus, dan sayangnya pada 2018 jumlahnya meningkat lagi menjadi 2.075 kasus[3]

Dengan permasalahan di atas, maka perlu dilakukan penanganan infeksi TB secara dini sehingga dapat dilakukan tindakan pencegahan dini sebelum penyakit semakin parah. Salah satu cara untuk mengetahui jumlah penderita TB di masa depan adalah dengan memprediksi jumlah penderita TB dengan menggunakan Algoritma *Long-Short Term Memory*.

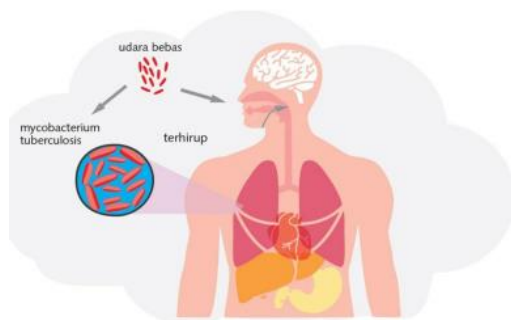
LSTM merupakan evolusi dari algoritma *Recurrent Neural Network* (RNN) untuk mengatasi permasalahan pada RNN dalam mengelola data untuk periode yang lama. LSTM dianggap lebih unggul dibandingkan algoritma lainnya dalam mengelola data yang bersifat *time series*.. LSTM menghasilkan lebih banyak proses sukses dan belajar cepat [4].

Jaringan LSTM cocok untuk belajar dari pengalaman untuk mengklasifikasikan, proses dan memprediksi data time series. Ketika rentan waktunya sangat lama dari ukuran yang diketahui dengan kejadian yang penting, ini adalah salah satu alasan utama mengapa LSTM menjadi alternatif dari RNNs dan Hidden Markov model dan metode pembelajaran urutan lainnya dalam berbagai aplikasi [5]

Diharapkan penelitian ini dapat menjadi solusi masyarakat dan membantu Dinas Kesehatan Karawang untuk lebih baik dalam mengatasi penyebaran penyakit TBC dan metode yang digunakan.

II. KAJIAN TEORI

A. Penyakit Tuberkulosis



GAMBAR 2.1
GAMBARAN PENULARAN TBC

Tuberkulosis (TB) adalah penyakit menular yang disebabkan oleh kuman *Mycobacterium tuberculosis*, meskipun dapat menyerang organ apa pun di dalam tubuh. Ketika bakteri masuk melalui droplet di udara. Terjadi infeksi Tuberkulosis bisa berakibat fatal, tetapi dalam banyak kasus dapat dicegah dan diobati [1].

Berdasarkan letak anatomi tuberkulosis dapat dibedakan menjadi 2 yaitu:

1. Tuberkulosis Paru adalah tuberkulosis yang mengenai parenkim paru. Pleura tidak termasuk sedangkan TB Milier diklasifikasikan sebagai Tuberkulosis paru karena berada dalam paru.
2. Tuberkulosis ekstra paru adalah tuberkulosis yang mengenai bagian tubuh lain selain paru [7].

Gejala utama pasien Tuberkulosis adalah batuk berdahak lebih dari 2 minggu. Batuk dapat diikuti dengan gejala tambahan yaitu dahak bercampur darah, batuk darah, sesak nafas, lemas, kehilangan nafsu makan, penurunan berat badan, malaise, berkeringat malam hari tanpa kegiatan fisik, demam meriang lebih dari sebulan. Gejala-gejala tersebut dapat dijumpai pula pada penyakit paru selain TB, seperti Bronkiektasi, bronkitis kronis, asma, kanker paru dan lain-lain, mengingat penderita Tuberkulosis di Indonesia saat ini masih tinggi, maka setiap orang yang datang dengan gejala tersebut diatas [7].

a. Pemeriksaan Tuberkulosis

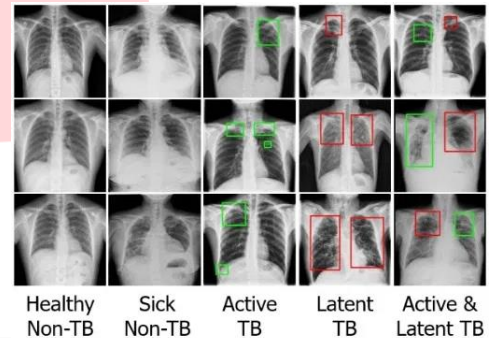
Pada penderita Tuberkulosis, terdapat beberapa macam pemeriksaan yang dilakukan yaitu :

a) Pemeriksaan Bakteriologi [21]

Pemeriksaan ini merupakan pemeriksaan dilakukan dengan melihat serta menghitung jumlah bakteri TBC yang berada pada tubuh pasien. Pada umumnya pemeriksaannya yang digunakan adalah berupa dahak, urin, feces dan jaringan biopsy dari pasien tersebut.

b) Pemeriksaan Radiologi [21]

Pemeriksaan radiologi adalah pemeriksaan toraks paru-paru pasien. Pemeriksaan ini memiliki tujuan untuk mengetahui kondisi bagian dalam tubuh. Gambaran mengenai TB paru aktif yaitu terdapat bayangan berawan atau noduler disegmen apical dan posterior lobus atas segmen superior lobus di bawah paru, terdapat opak berawan atau noduler.



GAMBAR 2.2
RONTGEN PASIEN TUBERKULOSIS

B. Machine Learning

Machine learning adalah seperangkat teknik yang membantu untuk memproses dan memprediksi data yang sangat besar dengan menggunakan algoritma pembelajaran untuk mewakili data [8]. Machine learning memungkinkan mesin untuk belajar sendiri tanpa pemrograman. Ini adalah bidang penelitian berdasarkan ide bahwa itu mungkin. Data yang digunakan sistem untuk pembelajaran disebut dataset, dan setiap contoh pelatihan disebut trainingset atau sampel pelatihan. Semakin banyak data, semakin baik pembelajaran [12]

Secara umum, ada tiga jenis pembelajaran dalam machine learning: Supervised Learning, Unsupervised Learning, dan Reinforcement Learning.

Supervised Learning adalah pembelajaran ML yang memecahkan masalah kompleks dengan menggunakan data berlabel untuk melatih sistem saraf buatan. Label adalah tag data yang digunakan dalam model machine learning. Diawasi lebih umum digunakan dalam tugas klasifikasi dan regresi [13].

Unsupervised Learning adalah pembelajaran ML menggunakan pola data yang belum pernah dilihat atau dipikirkan sebelumnya. Teknik ini digunakan untuk mengidentifikasi pola tersembunyi dalam data input yang tidak berlabel dan memproses informasi melalui fitur-fitur yang terdapat dalam pola tersebut, dan umumnya digunakan dalam tugas pengelompokan [13]

Reinforcemen Learninga dalah pembelajaran ML yang mempelajari informasi melalui interaksi pengumpulan dan pemrosesan . Lingkungan dapat belajar dari kesalahan untuk mendapatkan umpan balik positif atau negatif dan untuk mengembangkan umpan balik. Biasanya digunakan dalam permainan logika dan mobil self-driving[13]

C. Deep Learning

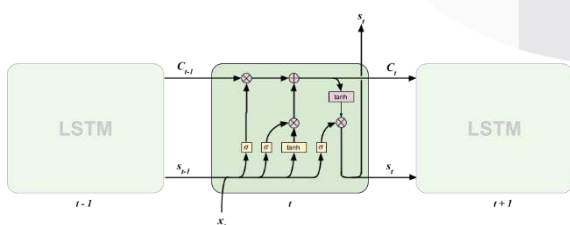
Deep Learning merupakan sub-bidang dari machine learning yang di perkenalkan pada tahun 1986, lalu diterapkan pada tahun 2000 pada jaringan saraf tiruan atau Artificial Neural Network (ANN) [14]. Deep learning dapat mengajarkan komputer untuk melakukan pekerjaan seperti manusia. Deep learning memiliki jaringan yang berjumlah ratusan lapisan atau layer bahkan lebih banyak lagi, itu sebabnya disebut deep learning. Salah satu potensi dari deep learning adalah mengganti fitur buatan tangan dengan algoritma yang efisien untuk pembelajaran hirarkis unsupervised atau semi-supervised feature learning dan hierarchical feature extraction[15]. Penerapan deep learning telah digunakan dalam beberapa bidang seperti klasifikasi gambar, klasifikasi video, deteksi objek, pengenalan pola, text-to-speech, natural processing, robotik, dan klasifikasi teks.

D. Recurrent Neural Network

RNN adalah jenis jaringan saraf tiruan yang sangat baik dalam menemukan korelasi tersembunyi dalam data. Ini berguna untuk aplikasi seperti pengenalan suara, pemrosesan bahasa alami, prediksi deret waktu, dll. RNN ideal untuk masalah pemodelan urutan karena mereka dapat memproses informasi input dan melacak informasi yang dikumpulkan sebelumnya melalui koneksi berulang.

Recurrent Neural Networks (RNNs)) pertama kali dikembangkan oleh Jeff Elman pada tahun 1990-an. RNN adalah singkatan dari berbagai jaringan saraf tiruan yang dirancang khusus untuk pemrosesan data secara terus menerus atau berurutan. Jaringan saraf iteratif dapat digunakan untuk memecahkan masalah dengan data deret waktu. Jaringan saraf iteratif adalah model yang terdiri dari unit input, unit output, dan unit tersembunyi. Model RNN bertindak sebagai aliran informasi satu arah dari entitas input ke entitas tersembunyi dan dari entitas tersembunyi ke entitas tersembunyi saat ini.

E. Long Short-Term Memory



GAMBAR 2.3 ARSITEKTUR LSTM

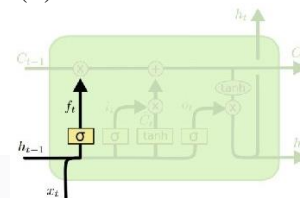
Long Short-Term Memory pertama kali disebutkan oleh Hochreiter dan Schimidhuber pada tahun 1997 [4]. LSTM dikenal juga sebagai jaringan saraf dengan arsitektur yang dapat beradaptasi dan bentuknya dapat disesuaikan,

tergantung aplikasinya. Long Short-Term Memory merupakan turunan dari metode RNN (Recurrent Neural Network). Recurrent Neural Network merupakan jaringan saraf berulang yang didesain khusus untuk handle data berurutan (sequence data). Akan tetapi RNN memiliki masalah vanishing dan exploding gradient yaitu terjadi perubahan pada jangkauan nilai dari satu lapisan menuju lapisan berikutnya pada sebuah arsitektur. LSTM dirancang dan dibangun untuk mengatasi masalah gradien menghilang dari RNN Ketika berhadapan dengan Vanishing dan Exploding gradient tersebut. Arsitektur LSTM terdiri dari lapisan input, lapisan output, dan lapisan tersembunyi.

Lapisan tersembunyi terdiri dari sel memori yang memiliki tiga Gate yaitu Input Gate, Forget Gate, dan Output Gate [9]. Input Gate memiliki fungsi untuk mengontrol berapa banyak informasi yang harus disimpan dalam sel. Hal ini untuk mencegah sel yang tidak perlu. Forget gate memiliki fungsi untuk mengontrol sejauh mana nilai tetap dalam sel memori. Sedangkan Output Gate memiliki fungsi untuk memutuskan berapa banyak nilai dalam sela memori, menghitung Output [9].

Berikut gerbang gerbang yang ada pada sel memori LSTM:

1. Forget Gate (ft)



GAMBAR 2.4 FORGET GATE

Forget gate adalah lapisan sigmoid yang mengambil output pada waktu t-1 dan input pada waktu t dan menggabungkannya serta menerapkan fungsi aktivasi sigmoid. Karena sigmoid, output dari gate ini adalah 0 atau 1. Jika $f_t = 0$ maka keadaan (state) sebelumnya akan dilupakan, sementara jika f_t adalah :

$$f_t = \sigma(U_f x_t + w_f h_{t-1} + b_f) \quad (2.1)$$

Keterangan:

f_t = Forget cell output

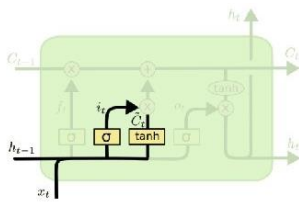
w_f, U_f = bobot

h_{t-1} = output sebelumnya

σ = Sigmoid Function

x_t = data input

2. Input Gate (it)



GAMBAR 2.5
INPUT GATE

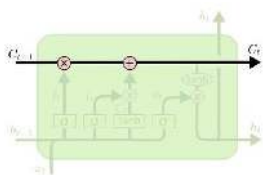
Input Gate memiliki fungsi untuk mengambil *output* sebelumnya dan input baru serta melewatkan mereka melalui lapisan Sigmoid. Gate ini mengembalikan nilai 0 atau 1. Rumus dari it adalah

$$i_t = \sigma(W_i x_t + U_f h_{t-1} + b_i) \quad (2.2)$$

Keterangan :

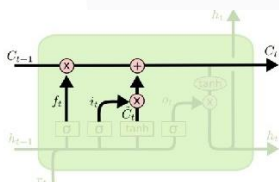
i_t = Input cell output
wt, U_i = bobot

- Cell State berfungsi sebagai tempat masukan data lama sehingga bisa dipakai untuk menentukan *output* dari LSTM, dengan \tilde{C}_t adalah value dari cell state yang akan diupdate dan C_t adalah cell state yang memiliki memori data sebelumnya.



GAMBAR 2.6
CELL STATE

- Candidate



GAMBAR 2.7
CANDIDATE CELL STATE

Dengan

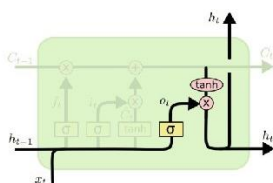
w_c, U_c = bobot

\tilde{C}_t = Candidate cell state

Tanh = tanh Activation Function

$$C_t = f_t \times C_{t-1} + i_t \times \tilde{C}_t \quad (2.4)$$

- Output Gate (o_t)



GAMBAR 2.8
OUTPUT GATE

Output gate mengontrol seberapa banyak state yang lewat ke *output* dan bekerja dengan cara yang sama dengan gate lainnya. Dan terakhir menghasilkan cell state yang baru (h_t). Rumus dari O_t dan h_t adalah :

$$o_t = \sigma(U_o W_o h_{t-1} + b_o) \quad (2.5)$$

$$h_t = O_t \times \tanh(C_t) \quad (2.6)$$

Dengan:

h_t = Output Prediksi

o_t = output cell

Optimasi adalah algoritma yang dapat digunakan untuk mencari nilai lebih baik, apakah mengurangi atau menambahkan fungsi tujuan (fungsi kesalahan). Algoritma optimasi sangat berguna untuk menemukan parameter jaringan yang diinginkan cepat konvergen ke sampel data yang tidak terlihat dari distribusi dasar [9]. Singkatnya, hasil dari semua perhitungan di jaringan saraf dipelajari dan kemudian diperbarui ke bentuk optimal, dengan cara mengurangi loss saat data training. Memiliki beberapa jenis optimasi pada library Keras yaitu Adam, Adamax, SGD dan RMSProp,

F. Uji Validitas

Uji validitas adalah analisis yang mengukur benar tidaknya data. Suatu alat ukur dikatakan valid jika alat tersebut mengukur apa yang seharusnya diukur oleh alat tersebut. Validitas adalah ukuran validitas dan tingkat validitas sarana. Suatu ukuran dianggap valid jika dapat secara memadai menjelaskan variabel data yang sedang diselidiki.[19]

Teknik yang umum digunakan dalam uji validitas adalah menggunakan korelasi *Bivariate Pearson* (Produk Momen Pearson). Berikut adalah persamaan teknik *Bivariate Pearson* [20]:

$$r_{xy} = \frac{N \sum XY - (\sum X)(\sum Y)}{\sqrt{(N \sum X^2 - (\sum X)^2)(N \sum Y^2 - (\sum Y)^2)}} \quad (2.9)$$

Keterangan:

r_{xy} = Koefisien korelasi antara variabel X dan Y

$\sum XY$ = Jumlah perkalian antara variabel X dan Y

N = Jumlah responden $\sum X$ = jumlah butir soal

$\sum X$ = jumlah skor butir soal

$\sum Y$ = jumlah skor total soal

$\sum X^2$ = jumlah skor kuadrat butir soal

$\sum X^2$ = jumlah skor total kuadrat butir soal

G. Uji Reliabilitas

Reliabilitas adalah indikator yang menunjukkan seberapa andal atau dapat dipercayanya suatu alat ukur. Oleh karena itu, uji reliabilitas dapat digunakan untuk mengetahui konsistensi alat ukur, apakah alat ukur tersebut akan tetap konsisten jika dilakukan pengulangan. Suatu alat ukur dikatakan reliabel jika memberikan hasil yang sama walaupun pengukuran dilakukan berulang-ulang. Sebelum melakukan uji reliabilitas, perlu dilakukan uji validitas data, karena data yang diukur harus valid. Namun, jika datanya

tidak valid, maka tidak perlu dilakukan pemeriksaan reliabilitas. Teknik yang digunakan dalam uji reliabilitas dengan metode Cronbach's Alpha dengan rumus sebagai berikut[20]:

$$r_{11} = \left[\frac{k}{(k-1)} \right] \left[1 - \frac{\sum \sigma_{\frac{2}{b}}^2}{\sigma_{\frac{2}{t}}^2} \right] \quad (2.10)$$

Keterangan:

r_{11} = Koefisien reliabilitas instrument (total tes)

k = Jumlah butir pertanyaan sah

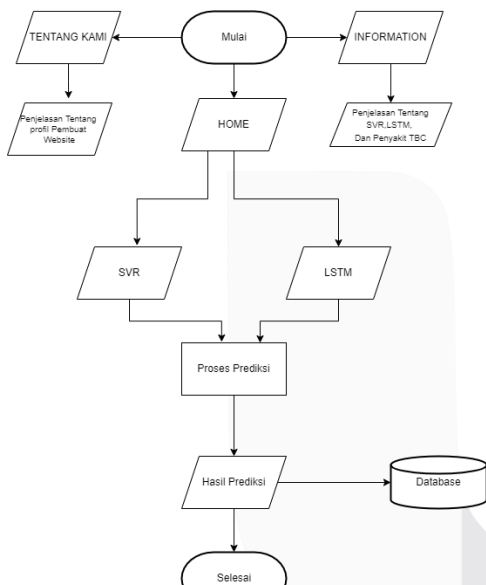
$\sum \sigma_{\frac{2}{b}}^2$ = jumlah varian butir

$\sum \sigma_{\frac{2}{t}}^2$ = varian skor total

III. METODE

A. Gambaran Umum Sistem

Dalam sebuah perancangan sistem prediksi penderita Penyakit Tuberkulosis maka di rancang gambaran umum untuk memudahkan dalam melihat proses kerja sistem yang dibuat. secara rincinya dapat dilihat pada gambar di bawah

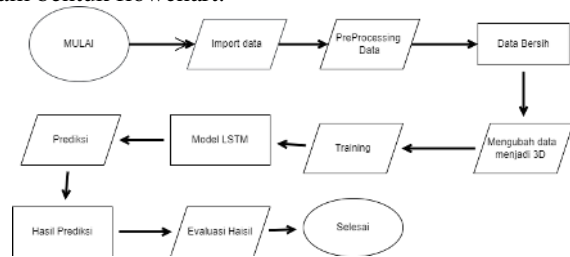


GAMBAR 3.1
GAMBARAN UMUM SISTEM

. Pada Gambar 3.1 terdapat gambaran umum pada sistem. Setelah website terbuka akan langsung di arahkan ke halaman home, pada halaman home pengguna dapat memilih algoritma yang inginkan (LSTM dan SVR) untuk melakukan prediksi penderita Tuberkulosis. Pada halaman prediksi algoritma yang dipilih pengguna dapat mengklik tombol start untuk melakukan prediksi penderita Tuberkulosis, setelah mendapat hasil prediksi akan ditampilkan dalam grafik dan tabel, lalu hasil prediksi tersebut disimpan dalam database. Selain itu terdapat halaman tentang kami, dimana pada halaman tersebut terdapat profil dari pembuat website dan juga terdapat halaman information. Halaman information merupakan halaman yang berisikan informasi singkat tentang algoritma yang digunakan pada website dan informasi singkat tentang penyakit Tuberkulosis

B. Rancangan Metode Long Short-Term Memory

Dalam pembuatan sistem prediksi dimulai dari pengumpulan data, merapihkan data, membersihkan data, melatih data dengan model algoritma hingga hasil dari prediksi. Berikut penjelasan mengenai proses yang terjadi dalam bentuk flowchart:



GAMBAR 3.2
FLOW CHART METODE LSTM

C. Sumber Data

Data yang digunakan untuk melatih model adalah data dari Dinas Kesehatan Karawang di mulai dari tanggal 1 Januari 2020 hingga 31 Desember 2021.

D. Preprocessing Data

Data yang telah didapatkan dan direkap belum bisa di masukkan dalam model algoritma dan perlu dilakukan pembersihan data, dengan menggunakan proses berikut

1. Setelah mendapatkan Raw Data, selanjutnya merekap keseluruhan data dengan hanya menunjukkan tanggal dan jumlah kasus perhari dan data tersebut di ubah kedalam format csv.
2. Mengubah format penulisan pada *Date* Menjadi "Year-Month-Day"
3. Mengubah data yang memiliki nilai kosong Menjadi NaN
4. Setelah itu menginterpolasi data atau pengisian data yang kosong sehingga data terisi. Data terisi merupakan syarat dari modeling. Untuk interpolasi yang di gunakan adalah interpolasi linier.
5. Transformasi data, setelah data di interpolasi selanjutnya dilakukan proses transformasi dengan cara melakukan normalisasi data. Normalisasi data memiliki tujuan yaitu menyederhanakan angka dalam parameter menjadi skala 0-1 sehingga proses pembelajaran mesin menjadi lebih cepat dan lebih baik untuk akurasi nya. Normalisasi yang digunakan adalah *MinMax Scaler*:

E. Proses Pelatihan LSTM

Proses ini dilakukan setelah dilakukan tahap PreProcessing, data di pelatihan sebanyak nilai epoch yang telah ditentukan sehingga menghasilkan hasil yang optimal. Berikut merupakan contoh pemisalan perhitungan manual dari LSTM:

$$\text{Input } 1 = x_{t-1} = [3 \ 0]$$

$$C_{(t-1)} = 0.716$$

$$h_{(t-1)} = 0.482$$

Diketahui bahwa bobot awal sebagai contoh :

$$w_f = 0.225 \quad w_i = 0.421 \quad w_c = 0.891 \quad w_o = 0.741$$

$$U_f = \begin{bmatrix} 0.321 \\ 0.354 \end{bmatrix} \quad U_i = \begin{bmatrix} 0.451 \\ 0.134 \end{bmatrix} \quad U_c = \begin{bmatrix} 0.539 \\ 0.165 \end{bmatrix} \quad U_o = \begin{bmatrix} 0.764 \\ 0.378 \end{bmatrix}$$

Berdasarkan persamaan (2.1) maka hasil dari Forget Gate yaitu

$$f_t = \sigma \left(\begin{bmatrix} 0.321 \\ 0.354 \end{bmatrix} \right) * [3 \ 0] + 0.225 * 0.482 = 0.99$$

Berdasarkan persamaan (2.2) maka hasil dari Input Gate yaitu

$$f_t = \sigma \left(\begin{bmatrix} 0.451 \\ 0.134 \end{bmatrix} \right) * [3 \ 0] + 0.421 * 0.482 = 0.82$$

Berdasarkan persamaan (2.3) maka hasil dari candidate value cell yaitu

$$\tilde{C} = \tanh \left(\begin{bmatrix} 0.539 \\ 0.165 \end{bmatrix} \right) * [3 \ 0] + 0.891 * 0.482 = 0.57$$

Berdasarkan persamaan (2.4) maka hasil dari candidate cell state adalah

$$C_t = 0.99 * 0.716 + 0.82 * 0.57 = 1,176$$

Berdasarkan persamaan (2.5) maka hasil dari Output Gate adalah

$$bracketso_t = \sigma \left(\begin{bmatrix} 0.764 \\ 0.378 \end{bmatrix} \right) * [3 \ 0] + 0.741 * 0.482 = 0.93$$

Berdasarkan persamaan (2.6) maka hasil dari Output Gate adalah

$$h_t = 0.93 * \tanh(1,176) = 0.725$$

F. Perhitungan Evaluasi

Setelah semua proses pelatihan dilakukan, selanjutnya adalah melakukan proses pengujian dengan melakukan perhitungan error menggunakan RMSE (Root Mean Square Error), MAE (Mean Absolute Error), dan R²(koefisien determinasi).

Dengan bandingkan hasil prediksi dan data asli.

1. Root Mean Square Error:

$$RMSE = \sqrt{\frac{\sum_{t=1}^n (A_t - F_t)^2}{N}} \quad (3.3)$$

Keterangan:

- At = nilai asli
- Ft = nilai prediksi
- N = jumlah dari keseluruhan data

2. MAE (Mean Absolute Error)

$$MAE = \frac{1}{m} \sum_{i=1}^m |X_i - Y_i| \quad (3.4)$$

Keterangan:

- X_i = Nilai Asli
- Y_i = Nilai Prediksi
- M = Jumlah dari keseluruhan data

3. R² (Koefisien Determinasi)

$$R^2 = 1 - \frac{\sum_{i=1} (X_i - Y_i)^2}{\sum_{i=1} (\bar{Y} - Y_i)^2} \quad (3.5)$$

Keterangan:

- Y_i = observasi respon ke-i
- X_i = Prediksi respon ke-i
- Ȳ = Rata-Rata

IV. HASIL DAN PEMBAHASAN

A. Pengujian Parameter Terbaik

Pada Pengujian Parameter ini memiliki tujuan yaitu untuk mencari parameter terbaik dari setiap parameter agar mencapai keakuratan yang baik. Pengujian yang akan dilakukan adalah mencari partisi data latih dan data uji, pencarian epoch, pencarian kombinasi hidden layer dan neuron dalam setiap layer dan yang terakhir adalah optimasi.

1. Pencarian Parameter terbaik

Pada pengujian pertama, untuk parameter pertama yang akan digunakan merupakan parameter umum yang digunakan untuk metode LSTM. Berikut merupakan Parameter dan hasil pengujian:

TABEL 4.1
PARAMETER UNTUK INISIALISASI

Modeling	Jumlah Parameter
hidden neuron (per 1 layer)	64
learning rate	0.001
Activation	sigmoid
Epoch	50
Optimizer	Adam
Output	1

Maka hasil dari pencarian parameter sebagai berikut :

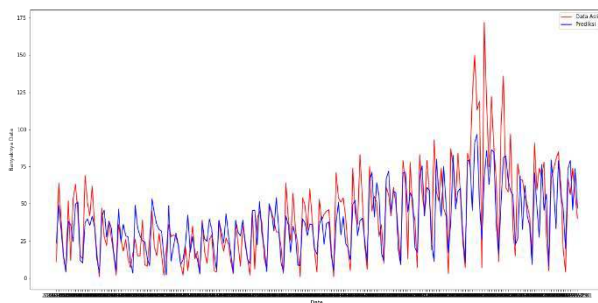
TABEL 4.2
HASIL PENGUJIAN KESELURUHAN

	P D	Epoch	Hidden Layer & Neuron	Optimasi	Learning Rate	MAE	RMSE	R2
Jumlah Pend erita	70%	900	256	Adam	0.001	0.0819	0.12019341	0.53508423

2. Penggunaan Parameter untuk Prediksi

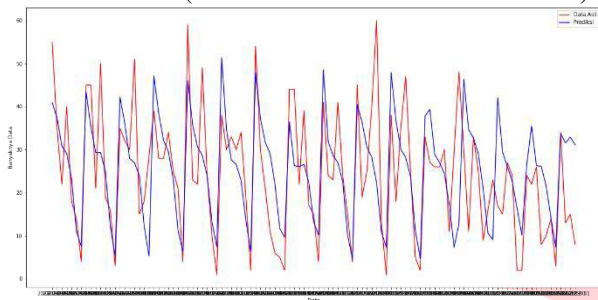
Dari pengujian yang dilakukan sebelumnya, mendapatkan hasil prediksinya dan melakukan prakiran nilai di masa mendatang (forecast future value). Dengan grafik antara nilai Asli dan nilai prediksi sebagai berikut:

- a. Dataset 2 Tahun (01 Januari 2020- 31 Desember 2021)



GAMBAR 4.1
PREDIKSI DAN DATA ASLI

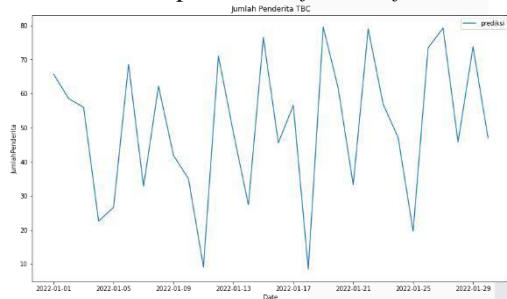
b. Dataset 1 Tahun (01 Januari 2020- 31 Desember 2020)



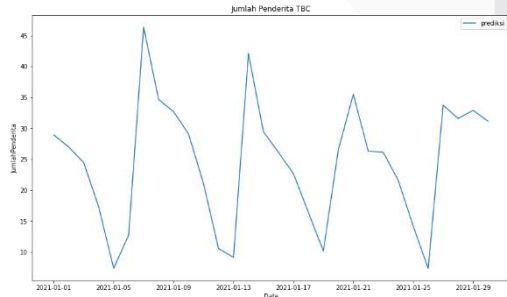
GAMBAR 4.2
PREDIKSI DAN DATA ASLI

Pada gambar 4.12 adalah perbandingan data asli dan prediksi dengan menggunakan dataset 2 tahun, sedangkan pada gambar 4.13 adalah perbandingan data asli dan prediksi dengan menggunakan dataset 1 tahun dimana pada gambar 4.13 parameter terbaik yang digunakan adalah Partisi data : 70% training data 30% testing data , epoch :30, hidden layer 2 dengan neuron masing masing layer 256 neuron dan untuk optimasi yang digunakan adalah adam

Dan berikut merupakan hasil *forecast future value*:



GAMBAR 4.3
PREDIKSI 01 JANUARI SAMPAI 30 JANUARI 2022



GAMBAR 4.4
PREDIKSI 01 JANUARI SAMPAI 30 JANUARI 2021

V. KESIMPULAN

Berdasarkan penelitian, pengujian dan analisa yang telah dilakukan pada Tugas Akhir ini, maka dapat disimpulkan bahwa :

1. *Website* untuk menampilkan hasil prediksi penderita penyakit tuberkulosis menggunakan metode LSTM berjalan dengan baik dengan hasil pengujian alpha sebesar 100%. Dan hasil pengujian beta didapat dari 42 responden sebesar 83,6%
2. Hasil dari proses pengujian partisi data dengan 70%:30%, *epoch* sebesar 900, jumlah *hidden layer* sebanyak 1 layer LSTM dengan *neuron* serta tipe optimasi Adam dengan learning rate sebesar 0.001. selain itu, hasil pengujian parameter terbaik menghasilkan nilai RMSE = 0.12019341, MAE = 0.0819 Dan R2 = 0.53508423.

REFERENCES

- [1] Kementerian Kesehatan RI. 2018. Infodatin Tuberkulosis. Jakarta
- [2] World Health Organization Global Tuberculosis Report ,2020.
- [3] Dinas Kesehatan Kabupaten Karawang, Profil Kesehatan Kabupaten Karawang Tahun 2018. Karawang: Dinas Kesehatan Kabupaten Karawang, 2019.
- [4] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735– 1780, 1997.
- [5] P. Sugiartawan, A. A. Jiwa Permana, and P. I. Prakoso, "Forecasting Kunjungan Wisatawan Dengan Long Short-Term Memory (LSTM)," *J. Sist. Inf. dan Komput. Terap. Indones.*, vol. 1, no. 1, pp. 43–52, 2018.
- [6] J. Schmidhuber, "Deep Learning in neural networks: An overview," *Neural Networks*, vol. 61, pp. 85–117, 2015, doi: 10.1016/j.neunet.2014.09.003.
- [7] Kementrian RI Direktorat Jendral Pengendalian Penyakit dan Penyehatan Lingkungan (2014) Pedoman Nasional Pengendalian Tuberkulosis. Jakarta: Kementrian Kesehatan RI.
- [8] Danukusumo, Kefin Pudi. 2017. "Implementasi Deep Learning Menggunakan Convolutional Neural Network Untuk Klasifikasi Citra Candi Berbasis GPU".
- [9] J. R. Vinayakumar, K. P. Soman, and P. Poornachandran, "Long short-term memory based operation log anomaly detection," 2017 Int. Conf. Adv. Comput. Commun. Informatics, ICACCI 2017, vol. 2017-Janua, pp. 236–242, 2017.
- [10] G. Mathisen, "Forecasting Multivariate Time Series Data Using Neural Networks Sigurd Øyen," no. June, 2018.
- [11] S. Chaudhury and T. Yamasaki, "Robustness of Adaptive Neural Network Optimization Under Training Noise," *IEEE Access*, vol. 9, pp. 37039–37053, 2021.
- [12] Hahn, P. (2019). Artificial intelligence and machine learning. *Handchirurgie Mikrochirurgie Plastische Chirurgie*, 51(1), 62–67. <https://doi.org/10.1055/a-0826-4789>
- [13] Mitchell, Tom A. *Machine Learning*. McGraw-Hill.1997

- [14] Schmidhuber, J. (2015). Deep Learning in neural networks: An overview. *Neural Networks*, 61, 85–117. <https://doi.org/10.1016/j.neunet.2014.09.003>
- [15] Manik, H. B. (2021). Penerapan Algoritma C4 . 5 Dalam Memprediksi Kebutuhan Pembibitan Pohon. 1(1):14–19.
- [16] Jo, J.-M. (2019). Effectiveness of Normalization Pre-Processing of Big Data to the Machine Learning Performance. 14:547–552.
- [17] Syilfi, D. Ispriyanti, and D. Safitri, "Analisis Regresi Linier Piecewise Dua Segmen," vol. 1, 2012.
- [18] Arief, M.Rudyanto., 2011, *Pemrograman Web Dinamis Menggunakan PHP dan MYSQL*, Andi, Yogyakarta.
- [19] Nasution, *Metode Research*, (Jakarta: Bumi Aksara, 2011), hal. 74.
- [20] F. Yusup, "Uji Validitas dan Reliabilitas Instrumen Penelitian Kuantitatif," *Jurnal Tarbiyah : Jurnal Ilmiah Kependidikan*, vol. 7, no. 1, 2018.
- [21] R. Kuloningtyas. A. B. Suksmono, T. L. R. Mengko, G. A. P. Saptawati. "Scg mentasi Citra Berwarna dengan Menggunakan Metode Clustering Berbasis Patch untuk Identifikasi Mycobacterium Tuberculosis" *Prosiding Jurnal BIO SAINS*, Vol. 17, No. 1, 2014.