

SKRIPSI

**IMPLEMENTASI DAN ANALISIS *DATA MINING* UNTUK PENCARIAN
POLA KECELAKAAN LALU LINTAS DI KOTA TERNATE DENGAN
METODE *LONG SHORT TERM MEMORY (LSTM)***



**OLEH
Vivi Fitriani Duwila
07351811051**

**PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS KHAIRUN
TERNATE
2024**

LEMBAR PENGESAHAN

ANALISIS DAN IMPLEMENTASI DATA MINING UNTUK PENCARIAN POLA KECELAKAAN LALU LINTAS DI KOTA TERNATE DENGAN METODE LONG SHORT TERM MEMORY (LSTM)

Oleh
Vivi Fitriani Duwila
07351811051

Skripsi ini telah disahkan
Tanggal 01 Maret 2024

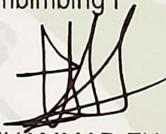
Menyetujui
Tim Penguji

Ketua Penguji



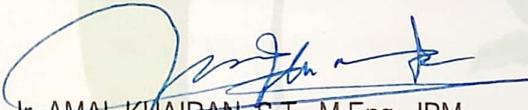
Ir. SALKIN LUTFI, S.Kom., M.T.
NIP. 198601112014041002

Pembimbing I



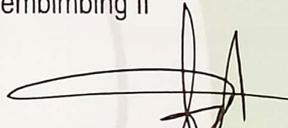
MUHAMMAD FHADLI, S.Kom., M.Sc.
NIP. 199611232023211012

Anggota Penguji



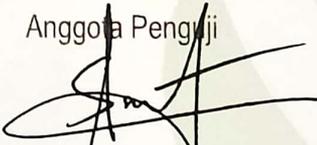
Ir. AMAL KHAIRAN, S.T., M.Eng., IPM.
NIP. 197401112003121003

Pembimbing II



SYARIFUDDIN W. KAPITA, S.Pd., M.Si.
NIP. 199103122024211001

Anggota Penguji



Dr. ASSAF ARIEF, S.T., M.Eng.
NIP. 198307102008121001

Mengetahui/Menyetujui

Koordinator Program Studi
Informatika



ROSIHAN S.T., M.Cs.
NIP. 197607192010121001



Dekan Fakultas Teknik
Universitas Khairun

Ir. ENDAH HARISUN, S.T., M.T., CRP.
NIP. 197511302005011013

LEMBAR PERNYATAAN KEASLIAN

Yang bertanda tangan dibawah ini:

Nama : Vivi Fitriani Duwila
NPM : 07351811051
Fakultas : Teknik
Jurusan/Program Studi : Informatika
Judul : Implementasi Dan Analisis Data Mining Untuk
Pencarian Pola Kecelakaan Lalu Lintas Di Kota
Ternate Dengan Metode Long Short Term
Memory (LSTM)

Dengan ini menyatakan bahwa penulisan Skripsi yang telah saya buat ini merupakan hasil karya sendiri dan benar keasliannya. Apabila ternyata di kemudian hari penulisan Skripsi ini merupakan hasil plagiat atau penjiplakan terhadap karya orang lain, maka saya bersedia mempertanggung jawabkan sekaligus bersedia menerima sanksi berdasarkan aturan tata tertib di Universitas Khairun.

Demikian pernyataan ini saya buat dalam keadaan sadar dan tidak dipaksakan.

Penulis



Vivi Fitriani Duwila

HALAMAN PERSEMBAHAN

Bismillahirrohmannirrahim

Dengan rahmat Allah SWT yang maha pengasih lagi maha penyayang dengan ini saya persembahkan skripsi ini untuk

Keluarga tercinta terumata mama dan papa, terima kasih atas limpahan doa dan kasih sayang yang tak terhingga dan selalu memberikan dorongan baik secara moril maupun material sehingga dapat menyelesaikan skripsi ini.

Dosen-dosen dan staf pengurus prodi informatika yang telah membagikan ilmu dan pengalamannya hingga penulis dapat sampai ke tahap ini.

Teman-teman angkatan 2018, terkhususnya ika, iwid isty dan pragos kakak senior dan juga adik junior yang selalu memberikan warna dalam perkuliahan, selalu memberikan dukungan dan semangat untuk sampai ke tahap ini.

MOTTO

“para pemenang tak pernah menyerah, dan orang-orang yang menyerah tak pernah menang”

LEMBAR PERNYATAAN KEASLIAN

Yang bertanda tangan dibawah ini:

Nama : Vivi Fitriani Duwila
NPM : 07351811051
Fakultas : Teknik
Jurusan/Program Studi : Informatika
Judul : Implementasi Dan Analisis Data Mining Untuk
Pencarian Pola Kecelakaan Lalu Lintas Di Kota
Ternate Dengan Metode Long Short Term
Memory (LSTM)

Dengan ini menyatakan bahwa penulisan Skripsi yang telah saya buat ini merupakan hasil karya sendiri dan benar keasliannya. Apabila ternyata di kemudian hari penulisan Skripsi ini merupakan hasil plagiat atau penjiplakan terhadap karya orang lain, maka saya bersedia mempertanggung jawabkan sekaligus bersedia menerima sanksi berdasarkan aturan tata tertib di Universitas Khairun.

Demikian pernyataan ini saya buat dalam keadaan sadar dan tidak dipaksakan.

Penulis

Vivi Fitriani Duwila

HALAMAN PERSEMBAHAN

Bismillahirrohmannirrahim

Dengan rahmat Allah SWT yang maha pengasih lagi maha penyayang dengan ini saya persembahkan skripsi ini untuk

Keluarga tercinta terumata mama dan papa, terima kasih atas limpahan doa dan kasih sayang yang tak terhingga dan selalu memberikan dorongan baik secara moril maupun material sehingga dapat menyelesaikan skripsi ini.

Dosen-dosen dan staf pengurus prodi informatika yang telah membagikan ilmu dan pengalamannya hingga penulis dapat sampai ke tahap ini.

Teman-teman angkatan 2018, terkhususnya ika, iwid isty dan pragos kakak senior dan juga adik junior yang selalu memberikan warna dalam perkuliahan, selalu memberikan dukungan dan semangat untuk sampai ke tahap ini.

MOTTO

“para pemenang tak pernah menyerah, dan orang-orang yang menyerah tak pernah menang”

KATA PENGANTAR

Puji syukur penulis panjatkan kepada Allah Subhanahu Wata'ala yang telah melimpahkan rahmat, taufik serta hidayah-Nya sehingga penulis dapat menyelesaikan skripsi dengan judul: "Implementasi Dan Analisis *Data Mining* Untuk Pencarian Pola Kecelakaan Lalu Lintas Di Kota Ternate Dengan Metode *Long Short Term Memory (LSTM)*".

Laporan Penelitian skripsi Ini Disusun Untuk Memenuhi Salah Satu Persyaratan Untuk Mendapatkan Gelar Sarjana Pada Jurusan Informatika Universitas Khairun. Selama Proses Dalam Menyelesaikan Skripsi Tugas Akhir Ini, Penulis Telah Mendapatkan Bantuan, Bimbingan, Dukungan, Serta Motivasi, Baik Secara Langsung Ataupun Tidak Langsung. Untuk Itu, Pada Kesempatan Ini Penulis Ingin Menyampaikan Ucapan Terima Kasih Kepada:

1. Bapak Dr. M. Ridha Ajam, M.H., selaku Rektor Universitas Khairun, beserta segenap pimpinan Universitas Khairun.
2. Bapak Ir., Endah Harisun, S.T., M.T., CPR., selaku Dekan Fakultas Teknik Universitas Khairun.
3. Bapak Rosihan, S.T., M.Cs., selaku Koordinator Program Studi Informatika
4. Bapak Muhammad Fhadli, S.Kom., M.Cs., selaku Dosen Pembimbing I, terima kasih telah memberikan arahan dan bimbingannya selama penulis melakukan penelitian dan penyelesaian skripsi ini.
5. Bapak Syarifuddin N. Kapita, S.pd., M.Si., selaku Dosen Pembimbing II, terima kasih atas bimbingannya dalam penyelesaian skripsi ini.
6. Terima kasih kepada papa, mama dan adikku M. Subandrio dan Firly Nurzahra yang banyak memberikan dukungan kepada penulis selama kuliah hingga menyelesaikan skripsi ini.
7. Terima kasih kepada sepupu ku M. Fhauzy, Rila, Nurfirsta dan keluarga besar yang telah bersedia membantu penulis dan memberikan semangat dalam penyelesaian skripsi ini.
8. Beasti healing Yeyen, Rina, Ka fit, Ka dila, Ka ghalib, Ka swa, Ifta, Ka hizbullah, Agus, Yusril, Terima kasih telah memberi semangat dan dukungan dalam penyelesaian skripsi ini.

9. Teman dekat yang selalu ada Juni, Ka Putri, Risty, Widy yang telah membantu penulis dan memberi semangat dalam penyelesaian skripsi ini.
10. Untuk teman-teman seperjuangan khususnya, Widy, Amalia sari, Kiki, Alwia, Andhika Budy, Amalia kurniawati, dan Angkatan 2018 Prodi Informatika yang tidak dapat disebutkan satu-persatu.

Ternate, 20 Agustus 2024

Penulis

DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
HALAMAN PENGESAHAN	ii
HALAMAN PERNYATAAN KEASLIAN	iii
HALAMAN PERSEMBAHAN	iv
KATA PENGANTAR	v
DAFTAR ISI	viii
DAFTAR GAMBAR	viii
DAFTAR TABEL	viii
ABSTRAK	xii
BAB I PENDAHULUAN	
1.1. Latar Belakang	1
1.2. Rumusan Masalah	2
1.2. Batasan Masalah.....	3
1.4. Tujuan Penelitian.....	3
1.5. Manfaat Penelitian.....	3
1.6. Sistematika Penulisan	3
BAB II TINJAUAN PUSTAKA	
2.1. Penelitian Terdahulu	5
2.2. Kecelakaan Lalu Lintas	6
2.3. <i>Analisis Data Mining</i>	7
2.4. <i>Decision Tree</i>	7
2.5. <i>Long Short Term Memory (LSTM)</i>	7
2.6. <i>Python</i>	13
2.7. <i>Flowchart</i> (Diagram Alir).....	13
BAB III METODE PENELITIAN	
3.1. Tahapan Penelitian.....	15
3.2. Objek dan Waktu Penelitian	16
3.3. Data Penelitian	16

3.4.	Perumusan Masalah	16
3.5.	Pengumpulan Data.....	16
3.6.	Proses <i>Data Mining</i>	17
3.7.	Proses Pemodelan (<i>Modelling</i>)	17
3.7.1.	<i>Long Short-term Memory</i> (LSTM).....	17
3.7.2.	Contoh Perhitungan LSTM.....	18
3.8.	Variabel Penelitian	22
3.9.	Evaluasi Faktor Kecelakaan	22
BAB IV HASIL DAN PEMBAHASAN		
4.1.	Proses Pengumpulan Data.....	25
4.2.	Membaca Data dari File	26
4.3.	Menampilkan Informasi Umum Tentang Data	26
4.4.	Menampilkan Beberapa Baris Pertama Data	27
4.5.	Statistik Deskriptif Tentang Data Numerik	28
4.6.	Visualisasi Histogram Jumlah Luka Ringan.....	28
4.7.	Visualisasi Histogram Jumlah Luka Berat	29
4.8.	Visualisasi Histogram Jumlah Kecelakaan Berdasarkan Tahun.....	30
4.9.	Visualisasi Histogram Jumlah kecelakaan Berdasarkan Jenis Kecelakaan	31
4.10.	Visualisasi Jumlah Luka Ringan Per Bulan	32
4.11.	Visualisasi Jumlah Luka Berat Per Bulan	33
4.12.	Visualisasi Jumlah Meninggal Dunia Per Bulan	34
4.13.	Visualisasi Jumlah Luka Ringan Per Lokasi	35
4.14.	Visualisasi jumlah Luka Berat Per Lokasi.....	36
4.15.	Visualisasi Jumlah Meninggal Dunia Per Lokasi	37
4.16.	Visualisasi Jumlah Kerugian Material Per Lokasi	38
4.17.	Histogram Berdasarkan Tahun Dan bulan Kejadian.....	39
4.18.	Visualisasi Korelasi Antara Variabel Numerik.....	39
4.19.	Implementasi Model <i>Long Short Term Memory</i>	40
4.19.1.	<i>Preprocessing Data</i>	40
4.19.2.	Memilih Atribut Yang Akan Digunakan.....	41
4.19.3.	Model Pada data Testing.....	42

4.20. Tampilkan Grafik <i>Dession Tree</i>	42
--------------------------------------------------	----

BAB V PENUTUP

5.1. Kesimpulan.....	45
----------------------	----

5.2. Saran.....	46
-----------------	----

DAFTAR PUSTAKA

LAMPIRAN

DAFTAR TABEL

	Halaman
Tabel 2.1. Penelitian Terdahulu	5
Tabel 2.2. Simbol-simbol pada <i>flowchart</i>	13
Tabel 3.1. <i>Data Input</i>	18
Tabel 3.2. Bobot tiap matriks dan nilai <i>ht</i> – 1 serta <i>ct</i> – 1	19
Tabel 4.1. Proses Pengumpulan Data	25

DAFTAR GAMBAR

	Halaman
Gambar 2.1. Arsitektur LSTM (Olah, 2015)	11
Gambar 2.2. Notasi LSTM (Olah, 2015)	12
Gambar 3.1. Tahapan Penelitian	15
Gambar 3.2. Arsitektur LSTM	15
Gambar 3.3. Jaringan LSTM	18
Gambar 4.1. Membaca Data dari File	26
Gambar 4.2. Menampilkan Informasi Umum Tentang Data	27
Gambar 4.3. Menampilkan Beberapa Baris Pertama Data.....	27
Gambar 4.4. Statistik Deskriptif Data Numerik	28
Gambar 4.5. Visualisasi Histogram Jumlah Luka Ringan.....	29
Gambar 4.6. Visualisasi Histogram Jumlah Luka Berat.....	30
Gambar 4.7. Visualisasi Histogram Berdasarkan Tahun	31
Gambar 4.8. Histogram Jumlah Kecelakaan Berdasarkan Jenis Kecelakaan.....	32
Gambar 4.9. Visualisasi Jumlah Luka Ringan Per Bulan	32
Gambar 4.10. Visualisasi Jumlah Luka Berat Per Bulan	33
Gambar 4.11. Visualisasi Jumlah Meninggal Dunia	34
Gambar 4.12. Visualisasi Jumlah Luka Ringan Per Lokasi	35
Gambar 4.13. Visualisasi Jumlah Luka Berat Per Lokasi.....	36
Gambar 4.14. Visualisasi Jumlah Meninggal Dunia Per Lokasi	37
Gambar 4.15. Kerugian Material Per Lokasi.....	38
Gambar 4.16. Histogram Berdasarkan tahun Dan Bulan Kecelakaan	39
Gambar 4.17. Visualisasi Korelasi Variabel Numerik	40
Gambar 4.18. <i>Preprocessing</i> Data	41
Gambar 4.19. Memilih Atribut.....	41
Gambar 4.20. Evaluasi Model Pada Data Testing.....	42
Gambar 4.21. Grafik Dession Tree.....	44

ABSTRAK

IMPLEMENTASI DAN ANALISIS DATA MINING UNTUK PENCARIAN POLA KECELAKAAN LALU LINTAS DI KOTA TERNATE DENGAN METODE *LONG SHORT TERM MEMORY* (LSTM)

Vivi Fitriani Duwila¹, Mfhadli², Syarifudin³
Program Studi Informatika, Fakultas Teknik, Universitas Khairun
Jl. Jati Metro, Kota Ternate

E-mail: vivifitrianduwila@gmail.com¹, mfhadli@unkhair.ac.id², syarifudin@gmail.com³

Meningkatnya jumlah kecelakaan lalu lintas di Kota Ternate dari tahun ke tahun membutuhkan perhatian dan penanggulangan yang serius. Upaya yang dapat dilakukan untuk mengurangi angka kecelakaan lalu lintas yaitu melihat proyeksi kedepan atau kecenderungan pola sehingga dapat meminimalisir terjadinya kecelakaan lalu lintas. Namun analisis yang dilakukan untuk hal ini sangatlah sedikit, maka sebuah teknologi data mining dilakukan untuk mengetahui tingkat kecelakaan lalu lintas di Kota Ternate dengan harapan dapat mengurangi angka kecelakaan. *Data Mining* adalah proses analisis yang dilakukan secara otomatis pada data yang berjumlah besar untuk memperoleh sebuah pola. LSTM merupakan salah satu pengembangan *neural network* yang dapat digunakan untuk pemodelan data *time series*. Dengan menggunakan algoritma LSTM maka dapat diperoleh hasil prediksi kecelakaan lalu lintas yang sering muncul sehingga dapat diperoleh pola dari algoritma *decision tree*.

Kata Kunci: Kecelakaan Lalu Lintas, Data Mining, LSTM, Decision Tree.

BAB I

PENDAHULUAN

1.1. Latar Belakang

Kecelakaan lalu lintas di Indonesia terus meningkat seiring dengan meningkatnya kendaraan bermotor yang mengakibatkan kerugian materi, luka ringan, luka berat sampai dengan meninggal dunia. Faktor penyebab kecelakaan lalu lintas diantaranya pengemudi, kendaraan dan prasarana maupun lingkungan. Berdasarkan data dari Polda Maluku Utara dan Polres Kota Ternate juga mengalami hal yang sama yaitu kepemilikan kendaraan bermotor pribadi semakin tinggi yang disebabkan oleh bertambahnya penduduk Provinsi Maluku Utara sehingga tidak menutup kemungkinan bertambahnya angka kecelakaan lalu lintas yang terjadi. Kecelakaan lalu lintas merupakan hal serius yang perlu ditangani karena dapat menyebabkan kematian, luka berat, luka ringan, cacat tubuh serta trauma bagi korban kecelakaan, dan kerugian material Efendi, M. F. (2019). Upaya yang dapat dilakukan untuk mengurangi angka kecelakaan lalu lintas yaitu melihat proyeksi kedepan atau kecenderungan pola sehingga dapat digunakan merencanakan strategi ataupun kebijakan agar dapat meminimalisir terjadinya kecelakaan lalu lintas. Proyeksi dapat dilakukan dengan melakukan peramalan pada data runtun atau *time-series*. Ada banyak faktor terjadinya kecelakaan lalu lintas ini. Namun analisis yang dilakukan untuk hal ini sangatlah sedikit. Maka sebuah teknologi *data mining* berkembang untuk menjawab tantangan ini.

Meningkatnya jumlah kecelakaan lalu lintas di kota Ternate dari tahun ke tahun membutuhkan perhatian dan penanggulangan yang serius yang mana mengingat kecelakaan lalu lintas menjadi faktor pembunuh yang cukup tinggi. Kecelakaan lalu lintas sangat sering terjadi yang mana tidak terlepas dari meningkatnya pertumbuhan penduduk

sehingga meningkatkan mobilitas masyarakat. Penanggulangan kecelakaan lalu lintas dapat dilakukan dengan melakukan langkah-langkah pendekatan seperti sosialisasi dan pengawasan terhadap rambu-rambu serta trafik pada jalan raya. Untuk dapat mencari solusi penanggulangan yang tepat maka diperlukan informasi mengenai kecelakaan lalu lintas yang terjadi. Dengan informasi kecelakaan lalu lintas yang tepat maka dapat diambil langkah-langkah penanggulangan yang efektif sehingga dapat menurunkan angka kecelakaan lalu lintas yang terjadi di Kota Ternate.

Long Short Term Memory (LSTM) merupakan salah satu jenis RNN. LSTM menyimpan informasi terhadap pola-pola pada data. LSTM dapat mempelajari data mana saja yang akan disimpan dan data mana saja yang akan dibuang, karena pada setiap *neuron* LSTM memiliki beberapa gates yang mengatur memori pada setiap *neuron* itu sendiri. LSTM banyak digunakan untuk pemrosesan teks, video, dan data *time series*.

Berdasarkan latar belakang tersebut, maka dalam penelitian ini akan dilakukan proses *data mining* untuk mengetahui tingkat kecelakaan lalu lintas di Kota Ternate dengan harapan dapat mengurangi angka kecelakaan lalu lintas yang dari tahun ke tahun semakin meningkat. Penelitian ini diharapkan dapat menjadi acuan kepolisian untuk dapat melakukan tindakan antisipasi agar terjadi penurunan angka kecelakaan lalu lintas di wilayah Kota Ternate.

1.2. Rumusan Masalah

Berdasarkan latar belakang diatas, maka perumusan masalah dalam penelitian ini adalah:

1. Bagaimana analisis data kecelakaan lalu lintas di Kota Ternate berdasarkan visualisasi grafik dengan menerapkan metode *Long Short Term Memory* (LSTM)
2. Bagaimana tampilan grafik *decision tree* pada data kecelakaan lalu lintas.

1.3. Batasan Masalah

Batasan masalah dari penelitian ini adalah penelitian ini dilakukan di Polda Maluku Utara, dimana data penelitian yang diambil yaitu data 4 tahun terakhir dan metode yang digunakan adalah *Long Short Term Memory* (LSTM).

1.4. Tujuan Penelitian

1. Mengetahui hasil analisis pada data kecelakaan lalu lintas di Kota Ternate berdasarkan tampilan visualisasi grafik.
2. Dapat menampilkan grafik pola kecelakaan lalu lintas di Kota Ternate dengan menggunakan algoritma Decision Tree.

1.5. Manfaat Penelitian

1. Untuk melihat bagaimana pola suatu kecelakaan lalu lintas di Kota Ternate itu terjadi
2. Dapat melakukan tindakan antisipasi agar terjadi penurunan angka kecelakaan lalu lintas di wilayah Kota Ternate.

1.6. Sistematika Penulisan

Sistematika penulis dibagi menjadi 3 (tiga) bab yang terdiri dari:

BAB I PENDAHULUAN

Terdiri dari latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, serta sistematika penulisan.

BAB II LANDASAN TEORI

Memaparkan teori-teori yang didapat dari sumber-sumber yang relevan untuk digunakan sebagai panduan dalam penelitian serta penyusunan skripsi.

BAB III METODOLOGI PENELITIAN

Bab ini membahas tentang metode penelitian yang telah dilakukan oleh penulis dengan permasalahan yang diangkat.

BAB IV HASIL DAN PEMBAHASAN

Pada bab ini menjelaskan implementasi dan analisis kecelakaan lalu lintas di Kota Ternate

BAB V PENUTUP

Pada bab ini memuat kesimpulan dari hasil penelitian dan saran. Untuk penelitian selanjutnya mengenai topik terkait.

BAB II

TINJAUAN PUSTAKA

2.1. Penelitian Terdahulu

Pada bagian ini peneliti mencantumkan berbagai hasil penelitian terdahulu terkait dengan penelitian yang hendak dilakukan dengan topik pola penyebab kecelakaan lalu lintas. Berikut merupakan penelitian terdahulu yang masih terkait dengan tema yang penulis kaji. Dapat dilihat pada tabel 2.1.

Tabel 2.1 Penelitian Terdahulu

NO	Judul, Nama dan Tahun	Metode	Hasil
1	Analisis Pola Kecelakaan Lalu Lintas Menggunakan Algoritma <i>Decision Tree</i> Berdasarkan Ekstraksi Informasi Dari Berita Online Menggunakan <i>Named Entity Recognition</i> (NER). (Hardi Dwi Susanto, Budi Yuniarto 2023)	<i>Decision Tree</i>	Hasil penelitian menunjukkan bahwa ekstraksi informasi kecelakaan lalu lintas di jalan tol di Provinsi Jawa Barat dari artikel berita detik.com dilakukan menggunakan model NER Bi.LSTM-CNN yang menghasilkan data akhir sebanyak 327 kejadian kecelakaan. Hasil penelitian menunjukkan bahwa CART dapat memodelkan pola kejadian kecelakaan dengan cukup baik.
2	Pengelompokan Data Kecelakaan Lalu Lintas Di Kota Tasikmalaya Menggunakan Algoritma <i>KMeans</i> (ledam Fardian Anshor1 2020)	Algoritma <i>KMeans</i>	Berdasarkan Validasi menggunakan metrik <i>Davies Bouldin Index</i> maka diperoleh 4 <i>cluster</i> yang dinilai dapat mengelompokan data dengan baik. <i>Performance Vector</i> hasil evaluasi <i>cluster</i> yang dibentuk sebanyak 4 <i>cluster</i> dengan nilai sebesar 0,134. <i>Cluster</i> 1 dengan jumlah data kecelakaan sebanyak 74 kecelakaan yang paling banyak terjadi pada malam hari, <i>Cluster</i> 2 dengan jumlah kecelakaan sebanyak 16 kecelakaan yang paling banyak terjadi pada siang hari, <i>Cluster</i> 3 dengan jumlah kecelakaan sebanyak 6 kecelakaan yang paling banyak terjadi

			pada sore hari dan <i>Cluster 4</i> dengan jumlah kecelakaan sebanyak 113 kecelakaan yang paling banyak terjadi pada pagi hari.
3	Menentukan Pola Hubungan Kecelakaan Lalu Lintas Dengan Algoritma Apriori (Endang Utari dan Paska Marto Hasugian 2021)	Algoritma Apriori	Dari tahapan yang dilakukan maka item yang memenuhi minimum support = 30% dan minimum confident = 50%.
4	Implementasi Data Mining Untuk Menentukan Pola Penyebab Kecelakaan Lalu Lintas Di Wilayah Kota Palembang Menggunakan Algoritma <i>FP-Growth</i> (fathan Pangestu dan Andri 2021)	Algoritma <i>FP-Growth</i>	Algoritma <i>fp-growth</i> dapat diterapkan dengan baik untuk menentukan pola penyebab kecelakaan lalu lintas di wilayah kota Palembang dengan menggunakan 2 minimum support sebesar 40% dan 50% serta 2 minimum <i>confidence</i> sebesar 70% dan 90%.
5	Penerapan Metode <i>Long Short Term Memory</i> (LSTM) Dalam Memprediksi Harga Saham PT Bank Central Asia (Abdul Rosyd, ade Irma Purnamasari, Irfan ali 2024)	Algoritma <i>Long short Term Memory</i> (LSTM)	Berdasarkan evaluasi terhadap prediksi harga saham PT Bank Central Asia menggunakan metode LSTM dengan data histori harga saham dari 2020 hingga 2023, dapat disimpulkan bahwa metode tersebut efektif dalam menganalisis pergerakan harga saham tersebut. grafik prediksi menunjukkan kemiripan yang relatif tinggi dengan data aktual.

2.2. Kecelakaan Lalu Lintas

Berdasarkan Undang-undang Nomor 22 tahun 2009 pasal 1 ayat 24 tentang lalu lintas dan angkutan jalan, Kecelakaan lalu lintas merupakan suatu peristiwa di jalan yang tidak terduga dan tidak disengaja yang melibatkan kendaraan dengan atau tanpa pengguna jalan lain yang mengakibatkan korban manusia dan/atau kerugian harta benda. Faktor penyebab terjadinya kecelakaan lalu lintas yang sering diabaikan oleh pengguna jalan yaitu:

mengendara saat keadaan mengantuk, menggunakan handphone saat berkendara, tidak mematuhi rambu-rambu lalu lintas, dan sebagainya.

2.3. Data Mining

Data mining adalah proses analisa yang dilakukan secara otomatis pada data yang kompleks dan berjumlah besar untuk memperoleh sebuah pola atau kecenderungan yang umumnya tidak disadari.

Data mining mulai ada sejak 1990-an sebagai cara yang benar dan tepat untuk mengambil pola dan informasi yang digunakan untuk menemukan hubungan antar data dalam melakukan pengelompokan ke dalam satu atau lebih cluster sehingga objek-objek yang berada dalam satu cluster akan mempunyai kesamaan yang tinggi antara satu dengan lainnya.

Data mining bertujuan untuk menemukan pola dan aturan dalam basis data yang berukuran besar sehingga dapat dipakai dalam mengambil suatu keputusan (Rubiyanti,2018).

2.4. Decision Tree

Decision tree adalah sebuah metode *machine learning* dengan kemampuan yang digunakan untuk membuat sebuah keputusan yang divisualisasikan dalam bentuk pohon keputusan.

2.5. Long Short Term Memory (LSTM)

Long Short Term Memory (LSTM) merupakan salah satu pengembangan *neural network* yang dapat digunakan untuk pemodelan data *time series* (Wiranda, 2019). Komponen utama lapisan LSTM disebut sel memori. Sel memori terdiri dari empat elemen utama, yaitu *input gate*, *neuron* dengan koneksi berulang, *forget gate*, dan *output gate*.

LSTM mempunyai *memory block* yang akan menentukan nilai mana yang akan dipilih sebagai keluaran yang relevan terhadap masukan yang diberikan. Hal ini adalah keunggulan yang dimiliki oleh LSTM (Wiranda, 2019).

Arsitektur LSTM memungkinkan untuk menambah atau menghapus informasi dari *cell state*. Hal ini disebut sebagai *gates* yang merupakan pengatur apakah informasi akan diteruskan atau diberhentikan. *Gates* terdiri dari lapisan *sigmoid* dan *pointwise multiplication operation*, yang mana *output* dari lapisan *sigmoid* adalah angka 1 atau 0 yang menunjukkan apakah informasi tersebut akan diteruskan atau diberhentikan. Lebih lanjut bahwa jika bernilai 0 menunjukkan tidak ada informasi yang akan diteruskan, sebaliknya jika bernilai 1 maka informasi akan diteruskan. Dalam arsitektur LSTM terdapat tiga *gate* utama yang mengatur alur informasi, yaitu sebagai berikut:

- a. *Forget gate*, merupakan *gate* yang memutuskan informasi mana yang akan dihapus dari *cell*.
- b. *Input gate*, merupakan *gate* yang memutuskan nilai dari *input* untuk di perbarui pada *state* memori.
- c. *Output gate*, merupakan *gate* yang memutuskan apa yang akan dihasilkan sesuai dengan *input* dan memori pada *cell*

Berikut dijelaskan terkait gerbang-gerbang yang terdapat dalam sel memori LSTM:

1. *Forget gate*

Forget gate berfungsi untuk mengolah informasi setiap data *input*-an dan memilih data mana saja yang perlu disimpan atau dibuang pada *memory cells*. Pada *forget gate*, fungsi aktivasi yang digunakan ialah fungsi aktivasi *sigmoid* (σ). Hasil keluaran yang dihasilkan dari fungsi aktivasi *sigmoid* ialah antara 0 dan 1. Jika hasil keluarannya

bernilai 1 maka semua data akan disimpan dan sebaliknya jika keluarannya 0 maka semua data akan dibuang (Aldi, 2018). Rumus dari f_t dapat dilihat pada persamaan 2.1.

$$f_t = (U_f X_t + W_f h_{t-1} + b_f) \dots \dots \dots (2.1)$$

Dengan:

f_t = Forget gate.

σ = Fungsi aktivasi *sigmoid*.

U_f = Bobot *input* pada *forget gate*.

X_t = *Input* pada waktu t .

W_f = Bobot *hidden state* pada *forget gate*.

h_{t-1} = *Output hidden state* sebelumnya atau *state* pada waktu $t-1$.

b_f = Bias pada *forget gate*.

2. *Input Gate (it)*

Input gate memiliki dua *gates* yang akan dilaksanakan, pertama akan diputuskan nilai mana yang akan diperbarui menggunakan fungsi aktivasi *sigmoid*, kemudian menggunakan fungsi aktivasi *tanh* untuk membuat vektor nilai baru yang akan disimpan pada *memory cell* (Aldi, 2018). Rumus dari *input gate* dapat dilihat pada persamaan 2.2 dan 2.3.

$$it = (U_i X_t + W_i h_{t-1} + b_i) \dots \dots \dots (2.2)$$

$$\tilde{C} = \tanh(U_c X_t + W_c h_{t-1} + b_c) \dots \dots \dots (2.3)$$

Dengan:

it = *Input gate*.

\tilde{C} = *Cell* aktivasi

σ = Fungsi aktivasi *sigmoid*.

\tanh = Fungsi aktivasi *tanh*

X_t = *Input* pada waktu t .

h_{t-1} = *Ouput hidden state* sebelumnya atau *state* pada waktu $t-1$.

U_i = Bobot *input* pada *input gate*.

U_c = Bobot *input* pada *cell* aktivasi.

W_i = Bobot *hidden state* pada *input gate*.

W_c = Bobot *hidden state* pada *cell* aktivasi.

b_i = Bias pada *input gate*.

b_c = Bias pada *cell* aktivasi.

3. *Cell gate* (ct)

Pada *cell gate* nilai yang ada pada *memory cell* sebelumnya akan diganti menggunakan nilai *memory cell* yang baru. Nilai ini didapatkan dengan cara menggabungkan nilai yang ada pada *forget gate* dan *input gate* (Aldi, 2018). Rumus dari ct dapat dilihat pada persamaan 2.4.

$$ct = ft * ct-1 + it * \tilde{C} \dots\dots\dots (2.4)$$

ct = *Cell gate*.

ft = *Forget gate*.

$ct-1$ = *Ouput hidden state* sebelumnya atau *state* pada waktu $t-1$.

it = *Input gate*.

\tilde{C} = *Cell* aktivasi

4. *Output gate* (ht)

Output gate memiliki dua *gates* yang akan dilaksanakan, pertama ditentukan nilai pada bagian *memory cell* mana yang akan dikeluarkan menggunakan fungsi aktivasi *sigmoid*. Kemudian nilai ditempatkan pada *memory cell* dengan menggunakan fungsi aktivasi *tanh*. Tahapan akhir, kedua *gate* ini dikalikan sehingga menghasilkan nilai yang akan dikeluarkan (Aldi, 2018). Rumus dari *ot* dapat dilihat pada persamaan 2.5 dan 2.6.

$$ot=(UoXt+Woht-1+bo) \dots\dots\dots (2.5)$$

$$ht=ot*\tanh(ct)\dots\dots\dots (2.6)$$

Dengan:

ot = *Output gate*.

ct = *cell gate*.

σ = Fungsi aktivasi *sigmoid*.

tanh = Fungsi aktivasi *tanh*

Uo = Bobot *input* pada *output gate*.

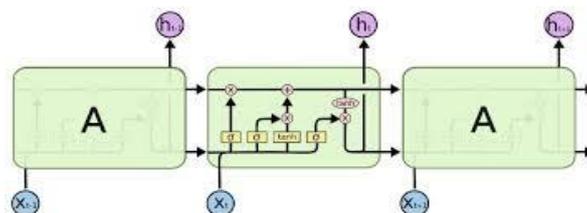
Xt = *Input* pada waktu *t*.

Wo = Bobot *hidden state* pada *output gate*.

ht-1 = *Output hidden state* sebelumnya atau *state* pada waktu *t-1*.

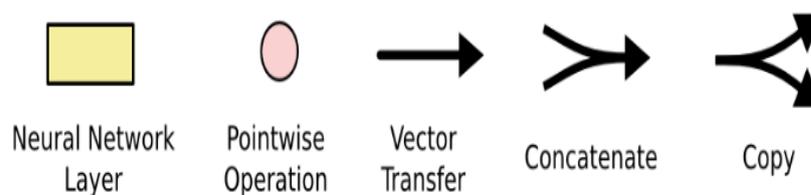
bo = Bias pada *output gate*.

Arsitektur LSTM dapat dilihat pada gambar 2.1.



Gambar 2.1 Arsitektur LSTM (Cristopher Olah, 2015)

Long Short Term Memory (LSTM) merupakan salah satu pengembangan *neural network* yang dapat digunakan untuk pemodelan data *time series* (Wiranda, 2019). LSTM diusulkan sebagai solusi untuk mengatasi terjadinya *vanishing gradient* pada RNN saat memproses data *sequential* yang panjang. LSTM dirancang untuk meningkatkan algoritma RNN yang mampu memecahkan masalah gradien yang hilang dengan menambahkan *cell state* untuk mengingat atau melupakan data. *Cell state* memuat struktur yang disebut *cell gate* atau gerbang sel. Setiap *gate* terdiri dari empat bagian yaitu *input gate*, *forget gate*, *memory-cell state gate* dan *output gate*. *Input gate* merupakan *gate* yang berfungsi mengontrol *input-an* data yang layak atau tidak untuk disimpan. *Forget gate* berfungsi mengontrol *hidden state* sebelumnya yang akan disimpan di *cell-memory* dari *hidden state* saat ini. *Memory-cell state gate* untuk memperbarui data berdasarkan informasi *input gate* dan *forget gate*. *Output gate* berfungsi untuk menghitung data keluaran dari jaringan berdasarkan status *cell memory*. Dalam setiap *cell* jaringan LSTM terdapat sebuah proses aliran data yang diwakilkan oleh notasi. Seperti pada gambar 2.2.



Gambar 2.2 Notasi LSTM (Cristopher, 2015)

Pada diagram di atas, setiap baris membawa seluruh vektor, dari *output* satu node ke *input* node lainnya. Lingkaran merah muda mewakili operasi titik, seperti penjumlahan vektor, sedangkan kotak kuning adalah lapisan jaringan saraf yang dipelajari. Penggabungan garis menunjukkan penggabungan, sedangkan percabangan garis menunjukkan kontennya sedang disalin dan salinannya pergi ke lokasi yang berbeda.

2.6. Python

Python adalah sebuah bahasa pemrograman yang digunakan untuk membuat aplikasi, perintah komputer, dan melakukan analisis data. Sebagai *general-purpose language*, *Python* bisa digunakan untuk membuat program apa saja dan menyelesaikan berbagai permasalahan. Selain itu, *Python* juga dinilai mudah untuk dipelajari.

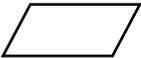
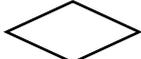
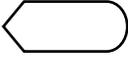
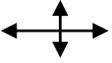
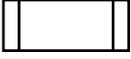
2.7. Flowchart (Diagram Alir)

flowchart adalah sebuah bagan alur yang mendeskripsikan prosesnya melalui simbol-simbol tertentu untuk digunakan sebagai alur sistem (Wibawanto 2018). *Flowchart* berperan penting dalam memutuskan sebuah langkah atau fungsionalitas dari sebuah proyek pembuatan program yang melibatkan banyak orang sekaligus. Selain itu dengan menggunakan bagan alur proses dari sebuah program akan lebih jelas, ringkas, dan mengurangi kemungkinan untuk salah penafsiran. Penggunaan *flowchart* dalam dunia pemrograman juga merupakan cara yang bagus untuk menghubungkan antara kebutuhan teknis dan non-teknis.

Fungsi utama dari *flowchart* adalah memberi gambaran jalannya sebuah program dari satu proses ke proses lainnya. Sehingga, alur program menjadi mudah dipahami oleh semua orang. Selain itu, fungsi lain dari *flowchart* adalah untuk menyederhanakan rangkaian prosedur agar memudahkan pemahaman terhadap informasi tersebut. Simbol *Flowchart* bisa dilihat pada tabel 2.2.

Tabel 2.2 Simbol-simbol pada *flowchart*

No	Simbol	Fungsi
1		Terminal, untuk memulai dan mengakhiri suatu proses / kegiatan.

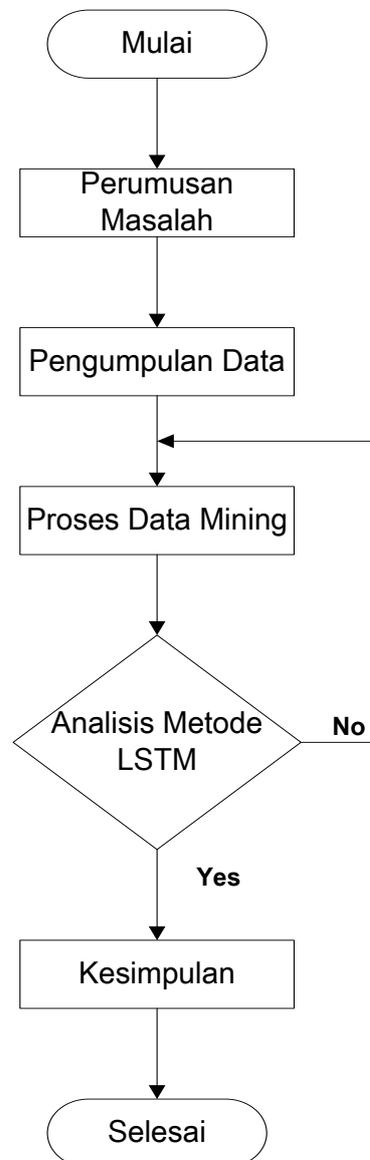
2		Proses, suatu yang menunjukkan setiap pengolahan yang dilakukan oleh komputer.
3		<i>Input</i> , untuk memasukan hasil dari suatu proses.
4		<i>Decision</i> , suatu kondisi yang akan menghasilkan beberapa kemungkinan jawaban atau pilihan
5		<i>Display, output</i> yang ditampilkan dilayar terminal
6		<i>Connector</i> , suatu prosedur akan masuk atau keluar melalui simbol ini dalam lembar yang sama.
7		<i>Off Page Connector</i> , merupakan simbol masuk atau keluarnya suatu prosedur pada kertas lembar lain.
8		Arus <i>Flow</i> , simbol ini digunakan untuk menggambarkan arus proses dari suatu kegiatan lain.
9		<i>Hard Disk Storage, input output</i> yang menggunakan <i>hard disk</i> .
10		<i>Predified Process</i> , untuk menyatakan sekumpulan langkah proses yang ditulis sebagai prosedur.
11		<i>Stored Data, input, output</i> yang menggunakan disket.
12		<i>Printer</i> , simbol ini digunakan untuk menggambarkan suatu dokumen atau kegiatan untuk mencetak suatu informasi dengan mesin <i>printer</i> .

BAB III

METODE PENELITIAN

3.1. Tahapan Penelitian

Tahapan penelitian merupakan langkah-langkah yang dilakukan oleh penulis dalam proses penelitian. Tahapan penelitian memiliki peranan penting untuk mengetahui alur terhadap penelitian yang dilakukan. Tahapan pada penelitian dapat dilihat pada gambar 3.1.



Gambar 3.1 Tahapan Penelitian

3.2. Objek dan Waktu Penelitian

Objek yang digunakan dalam penelitian ini ialah data kecelakaan lalu lintas dalam 4 tahun terakhir. Penelitian ini dilakukan pada semester ganjil tahun ajaran 2022/2023.

3.3. Data Penelitian

Data yang akan digunakan pada penelitian ini merupakan data kecelakaan lalu lintas di Kota Ternate yang diperoleh dari Polres Kota Ternate.

3.4. Perumusan Masalah

Pada tahap ini peneliti melakukan pemahaman mengenai permasalahan yang menjadi fokus dalam penelitian. Pada tahap ini juga jenis data dan lingkup yang digunakan penelitian dapat ditentukan. Berdasarkan pemahaman mengenai permasalahan yang ada, maka rumusan masalah pada penelitian ini adalah bagaimana penerapan metode *Long Short Term Memory* (LSTM) pada data kecelakaan lalu lintas di Kota Ternate

3.5. Pengumpulan Data

Pengumpulan data merupakan tahapan selanjutnya yang dilakukan oleh peneliti guna mendapatkan data-data yang dibutuhkan dalam penelitian ini. Data-data yang dibutuhkan antara lain:

a. Studi Pustaka

Studi pustaka yaitu dengan mencari bahan-bahan berdasarkan penelitian sebelumnya atau pada literatur buku yang ada kaitannya dengan penelitian ini.

b. Wawancara

Untuk memperkuat data-data yang diperoleh pada tahap studi pustaka, maka akan dilakukan wawancara dan observasi.

c. Data Sekunder

Data yang dikumpulkan dalam penelitian ini adalah data yang bersumber dari laporan polisi di Unit Laka Lantas Polres Kota Ternate dari Januari tahun 2019 sampai Oktober tahun 2022

3.6. Proses Data Mining

Tahapan *data mining* adalah tahapan yang paling penting dalam analisa KDD.

3.6.1. Tahap Data Preprocessing

Tahap *preprocessing* merupakan langkah awal sebelum data diolah, dimana kita akan melakukan analisis terhadap variabel pada dataset. Analisis data dilakukan untuk mengurangi kompleksitas data yang tidak penting, menghindari ada *noise*, mendeteksi dan menghapus elemen yang tidak relevan. Pada tahap ini kita akan melakukan proses *cleaning* dan *data transformation*.

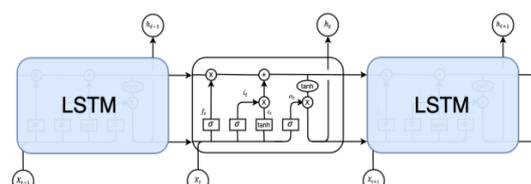
3.7. Proses Pemodelan (Modelling)

Tahap pemodelan yaitu proses membuat model dari dataset. Pada penelitian ini kita akan membuat pemodelan *Long Long Short-Term Memory* (LSTM) dengan memanfaatkan *library* pada *Python*.

3.7.1. Long Short-term Memory (LSTM)

Secara umum, tahapan dalam membangun model LSTM terdiri dari *Preprocessing Data*, inialisasi parameter, *training LSTM Network*, dan pengujian terhadap *data testing*.

Arsitektur LSTM terdiri dari tiga lapisan, yaitu lapisan *input*, lapisan tersembunyi, dan lapisan *output*. Dapat dilihat pada gambar 3.2.



Gambar 3.2 Arsitektur LSTM

Rumus LSTM dapat dilihat pada persamaan 3.2, 3.3, 3.4, 3.5, dan 3.6.

Forget Gate: $f_t = \sigma(U_f X_t + W_f h_{t-1} + b_f)$(3.2)

Input Gate: $i_t = \sigma(U_i X_t + W_i h_{t-1} + b_i)$(3.3)

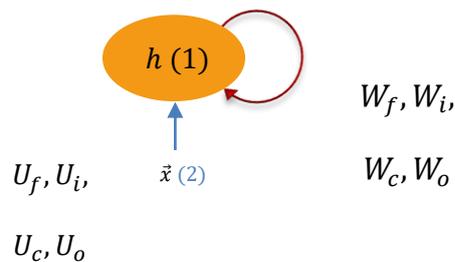
Memory Update: $c_t = f_t * c_{t-1} + i_t * \tilde{C}$ (3.4)

Output Gate: $o_t = \sigma(U_o X_t + W_o h_{t-1} + b_o)$ (3.5)

$h_t = o_t * \tanh(c_t)$ (3.6)

3.7.2 Contoh Perhitungan LSTM

Berikut ini merupakan contoh perhitungan pada jaringan LSTM.



Gambar 3.3 Jaringan LSTM

Pada gambar 3.3 menunjukkan pada jaringan LSTM tersebut memiliki 2 dimensi *input* ($\vec{x}(2)$) dengan 1 *unit* sel LSTM $h(1)$. Dalam contoh perhitungan ini, kita telah memiliki data *input* yang akan kita proses dengan menggunakan rumus-rumus LSTM yang telah dijelaskan pada poin sebelumnya. Data *input* dapat dilihat pada tabel 3.1 dimana nilai x_t berdimensi 1x2 berupa A1 dan A2 serta nilai y_t yang merupakan target. Dapat dilihat pada tabel 3.1.

Tabel 3.1 Data *Input*

A1	A2	Target
1	2	0.5
0.5	3	1

Diasumsikan kita sudah memiliki model yang siap untuk digunakan berupa kumpulan matriks bobot U , W , dan b . Setiap matriks U memiliki ukuran 1×2 yang menyatakan jumlah unit LSTM dikali dengan dimensi *input*-nya. Matriks W dan b masing-masing memiliki ukuran 1×1 serta terdapat nilai h_{t-1} dan c_{t-1} . Untuk bobot tiap matriks serta nilai h_{t-1} dan c_{t-1} dapat dilihat pada tabel 3.2.

Tabel 3.2 Bobot tiap matriks dan nilai h_{t-1} serta c_{t-1}

Bobot			
U_f		W_f	b_f
0.700	0.450	0.100	0.150
U_i		W_i	b_i
0.950	0.800	0.800	0.650
U_c		W_c	b_c
0.450	0.250	0.150	0.200
U_o		W_o	b_o
0.600	0.400	0.250	0.100
h_{t-1}		c_{t-1}	
0		0	

Selanjutnya kita akan melakukan perhitungan untuk menentukan nilai h_t dan c_t untuk *timestep* t1.

$$t1 = \langle \frac{1}{2}, 0.5 \rangle \quad h_{t-1} = 0, \quad c_{t-1} = 0$$

- Perhitungan h_t dan c_t : *timestep* t1

Langkah pertama yaitu menghitung nilai *Forget Gate* (f_t)

$$\begin{aligned}
 f_t &= \sigma(U_f X_t + W_f h_{t-1} + b_f) \\
 &= \sigma(1.600 + (0.100 \times 0) + 0.150) \\
 &= \sigma(1.750) \\
 &= \frac{1}{1+e^{-1.750}} = 0.852
 \end{aligned}$$

Selanjutnya menghitung nilai *Input Gate* (i_t) dan nilai \tilde{C}_t

$$\begin{aligned}
 i_t &= \sigma(U_i X_t + W_i h_{t-1} + b_i) \\
 &= \sigma(2.550 + (0.800 \times 0) + 0.650) \\
 &= \sigma(3.200) \\
 &= \frac{1}{1+e^{-3.200}} = 0.961
 \end{aligned}$$

$$\begin{aligned}
 \tilde{C}_t &= \tanh(U_c X_t + W_c h_{t-1} + b_c) \\
 &= \tanh(0.950 + (0.150 \times 0) + 0.200) \\
 &= \tanh(1.150) = \frac{e^{2 \times 1.150} - 1}{e^{2 \times 1.150} + 1} = 0.818
 \end{aligned}$$

Untuk tahap selanjutnya kita perlu menghitung nilai *Output Gate* (o_t)

$$\begin{aligned}
 o_t &= \sigma(U_o X_t + W_o h_{t-1} + b_o) \\
 &= \sigma(1.400 + (0.250 \times 0) + 0.100) \\
 &= \sigma(1.500) \\
 &= \frac{1}{1+e^{-1.500}} = 0.818
 \end{aligned}$$

Setelah mendapatkan nilai f_t , i_t , o_t dan \tilde{C}_t , kemudian hitung nilai *cell state* (c_t) dan

nilai h_t .

$$\begin{aligned}
 c_t &= f_t \times c_{t-1} + i_t \times \tilde{C}_t \\
 &= (0.852 \times 0) + (0.961 \times 0.818) \\
 &= 0.786
 \end{aligned}$$

Menghitung nilai h_t

$$\begin{aligned} h_t &= o_t * \tanh(c_t) \\ &= 0.818 \times \tanh(0.786) = 0.536 \end{aligned}$$

- Perhitungan h_t dan c_t : *timestep* t2

Perhitungan dilanjutkan ke *timestep* t2 menggunakan nilai h_{t-1} dan c_{t-1} dari *timestep* sebelumnya yaitu *timestep* t1 dan lakukan perhitungan untuk mencari nilai f_t, i_t, \tilde{C}_t dan o_t lalu hitung nilai c_t dan h_t .

$$t2 = \langle \frac{0.5}{3}, 1 \rangle \quad h_{t-1} = 0.536 \quad c_{t-1} = 0.786$$

Menghitung nilai *Forget Gate* (f_t)

$$\begin{aligned} f_t &= \sigma(U_f X_t + W_f h_{t-1} + b_f) \\ &= \sigma(1.700 + (0.100 \times 0.536) + 0.150) \\ &= \sigma(1.904) \\ &= \frac{1}{1+e^{-1.904}} = 0.870 \end{aligned}$$

Menghitung nilai *Input Gate* (i_t) dan nilai \tilde{C}_t

$$\begin{aligned} i_t &= \sigma(U_i X_t + W_i h_{t-1} + b_i) \\ &= \sigma(2.875 + (0.800 \times 0.536) + 0.650) \\ &= \sigma(3.954) \\ &= \frac{1}{1+e^{-3.954}} = 0.981 \end{aligned}$$

$$\begin{aligned} \tilde{C}_t &= \tanh(U_c X_t + W_c h_{t-1} + b_c) \\ &= \tanh(0.975 + (0.150 \times 0.536) + 0.200) \\ &= \tanh(1.255) = \frac{e^{2 \times 1.255} - 1}{e^{2 \times 1.255} + 1} = 0.850 \end{aligned}$$

Menghitung nilai *Output Gate* (o_t)

$$\begin{aligned}
o_t &= (U_o X_t + W_o h_{t-1} + b_o) \\
&= \sigma(1.500 + (0.250 \times 0.536) + 0.100) \\
&= \sigma(1.734) \\
&= \frac{1}{1+e^{-1.734}} = 0.850
\end{aligned}$$

Setelah mendapatkan nilai f_t , i_t , o_t dan \tilde{C}_t , kemudian hitung nilai *cell state* (c_t) dan nilai h_t

$$\begin{aligned}
c_t &= f_t \times c_{t-1} + i_t \times \tilde{C}_t \\
&= (0.870 \times 0.786) + (0.961 \times 0.850) \\
&= 1.518
\end{aligned}$$

Menghitung nilai h_t

$$\begin{aligned}
h_t &= o_t * \tanh(c_t) \\
&= 0.850 \times \tanh(1.518) = 0.772
\end{aligned}$$

3.8. Variabel Penelitian

Variabel penelitian merupakan suatu objek atau faktor yang berperan dalam peristiwa atau gejala yang akan diteliti. Variabel penelitian ini mengacu pada penelitian terdahulu yang telah dilakukan. Berikut adalah variabel yang akan diteliti:

1. Waktu Kejadian Pada penelitian ini tidak digunakan format waktu 24:00 /24 jam tetapi waktu terjadinya kecelakaan lalu lintas dibagi menjadi lima bagian yaitu:
 - a. Dini Hari (22.01–06.29)
 - b. Pagi Hari (6.30–10.00)
 - c. Siang Hari (10.01–14.59)
 - d. Sore Hari (15.00–18.00)
 - e. Malam Hari (18.01–22.00)

2. Status Jalan Variabel Status Jalan terbagi menjadi 2 bagian yaitu:
 - a. Jalan Nasional
 - b. Jalan Kota
3. Bentuk Geometri Variabel Bentuk Geometri menjabarkan bentuk/model jalan dimana terjadi kecelakaan lalu lintas, variable ini terbagi menjadi 5 yaitu:
 - a. Lurus
 - b. Bundaran
 - c. Simpang 3
 - d. Simpang 4
 - e. Jembatan
4. Kerugian Material

Varibel Kerugian Material menjabarkan tentang kerugian pada saat terjadinya kecelakaan lalu lintas,
5. Tingkat Kecelakaan Variabel ini menjelaskan seberapa besar tingkat kecelakaan yang dialami korban/pelaku saat terjadi kecelakaan lalu lintas, yaitu:
 - a. Luka Ringan
 - b. Luka Berat
 - c. Meninggal Ditempat

Ketika terjadi dua kecelakaan yang berbeda, maka diambil tingkat kecelakaan yang terparah.
6. Pihak yang terlibat Kecelakaan

Variabel pihak yang terlibat kecelakaan merupakan model dari setiap kecelakaan yang terjadi di lokasi dengan pembagian sebagai berikut:

- a. Roda 2 x Roda 2: ketika kecelakaan melibatkan kendaraan roda dua dengan roda dua, dimana termasuk seluruh jenis kendaraan roda dua, seperti sepeda motor, sepeda angin, dan sejenisnya.
- b. Roda 2 x Roda 4: ketika kecelakaan melibatkan kendaraan roda dua dengan roda empat dan termasuk seluruh kendaraan yang memiliki roda lebih dari empat, seperti antara sepeda motor dengan truk.
- c. Roda 2 x Pejalan Kaki: ketika kecelakaan melibatkan antara roda dua dengan pejalan kaki.
- d. Roda 4 x Pejalan Kaki: ketika kecelakaan melibatkan roda empat dengan pejalan kaki.
- e. Roda 2 Tunggal: ketika kecelakaan terjadi hanya melibatkan satu pihak, seperti terjadinya kecelakaan yang disebabkan karena mengantuk atau kelelahan.
- f. Roda 4 Tunggal: ketika kecelakaan terjadi hanya melibatkan satu pihak, seperti terjadinya kecelakaan yang disebabkan karena mengantuk, kelelahan atau mengalami kerusakan pada fungsi pengeremannya.

3.9. Evaluasi Faktor Kecelakaan

Pada sub bab ini menjelaskan tentang faktor apa saja yang menjadi penyebab kecelakaan lalu lintas. Faktor penyebab kecelakaan dibagi menjadi tiga bagian, yaitu:

1. Faktor Kerugian Material

Pada faktor ini akan menjelaskan hubungan antara variabel Waktu Kejadian, Status Jalan, dan Geometri dengan variabel Kerugian Material.

2. Faktor Tingkat Kecelakaan

Pada faktor ini akan menjelaskan hubungan antara variabel Waktu Kejadian, Status Jalan, dan Geometri dengan variabel Tingkat Kecelakaan.

3. Faktor Pihak yang Terlibat

Pada faktor ini akan menjelaskan hubungan antara variabel Waktu Kejadian, Status Jalan, dan Geometri dengan variabel Pihak yang Terlibat.

BAB IV

HASIL DAN PEMBAHASAN

4.1. Proses Pengumpulan Data

Pada bab ini kita membahas tentang bagaimana menganalisis data kecelakaan lalu lintas dengan menggunakan metode Exploratory Data Analysis (EDA), memprediksi jumlah kerugian material pada data kecelakaan lalu lintas, dan menampilkan pola kecelakaan lalu lintas menggunakan decision tree. Dataset yang digunakan adalah data kecelakaan yang di peroleh langsung di Polda Maluku Utara. Dataset ini diambil dari tahun 2019-2022 yang terdiri dari 753 data dengan 7 atribut yang terdiri dari luka ringan, luka berat, meninggal dunia, jenis kecelakaan, kerugian material, lokasi, dan kategori kecelakaan. Berikut proses pengumpulan data kecelakaan lalu lintas yang digunakan pada penelitian ini dapat dilihat pada tabel 4.1.

Tabel 4.1 Proses Pengumpulan Data

	luka_ringan	luka_berat	meninggal_dunia	jenis_kecelakaan	kerugian_material	lokasi	kategori_kecelakaan
2	2	0	0	tabrakan	500000	HALSEL	Ringan
3	0	0	1	tunggal	500	HALUT	Ringan
4	8	1	2	tunggal	30000000	SULA	Berat
5	1	0	1	tunggal	300000	HALSEL	Ringan
6	2	0	1	tabrakan	10000000	HALBAR	Berat
7	0	0	1	tabrakan	500000	SULA	Ringan
8	3	0	1	tabrakan	10000000	HALBAR	Berat
9	2	0	1	tabrakan	2500000	HALTIM	Ringan
10	4	0	1	tabrakan	1500000	HALUT	Ringan
11	1	1	1	tabrakan	1500000	TERNATE	Ringan
12	0	0	1	tabrakan	300000	HALSEL	Ringan
13	0	0	0	tabrakan	60000000	TIDORE	Berat
14	1	0	0	tabrakan	1500001	HALUT	Ringan
15	1	0	0	tabrakan	0	TERNATE	Ringan
16	0	0	1	tunggal	3000000	TIDORE	Ringan
17	0	0	1	tabrakan	1000000	TIDORE	Ringan
18	0	0	3	tabrakan	9000000	HALTIM	Berat
19	0	0	2	tunggal	2000000	TIDORE	Berat
20	0	0	1	tunggal	500	HALUT	Ringan
21	0	1	1	tabrakan	2000000	TERNATE	Ringan
22	0	0	1	tabrakan	500	HALTENG	Ringan

4.2. Membaca Data dari File

Pada tahap ini yaitu membaca data dari excel dengan nama file “data_accidents” kemudian disimpan kedalam variabel yang di beri nama data, dimana terdapat 753 baris dan 9 colom. Untuk lebih jelasnya dapat dilihat pada gambar 4.1.

```
data = pd.read_excel("data_accidents.xlsx")
data
```

	tahun	tanggal	luka_ringan	luka_berat	meninggal_dunia	jenis_kecelakaan	kerugian_material	lokasi	kategori_kecelakaan
0	2012	2012-11-21	2	0	0	tabrakan	500000	HALSEL	Ringan
1	2019	2019-01-03	0	0	1	tunggal	500	HALUT	Ringan
2	2019	2019-01-04	8	1	2	tunggal	30000000	SULA	Berat
3	2019	2019-01-05	1	0	1	tunggal	300000	HALSEL	Ringan
4	2019	2019-01-06	2	0	1	tabrakan	10000000	HALBAR	Berat
...
747	2022	2022-05-16	3	0	0	tabrakan	500	HALTIM	Ringan
748	2022	2022-09-25	0	0	1	tabrakan	500	HALTIM	Ringan
749	2022	2022-10-20	1	0	0	tunggal	200	HALTIM	Ringan
750	2022	2022-10-28	0	0	2	tunggal	500	HALTIM	Berat
751	2022	2022-10-30	0	0	1	tabrakan	5000000	HALTIM	Berat

752 rows × 9 columns

Gambar 4.1 Membaca Data Dari File

4.3. Menampilkan Informasi Umum Tentang Data

Pada tahap ini informasi umum dari data ditampilkan, dimana terdapat 9 colom diantaranya, tahun, tanggal, luka_ringan, luka_berat, meninggal_dunia, jenis_kecelakaan, kerugian_material, lokasi dan kategori_kecelakaan. Dengan dtype int dan object. Untuk lebih jelasnya dapat dilihat pada gambar 4.2.

```
print(data.info())

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 752 entries, 0 to 751
Data columns (total 9 columns):
#   Column                Non-Null Count  Dtype
---  -
0   tahun                 752 non-null    int64
1   tanggal               752 non-null    object
2   luka_ringan           752 non-null    int64
3   luka_berat            752 non-null    int64
4   meninggal_dunia       752 non-null    int64
5   jenis_kecelakaan     752 non-null    object
6   kerugian_material     752 non-null    int64
7   lokasi                752 non-null    object
8   kategori_kecelakaan  752 non-null    object
dtypes: int64(5), object(4)
memory usage: 53.0+ KB
None
```

Gambar 4.2 Menampilkan Informasi Umum Tentang Data

4.4. Menampilkan Beberapa Baris Pertama Data

Pada tahap ini dimana terdapat 752 data yang diambil dari tahun 2019-2022, kemudian disini hanya ditampilkan baris pertama dari data tersebut. Untuk lebih jelasnya dapat dilihat pada gambar 4.3.

```
print(data.head())
```

	tahun	tanggal	luka_ringan	luka_berat	meninggal_dunia	\
0	2012	2012-11-21	2	0	0	
1	2019	2019-01-03	0	0	1	
2	2019	2019-01-04	8	1	2	
3	2019	2019-01-05	1	0	1	
4	2019	2019-01-06	2	0	1	

	jenis_kecelakaan	kerugian_material	lokasi	kategori_kecelakaan
0	tabrakan	500000	HALSEL	Ringan
1	tunggal	500	HALUT	Ringan
2	tunggal	30000000	SULA	Berat
3	tunggal	300000	HALSEL	Ringan
4	tabrakan	10000000	HALBAR	Berat

Gambar 4.3 Menampilkan Baris Pertama Data

4.5. Statistik Deskriptif Tentang Data Numerik

Pada tahap ini adalah tahap informasi tentang data. Dimana kita bisa melihat nilai count dan mean pada colom atau atribut yang di gunakan. Terdapat beberapa atribut diantaranya, tahun, luka_ringan, luka_berat, meninggal_dunia dan kerugian_material. Untuk lebih jelasnya dapat dilihat pada gambar 4.4.

```
print(data.describe())
```

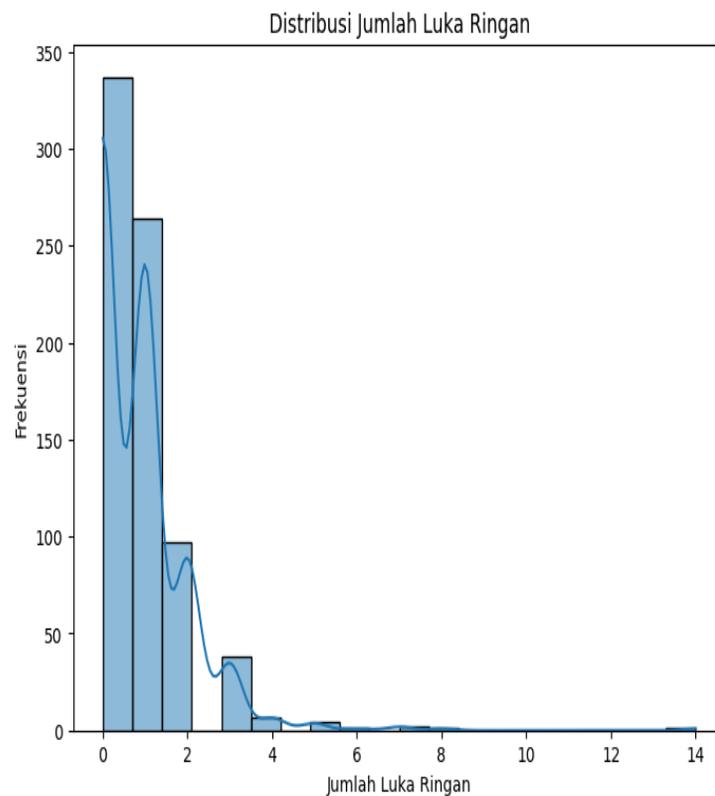
	tahun	luka_ringan	luka_berat	meninggal_dunia
count	752.000000	752.000000	752.000000	752.000000
mean	2020.631649	0.880319	0.400266	0.617021
std	1.191943	1.163829	0.649203	0.658552
min	2012.000000	0.000000	0.000000	0.000000
25%	2020.000000	0.000000	0.000000	0.000000
50%	2021.000000	1.000000	0.000000	1.000000
75%	2022.000000	1.000000	1.000000	1.000000
max	2022.000000	14.000000	5.000000	4.000000
	kerugian_material			
count	7.520000e+02			
mean	4.946618e+06			
std	1.104854e+07			
min	0.000000e+00			
25%	2.000000e+05			
50%	1.000000e+06			
75%	5.000000e+06			
max	1.000000e+08			

Gambar 4.4 Statistik Deskriptif Data Numerik

4.6. Visualisasi Histogram Jumlah Luka Ringan

Visualisasi histogram jumlah luka ringan merupakan salah satu dari Exploratory Data Analysis (EDA).

Terdapat beberapa jumlah luka ringan dengan jumlah data dari grafik yang berbeda. Jumlah luka ringan adalah 1 – 14 dengan jumlah frekuensi 0 - 350. Pada grafik jumlah luka ringan 0 menunjukkan jumlah frekuensi sebanyak 340, jumlah luka ringan pada grafik 1 menunjukkan jumlah frekuensi sebanyak 260, Jumlah luka ringan pada grafik 2 sebanyak 90, jumlah luka ringan pada grafik 3 sebanyak 35. Untuk lebih jelasnya dapat dilihat pada gambar 4.5.

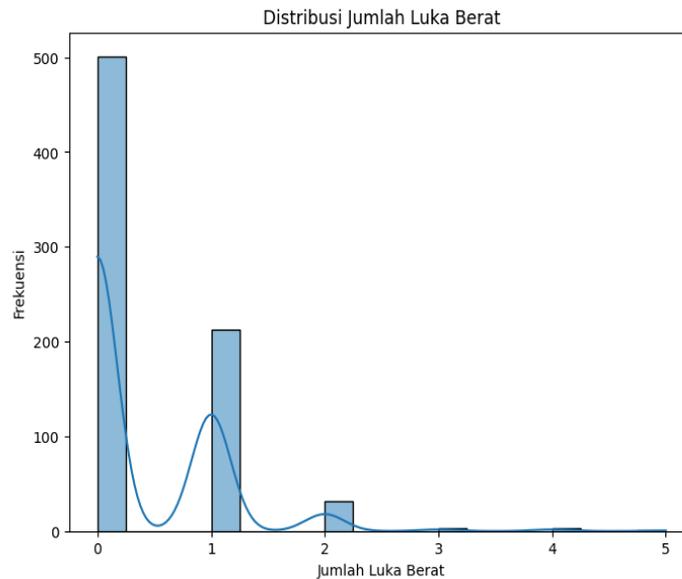


Gambar 4.5 Visualisasi Histogram Jumlah Luka Ringan

4.7. Visualisasi Histogram Jumlah Luka Berat

Pada tahap ini terdapat beberapa grafik jumlah luka berat dengan jumlah data dari grafik yang berbeda dengan jumlah frekuensi 0 - 500. Jumlah luka berat pada grafik 0 menunjukkan bahwa jumlah frekuensi sebanyak 495, jumlah luka berat pada grafik 1 menunjukkan jumlah

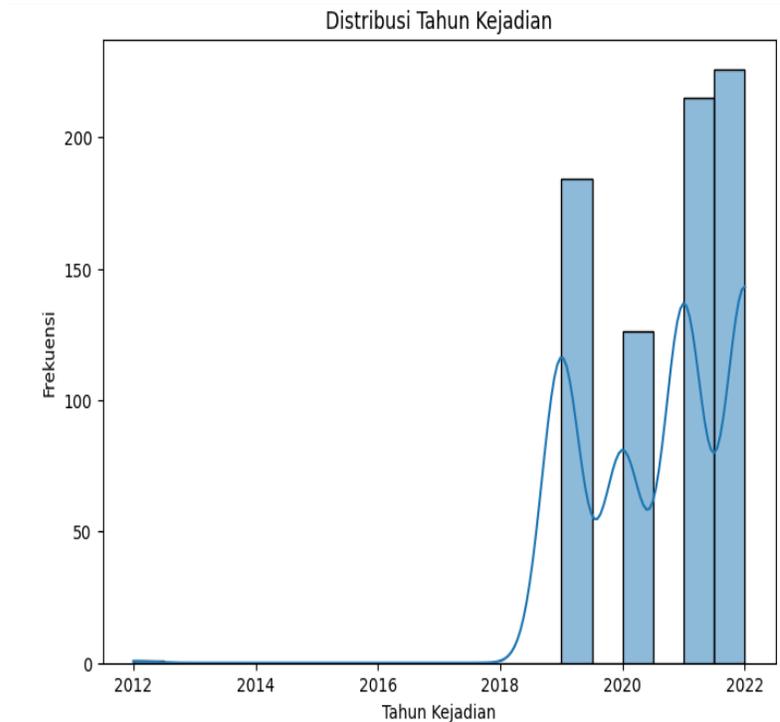
frekuensi sebanyak 205, Jumlah luka berat pada grafik 2 sebanyak 20. Untuk lebih jelasnya dapat dilihat pada gambar 4.6.



Gambar 4.6 Visualisasi Histogram Jumlah Luka Berat

4.8. Visualisasi Histogram Jumlah Kecelakaan Berdasarkan Tahun

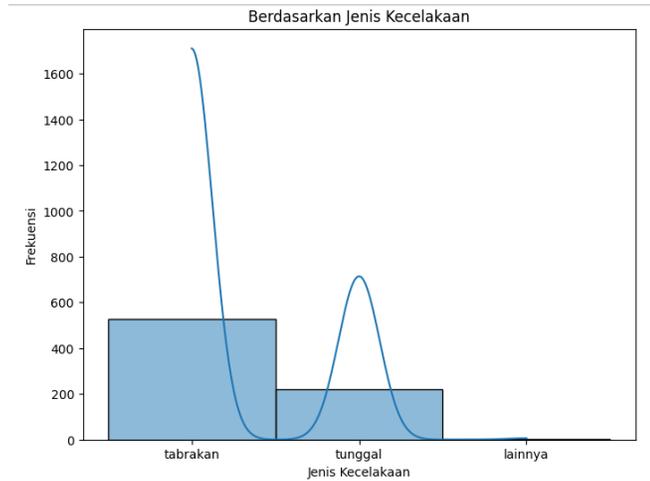
Pada tahap ini dilakukan visualisasi histogram pada data jumlah kecelakaan berdasarkan tahun. Jadi atribut yang di ambil yaitu pada tahun kejadian. Terdapat beberapa grafik histogram kecelakaan berdasarkan tahun dari tahun 2012-2022. Seperti pada tahun 2019 jumlah kecelakaan mencapai angka sebanyak 175, di tahun 2020 jumlah kecelakaan mengalami penurunan hingga mencapai angka dengan jumlah sebanyak 127, pada tahun 2021 jumlah kecelakaan mengalami kenaikan dengan menghasilkan jumlah data sebanyak 210 dan pada tahun 2022 mengalami tingkat kenaikan yang cukup tinggi hingga mencapai angka dengan jumlah data sebanyak 240. Hal ini menunjukkan bahwa jumlah kecelakaan dari tahun ke tahun semakin meningkat. Untuk lebih jelasnya dapat dilihat pada gambar 4.7.



Gambar 4.7 Visualisasi Histogram Berdasarkan Tahun

4.9. Visualisasi Histogram Jumlah Kecelakaan Berdasarkan Jenis Kecelakaan

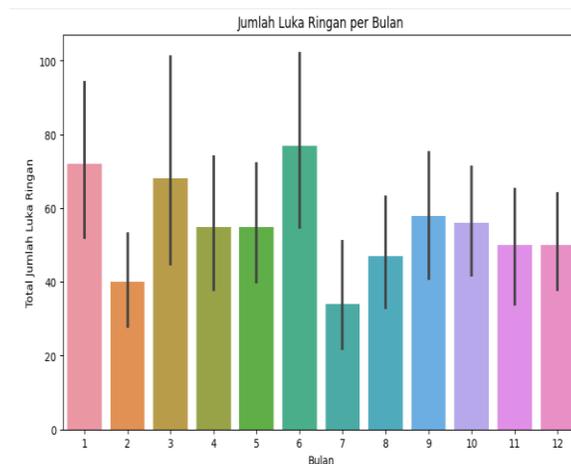
Pada tahap ini dilakukan visualisasi histogram pada data jumlah kecelakaan berdasarkan jenis kecelakaan. Jadi atribut yang di ambil hanya pada jenis kecelakaan, Terdapat beberapa grafik jenis kecelakaan dengan jumlah data yang berbeda. Terdapat tiga grafik jenis kecelakaan yaitu grafik tabrakan dan tunggal jenis kecelakaan. Dimana pada grafik dengan berdasarkan jenis kecelakaan tabrakan menghasilkan jumlah frekuensi sebanyak 531 data. Sedangkan pada grafik tunggal menunjukkan bahwa nilai frekuensi sebanyak 221 data. Hal ini menunjukkan bahwa jumlah kecelakaan pada grafik jenis kecelakaan tabrakan lebih banyak nilai frekuensi nya dibandingkan dengan jenis kecelakaan tunggal. Untuk lebih jelasnya dapat dilihat pada gambar 4.8.



Gambar 4.8 Histogram Jumlah Kecelakaan Berdasarkan Jenis Kecelakaan

4.10. Visualisasi Jumlah Luka Ringan Per Bulan

Pada tahap ini dilakukan visualisasi pada data jumlah luka ringan per bulan. Terdapat beberapa jumlah luka ringan per bulan dengan jumlah data dari grafik yang berbeda dari bulan 1 sampai dengan bulan 12. Untuk lebih jelasnya dapat dilihat pada gambar 4.9.



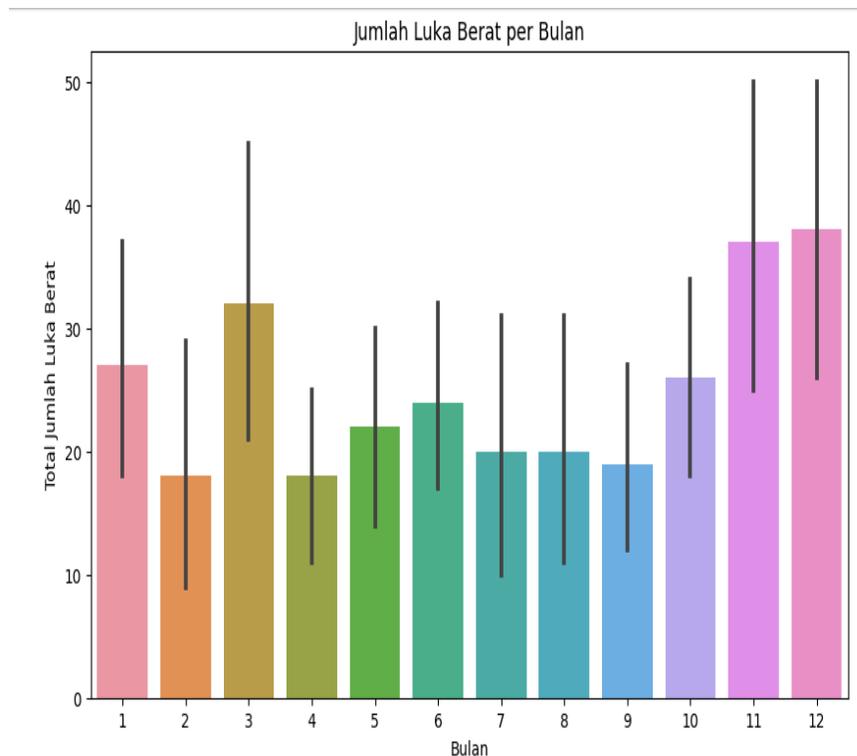
Gambar 4.9 Visualisasi Jumlah Luka Ringan Per Bulan

Pada gambar 4.9 menunjukkan bahwa jumlah luka ringan setiap bulan berbeda-beda. Dimana pada grafik bulan 1 jumlah luka ringan mencapai 69, dan pada grafik bulan ke 7 jumlah luka

ringan sebanyak 33. Hasil dari grafik visualisasi jumlah luka ringan per bulan ini menunjukkan bahwa jumlah luka ringan yang paling meningkat terdapat pada bulan 6 dengan jumlah data sebanyak 70.

4.11. Visualisasi Jumlah Luka Berat Per Bulan (Berdasarkan Tanggal Kejadian)

Pada tahap ini dilakukan visualisasi pada data jumlah luka berat per bulan, tahap ini juga sama seperti pada tahap sebelumnya. Terdapat beberapa jumlah luka berat per bulan dengan jumlah data dari grafik yang berbeda dari bulan 1 sampai dengan bulan 12. Untuk lebih jelasnya dapat dilihat pada gambar 4.10.



Gambar 4.10 visualisasi jumlah luka berat per bulan

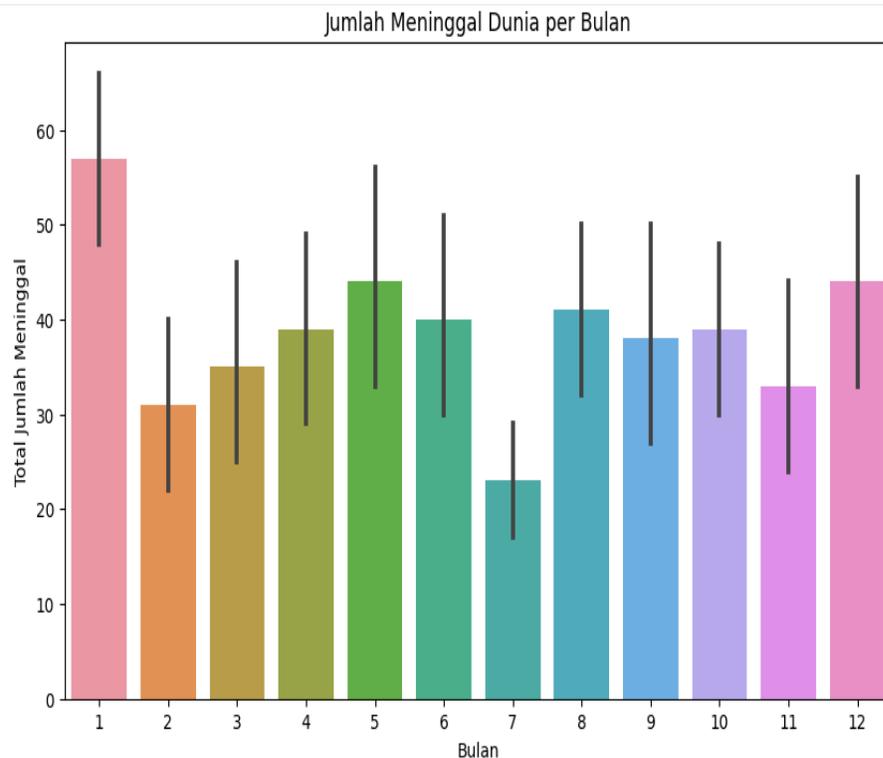
Pada gambar 4.10 menunjukkan bahwa jumlah luka berat pada setiap bulan berbeda-beda.

Dimana pada grafik bulan 1 jumlah luka berat mencapai 21 data, dan pada grafik bulan ke 4

jumlah luka berat sebanyak 18. Hasil dari grafik visualisasi jumlah luka berat per bulan ini menunjukkan bahwa jumlah luka berat yang paling meningkat terdapat pada bulan 12 dengan jumlah data sebanyak 37.

4.12. Visualisasi Jumlah Meninggal Dunia Per Bulan (Berdasarkan Tanggal Kejadian)

Pada tahap ini dilakukan visualisasi pada data jumlah meninggal dunia per bulan, tahap ini juga sama seperti pada tahap sebelumnya. Terdapat beberapa jumlah meninggal dunia per bulan dengan jumlah data dari grafik yang berbeda dari bulan 1 sampai dengan bulan 12. Untuk lebih jelasnya dapat dilihat pada gambar 4.11.



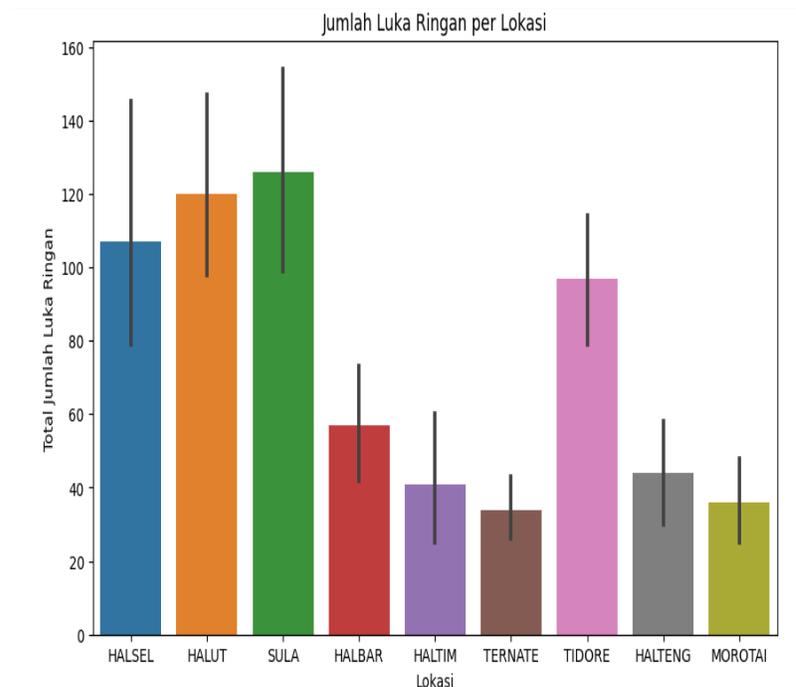
Gambar 4.11 Visualisasi Jumlah Meninggal Dunia

Pada gambar 4.11 menunjukkan bahwa jumlah meninggal dunia pada setiap bulan berbeda-beda. Dimana pada grafik bulan 1 jumlah meninggal dunia mencapai 55 dan pada grafik bulan

ke 3 jumlah meninggal dunia sebanyak 33. Hasil dari grafik visualisasi jumlah meninggal dunia per bulan ini menunjukkan bahwa jumlah meninggal dunia yang paling sedikit terdapat pada bulan 7 dengan jumlah data sebanyak 23.

4.13. Visualisasi Jumlah Luka Ringan Per Lokasi (Berdasarkan Tanggal Kejadian)

Pada tahap ini dilakukan visualisasi jumlah luka ringan per lokasi. Dimana pada tahap ini atribut yang digunakan yaitu lokasi dan total jumlah luka ringan. Jadi pada data yang diambil terdapat beberapa lokasi yang ada di Maluku Utara diantaranya Halsel, Halut, Sula, Halbar, Haltim, Ternate Tidore, Halteng dan Morotai. Untuk lebih jelasnya dapat dilihat pada gambar 4.12.



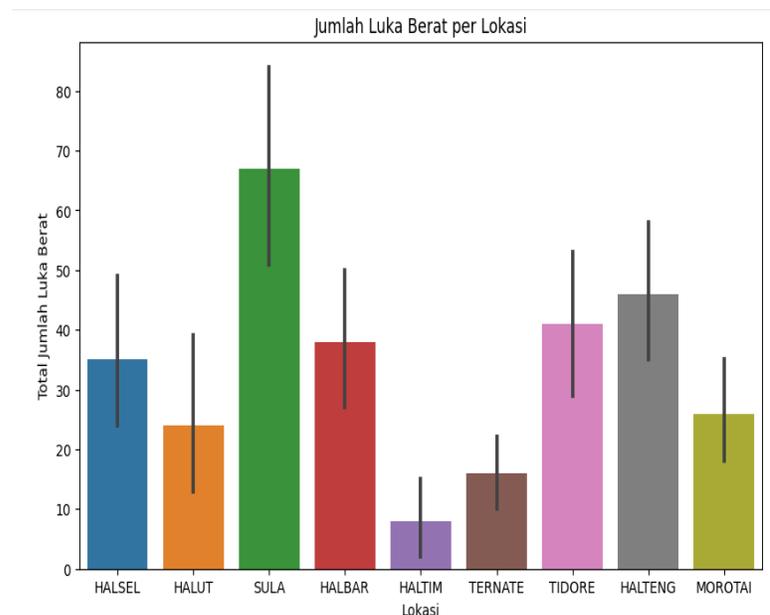
Gambar 4.12 visualisasi jumlah luka ringan per lokasi

Pada gambar 4.12 menunjukkan bahwa grafik jumlah luka ringan per lokasi berbeda-beda. Dimana jumlah luka ringan di Halsel pada grafik sebanyak 105, pada grafik jumlah luka ringan

di Halut sebanyak 120, grafik jumlah luka ringan di Halbar sebanyak 50, grafik jumlah luka ringan di Haltim sebanyak 40, grafik jumlah luka ringan di Halteng sebanyak 45, grafik jumlah luka ringan pada Kota Tidore sebanyak 45 dan Jumlah grafik pada pulau Morotai sebanyak 35. Hasil dari grafik visualisasi jumlah luka ringan ini menunjukkan bahwa jumlah luka ringan yang paling meningkat terdapat pada Kabupaten Kepulauan Sula dengan jumlah data sebanyak 125, dan jumlah luka ringan yang paling sedikit terjadi terdapat pada Kota Ternate.

4.14. Visualisasi Jumlah Luka Berat Per Lokasi (Berdasarkan Tanggal Kejadian)

Pada tahap ini dilakukan visualisasi jumlah luka berat per lokasi. Pada tahap ini juga sama seperti tahap sebelumnya, dimana atribut yang digunakan yaitu lokasi dan total jumlah luka berat. Untuk lebih jelasnya dapat dilihat pada gambar 4.13.



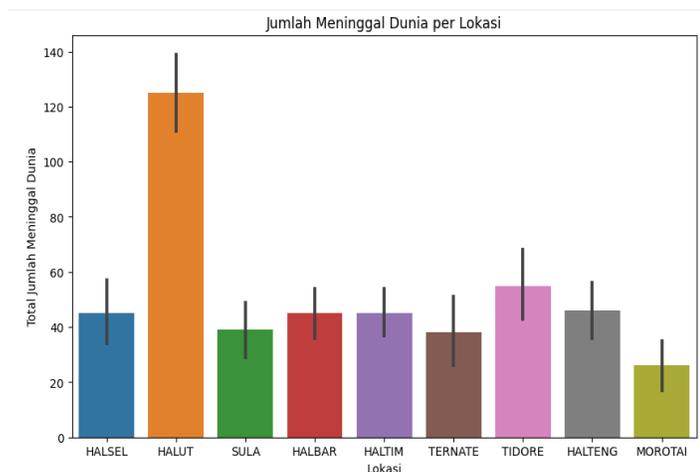
Gambar 4.13 Visualisasi Jumlah Luka Berat Per Lokasi

Pada gambar 4.13 menunjukkan bahwa grafik jumlah luka berat per lokasi berbeda-beda. Dimana jumlah luka berat di Halsel pada grafik sebanyak 35, pada grafik jumlah luka berat di

Halut sebanyak 22, grafik jumlah luka berat di Halbar sebanyak 39, grafik jumlah luka berat di Haltim sebanyak 5, grafik jumlah luka berat di Halteng sebanyak 50, grafik jumlah luka berat pada Kota Tidore sebanyak 45 dan Jumlah grafik luka berat pada pulau Morotai sebanyak 25. Hasil dari grafik visualisasi jumlah luka berat ini menunjukkan bahwa jumlah luka berat yang paling meningkat terdapat pada Kabupaten Kepulauan Sula dengan jumlah data sebanyak 65, dan jumlah luka berat yang paling sedikit terjadi terdapat di Haltim dengan jumlah sebanyak 5 jumlah luka berat.

4.15. Visualisasi Jumlah Meninggal Dunia Per Lokasi (Berdasarkan Tanggal Kejadian)

Pada tahap ini dilakukan visualisasi jumlah meninggal dunia per lokasi. Pada tahap ini dimana label yang digunakan yaitu lokasi dan total jumlah meninggal dunia. Jadi terdapat beberapa lokasi yang ada di Maluku Utara pada data yang di ambil. Untuk lebih jelasnya dapat dilihat pada gambar 4.14.



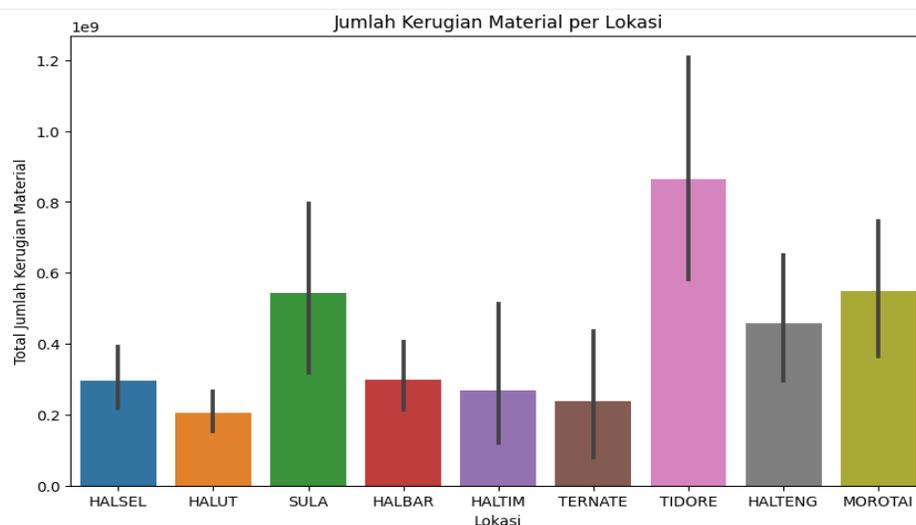
Gambar 4.14 Visualisasi Jumlah Meninggal Dunia Per Lokasi

Pada gambar 4.14 menunjukkan bahwa grafik jumlah meninggal dunia per lokasi berbeda-beda. Dimana jumlah meninggal dunia di Halsel pada grafik sebanyak 45, pada grafik jumlah

meninggal dunia di Halut sebanyak 120, grafik jumlah meninggal dunia di Halbar sebanyak 40, grafik jumlah meninggal dunia di Haltim sebanyak 40, grafik jumlah meninggal dunia di Halteng sebanyak 39, grafik jumlah meninggal dunia pada Kota Tidore sebanyak 55 dan Jumlah grafik meninggal dunia pada pulau Morotai sebanyak 25. Hasil dari grafik visualisasi jumlah meninggal dunia ini menunjukkan bahwa jumlah meninggal dunia yang paling meningkat terdapat pada Halmahera Utara dengan jumlah data sebanyak 120, dan jumlah meninggal dunia yang paling sedikit terjadi terdapat di Morotai dengan jumlah sebanyak 25.

4.16. Visualisasi Jumlah Kerugian Material Per Lokasi (Berdasarkan Tanggal Kejadian)

Pada tahap ini kita menggunakan atribut lokasi dan total jumlah kerugian material. Jadi berdasarkan tanggal kejadian kita dapat melihat jumlah kerugian material per lokasi, Untuk lebih jelasnya dapat dilihat pada gambar 4.15.

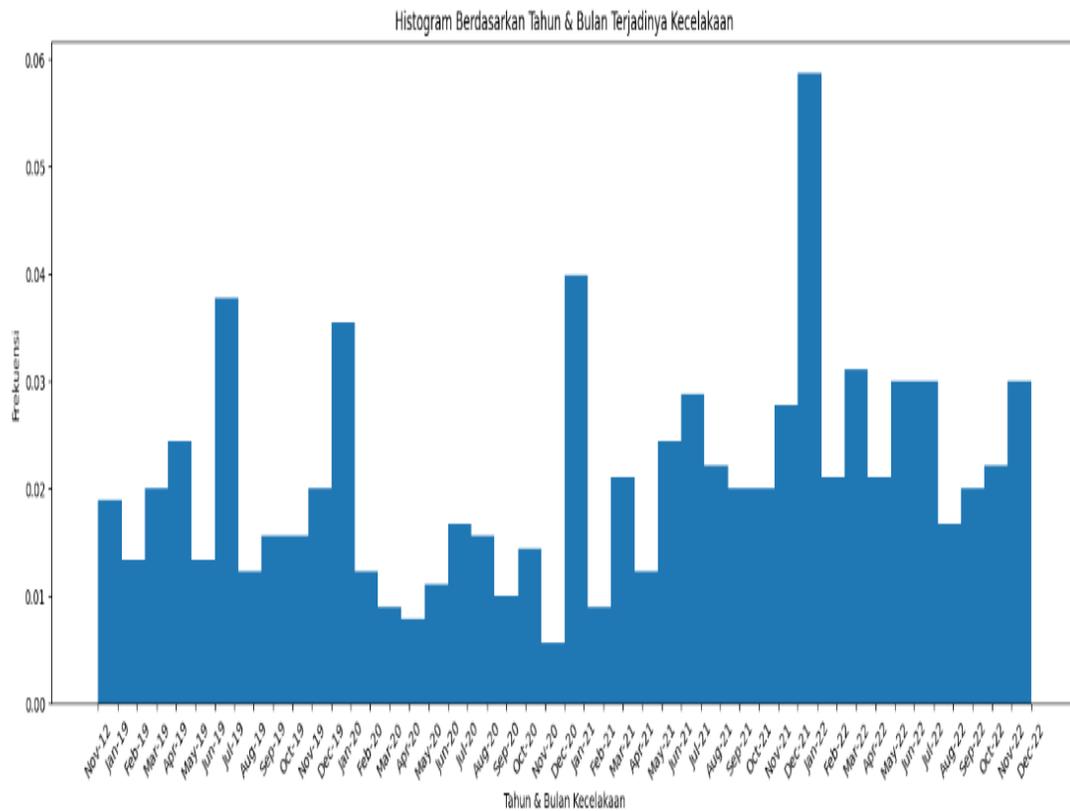


Gambar 4.15 Kerugian Material Per Lokasi

Hasil dari grafik visualisasi jumlah kerugian material dapat diperoleh perbandingan nilai hasil prediksi berdasarkan tanggal kejadian.

4.17. Histogram Berdasarkan tahun dan Bulan Kecelakaan

Dari histogram ini bisa dilihat bahwa trend frekuensi kecelakaan tertinggi terjadi di bulan desember tahun 2021 dan terendah di bulan november 2020. Untuk lebih jelasnya dapat dilihat pada gambar 4.16.

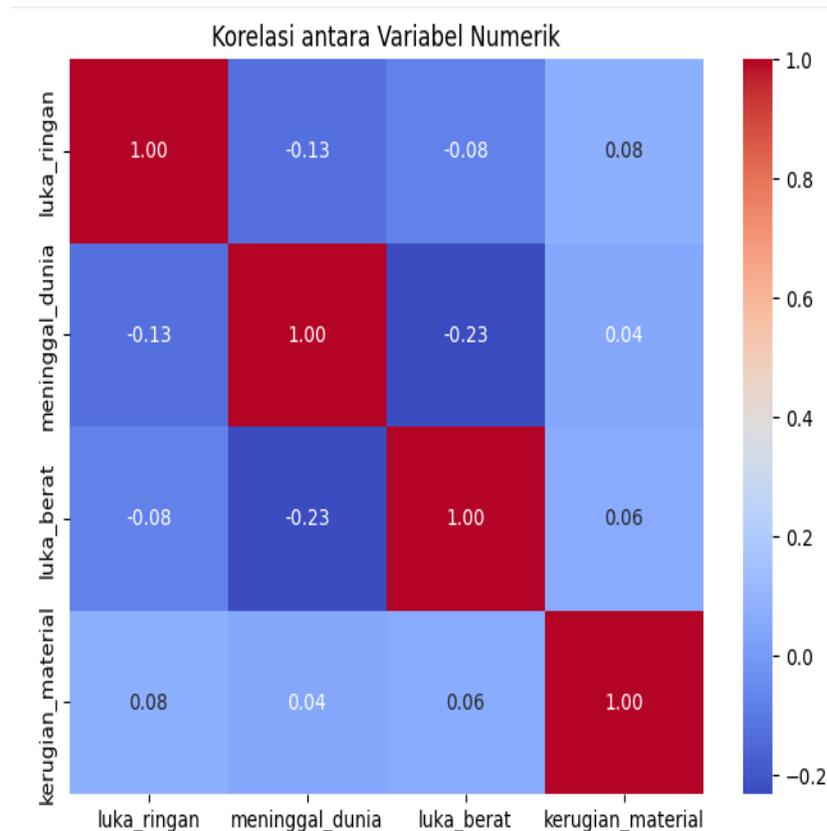


Gambar 4.16 Histogram Berdasarkan tahun Dan Bulan Kecelakaan

4.18. Visualisasi Korelasi Antara Variabel Numerik

Pada tahap ini dilakukan korelasi antara variabel numerik dengan menggunakan beberapa parameter yang ada. Disini diambil sebagian data dari Data Frame data yaitu kolom-kolom dengan nama "luka ringan, meninggal dunia, luka berat dan kerugian material", kemudian menghitung matriks korelasi antara kolom-kolom ini menggunakan metode `corr()` dari Pustaka

pandas, hasilnya disimpan dalam variabel `correlation_matrix`. Untuk lebih jelasnya dapat dilihat pada gambar 4.17.



Gambar 4.17 Visualisasi Korelasi Variabel Numerik

4.19. Implementasi Model *Long Short Term Memory*

LSTM merupakan algoritma Deep Learning yang populer dan cocok digunakan untuk membuat prediksi dan klasifikasi yang berhubungan dengan waktu.

4.19.1. Preprocessing Data

Pada tahap ini Misalnya, kita ingin melakukan klasifikasi untuk tingkat kecelakaan dimana: Jika jumlah luka berat > 0 , kita anggap sebagai kecelakaan berat (1), sebaliknya, kecelakaan ringan (0). Untuk lebih jelasnya dapat dilihat pada gambar 4.18.

	tahun	tanggal	luka_ringan	luka_berat	meninggal_dunia	jenis_kecelakaan	kerugian_material	lokasi	kategori_kecelakaan
0	2012	2012-11-21	2	0	0	tabrakan	500000	HALSEL	Ringan
1	2019	2019-01-03	0	0	1	tunggal	500	HALUT	Ringan
2	2019	2019-01-04	8	1	2	tunggal	30000000	SULA	Berat
3	2019	2019-01-05	1	0	1	tunggal	300000	HALSEL	Ringan
4	2019	2019-01-06	2	0	1	tabrakan	10000000	HALBAR	Berat
...
747	2022	2022-05-16	3	0	0	tabrakan	500	HALTIM	Ringan
748	2022	2022-09-25	0	0	1	tabrakan	500	HALTIM	Ringan
749	2022	2022-10-20	1	0	0	tunggal	200	HALTIM	Ringan
750	2022	2022-10-28	0	0	2	tunggal	500	HALTIM	Berat
751	2022	2022-10-30	0	0	1	tabrakan	5000000	HALTIM	Berat

752 rows × 9 columns

Gambar 4.18 Preprocessing Data

4.19.2. Memilih Atribut Yang Akan Digunakan

Pada tahap ini tanggal kejadian di abaikan. Atribut yang di gunakan yaitu luka ringan, luka berat, meninggal dunia, dan jenis kecelakaan. Untuk lebih jelasnya dapat dilihat pada gambar 4.19.

```
X = data(['luka_ringan', 'luka_berat', 'meninggal_dunia', 'jenis_kecelakaan_encoded', 'kerugian_material']).values
y = data(['kategori_kecelakaan_encoded']).values
```

Gambar 4.19 Memilih Atribut

4.19.3. Evaluasi Model Pada Data Testing

Pada tahap ini *accuracy* yang di dapat adalah 98,68%. Untuk lebih jelasnya dapat dilihat pada gambar 4.20.

```

loss, accuracy = model.evaluate(X_test, y_test)
print(f"Testing Accuracy: {accuracy*100:.2f}%")

5/5 [=====] - 0s 4ms/step - loss: 0.0651 - accuracy: 0.9868
Testing Accuracy: 98.68%

```

Gambar 4.20 Evaluasi Model Pada Data Testing

4.20. Tampilkan Grafik *Decision Tree*

Pada tahap ini kita tampilkan grafik decision tree menggunakan matplotlib. Visualisasi hasil dari pohon keputusan yang digunakan dalam kode di atas akan membantu memahami bagaimana model pohon keputusan membuat keputusan berdasarkan fitur-fitur yang ada dalam dataset.

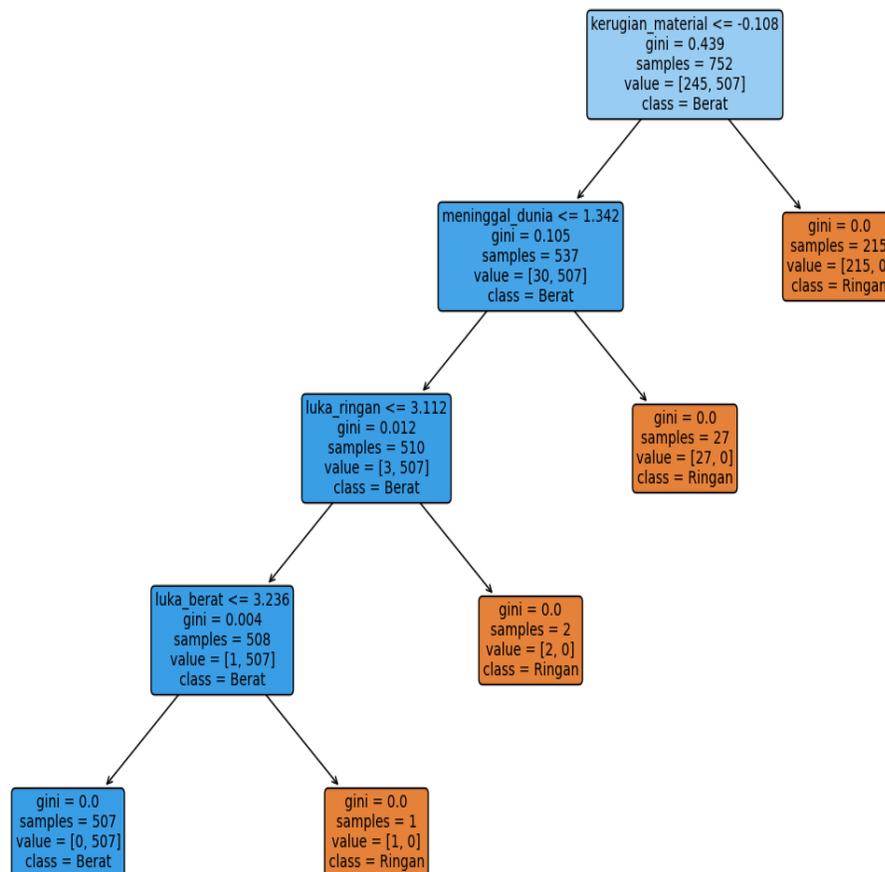
Node (simpul) Pohon Keputusan: Setiap persegi dalam visualisasi adalah simpul (node) dalam pohon keputusan. Simpul pertama yang Anda lihat pada visualisasi adalah simpul akar (root node).

Aturan Pemisahan: Setiap simpul memuat sebuah aturan pemisahan yang digunakan untuk membagi data ke dalam dua atau lebih cabang (branch). Aturan ini berdasarkan salah satu fitur dalam dataset.

Gini Impurity atau Entropy: Terkadang, Anda akan melihat angka di dalam simpul. Ini adalah ukuran impuritas (Gini impurity) atau entropi yang mengukur ketidakmurnian dataset pada simpul tersebut. Semakin rendah impuritas atau entropi, semakin baik pemisahan pada simpul tersebut. **Nilai Keputusan (Class):** Di simpul-simpul akhir (simpul daun) pohon, Anda akan melihat nilai-nilai yang menunjukkan kelas atau label yang diprediksi oleh pohon untuk data yang sampai pada simpul tersebut.

Cabang: Garis-garis panah yang menghubungkan simpul-simpul adalah cabang (branch) yang mengindikasikan jalur yang harus diikuti oleh data sesuai dengan aturan pemisahan. Jika data memenuhi aturan tertentu, mereka akan mengikuti cabang ke simpul anak yang sesuai.

Berhenti atau Daun: Simpul-simpul terakhir dalam pohon (simpul daun) adalah simpul di mana model membuat prediksi kelas. Prediksi kelas ini ditunjukkan oleh nilai-nilai dalam simpul daun tersebut. Untuk lebih jelasnya dapat dilihat pada gambar 4.21.



Gambar 4.21 Grafik Decision Tree

Dalam konteks decision tree, istilah "Gini" mengacu pada indeks Gini atau Gini impurity, yang digunakan untuk mengukur seberapa homogen atau campur aduk suatu set data. Nilai Gini berkisar antara 0 dan 1, di mana nilai 0 menunjukkan bahwa semua elemen pada set data tersebut memiliki label yang sama, sementara nilai 1 menunjukkan bahwa set data tersebut memiliki campuran label yang merata.

Secara spesifik, jika suatu node memiliki Gini impurity sebesar 0, itu berarti semua sampel pada node tersebut memiliki label yang sama. Sebaliknya, jika Gini impurity sebesar 0.5, itu menunjukkan adanya campuran label yang merata pada node tersebut.

Dalam konteks decision tree, node pada setiap level tree akan memiliki nilai Gini impurity, dan tujuan dari algoritma adalah membagi data pada setiap node menjadi dua subset sehingga nilai Gini impurity pada kedua subset tersebut semakin rendah.

Sebagai contoh, jika Anda melihat output dari decision tree yang menggunakan indeks Gini, seperti pada plot tree yang dihasilkan dari kodingan yang Anda berikan, mungkin akan melibatkan baris informasi yang menyertakan nilai Gini dan jumlah sampel pada suatu node. Contoh seperti "Gini = 0.42" dan "Sample = 759" pada node tertentu menunjukkan bahwa pada node tersebut, indeks Gini adalah 0.42 dan terdapat 759 sampel pada node tersebut.

Jumlah sampel pada node (Sample) menunjukkan berapa banyak data yang termasuk dalam node tersebut, dan nilai Gini memberikan indikasi seberapa baik data tersebut terpisah menjadi kelas-kelas yang berbeda pada node tersebut. Semakin rendah nilai Gini, semakin baik pemisahan kelasnya.

BAB V

PENUTUP

5.1. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan dengan menerapkan metode *Long Short Term Memory* (LSTM) untuk mengetahui hasil implementasi dan analisis data kecelakaan lalu lintas dan dapat menampilkan pola kecelakaan lalu lintas dengan menggunakan metode *Decision Tree* maka dapat diperoleh beberapa kesimpulan sebagai berikut :

1. Penulis melakukan analisa berdasarkan dataset dengan atribut tanggal kecelakaan, lokasi kecelakaan, nominal kecelakaan, lokasi kecelakaan, nominal kerugian, korban luka ringan, berat dan meninggal dunia, penulis menyimpulkan kecelakaan dengan jenis tabrakan berpotensi menghasilkan kerugian material lebih besar disbanding kecelakaan jenis tunggal atau lainnya, tabrakan juga berpotensi menimbulkan korban meninggal dunia lebih besar. Jika melihat berdasarkan tanggal pola kecelakaan sempat mengalami penurunan di tahun 2020 akan tetapi setelahnya terus mengalami kenaikan, berdasarkan data Kota Tidore mengalami kerugian material paling besar dari kecelakaan lalu lintas dan Kota Halut menjadi Kota paling banyak korban meninggal dunia.
2. Analisis yang dilakukan dengan menggunakan metode *Exploratory Data Analysis* (EDA), dapat menampilkan hasil analisis dengan menggunakan grafik pada data kecelakaan lalu lintas di Kota Ternate.
3. Berdasarkan hasil dari *data mining* kecelakaan berat atau ringan dipengaruhi dari besarnya nilai kerugian material karena jika nilai kerugian material besar maka sudah dipastikan juga menimbulkan korban meninggal dunia.

4. Dengan menggunakan algoritma *Decision Tree* dapat membantu memahami bagaimana model pohon keputusan membuat keputusan berdasarkan fitur-fitur yang ada dalam dataset.

5.2. SARAN

Agar penelitian yang berjudul implementasi dan analisis data mining untuk pencarian pola kecelakaan menggunakan metode *Long Short Term Memory* ini bermanfaat dimasa sekarang dan yang akan datang, maka penulis dapat memberikan saran sebagai berikut:

1. Harapan penulis untuk pemerintah Kota Ternate dapat memfokuskan untuk meningkatkan keamanan lalu lintas di Kota Tidore dan Halmahera Utara.
2. Pada peneliti selanjutnya dapat menambahkan perbandingan metode algoritma lainnya dengan studi kasus yang berbeda untuk melihat hasil yang di dapatkan
3. Dapat menampilkan pola penyebab kecelakaan lalu lintas dengan menggunakan metode yang lain agar dapat melihat hasil yang baik.
4. Menambahkan manualisasi perhitungan *Decision Tree* pada pencarian pola kecelakaan lalu lintas dengan studi kasus yang berbeda

DAFTAR PUSTAKA

- Aldi, M. W. P., Jondri, & Aditsania, A. 2018, Analisis dan Implementasi *Long Short Term Memory Neural Network* untuk Prediksi Harga *Bitcoin*. *Jurnal Informatika*, 5, No(2), 3548.
- Alimuddin, W., Tungadi, E., & Saharuna, Z. (2020). Analisis Tingkat Kecelakaan Lalu Lintas dengan Metode *Association Rule* Menggunakan Algoritma Apriori Analisis Tingkat Kecelakaan Lalu Lintas dengan Metode *Association Rule*. *Researchgate.Net, January*,206–210.
- Alwie, rahayu deny danar dan alvi furwanti, Prasetio, A. B., Andespa, R., Lhokseumawe, P. N., & Pengantar, K. 2020, Tugas Akhir. *Jurnal Ekonomi Volume 18*, Nomor 1 Maret201, 2(1), 41–49.
- Aprian, B. A., Azhar, Y., & Setya Nastiti, V. R. 2020, Prediksi Pendapatan Kargo Menggunakan Arsitektur *Long Short Term Memory*. *Jurnal Komputer Terapan*, 6(Vol. 6 No. 2 (2020)), 148–157.
- Cristopher, Olah. (2015, 27 Agustus). *Understanding LSTM Networks*.
- David.(2008). Kajian Kecelakaan Lalu Lintas Di Jalan Perintis Kemerdekaan Jalan Kartini. 174.
- Efendi, M. F. (2019). Analisis pola kecelakaan lalu lintas di surabaya menggunakan algoritma *fp-growth*.
- I swari, L., & Ayu, E. G. (2015). Pemanfaatan Algoritma *K-Means* Untuk Pemetaan Hasil Klasterisasi Data Kecelakaan Lalu Lintas. *Teknoin*, 21(1), 1–13.
- Khumaidi, A., Raafi'udin, R., & Solihin, I. P. 2020, Pengujian Algoritma *Long Short Term Memory* untuk Prediksi Kualitas Udara dan Suhu Kota Bandung. *Jurnal Telematika*, 15(1), 13–18.
- Rubiyanti.(2018). Analisa Kecelakaan Lalu Lintas Menggunakan Metode Algoritma C4.5 Dan *Naïve Bayes* (Studi Kasus Di Kabupaten Ponorogo). *Komputek*, 2(1), 69.
- Wibawanto. 2018, Bab li T. Pustaka. Sumber *Elektronika*, VI(7), 4–28.
<http://electrozone94.blogspot.co.id/2013/10/panel-surya->
- Wiranda. 2019, Penerapan *Long Short Term Memory* Pada Data *Time Series* Untuk Memprediksi Penjualan Produk Pt. Metiska Farma. *Jurnal Nasional Pendidikan Teknik Informatika (JANAPATI)*, 8(3), 184–196.

SOURCE CODE

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.preprocessing import LabelEncoder
data = pd.read_excel("data_accidents.xlsx")
data
print(data.info())
print(data.head())
print(data.describe())
plt.figure(figsize=(8, 6))
sns.histplot(data['luka_ringan'], bins=20, kde=True)
plt.xlabel('Jumlah Luka Ringan')
plt.ylabel('Frekuensi')
plt.title('Distribusi Jumlah Luka Ringan')
plt.show()
plt.figure(figsize=(8, 6))
sns.histplot(data['luka_berat'], bins=20, kde=True)
plt.xlabel('Jumlah Luka Berat')
plt.ylabel('Frekuensi')
plt.title('Distribusi Jumlah Luka Berat')
plt.show()
plt.figure(figsize=(8, 6))
sns.histplot(data['tanggal'], bins=20, kde=True)
plt.xlabel('Tanggal Kejadian')
plt.ylabel('Frekuensi')
plt.title('Distribusi Tanggal Kejadian')
plt.show()
```

```
plt.figure(figsize=(8, 6))
sns.histplot(data['tahun'], bins=20, kde=True)
plt.xlabel('Tahun Kejadian')
plt.ylabel('Frekuensi')
plt.title('Distribusi Tahun Kejadian')
plt.show()

plt.figure(figsize=(8, 6))
sns.histplot(data['jenis_kecelakaan'], bins=20, kde=True)
plt.xlabel('Jenis Kecelakaan')
plt.ylabel('Frekuensi')
plt.title('Berdasarkan Jenis Kecelakaan')
plt.show()

data['tanggal'] = pd.to_datetime(data['tanggal'])
data['bulan'] = data['tanggal'].dt.month

plt.figure(figsize=(10, 6))
sns.barplot(x='bulan', y='luka_ringan', data=data, estimator=sum)
plt.xlabel('Bulan')
plt.ylabel('Total Jumlah Luka Ringan')
plt.title('Jumlah Luka Ringan per Bulan')
plt.show()

plt.figure(figsize=(10, 6))
sns.barplot(x='bulan', y='luka_berat', data=data, estimator=sum)
plt.xlabel('Bulan')
plt.ylabel('Total Jumlah Luka Berat')
plt.title('Jumlah Luka Berat per Bulan')
plt.show()

plt.figure(figsize=(10, 6))
sns.barplot(x='bulan', y='meninggal_dunia', data=data, estimator=sum)
```

```
plt.xlabel('Bulan')
plt.ylabel('Total Jumlah Meninggal')
plt.title('Jumlah Meninggal Dunia per Bulan')
plt.show()
plt.figure(figsize=(10, 6))
sns.barplot(x='lokasi', y='luka_ringan', data=data, estimator=sum)
plt.xlabel('Lokasi')
plt.ylabel('Total Jumlah Luka Ringan')
plt.title('Jumlah Luka Ringan per Lokasi')
plt.show()
plt.figure(figsize=(10, 6))
sns.barplot(x='lokasi', y='luka_berat', data=data, estimator=sum)
plt.xlabel('Lokasi')
plt.ylabel('Total Jumlah Luka Berat')
plt.title('Jumlah Luka Berat per Lokasi')
plt.show()
plt.figure(figsize=(10, 6))
sns.barplot(x='lokasi', y='meninggal_dunia', data=data, estimator=sum)
plt.xlabel('Lokasi')
plt.ylabel('Total Jumlah Meninggal Dunia')
plt.title('Jumlah Meninggal Dunia per Lokasi')
plt.show()
plt.figure(figsize=(10, 6))
sns.barplot(x='lokasi', y='kerugian_material', data=data, estimator=sum)
plt.xlabel('Lokasi')
plt.ylabel('Total Jumlah Kerugian Material')
plt.title('Jumlah Kerugian Material per Lokasi')
plt.show()
```

```

correlation_matrix = data[['luka_ringan', 'meninggal_dunia', 'luka_berat', 'kerugian_material']].corr()
plt.figure(figsize=(8, 6))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt=".2f")
plt.title('Korelasi antara Variabel Numerik')
plt.show()

import numpy as np
import pandas as pd

from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder, StandardScaler
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import LSTM, Dense, Dropout
from tensorflow.keras.utils import to_categorical

import tensorflow as tf

label_encoder = LabelEncoder()

data['jenis_kecelakaan_encoded'] = label_encoder.fit_transform(data['jenis_kecelakaan'])
data['kategori_kecelakaan_encoded'] = label_encoder.fit_transform(data['kategori_kecelakaan'])

data

X = data[['luka_ringan', 'luka_berat', 'meninggal_dunia', 'jenis_kecelakaan_encoded',
'kerugian_material']].values

y = data['kategori_kecelakaan_encoded'].values

model = Sequential()

model.add(LSTM(64, input_shape=(1, X_train.shape[2]), activation='relu'))

model.add(Dense(32, activation='relu'))

model.add(Dense(2, activation='softmax')) # Jumlah kelas untuk output, dalam contoh ini 2 (0 dan 1)

model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])

loss, accuracy = model.evaluate(X_test, y_test)

print(f"Testing Accuracy: {accuracy*100:.2f}%")

from sklearn.tree import DecisionTreeClassifier, export_graphviz

from sklearn.datasets import load_iris

```

```

import graphviz
decision_tree_model = DecisionTreeClassifier(criterion='entropy')
import matplotlib.pyplot as plt
from sklearn.tree import DecisionTreeClassifier
from sklearn.datasets import load_iris
from sklearn import tree
# Tampilkan grafik Decision Tree menggunakan matplotlib
plt.figure(figsize=(10, 6))
tree.plot_tree(decision_tree_model, filled=True, feature_names=['luka_ringan', 'luka_berat',
'meninggal_dunia', 'jenis_kecelakaan_encoded', 'kerugian_material'])
plt.show()
# Fungsi untuk menampilkan grafik Decision Tree menggunakan graphviz
def plot_decision_tree(tree_model):
    dot_data = export_graphviz(tree_model, out_file=None,
                               feature_names=['luka_ringan', 'luka_berat', 'meninggal_dunia',
'jenis_kecelakaan_encoded', 'kerugian_material'],
                               class_names=['ringan', 'berat'],
                               filled=True, rounded=True, special_characters=True)

    graph = graphviz.Source(dot_data)
    graph.render('random_forest_tree') # Simpan grafik dalam file 'random_forest_tree.pdf'
    return graph
# Tampilkan grafik Decision Tree untuk salah satu pohon dalam Random Forest
plot_decision_tree(selected_tree)

```



UNIVERSITAS KHAIRUN
FAKULTAS TEKNIK
PROGRAM STUDI INFORMATIKA

DAFTAR PERBAIKAN UJIAN SKRIPSI/TUTUP

Dengan ini dinyatakan bahwa pada

Hari / tanggal : JUMAT, 01 MARET 2024
Pukul : 07:30 - 09:00
Tempat : RUANG PRODI

telah berlangsung Ujian Skripsi/Tutup dengan Peserta:

Nama Mahasiswa : VIVI FITRIANI DUWILA
NPM : 07351811051
Judul : IMPLEMENTASI DAN ANALISIS DATA MINING UNTUK
PENCARIAN POLA PENYEBAB KECELAKAAN LALU LINTAS DI
KOTA TERNATE DENGAN METODE LONG SHORT TERM
MEMORY (LSTM)

dinyatakan HARUS menyelesaikan perbaikan, yaitu:

- Pahami inti dari penelitian anda

Ace 01-03-2024

Dosen Pembimbing I,

MUHAMMAD FHADLI, S.Kom., M.Sc.
NIP. 199611232023211012



**UNIVERSITAS KHAIRUN
FAKULTAS TEKNIK
PROGRAM STUDI INFORMATIKA**

DAFTAR PERBAIKAN UJIAN SKRIPSI/TUTUP

Dengan ini dinyatakan bahwa pada

Hari / tanggal : JUMAT, 01 MARET 2024
Pukul : 07:30 - 09:00
Tempat : RUANG PRODI

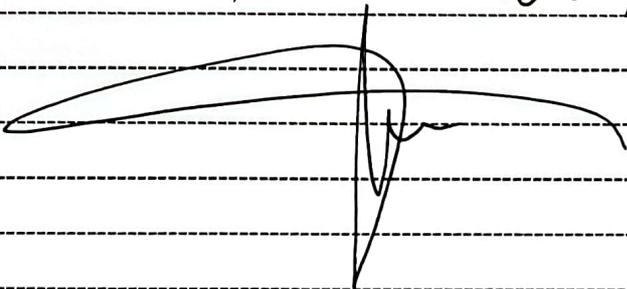
telah berlangsung Ujian Skripsi/Tutup dengan Peserta:

Nama Mahasiswa : VIVI FITRIANI DUWILA
NPM : 07351811051
Judul : IMPLEMENTASI DAN ANALISIS DATA MINING UNTUK
PENCARIAN POLA PENYEBAB KECELAKAAN LALU LINTAS DI
KOTA TERNATE DENGAN METODE LONG SHORT TERM
MEMORY (LSTM)

dinyatakan HARUS menyelesaikan perbaikan, yaitu:

selesaikan semua dari pengunji

Acc, 19/08/2024



Dosen Pembimbing II,



SYARIFUDDIN N. KAPITA, S.Pd., M.Si.
NIDN. 0012039105



UNIVERSITAS KHAIRUN
FAKULTAS TEKNIK
PROGRAM STUDI INFORMATIKA

DAFTAR PERBAIKAN UJIAN SKRIPSI/TUTUP

Dengan ini dinyatakan bahwa pada

Hari / tanggal : JUMAT, 01 MARET 2024

Pukul : 07:30 - 09:00

Tempat : RUANG PRODI

telah berlangsung Ujian Skripsi/Tutup dengan Peserta:

Nama Mahasiswa : VIVI FITRIANI DUWILA

NPM : 07351811051

Judul : IMPLEMENTASI DAN ANALISIS DATA MINING UNTUK
PENCARIAN POLA PENYEBAB KECELAKAAN LALU LINTAS DI
KOTA TERNATE DENGAN METODE LONG SHORT TERM
MEMORY (LSTM)

dinyatakan HARUS menyelesaikan perbaikan, yaitu:

Pelajari algoritma LSTM serta memastikan dan menunjukkan penerapan algoritma di pemrograman

Tentukan pola dan kuasai decision tree yang digunakan

Kuasa pemrograman dan algoritma yang digunakan dalam penelitian ini

Pola lokasi secara spesifik dari LSTM perlu di tunjukkan untuk menjadi data yang perlu di tindaklanjuti oleh pihak kepolisian

Kuasai office ✓

21/01/2024

Au Rizki

Dosen Penguji I,

Ir. SALKIN LUTFI, S.Kom., M.T., IPM

NIP. 198601112014041002



DAFTAR PERBAIKAN UJIAN SKRIPSI/TUTUP

Dengan ini dinyatakan bahwa pada

Hari / tanggal : JUMAT, 01 MARET 2024
Pukul : 07:30 - 09:00
Tempat : RUANG PRODI

telah berlangsung Ujian Skripsi/Tutup dengan Peserta:

Nama Mahasiswa : VIVI FITRIANI DUWILA
NPM : 07351811051
Judul : IMPLEMENTASI DAN ANALISIS DATA MINING UNTUK
PENCARIAN POLA PENYEBAB KECELAKAAN LALU LINTAS DI
KOTA TERNATE DENGAN METODE LONG SHORT TERM
MEMORY (LSTM)

dinyatakan HARUS menyelesaikan perbaikan, yaitu:

⊙ Pola penyebab kecelakaan lalu lintas yg dicari,
belum muncul pd Gb 4 dan kesimpulan

⊙ Di abstrak, minimal berisi: Tujuan penelitian,
Metode yg dipakai dan hasil penelitian.
→ Perbaiki abstrak

Ace Perbaikan Skripsi
31/2024
07

Dosen Penguji II,

Ir. AMAL KHAIRAN, S.T., M.Eng., IPM
NIP. 197401112003121003



DAFTAR PERBAIKAN UJIAN SKRIPSI/TUTUP

Dengan ini dinyatakan bahwa pada

Hari / tanggal : JUMAT, 01 MARET 2024
Pukul : 07:30 - 09:00
Tempat : RUANG PRODI

telah berlangsung Ujian Skripsi/Tutup dengan Peserta:

Nama Mahasiswa : VIVI FITRIANI DUWILA
NPM : 07351811051
Judul : IMPLEMENTASI DAN ANALISIS DATA MINING UNTUK
PENCARIAN POLA PENYEBAB KECELAKAAN LALU LINTAS DI
KOTA TERNATE DENGAN METODE LONG SHORT TERM
MEMORY (LSTM) }

dinyatakan HARUS menyelesaikan perbaikan, yaitu:

PERBAIKI SESUAI MASUKAN SIDANG

1) Babu Kerangka & Grafik. Juku
Acc 5/10 2024

Dosen Penguji III,

Dr. ASSAF ARIEF, S.T., M.Eng.
NIP. 198307102008121001