

**ANALISIS FUNGSI AKTIVASI PADA ALGORITMA  
*LONG SHORT TERM MEMORY* UNTUK KLASIFIKASI TEKS**

**SKRIPSI**

**Oleh:**

**DANIEL ROYHAN SUHEGA HARAHAHAP**

**198160050**



**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK**

**UNIVERSITAS MEDAN AREA**

**2024**

**UNIVERSITAS MEDAN AREA**

© Hak Cipta Di Lindungi Undang-Undang

1. Dilarang Mengutip sebagian atau seluruh dokumen ini tanpa mencantumkan sumber
2. Pengutipan hanya untuk keperluan pendidikan, penelitian dan penulisan karya ilmiah
3. Dilarang memperbanyak sebagian atau seluruh karya ini dalam bentuk apapun tanpa izin Universitas Medan Area

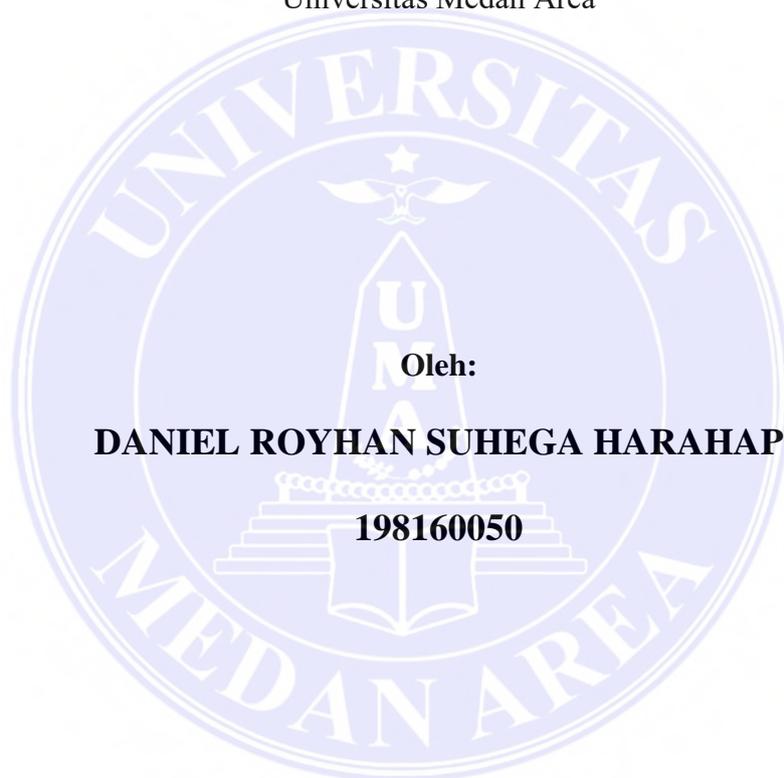
Document Accepted 20/1/25

Access From (repository.uma.ac.id)20/1/25

**ANALISIS FUNGSI AKTIVASI PADA ALGORITMA  
LONG SHORT TERM MEMORY UNTUK KLASIFIKASI TEKS**

**SKRIPSI**

Diajukan sebagai Salah Satu Syarat untuk Memperoleh  
Gelar Sarjana di Fakultas Teknik  
Universitas Medan Area



**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK**

**UNIVERSITAS MEDAN AREA**

**2024**

UNIVERSITAS MEDAN AREA

© Hak Cipta Di Lindungi Undang-Undang

1. Dilarang Mengutip sebagian atau seluruh dokumen ini tanpa mencantumkan sumber
2. Pengutipan hanya untuk keperluan pendidikan, penelitian dan penulisan karya ilmiah
3. Dilarang memperbanyak sebagian atau seluruh karya ini dalam bentuk apapun tanpa izin Universitas Medan Area

Document Accepted 20/1/25

Access From (repository.uma.ac.id)20/1/25

## LEMBAR PENGESAHAN

Judul Skripsi : Analisis Fungsi Aktivasi Pada Algoritma *Long Short Tem Memory*

Untuk Klasifikasi Perkara.

Nama : Daniel Royhan Suhega Harahap

NPM : 198160050

Fakultas : Teknik

Prodi : Teknik Informatika

Disetujui Oleh  
Pembimbing

Susilawati, S.Kom., M.Kom  
Mengetahui



Dr. Eng. Supriatno, ST, MT  
Dekan



Rizki Muliono, S.Kom., M.Kom  
Ka. Prodi

Tanggal Lulus : 30 Agustus 2024

iii

## HALAMAN PERNYATAAN

Saya menyatakan bahwa skripsi yang saya buat merupakan karya pribadi yang disusun sebagai persyaratan untuk mendapatkan gelar sarjana. Adapun bagian-bagian tertentu yang saya ambil dari karya orang lain telah saya sertakan dengan jelas asal sumbernya, sesuai dengan norma dan etika penulisan ilmiah. Saya siap menerima konsekuensi akademik dan sanksi lainnya sesuai peraturan yang berlaku, jika di masa mendatang terbukti terdapat plagiarisme dalam skripsi ini.

Medan, 30 Agustus 2024  
Yang membuat pernyataan



Daniel Royhan Suhega Harahap  
198160050

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI  
TUGAS AKHIR/SKRIPSI/TESIS UNTUK KEPENTINGAN  
AKADEMIS**

Sebagai sivitas akademik Universitas Medan Area, saya yang bertanda tangan di bawah ini:

Nama : Daniel Royhan Suhega Harahap  
NPM : 198160050  
Program Studi : Teknik Informatika  
Fakultas : Teknik  
Jenis karya : Tugas Akhir/Skripsi/Tesis

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Medan Area **Hak Bebas Royalti Noneksklusif** (*Non-exclusive Royalty-Free Right*) atas karya ilmiah saya yang berjudul : “**Analisis Fungsi Aktivasi Pada Algoritma Long Short Term Memory Untuk Klasifikasi Teks**” Beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini Universitas Medan Area berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan memublikasikan tugas akhir/skripsi/tesis saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta. Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Medan  
Pada tanggal : 30 Agustus 2024  
Yang menyatakan

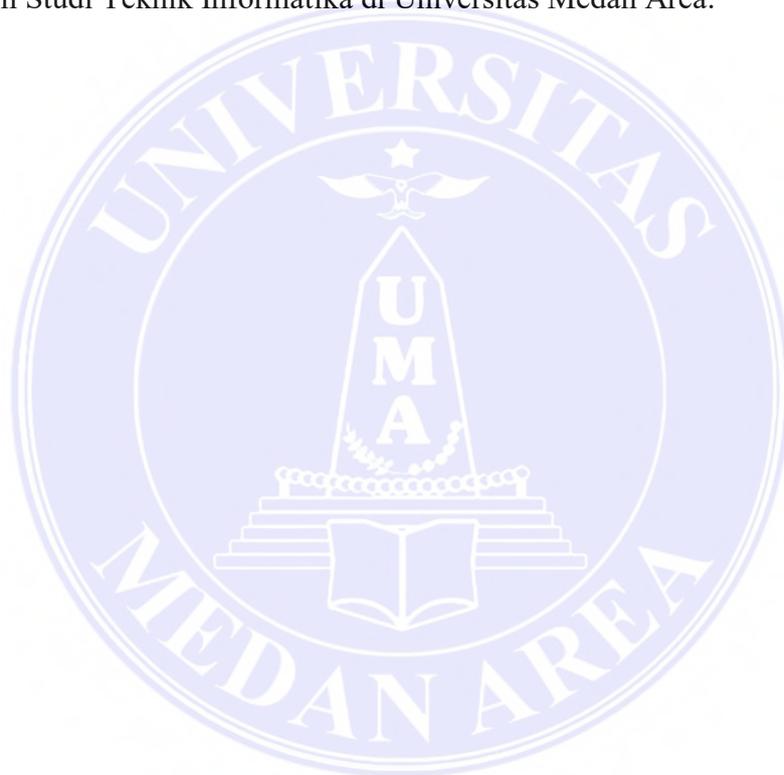


Daniel Royhan Suhega Harahap  
198160050

v

## RIWAYAT HIDUP

Penulis dilahirkan di Sosopan pada tanggal 24 Desember 2001 dari Bapak Himsaruddin Harahap dan Ibu Dewarna Lubis, S.Pd. Penulis merupakan anak pertama dari empat bersaudara, memiliki dua adik laki-laki dan satu adik perempuan. Penulis pertama kali mengenyam pendidikan di bangku SD Negeri 0310 Simaninggir pada tahun 2007 dan lulus pada tahun 2013. Kemudian Penulis melanjutkan pendidikan ke jenjang SMP pada tahun 2013 di MTS Swasta Al-Muttaqin dan lulus pada tahun 2016. Pada tahun yang sama, Penulis melanjutkan ke jenjang selanjutnya yaitu di MAN 1 Padangsidempuan dan lulus pada tahun 2019. Pada bulan September tahun 2019, Penulis melanjutkan pendidikan di bangku kuliah dan terdaftar sebagai mahasiswa Fakultas Teknik dengan mengambil Program Studi Teknik Informatika di Universitas Medan Area.



## KATA PENGANTAR

Puji syukur penulis ucapkan kehadirat Allah SWT tuhan yang Maha Esa, atas berkat dan rahmat serta kasih sayang-Nya sehingga penulis masih diberi kesempatan dan kesehatan dalam menyelesaikan penyusunan Proposal Skripsi yang berjudul: Analisis Fungsi Aktivasi Pada Algoritma *Long Short Term Memory* Untuk Klasifikasi Teks.

Adapun penyusunan proposal skripsi ini sebagai salah satu syarat dalam menyelesaikan program sarjana (S1) Fakultas Teknik Informatika Universitas Medan Area. Kemudian penulis ingin menyampaikan rasa terimakasih yang sangat dalam kepada semua pihak yang telah membantu dan melancarkan penulisan proposal skripsi ini. Penulis ucapkan terimakasih kepada:

1. Allah SWT yang telah memberikan rahmatnya sehingga penulis dapat menyelesaikan proposal skripsi ini.
2. Orangtua penulis yaitu Bapak Himsaruddin Harahap dan Ibu Dewarna Lubis serta keluarga besar penulis yang senantiasa memberikan doa, semangat dan dukungan selama melakukan studi.
3. Bapak Prof. Dr. Dadan Ramdan M. Eng, M.Sc sebagai Rektor Universitas Medan Area.
4. Bapak Dr.Eng. Supriatno S.T., M.T sebagai Dekan Fakultas Teknik Universitas Medan Area.
5. Ibu Susilawati, S.Kom., M.Kom selaku Wakil Bidang Penjamin Mutu Akademik Fakultas Teknik Universitas Medan Area dan dosen Pembimbing yang sudah memberikan bimbingan serta arahan kepada penulis.

6. Bapak Rizki Muliono, S.Kom., M.Kom selaku Kepala Program Studi Teknik Informatika
7. Bapak/Ibu Dosen dan Staf Teknik Informatika Universitas Medan Area.
8. Teman-teman yang telah mendukung selama proses penulisan proposal skripsi ini, terlebih kepada teman-teman Teknik Informatika Angkatan 2019.
9. Senior yang telah memberi saya banyak ilmu dan pengalaman.

Penulis sadar akan penelitian ini masih memiliki kekurangan, oleh karena itu saran dan kritik yang bisa membangun sangat penulis harapkan demi kesempurnaan penelitian ini. Penulis berharap agar penelitian ini dapat dimanfaatkan kedepannya.

Akhir kata penulis ucapkan terimakasih.

Medan, 30 September 2024  
Penulis

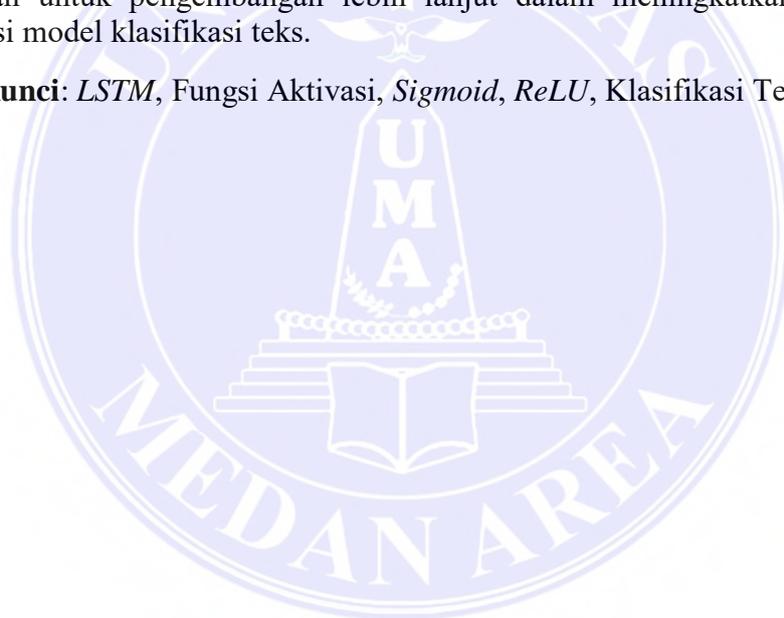


Daniel Royhan Suhega Harahap  
198160050

## ABSTRAK

Penelitian ini bertujuan untuk menganalisis pengaruh fungsi aktivasi pada model Long Short Term Memory *LSTM* dalam klasifikasi teks hukum. Long Short Term Memory (*LSTM*), yang digunakan untuk menangkap fitur lokal dan semantik dari teks. Studi ini menggunakan dua fungsi aktivasi utama, yaitu *Sigmoid* dan *ReLU*, untuk mengevaluasi kinerja model dalam mengklasifikasikan empat jenis perkara: Pidana, Perdata, PTUN, dan Agama. Dataset yang digunakan terdiri dari 400 kronologi perkara, yang diperoleh dari salah satu instansi hukum di Kota Medan. Data tersebut dibagi menjadi 80% untuk pelatihan dan 20% untuk pengujian. Proses pre-processing dilakukan untuk membersihkan dan menormalkan data teks, termasuk tokenisasi dan penghapusan kesalahan pengetikan. Hasil penelitian menunjukkan bahwa pemilihan fungsi aktivasi yang tepat sangat mempengaruhi akurasi model. Model dengan fungsi aktivasi *ReLU* menunjukkan kinerja terbaik dengan akurasi mencapai 83% pada beberapa kasus. Temuan ini memberikan kontribusi penting dalam pemahaman tentang peran fungsi aktivasi dalam model *LSTM* serta aplikasinya dalam klasifikasi teks hukum. Hasil ini juga memberikan landasan untuk pengembangan lebih lanjut dalam meningkatkan akurasi dan efisiensi model klasifikasi teks.

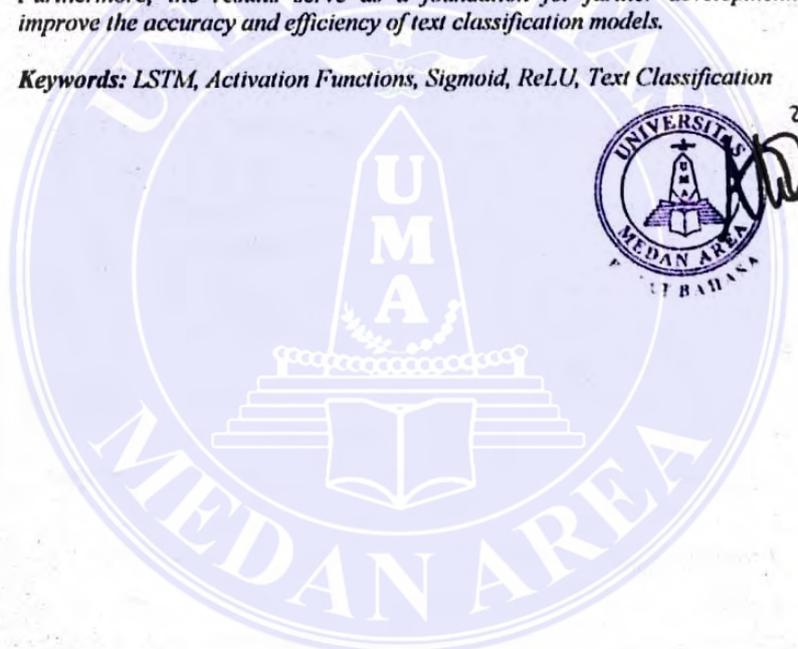
**Kata kunci:** *LSTM*, Fungsi Aktivasi, *Sigmoid*, *ReLU*, Klasifikasi Teks



### ABSTRACT

*This research aimed to analyze the impact of activation functions in the Long Short Term Memory (LSTM) model for legal text classification. LSTM is utilized to capture both local and semantic features of text. The research employed two main activation functions, namely Sigmoid and ReLU, to evaluate the model's performance in classifying four types of legal cases: Criminal, Civil, Administrative Court (PTUN), and Religious Court. The dataset consists of 400 case chronologies obtained from a legal institution in Medan City. The data was divided into 80% for training and 20% for testing. Pre-processing was conducted to clean and normalize the text data, including tokenization and typo corrections. The results indicate that selecting the appropriate activation function significantly affects the model's accuracy. The model using the ReLU activation function achieved the best performance, with an accuracy of up to 83% in certain cases. These findings provide valuable insights into the role of activation functions in LSTM models and their application in legal text classification. Furthermore, the results serve as a foundation for further developments to improve the accuracy and efficiency of text classification models.*

**Keywords:** LSTM, Activation Functions, Sigmoid, ReLU, Text Classification



x

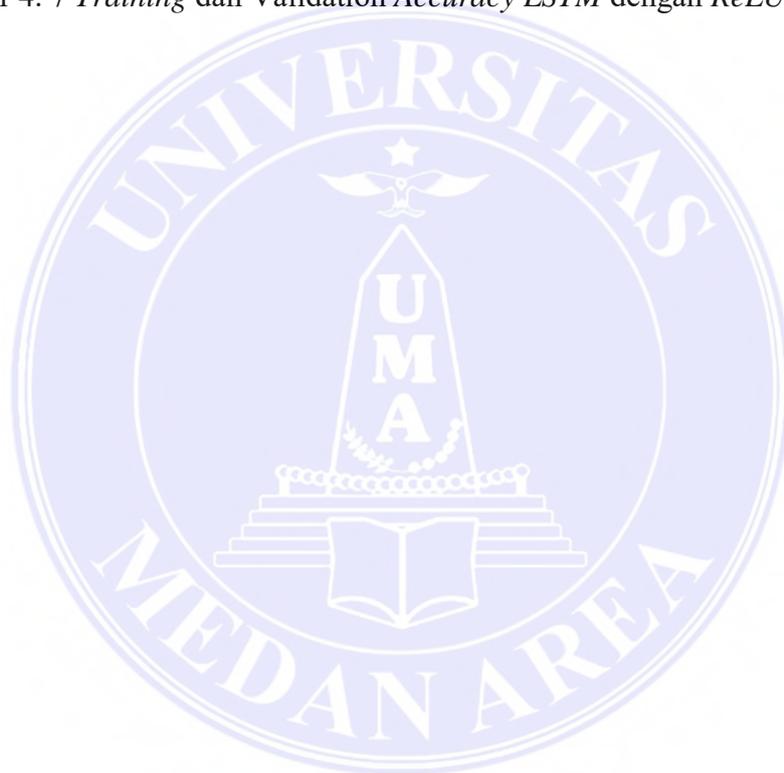
## DAFTAR ISI

<b>LEMBAR PENGESAHAN .....</b>	<b>iii</b>
<b>HALAMAN PERNYATAAN.....</b>	<b>iv</b>
<b>HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR/SKRIPSI/TESIS UNTUK KEPENTINGAN AKADEMIS .....</b>	<b>v</b>
<b>RIWAYAT HIDUP .....</b>	<b>vi</b>
<b>KATA PENGANTAR.....</b>	<b>vii</b>
<b>ABSTRAK .....</b>	<b>ix</b>
<b>ABSTRACT .....</b>	<b>x</b>
<b>DAFTAR ISI.....</b>	<b>xi</b>
<b>DAFTAR GAMBAR.....</b>	<b>xiii</b>
<b>DAFTAR TABEL.....</b>	<b>xiv</b>
<b>BAB I PENDAHULUAN .....</b>	<b>2</b>
1.1. Latar Belakang.....	2
1.2. Rumusan Masalah .....	4
1.3. Batasan Masalah.....	4
1.4. Tujuan Penelitian.....	4
1.5. Manfaat Penelitian.....	4
<b>BAB II TINJAUAN PUSTAKA.....</b>	<b>6</b>
2.1. <i>Long Short Term Memory</i> .....	6
2.2. Klasifikasi.....	10
2.3. <i>Data Mining</i> .....	11
2.4. Teks Mining.....	13
2.5. <i>Hyper Text</i> .....	14
2.6. Fungsi Aktivasi <i>Sigmoid</i> .....	15
2.7. Fungsi Aktivasi <i>ReLU</i> .....	16
2.7.1. Kelebihan <i>ReLU</i> .....	17
2.7.2. Kelemahan <i>ReLU</i> .....	18
2.8. Jenis-Jenis Perkara.....	18
2.9. Penelitian Terdahulu.....	20

<b>BAB III METODE PENELITIAN .....</b>	<b>22</b>
3.1. Alat dan Bahan Penelitian .....	22
3.1.1. Perangkat Keras .....	22
3.1.2. Perangkat Lunak.....	22
3.2. Alur Penelitian.....	23
3.3. Metode Pengumpulan Data .....	23
3.4. Pre-Processing Data.....	25
3.5. <i>Training Data</i> .....	27
3.5.1. <i>Sigmoid</i> .....	27
3.5.2. <i>ReLU</i> .....	27
3.6. Metode Evaluasi .....	28
<b>BAB IV HASIL DAN PEMBAHASAN.....</b>	<b>30</b>
4.1. Hasil.....	30
4.1.1. Hasil Pengujian Dengan <i>LSTM</i> .....	31
4.1.2. Hasil Pengujian Dengan <i>Sigmoid</i> .....	33
4.1.3. Hasil Pengujian Dengan <i>ReLU</i> .....	34
4.2. Pembahasan .....	37
4.2.1. Dataset.....	37
4.2.2. <i>Training dan Validation LSTM dengan Sigmoid</i> .....	39
4.2.3. <i>Training dan Validation LSTM dengan ReLU</i> .....	40
4.2.4. Perbandingan Dengan Penelitian Terdahulu.....	40
<b>BAB V KESIMPULAN DAN SARAN .....</b>	<b>42</b>
5.1. Kesimpulan.....	42
5.2. Saran.....	43
<b>DAFTAR PUSTAKA .....</b>	<b>44</b>
<b>LAMPIRAN.....</b>	<b>49</b>

## DAFTAR GAMBAR

Gambar 2. 1 Arsitektur <i>LSTM</i> .....	7
Gambar 2. 2 Fungsi <i>Sigmoid</i> .....	16
Gambar 2. 3 Fungsi <i>ReLU</i> .....	17
Gambar 3. 1 Alur Penelitian.....	23
Gambar 4. 1 <i>Confusion Matrix LSTM</i> .....	31
Gambar 4. 2 <i>Confusion matrix LSTM with Sigmoid</i> .....	34
Gambar 4. 3 <i>Confusion Matrix LSTM with ReLU</i> .....	35
Gambar 4. 4 Kode <i>Cleaning Data</i> .....	38
Gambar 4. 5 Kode Tokenisasi Teks .....	38
Gambar 4. 6 <i>Training &amp; Validation Accuracy LSTM dengan Sigmoid</i> .....	39
Gambar 4. 7 <i>Training dan Validation Accuracy LSTM dengan ReLU</i> .....	40



## DAFTAR TABEL

Tabel 2. 1 Penelitian Terdahulu .....	20
Tabel 3. 1 Keterangan Perangkat Keras .....	22
Tabel 3. 2 Keterangan Perangkat Lunak .....	22
Tabel 3. 3 Contoh Data .....	24
Tabel 3. 4 Contoh Hasil Tokenisasi .....	26
Tabel 4. 1 <i>Classification Report</i> .....	30
Tabel 4. 2 <i>Classification Report algoritma LSTM</i> .....	32
Tabel 4. 3 <i>Classification report Sigmoid Activation</i> .....	34
Tabel 4. 4 <i>Classification report ReLU Activation</i> .....	36
Tabel 4. 5 Perbandingan Penelitian Terdahulu .....	41



# BAB I

## PENDAHULUAN

### 1.1. Latar Belakang

*Long Short Term Memory (LSTM)* merupakan varian dari *Recurrent neural network (RNN)*. *LSTM* seringkali digunakan untuk memproses data yang berurutan sehingga seringkali digunakan untuk memproses data yang berbentuk bahasa alami dan data pengenalan suara. *LSTM* juga dapat langsung melakukan ekstraksi fitur dari data yang banyak dan mengatasi ketergantungan pada jangka panjang (Hao dkk., 2020).

Salah satu penerapan *LSTM* untuk klasifikasi teks yaitu pada penelitian klasifikasi kalimat berurutan dalam abstrak ilmiah medis. Pada penelitian ini terdapat dua jenis dataset yang digunakan, dataset yang pertama merupakan dataset teks berbasis abstrak ilmiah medis yang terbesar dan dipublikasikan. Jumlah dataset pertama sebanyak 200.000 data dan telah dibagi kedalam tiga subset, yaitu untuk pelatihan, validasi, dan data pengujian. Model *LSTM* yang diusulkan pada penelitian ini membuktikan peran penting dalam menghasilkan representasi kalimat yang komprehensif, sehingga dapat menjadi faktor pendukung untuk melakukan penelitian tentang klasifikasi kalimat berurutan dan klasifikasi teks (Lam dkk., 2024).

Pada penelitian lain juga menyatakan bahwa *LSTM* mengatasi masalah ketergantungan jangka panjang dari *RNN*. *LSTM* ini dapat memutus informasi mana yang akan diingat dan informasi mana yang akan dilupakan. Dataset yang dipakai berbentuk bahasa korea yang dikumpulkan dari beberapa kategori, diantaranya, IT, sains, budaya, kehidupan dan dunia. Ada tiga model

yang dipakai pada penelitian ini, yaitu *RNN*, *LSTM*, *Bi-LSTM* dengan masing masing nilai akurasi sebesar 90,24%, 92,99% dan 92,15% (Noh & Cho, 2021). Pada tahun 2019 algoritma *LSTM* dibandingkan dengan Random Forest dengan tugas mengklasifikasikan pesan sebagai pernyataan atau bukan pernyataan. Dilakukannya perbandingan ini dengan cara melatih dan mengoptimalkan model pada percakapan historis dari salah satu perusahaan asuransi yang berada di Swedia. Model *embedding* kata diuji juga pada penelitian ini, diantaranya adalah *Word2vec* dan *Bag Of Word*. Hasil pada penelitian ini menyatakan *LSTM* memperoleh hasil yang lebih tinggi jika dibandingkan dengan Random Forest pada nilai F1 dan akurasinya (Kindbom, 2019). Penelitian tentang algoritma *LSTM* sangatlah banyak, salah satunya adalah penelitian yang dilakukan (Reddy Prasad & Deepa, 2021). Namun pada penelitian ini menggunakan aplikasi yang mengubah ucapan menjadi format teks. Tujuan utama penelitian ini adalah mendapatkan akurasi dan presisi yang baik menggunakan *RNN-LSTM* dan menggunakan *ReLU* dan *softmax* sebagai parameter untuk menghindari *overfitting*. Jumlah data yang dipakai pada penelitian ini sebanyak 16.332 data dan terdiri dari 2 jenis data yaitu data dalam kalimat bahasa inggris dan dalam kalimat bahasa telugu.

Penelitian ini memiliki tujuan utama untuk menganalisis model-model yang dipakai dalam klasifikasi teks, sehingga menjadi pengetahuan baru tentang ke-efektifan fungsi aktivasi yang disandingkan pada algoritma *LSTM* dalam melakukan pada klasifikasi teks.

## 1.2. Rumusan Masalah

Berdasarkan latar belakang yang ada, rumusan masalah yang terbangun adalah bagaimana proses metode deep *LSTM* mengklasifikasi teks kronologi perkara dan bagaimana pengaruh fungsi aktivasi terhadap kinerja model.

## 1.3. Batasan Masalah

Batasan masalah dalam penelitian ini antara lain adalah:

1. Dataset yang dipakai merupakan kronologi perkara hukum.
2. Terdapat empat jenis perkara yang akan diklasifikasikan yaitu Pidana, Perdata, Ptun, dan Agama.
3. Dataset perkara yang digunakan adalah kronologi perkara berbentuk *file text*
4. Hanya memakai dua fungsi aktivasi yaitu *Sigmoid* dan *ReLU*.
5. Proses penelitian menggunakan metode deep *LSTM*.
6. Penerapan metode pada penelitian ini menggunakan *Google Colab*.

## 1.4. Tujuan Penelitian

Tujuan penelitian ini adalah menganalisis dan mengidentifikasi pengaruh penggunaan fungsi aktivasi terhadap kinerja model *LSTM* dalam klasifikasi teks, dengan tujuan mengetahui fungsi aktivasi yang paling efektif untuk meningkatkan akurasi dan kinerja model tersebut.

## 1.5. Manfaat Penelitian

Manfaat dari penelitian ini adalah meningkatkan kinerja klasifikasi teks dan akurasi model melalui analisis fungsi aktivasi yang paling sesuai untuk *LSTM*. Penelitian ini juga menjadi landasan untuk penelitian selanjutnya dalam pengembangan dan pemahaman lebih lanjut tentang fungsi aktivasi

pad;a algoritma *LSTM*. Penerapan di bidang hukum memberikan wawasan baru dalam pemilihan dataset untuk klasifikasi teks.



## BAB II

### TINJAUAN PUSTAKA

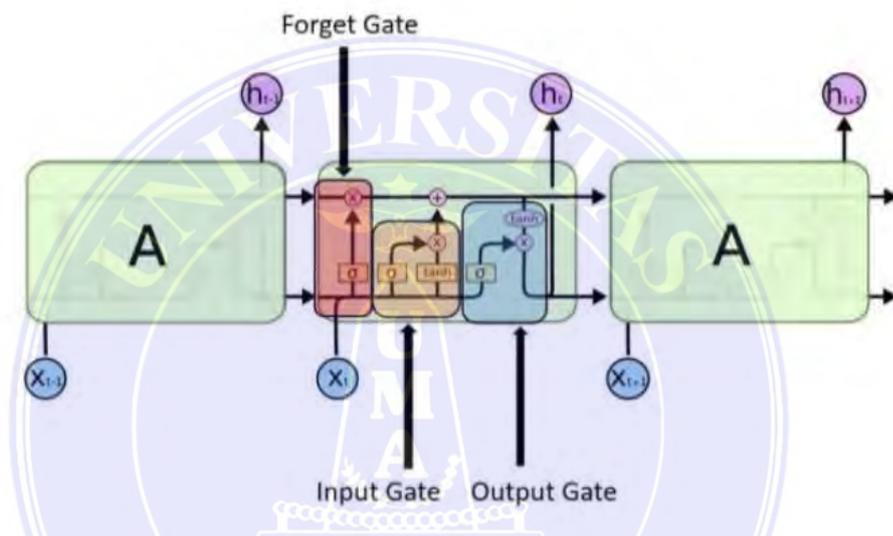
#### 2.1. Long Short Term Memory

*Long Short Term Memory* merupakan algoritma yang sangat sering digunakan karena *LSTM* mampu mengatasi masalah ketergantungan jangka panjang dan masalah *vanishing gradient* yang merupakan masalah sulit yang harus diatasi dalam pelatihan jaringan saraf berulang dan jaringan dalam (Van Houdt dkk., 2020).

*LSTM* merupakan jenis special dari *RNN* yang memiliki kemampuan spesial yaitu belajar ketergantungan jangka panjang. *LSTM* memiliki inti state sel, kegunaan dari state sel adalah menambahkan atau menghapus informasi kedalam sel dan membiarkan aliran informasi melalui mekanisme gerbang untuk mencapai tujuannya secara selektif. *LSTM* memiliki tiga jenis gerbang yaitu gerbang lupakan, gerbang masukan dan gerbang keluaran (Luan & Lin, 2019). Sedangkan menurut (Nagabushanam dkk., 2020) *LSTM* adalah bagian dari *deep learning* yang didasari pada *RNN* yang dimana *RNN* terdiri dari struktur rekuren yang ditinjau dari segi lokal mengalirkan kekuatan pantulan sehingga tidak membutuhkan memori untuk menyimpan output sebelumnya.

Penggunaan *LSTM* untuk klasifikasi telah dilakukan pada beberapa penelitian contohnya adalah sentimen klasifikasi yang dilakukan oleh (Tri Hermanto dkk., 2021), pada penelitian ini peneliti menggunakan 3 model yaitu *LSTM*, *LSTM-CNN*, dan *CNN-LSTM* dimana metode *LSTM* mendapatkan akurasi sebesar 51%, nilai *recall* sebesar 57%, dan nilai presisi sebesar 60%.

Arsitektur *LSTM* ini memiliki 3 lapisan yaitu, lapisan masukan (*input*), lapisan tersembunyi (*hidden*), dan lapisan keluaran (*output*) dan setiap sel memori dalam *LSTM* mempunyai tiga lapisan *Sigmoid* dan satu lapisan *tanh* atau lapisan tangen hiperbolik. Dalam satu sel memori terdiri atas beberapa gerbang, diantaranya adalah *forget gate* (gerbang penghapusan), gerbang masukan (*input gate*), dan gerbang keluaran (*output*).



**Gambar 2. 1** Arsitektur *LSTM*

Fungsi lapisan *Sigmoid* dan lapisan *tanh* adalah untuk mengaktifkan neuron.

1. *Sigmoid* mengkonversi nilai antara -1 dan 1 menjadi 0 dan 1, rumus *Sigmoid* adalah

$$\sigma(x) = \frac{1}{1 + e^{-x}} \dots\dots\dots 2.1$$

2. Lapisan *tanh* merupakan alternatif dari lapisan *Sigmoid*, nilai dari *tanh* adalah sekitar -1 sampai 1, rumus lapisan *tanh* adalah

$$\text{Tanh}(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \dots\dots\dots 2.2$$

Berikut ini merupakan penjelasan gerbang sel *LSTM*:

- a. Gerbang penghapusan (*Forget Gate*), fungsi dari gerbang ini untuk penentuan informasi yang kurang penting dan kurang bermakna untuk dihapus dengan fungsi *Sigmoid*. Rumus *forget gate* adalah:

$$f_t = \sigma(W_f * [h_{t-1}, x_t] + b_f) \dots \dots \dots 2.3$$

Penjelasan:

$f_t$  : Gerbang penghapusan

$\sigma$  : Fungsi *Sigmoid*

$W_f$  : Nilai *Forget Gate*

$h_{t-1}$  : Nilai *Output* sebelum orde -t

$x_t$  : Nilai *Input* untuk orde -t

$b_f$  : Nilai bias *Forget Gate*

- b. Gerbang masukan (*Input Gate*), berfungsi untuk memfilter dan menentukan informasi yang akan diupdate dalam status sel yang menggunakan fungsi aktivasi *Sigmoid*. Proses ini membentuk *vector* baru yang menggunakan *tanh hiperbolik function*, dimana rumus nya adalah:

$$i_t = \sigma(W_i * [h_{t-1}, x_t] + b_i) \dots \dots \dots 2.4$$

Penjelasan :

$i_t$  : Gerbang masukan

$\sigma$  : Fungsi *Sigmoid*

$W_i$  : Nilai gerbang masukan

$h_{t-1}$  : Nilai *output* sebelum orde-t

$x_t$  : Nilai *input* untuk urutan -t

$b_i$  : Nilai bias gerbang masukan

Rumus dari tahn *hiperbolik function*:

$$\tilde{c}_t = \tanh(W_c * [h_{t-1}, x_t] + bc) \dots \dots \dots 2.5$$

Penjelasan:

$\tilde{c}_t$  : Nilai baru yang diinput kedalam status sel

$\tanh$  : Fungsi  $\tanh$

$W_c$  : Nilai bobot status sel

$h_{t-1}$ : Nilai *output* sebelum urutan -t

$x_t$  : Nilai *input* sebelum urutan -t

$bc$  : Nilai bias status sel

- c. Gerbang Keluaran (*Output Gate*), gerbang ini memiliki peran sebagai pemutus nilai keluaran dimana hasilnya bersifat tersembunyi atau hidden state serta menempatkan status sel kedalam *tanh hiperbolik* berdasarkan *input* dan memori blok. Setelah memperoleh nilai keluaran *Sigmoid* dan *tanh*, kedua hasil tersebut dikalikan. Rumus nya adalah:

$$o_t = \sigma(W_o * [h_{t-1}, x_t] + b_o) \dots \dots \dots 2.6$$

Penjelasan:

$o_t$  : Gerbang Keluaran

$\sigma$  : Fungsi *Sigmoid*

$W_o$  : Nilai *output gate*

$h_{t-1}$  : Nilai *output* sebelum orde -t

$x_t$  : Nilai *input* orde -t

$b_o$  : Nilai bias *output gate* (Yusro, 2023).

## 2.2. Klasifikasi

Klasifikasi merupakan jenis analisis data yang berfungsi untuk menentukan kelas label dari suatu kasus yang ingin diklasifikasi, metode klasifikasi ini adalah *supervised learning*. Tujuan klasifikasi adalah meningkatkan ketepatan hasil yang diperoleh dari data (Hendrian, 2018). Metode klasifikasi telah dipergunakan diberbagai bidang ilmu, keluaran dari metode klasifikasi ini merupakan nilai diskrit seperti pohon keputusan. Performa klasifikasi ditunjukkan oleh nilai skalar yaitu akurasi, sensitivitas dan spesifitas (Tharwat, 2018). Dalam data mining klasifikasi dikenal sebagai suatu alat yang sangat populer dalam memahami hubungan antara berbagai kondisi dan fitur dari suatu objek yang berbeda. Klasifikasi ini dapat membantu dalam mengidentifikasi pengamatan-pengamatan baru yang termasuk kedalam sub-populasi atau kategori tertentu (Lan dkk., 2018).

Di dalam teknik klasifikasi, objek-objek yang berbeda menentukan kelas atau jenis yang cocok untuk data yang tidak memiliki label. Salah satu algoritma yang cocok untuk klasifikasi adalah *KNN (K-Nearest-Neighbour)*, algoritma ini dikenal karna efektivitas dan kesederhanaannya (Taunk dkk., 2019). Menurut (Salih & Abdulazeez, 2021) klasifikasi ini merupakan bagian dari tugas *machine learning* yang berbasis kelas biner dan multi, dimana model dari klasifikasi ini adalah pembelajaran terawasi yang digunakan untuk sistem deteksi intrusi yang mengaitkan setiap data ke golongan dataset tertentu. Proses klasifikasi data dalam dataset ada dua tahap yaitu data pengujian dan data pelatihan. Pada tahap tersebut klasifier dipelajari sebagai

target sedangkan pada tahap kedua, model dibuat untuk memprediksi label kelas data yang diberikan, tahap ke dua ini dinamakan tahap pengujian.

### **2.3. Data Mining**

Data Mining merupakan disiplin ilmu yang muncul yang digunakan untuk menganalisis data secara otomatis dengan cara menemukan pola dan hubungan dalam data mentah. Data mining ini telah mencapai kesuksesan yang sangat signifikan dalam banyak bidang, diantaranya adalah bidang kedokteran, bisnis, robotika, visi komputer dan lain lain (Yahya & Osman, 2019).

Data mining ini terdiri atas fungsionalitas, teknik, dan algoritma yang dalam penggunaannya untuk mencari, menemukan, dan mengekstraksi pola-pola unik dari repositori database (M. K. Gupta & Chandra, 2020). Data Mining muncul pada 20 tahun terakhir, digunakan untuk memecahkan masalah-masalah manufaktur, sistem pengambilan keputusan yang bagus, pengaturan perawatan mesin, serta program untuk menjadwalkan jalur produksi secara simultan (Dogan & Birant, 2021). Data mining adalah rangkaian dari prosedur dan teknik yang mampu dikembangkan dari berbagai sumber data contohnya basis data rasional atau data warehouse hingga berkas yang tidak memiliki format studi statistik untuk mengantisipasi dan memprediksi ukuran statistik kepastian berdasarkan fakta yang ada (Villanueva & Moreno, 2018).

Terdapat beberapa teknik dalam data mining. Menurut (Pratheebha dkk., 2021) ada enam teknik pengambilan data yaitu:

a. Klasifikasi

Klasifikasi merupakan teknik paling sering digunakan. Metode ini seringkali menggunakan algoritma klasifikasi model pohon keputusan atau jaringan syaraf. Ciri utama dari tugas klasifikasi adalah pembelajaran secara terbimbing, variabel dependen jenis kategori, dan pemberian data ke salah satu kelas yang sudah ditentukan dengan baik. Teknik klasifikasi ini dipergunakan untuk segmentasi klien, pemodelan bisnis, analisis kredit dan aplikasi lainnya.

b. *Clustering*

*Clustering* adalah metode pengelompokan suatu objek yang nyata maupun tidak menjadi kategori objek yang serupa. Contohnya mengelompokkan pembeli berdasarkan pola pembeli, dengan menggunakan teknik pengelompokan kita mampu menentukan wilayah yang padat dan langka diruang objek dan dapat juga menemukan pola-pola distribusi keseluruhan dan keterkaitan antara atribut-atribut data.

c. Pola Sekuensial

Teknik pola sekuensial berguna untuk memprediksi ketergantungan sekuensial dan subsekuensial, penggunaan metode ini adalah untuk menemukan pola sekuensial seperti *generalized sequential pattern (GSP)*, *free span*, *prefix span* dan *SPADE (Sequential pattern discovery using equivalent class)*.

d. Deteksi *Outlier*

Sebuah kumpulan data objek yang tidak sesuai dengan umum ataupun model dari data. *Outlier* adalah objek dari data, dalam beberapa teknik data

mining mengabaikan *outlier* sebagai *noise* atau gangguan. Analisis data *outlier* disebut dengan analisis anomali atau analisis *outlier* yang dapat membantu mendeteksi kecurangan serta memprediksi nilai yang bersifat abnormal.

e. Aturan Asosiasi

Teknik terbaik data mining adalah aturan asosiasi, dimana pola dipahami melalui hubungan antara suatu objek dengan objek lainnya dalam transaksi yang sama. Contohnya penggunaan teknik asosiasi dalam analisis keranjang belanja pasar untuk menemukan item yang sering dibeli oleh pelanggan.

f. Regresi

Regresi adalah salah satu teknik *mining* data bersifat statistik yang sering digunakan untuk memprediksi numerik. Teknik ini memodelkan hubungan antara satu variabel atau lebih yang mempunyai sifat dependen. Dalam data mining, variabel dependen merupakan atribut yang sudah diketahui dan variabel respons adalah variabel yang akan kita prediksi.

## 2.4. Teks Mining

Penggunaan teks mining semakin meningkat dalam bidang pelayanan manajemen dikarenakan pentingnya akses kedalam big data, dimana banyak platform digital yang mungkin menggunakan layanan tersebut. Analisis teks ataupun yang sering juga disebut *Text Mining* merupakan salah satu teknik kecerdasan buatan menggunakan *NLP (Natural Language Processing)* yang berproses mengubah data yang tidak terstruktur menjadi data yang terstruktur menggunakan algoritma pembelajaran mesin. *Text mining* ini sendiri merupakan Teknik yang sangat dikenal dikalangan ilmu komputer, ilmu

informasi, dan dibidang ekstrak kecerdasan buatan dari big data (Kumar dkk., 2021). Menurut data pada penelitian Louis menyatakan bahwa 80% data organisasi disimpan dengan bentuk bahasa alami yang tidak terstruktur dan dalam berbagai penelitian selama tiga dekade terakhir ini menggunakan teks mining dengan kosa kata yang tertutup untuk meringkas data teks tersebut dengan cara menghitung frasa dan kata yang terkait secara konseptual untuk memberi nilai pada konstruk (Hickman dkk., 2022).

Teks mining ini juga sudah merambat dibidang industrial keungan, teks mining ini sendiri merupakan sebuah turunan dari teknik *data mining*, *machine learning*, dan linguistik komputasional, dari gabungan beberapa teknik tersebutlah munculnya teks mining. Data yang bersifat tekstual pada big data keungan akan dianalisis teks mining tersebut secara keseluruhan. Hasil penelitian ini menyatakan teks mining dapat memprediksi pasar keungan dan juga berfungsi dibidang laporan korporat dan laporan keungan (A. Gupta dkk., 2020).

## 2.5. *Hyper Text*

*Hypertext* secara harpiah berarti bentuk penulisan yang diperluas dan umum. Menurut Theodor H. Nelson dalam penelitian (Wachter, 2022) menyebutkan *hypertext* merupakan penulisan teks yang tidak berurutan dan bercabang dan memungkinkan pembaca untuk membuat pengartian yang lebih baik yang dibaca pada tampilan yang interaktif. *Hypertext* ini terdiri dari node yang saling terhubung dan berisi format media dan pada umumnya dikaitkan dengan ekspresi pengetahuan yang luas.

Lingkungan *hypertext* memungkinkan pembaca untuk mengakses jumlah informasi yang luar biasa. Namun, multipelitas tautan dan fleksibilitas yang menjadi ciri khas lingkungan *hypertext* telah dikaitkan dengan tantangan dalam membaca yang dapat mempengaruhi kinerja pemahaman pembaca (Taky-eddine & Madaoui, 2024).

## 2.6. Fungsi Aktivasi *Sigmoid*

Fungsi aktivasi yang paling umum digunakan pada jaringan saraf salah satunya adalah *Sigmoid* hal ini dikarenakan sifat sifat dari fungsi aktivasi ini memiliki kegunaan yang baik dan dapat dideferensiasi sehingga mengalami peningkatan dan juga mewakili keseimbangan antara *linear* dan *nonlinear* (Roodschild., 2020).

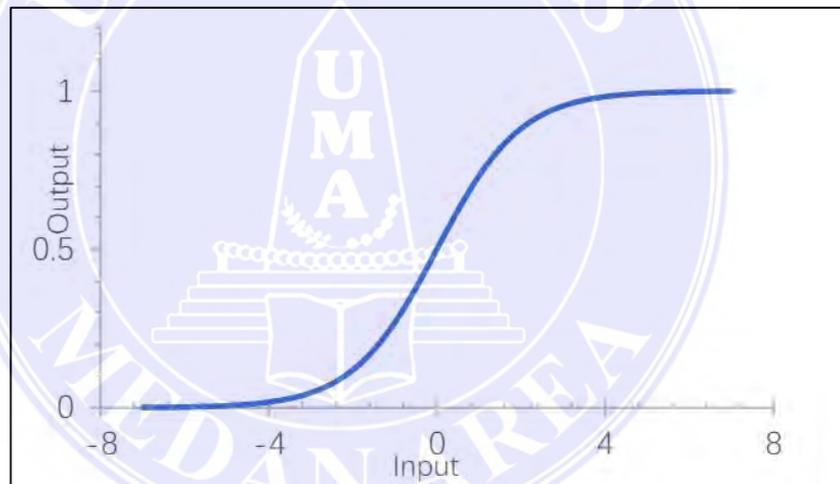
$$f(x) = \frac{1}{1+e^{-x}} \dots\dots\dots 2.7.$$

Dalam penelitian Liu W dkk mengemukakan keuntungan dalam penggunaan fungsi aktivasi *Sigmoid*. Keuntungan fungsi aktivasi *Sigmoid* adalah parameter kontrol dipilih secara adaptif yang membantu menjamin keragaman populasi. Alasan-alasan penggunaan aktivasi *Sigmoid* pada penelitian ini adalah:

- a. Fungsi aktivasi *Sigmoid* bersifat monotonik dan terbatas.
- b. Kurva fungsi aktivasi *Sigmoid* berbentuk S sehingga dapat menghindari perubahan yang tidak diinginkan secara tiba-tiba dari parameter kontrolnya.

c. Fungsi *Sigmoid* bersifat halus dan bisa didiferensiasi sehingga pencerminan sifat adaptif dari pembaruan bobot dari iterasi ke iterasi (Liu ., 2021).

Pendapat lain tentang umumnya penggunaan *Sigmoid* pada jaringan saraf tiruan terdapat pada penelitian xing pada tahun 2020. *Sigmoid* dengan kelebihan nonlinearitas dan kinerja turunannya menjadi alasan sering digunakan, namun pada penelitian ini mereka melakukan perancangan neuron *Sigmoid* AF yang memiliki presisi yang tinggi dan mempunyai struktur yang simpel menggunakan teknologi perangkat keras. Berikut adalah karakteristik dari *Sigmoid* (Xing & Wu, 2020).



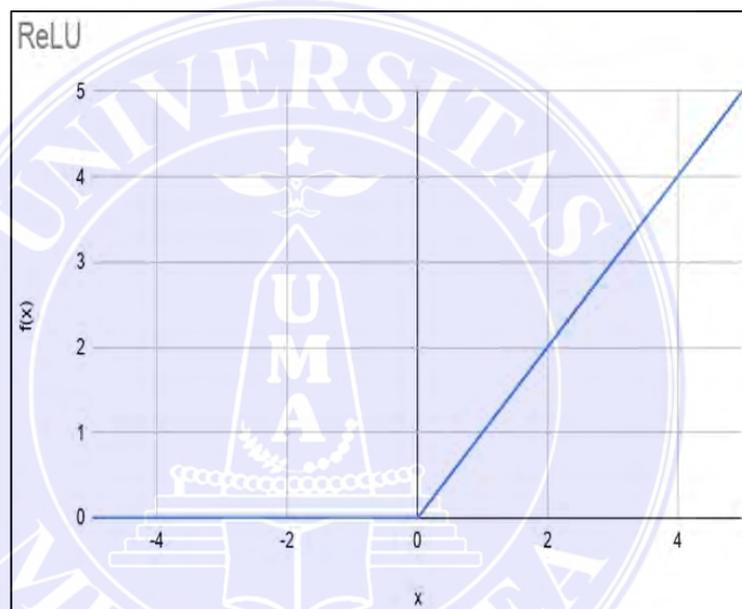
**Gambar 2. 2** Fungsi *Sigmoid*

### 2.7. Fungsi Aktivasi *ReLU*

*ReLU* (*Rectified Linear Ujnit*) merupakan salah satu jenis fungsi aktivasi yang sangat umum digunakan dalam jaringan saraf tiruan. Fungsi aktivasi ini merupakan fungsi yang sangat sederhana dan efektif dalam mempercepat proses pelatihan model (Peng dkk., 2023).

$$F(x) = \max(0, x) \dots\dots\dots 2.8$$

*ReLU* ini diusulkan untuk mengatasi masalah vanishing gradient pada jaringan dalam pada saat pelatihan. Fungsi ini merupakan identitas untuk *input non-negatif* dan fungsi nol untuk *input negatif*. Walaupun *ReLU* memiliki kinerja yang sangat baik, jumlah elemen sparsity yang berlebihan yang dikemukakan oleh *ReLU* itu sendiri dapat merugikan jaringan saraf (Yudkk., 2020). Berikut adalah karakteristik dari fungsi aktivasi *ReLU* (Dasgupta dkk., 2021).



**Gambar 2. 3** Fungsi *ReLU*

*ReLU* mempunyai beberapa keunggulan dan juga kelemahan dalam penggunaannya sebagai fungsi aktivasi pada jaringan saraf diantaranya adalah:

### 2.7.1. Kelebihan *ReLU*

- a. Daya Ekspresif, mampu untuk mempresentasikan banyak fungsi yang berbeda yang menjadikannya sebagai pengaproksimasi fungsi universal.

- b. Efisien Dalam Aproksimasi, jaringan pada *ReLU* dibangun untuk mengaproksimasi fungsi kontinu, cembung dan halus yang berguna untuk menentukan konfigurasi jaringan yang efisien dan praktis.
- c. Reprerentasi Fungsi Afine Bagian, jaringan *ReLU* juga mampu mempresentasi fungsi bagian affine dengan bagus. Hal tersebut penting dikarenakan dapat memecah fungsi kompleks menjadi segmen affine yang membuat mudah dalam pengembangan model dan pemahaman terhadap fungsi.

### 2.7.2. Kelemahan *ReLU*

- a. Lebar Terbatas, artinya adalah memiliki batasan lebar lapisan tersembunyi dan juga terjadi pertukaran antara lebar dan kedalaman jaringan dalam mengaproksimasi fungsi tertentu.
- b. Pertimbangan Kedalaman, ketergantungan *ReLU* pada kedalaman jaringan yang menentukan seberapa bagus fungsi tersebut diaproksimasikan.
- c. Kompleksitas Dalam Representasi, walaupun *ReLU* mampu merepresentasi banyak fungsi secara efektif, *ReLU* membutuhkan beberapa hal untuk menghindari desain jaringan yang rumit yang membuat jaringan tersebut tidak efisien, diantaranya adalah akurasi aproksimasi, kedalaman aproksimasi, kompleksitas arsitektur (Hanin, 2019).

## 2.8. Jenis-Jenis Perkara

1. Perkara Pidana: Kesatuan hukum sangatlah dibutuhkan dikarenakan para pihak yang membutuhkan keadilan dalam perkaranya memperoleh penyelesaian yang sama untuk permasalahan hukum yang sama sehingga memperoleh perlakuan yang sama juga. Perkara perdata merupakan salah

satu perkara yang diwewenangkan kepada Pengadilan Negeri untuk mengadilinya (Bilaldzy & Ariani, 2022).

2. Perkara Pidana Perilaku pidana adalah salah satu bagian yang dipelajari dalam hukum pidana, karena hukum pidana tidak hanya menjelaskan tentang perbuatan yang dilarang oleh aturan hukum, tapi juga larangan dan ancaman pidana bagi pelaku tindak pidana atau pihak yang melakukan pelanggaran tersebut dan mencakup hal yang berkaitan dengan pengenaan pidana dan bagaimana pelaksanaan tata cara pidana tersebut (Zaini, 2019).
3. Perkara PTUN. Dalam perkara PTUN, penyelesaian perkara ini sedikit memiliki perbedaan dengan penyelesaian perkara lainnya. Pada umumnya pengadilan Tata Usaha Negara memiliki tahapan persidangan seperti pengadilan lainnya, akan tetapi mempunyai tahapan yang berbeda yang disebut dengan tahapan pra-persidangan. Tahapan pra-persidangan ini ada dua yaitu rapat permusyawaratan dan pemeriksaan persiapan (Wira Prakasa, 2022).
4. Perkara Agama. Salah satu contoh dari perkara agama adalah perceraian, pada UU Nomor 7 Tahun 1989 tentang Peradilan Agama dan Kompilasi Hukum Islam menegaskan bahwa ada dua jenis perceraian yaitu cerai talaq dan cerai gugat. Cerai talaq adalah permohonan cerai kepada Pengadilan Agama yang dijatuhkan oleh pihak suami kepada istrinya, sedangkan cerai gugat yaitu pengajuan gugatan oleh istri untuk memutuskan perkawinan dengan suami (Mauliddina dkk., 2021).

## 2.9. Penelitian Terdahulu

Tabel dibawah ini menunjukkan penelitian terdahulu yang digunakan sebagai rujukan penelitian:

**Tabel 2. 1** Penelitian Terdahulu

No.	Judul	Metode	Keterangan
1	<i>RNNTd: An approach based on LSTM and tensor decomposition for classification of crimes in legal cases (Guo dkk., 2019).</i>	<i>RNNTd, RNN, LSTM, Bi-LSTM, GRU, dan Bi-GRU.</i>	Penelitian ini memiliki usulan untuk model klasifikasi kasus hukum berdasarkan <i>LSTM</i> dan lapisan dekomposisi tensor yang dinamakan <i>RNNTd</i> . Hasil penelitian ini menyatakan bahwa model <i>RNNTd</i> mendapatkan akurasi yang signifikan diantara model lainnya.
2	<i>Combining Domain Knowledge Extraction With Graph Long Short-Term Memory for Learning</i>	<i>GRAPH LSTM</i> dengan kombinasi ekstraksi pengetahuan domain	Penelitian ini menggunakan metode <i>GRAPH LSTM</i> yang sudah dikominasikan dengan ekstraksi domain, pada penelitian ini hasil yang didapatkan adalah nilai akurasi sebesar 90% pada dataset yang mencakup

	<i>Classification of Chinese Legal Documents</i> (Lidkk., 2019).		sekitar 50.000 dokumen hukum china.
3	<i>Machine &amp; Deep Learning Techniques for Detection of Fake Reviews: A Survey</i> (Rodrigues dkk., 2020).	<i>LSTM, Bi-LSTM, dan GRNN</i> dengan menggunakan fungsi aktivasi <i>Sigmoid, ReLU, dan Tanh</i>	Penelitian ini dilakukan untuk mendeteksi ulasan palsu menggunakan analisis sentimen dengan memanfaatkan teknik <i>machine learning</i> dan <i>deep learning</i> seperti <i>LSTM, Bi-LSTM</i> dan <i>GRNN</i> . Dari beberapa model yang digunakan, <i>Bi-LSTM</i> mendapatkan nilai akurasi tertinggi sebesar 98,8%

## BAB III

### METODE PENELITIAN

#### 3.1. Alat dan Bahan Penelitian

Berikut adalah daftar alat dan bahan yang digunakan dalam penelitian ini:

##### 3.1.1. Perangkat Keras

Berikut ini adalah keterangan perangkat keras yang digunakan pada penelitian ini.

**Tabel 3. 1** Keterangan Perangkat Keras

No.	Perangkat Keras	Keterangan
1	<i>Device</i>	Acer One 14 Z476
2	<i>Processor</i>	Intel®Core™i3-6006U CPU @ 2.00GHz (4 CPUs), ~2.0GHz
3	HDD	1 TB
4	RAM	4,00 GB

##### 3.1.2. Perangkat Lunak

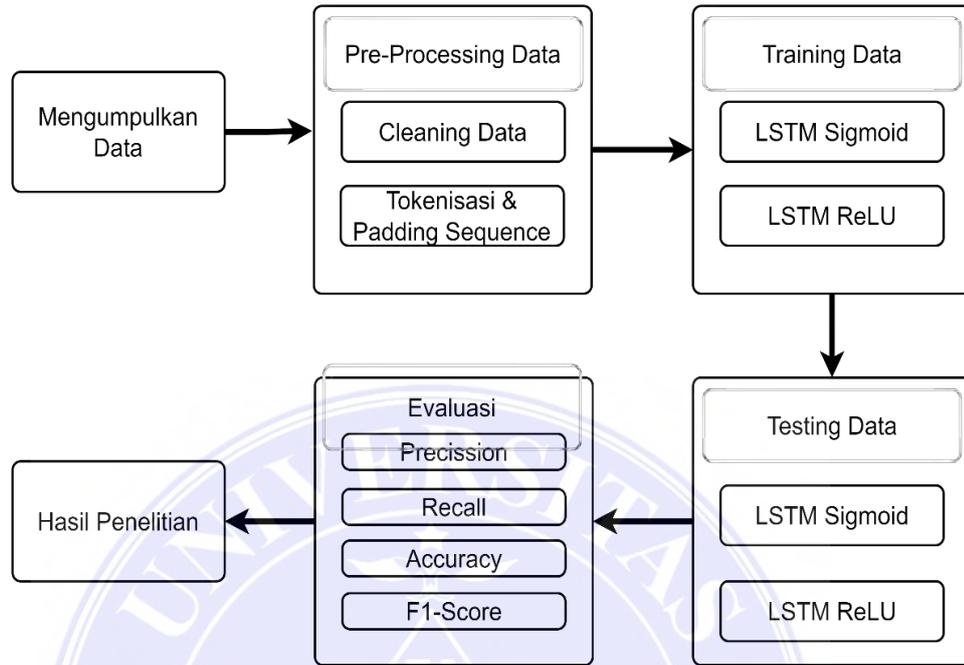
Dibawah ini merupakan keterangan perangkat lunak yang digunakan pada penelitian ini.

**Tabel 3. 2** Keterangan Perangkat Lunak

No.	Perangkat Lunak	Keterangan
1	Sistem Operasi	Windows 10 Home Single Language
2	Google Colab	-RAM: 12,7 GB -Disk: 107,7 GB
3	RAM	4,00 GB

### 3.2. Alur Penelitian

Berikut ini merupakan tahapan penelitian yang dilakukan:



**Gambar 3. 1** Alur Penelitian

Pada proses penelitian klasifikasi tentang teks sangatlah berbeda dengan penelitian klasifikasi biasanya dimana pada proses klasifikasi biasa terdapat proses *training* dan proses testing, akan tetapi pada klasifikasi teks akan ada proses lain seperti tokenisasi dan padding sequence pada teks yang dimana proses tokenisasi ini adalah mengubah serangkaian kalimat menjadi bagian kata kata atau token.

### 3.3. Metode Pengumpulan Data

Pengumpulan data terbagi kedalam dua bagian yaitu dataset dan data analis. Dataset yang digunakan adalah kronologi-kronologi perkara yang telah dikumpulkan dari salah satu instansi hukum yang bernama Kantor Hukum Perjuangan yang berada di kota medan.

**Tabel 3. 3** Contoh Data

<i>Content</i>	Label
menimbang bahwa penggugat dengan surat gugatannya tertanggal 11 desember 2023 yang diterima dan didaftarkan di kepaniteraan pengadilan negeri rembang pada tanggal 9 januari 2023 yang terdaftar diregister perkara nomor 1 pdt g s 2024 pn rbg yang isinya adalah sebagai berikut 1 bahwa penggugat merupakan hasil konsolidasi dari 27 pd bkk di jawa tengahberdasarkan peraturan daerah provinsi jawa tengah nomor 4 tahun 2017 tentang pembentukan perusahaan perseroan	Perdata
pada suatu waktu ditahun 2017 bertempat di kantor satuan polisi pamong praja satpol pp kabupaten trenggalek yang beralamat di kecamatan trenggalek kabupaten trenggalek atau setidak tidaknya pada suatu tempat yang masih termasuk dalam daerah hukum pengadilan negeri trenggalek yang berwenang memeriksa dan mengadili dengan sengaja dan tanpa hak menyebarkan informasi yang ditujukan untuk menimbulkan rasa kebencian atau permusuhan individu dan atau kelompok masyarakat tertentu berdasarkan atas suku agama ras dan antargolongan sara.	Pidana
bahwa pada tahun 2020 terjadi pertengkaran dikarenakan permasalahan kecil kemudian tergugat membesarkan hingga tergugat melakukan kdrt dengan memukul penggugat kemudian tergugat menthalak penggugat dan mempersilahkan	Agama

<p>penggugat untuk pergi dari kediaman bersama dan sejak saat itu antara tergugat dan penggugat sudah tidak hidup bersama</p>	
<p>sesuai dengan aturan yang sebenarnya adalah cacat hukum dan tidak dapat dibenarkan karena telah terbit di atas bidang tanah milik penggugat sehingga merugikan kepentingan penggugat dalam hal ini penggugat tidak dapat lagi menjual dan mengalihkan tanah yang telah dikuasai</p>	Tun

### 3.4. Pre-Processing Data

Data yang sudah dikumpulkan akan dibagi menjadi dua bagian. Pembagian dari dataset yaitu data *testing* dan data *training*. Data *testing* berjumlah 20% dari dataset sedangkan data *training* berjumlah 80% dari dataset. Dengan percobaan data analisis tersebut diharapkan mendapatkan hasil evaluasi yang bagus dalam penelitian atau lebih akurat dan handal dalam klasifikasi.

#### 3.4.1. Cleaning Data

Pada tahapan ini, file csv yang sudah diinput akan dibersihkan menggunakan fungsi `clean_csv`. Pada proses ini memastikan bahwa setiap baris di file memiliki jumlah kolom yang sesuai dengan header. Jika ada baris yang tidak sesuai, maka baris tersebut akan diabaikan. Data yang sudah valid akan dituliskan kembali kedalam file csv baru, sehingga data memiliki struktur yang benar. Tahapan selanjutnya, data diimpor kedalam dataframe, baris yang memiliki nilai kosong dikolom label akan dihapus untuk memastikan bahwa hanya data yang lengkap yang digunakan pada proses

*training*. Langkah berikutnya adalah menghapus konten duplikat, penghapusan duplikasi bertujuan untuk menghindari terjadinya *overfitting*.

### 3.4.2. Tokenisasi dan Padding Sequence

Data yang sudah dibersihkan akan dilanjutkan ketahap tokenisasi untuk mengubah kata-kata dalam teks menjadi urutan angka. Hal ini bertujuan untuk mengonversi kata menjadi representasi numerik yang dapat dipahami dan dipelajari oleh mesin. Dikarenakan model memerlukan *inputan* yang panjang yang konsisten, maka dilakukan padding pada urutan token. Proses padding berfungsi untuk menambahkan nilai nol pada awal atau akhir sehingga semua urutan memiliki panjang yang sama

Berikut beberapa contoh data yang sudah ditokenisasi dan padding.

**Tabel 3. 4** Contoh Hasil Tokenisasi

Keterangan	Contoh
Hasil Tokenisasi ( <i>Word Index</i> )	'dan': 1 'yang': 2 'dengan': 3 'penggugat': 4 'terdakwa': 5
Hasil Tokenisasi ( <i>Sequences</i> )	[794, 8, 4, 3, 31, 1787, 90, 271, 186, 24, 2, 357, 1, 835, 7, 711, 95, 179, 4963, 10]
Hasil <i>Padding Sequence</i>	[ 0 0 0 ... 2027 3514 4969], [ 0 0 0 ... 1793 836 59]

### 3.5. Training Data

#### 3.5.1. Sigmoid

Fungsi *Sigmoid* mempunyai peran penting dalam menentukan hasil *output* dari *LSTM* selama proses *training*. Fungsi ini mengonversi nilai numerik yang dihasilkan oleh neuron menjadi nilai probabilitas dalam rentang 0 hingga 1, yang sering kali digunakan dalam klasifikasi biner atau probabilitas kelas tertentu. Hal ini membantu model untuk menunjukkan seberapa besar kemungkinan sebuah *input* termasuk kedalam kelas yang ada.

Proses *training* menggunakan jumlah *epoch* 20, *batch size* 64, *optimizer adam*. Selama proses *training* berjalan, *output* yang dihasilkan oleh fungsi *Sigmoid* digunakan untuk menghitung nilai *loss*. Model memperkecil nilai *loss* ini dengan memperbaharui bobotnya menggunakan metode *backpropagation*, sehingga *Sigmoid* memainkan peran penting pada proses optimasi model.

#### 3.5.2. ReLU

Fungsi *ReLU* yang bersifat linear pada nilai positif, dimana *ReLU* akan membiarkan nilai *input* yang positif tetap seperti semula sementara nilai negatif *output*nya menjadi nol. Ini membantu *LSTM* dalam proses *training* dengan mempercepat konvergensi, terutama ketika memproses data yang kompleks. Proses *training* menggunakan jumlah *epoch* 20, *batch size* 64, *optimizer adam*.

*ReLU* bertugas untuk mengolah informasi yang diekstrak oleh layer sebelumnya yaitu layer *LSTM* dan mengubahnya menjadi representasi yang lebih bagus untuk *output* akhir.

### 3.6. Metode Evaluasi

Pada *Confusion Matrix* disetiap kolom mewakili contoh kelas yang diprediksi sedangkan disetiap baris mewakili contoh dalam kelas yang sebenarnya. *Confusion Matrix* ini menunjukkan bagaimana cara pemodelan klasifikasi “*confused*” saat membuat prediksi. Hal ini dapat memberikan wawasan tentang kesalahan yang dibuat pengklasifikasi dan juga dapat mengetahui jenis kesalahan dalam klasifikasi tersebut (Markoulidakis dkk., 2021).

Dalam tugas mengklasifikasi, ada 5 nilai yang digunakan dalam menentukan kemampuan dalam mengklasifikasi. Nilai-nilai tersebut adalah akurasi, presisi, *recall*, *f1-score*, dan *f2-score*.

a. Akurasi

Akurasi merupakan tolak ukur dalam klasifikasi tentang sejauh mana klasifikasi itu dapat mengidentifikasi kedua kelas yaitu positif dan negatif dan membandingkan dengan jumlah dataset yang ada. Rumus akurasi adalah sebagai berikut:

$$\text{Akurasi: } \frac{TP+TN}{TP+TN+FP+FN} \dots\dots\dots 3.1$$

b. Presisi

Presisi menunjukkan keakuratan prediksi positif dari hasil klasifikasi. Rumus presisi adalah:

$$\text{Presisi: } \frac{TP}{TP+FP} \dots\dots\dots 3.2$$

c. *Recall*

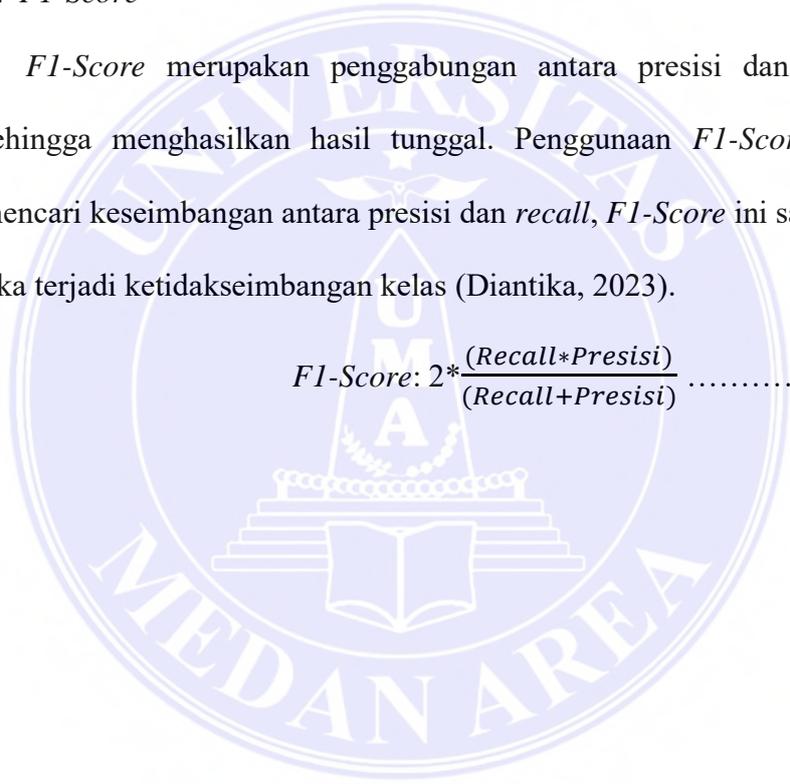
*Recall* merupakan salah satu matriks evaluasi yang sangat penting dalam klasifikasi data. *Recall* mengukur tingkat keberhasilan dalam mengenali data yang benar tanpa terpengaruh pada pengenalan data positif yang salah. Rumus dari *recall* adalah:

$$Recall: \frac{TP}{TP+FN} \dots\dots\dots 3.3$$

d. *F1-Score*

*F1-Score* merupakan penggabungan antara presisi dan *recall* yang sehingga menghasilkan hasil tunggal. Penggunaan *F1-Score* ini adalah mencari keseimbangan antara presisi dan *recall*, *F1-Score* ini sangat berguna jika terjadi ketidakseimbangan kelas (Diantika, 2023).

$$F1-Score: 2 * \frac{(Recall * Presisi)}{(Recall + Presisi)} \dots\dots\dots 3.4$$



## BAB V

### KESIMPULAN DAN SARAN

#### 5.1. Kesimpulan

Penelitian ini menganalisis kinerja fungsi aktivasi pada algoritma *LSTM* untuk klasifikasi teks kronologi perkara hukum. Pada penelitian ini model dilatih masing-masing selama 20 *epoch* . Hasil penelitian menunjukkan bahwa pemilihan fungsi aktivasi sangat mempengaruhi akurasi dan efektivitas model.

##### 1. Pengaruh Fungsi Aktivasi *Sigmoid*

Pada saat proses *epoch* dilakukan, awalnya sistem terlihat lambat karna gradien pada *Sigmoid* bisa sangat kecil jika *inputannya* jauh dari 0, sehingga dapat memperlambat pembaruan bobot. Namun pada *epoch* tengah dan akhir sistem mengalami perubahan besar yang tidak tetap yang dimungkinkan ini terjadi karna sifat dari *Sigmoid* yang bisa memperparah *overfitting*. Dengan fungsi aktivasi ini, hasil akurasi yang didapatkan adalah 85%

##### 2. Pengaruh Fungsi Aktivasi *ReLU*

Pada saat proses *epoch* awal, kesalahan model saat belajar dari data yang sudah dipelajari ataupun yang disebut dengan *training loss* menurun secara bertahap dan akurasi meningkat. Tetapi *vall loss* atau kesalahan pada data yang belum dipelajari menunjukkan hasil yang berubah-ubah. Pada fungsi aktivasi ini, hasil akurasi yang dicapai adalah 74%

##### 3. Kinerja Model *LSTM*

Akurasi pada data *training* dan data testing dengan model ini sangat bagus meskipun data testing tidak sebagus data *training*. Hal tersebut menunjukkan

model mampu mengenali pola pada data *training*. Ada beberapa tanda model mengalami *overfitting*, yaitu terjadinya perbandingan antara *training loss* dan *validation loss* selama beberapa *epoch*, terutama pada *epoch* terakhir.

## 5.2. Saran

Saran yang dapat disampaikan penulis untuk pengembangan dan penelitian selanjutnya adalah:

### 1. Penelitian Lanjutan

Melakukan penelitian yang lebih lanjut yang bertujuan untuk menjelajahi dampak variabel lain yang mungkin bisa mempengaruhi kinerja dari model yang dibangun.

### 2. Peningkatan Dataset

Mengumpulkan lebih banyak lagi dataset tentang kronologi hukum untuk melatih model atau menggunakan data dengan jenis yang berbeda dengan tujuan meningkatkan ke-akuratan dan generalisasi model serta mempertimbangkan variasi data yang lebih banyak.

### 3. Pengembangan Model

Melakukan optimisasi model dengan teknik lain dan menggunakan fungsi aktivasi lain dengan tujuan meningkatkan kinerja secara keseluruhan untuk klasifikasi teks kronologi perkara, contohnya menambahkan teknik regulasi untuk mengurangi terjadinya *overfitting*.

## DAFTAR PUSTAKA

- Apicella, A., Donnarumma, F., Isgrò, F., & Prevete, R. (2021). A survey on modern trainable activation functions. *Elsevier*, 138, 14–32. <https://doi.org/10.1016/j.neunet.2021.01.026>
- Bilaldzy, A., & Ariani, R. S. (2022). Tinjauan kritis urgensi pembentukan pengadilan Agraria : upaya menangani inefektivitas penyelesaian Konflik agraria pada peradilan tata usaha negara dan Peradilan umum. *Jurnal Hukum Lex Generalin*, 3. <https://lokadata.beritagar.id/chart/preview/konflik-agraria-di-Indonesia-2010->
- Dasgupta, R., Chowdhury, Y. S., & Nanda, S. (2021). Performance Comparison of Benchmark Activation Function ReLU, Swish and Mish for Facial Mask Detection Using Convolutional Neural Network. *ResearchGate*, 355–367. [https://doi.org/10.1007/978-981-16-2248-9\\_34](https://doi.org/10.1007/978-981-16-2248-9_34)
- Diantika, S. (2023). Penerapan Teknik Random Oversampling Untuk Mengatasi Imbalance Class Dalam Klasifikasi Website Phishing Menggunakan Algoritma Lightgbm. *Jurnal Mahasiswa Teknik Informatika*, 7(1).
- Dogan, A., & Birant, D. (2021). Machine learning and data mining in manufacturing. *Expert Systems with Applications*, 166. <https://doi.org/10.1016/j.eswa.2020.114060>
- Guo, X., Zhang, H., Ye, L., & Li, S. (2019). RNNTd: An approach based on LSTM and tensor decomposition for classification of crimes in legal cases. *Proceedings - 2019 IEEE 4th International Conference on Data Science in Cyberspace, DSC 2019*, 16–22. <https://doi.org/10.1109/DSC.2019.00012>
- Gupta, A., Dengre, V., Kheruwala, H. A., & Shah, M. (2020). Comprehensive review of text-mining applications in finance. *Financial Innovation*, 6(1). <https://doi.org/10.1186/s40854-020-00205-1>
- Gupta, M. K., & Chandra, P. (2020). A comprehensive survey of data mining. *International Journal of Information Technology (Singapore)*, 12(4), 1243–1257. <https://doi.org/10.1007/s41870-020-00427-7>
- Hanin, B. (2019). Universal function approximation by deep neural nets with bounded width and ReLU activations. *Mathematics*, 7(10). <https://doi.org/10.3390/MATH7100992>
- Hao, S., Ge, F. X., Li, Y., & Jiang, J. (2020). Multisensor bearing fault diagnosis based on one-dimensional convolutional long short-term memory networks. *Measurement: Journal of the International Measurement Confederation*, 159. <https://doi.org/10.1016/j.measurement.2020.107802>

- Hendrian, S. (2018). Algoritma Klasifikasi Data Mining Untuk Memprediksi Siswa Dalam Memperoleh Bantuan Dana Pendidikan. *Faktor Exacta*, 11(3). <https://doi.org/10.30998/faktorexacta.v11i3.2777>
- Hickman, L., Thapa, S., Tay, L., Cao, M., & Srinivasan, P. (2022). Text Preprocessing for Text Mining in Organizational Research: Review and Recommendations. *Organizational Research Methods*, 25(1), 114–146. <https://doi.org/10.1177/1094428120971683>
- Kindbom, H. (2019). LSTM vs Random Forest for Binary Classification of Insurance Related [Thesis]. In KTH Royal Institute of Technology . KTH Royal Institute of Technology.
- Kumar, S., Kar, A. K., & Ilavarasan, P. V. (2021). Applications of text mining in services management: A systematic literature review. In *International Journal of Information Management Data Insights* (Vol. 1, Issue 1). Elsevier Ltd. <https://doi.org/10.1016/j.jjime.2021.100008>
- Lam, P., Pham, L., Nguyen, T., Tang, H., Seidl, M., Andresel, M., & Schindler, A. (2024). LSTM-based Deep Neural Network With A Focus on Sentence Representation for Sequential Sentence Classification in Medical Scientific Abstracts. *ArXiv*. <http://arxiv.org/abs/2401.15854>
- Lan, K., Wang, D. tong, Fong, S., Liu, L. sheng, Wong, K. K. L., & Dey, N. (2018). A Survey of Data Mining and Deep Learning in Bioinformatics. In *Journal of Medical Systems* (Vol. 42, Issue 8). Springer New York LLC. <https://doi.org/10.1007/s10916-018-1003-9>
- Li, G., Wang, Z., & Ma, Y. (2019). Combining domain knowledge extraction with graph long short-term memory for learning classification of Chinese legal documents. *IEEE Access*, 7. <https://doi.org/10.1109/ACCESS.2019.2943668>
- Liu, W., Wang, Z., Yuan, Y., Zeng, N., Hone, K., & Liu, X. (2021). A Novel Sigmoid-Function-Based Adaptive Weighted Particle Swarm Optimizer. *IEEE Transactions on Cybernetics*, 51(2), 1085–1093. <https://doi.org/10.1109/TCYB.2019.2925015>
- Luan, Y., & Lin, S. (2019). Research on Text Classification Based on CNN and LSTM. *IEEE*.
- Lyu, Z., Wang, Y., Li, W., Guo, L., Yang, J., Sun, J., Liu, M., & Gui, G. (2020). Robust automatic modulation classification based on convolutional and recurrent fusion network. *Physical Communication*, 43. <https://doi.org/10.1016/j.phycom.2020.101213>
- Markoulidakis, I., Kopsiaftis, G., Rallis, I., & Georgoulas, I. (2021). Multi-Class Confusion Matrix Reduction method and its application on Net Promoter Score classification problem. *ACM International Conference Proceeding Series*, 412–419. <https://doi.org/10.1145/3453892.3461323>

- Mauliddina, S., Puspitawati, A., Aliffia, S., Kusumawardani, D. D., & Amalia, R. (2021). Analisis Faktor-Faktor Yang Mempengaruhi Tingginya Angka Perceraian Pada Masa Pandemi Covid-19: A Systematic Review. *JURNAL KESEHATAN TAMBUSAI*, 2(3).
- Nagabushanam, P., Thomas George, S., & Radha, S. (2020). EEG signal classification using LSTM and improved neural network algorithms. *Soft Computing*, 24(13), 9981–10003. <https://doi.org/10.1007/s00500-019-04515-0>
- Noh, Y.-D., & Cho, K.-C. (2021). A Text Content Classification Using LSTM For Objective Category Classification. *한국컴퓨터정보학회논문지 Journal of The Korea Society of Computer and Information*, 26(5), 39–46. <https://doi.org/10.9708/jksci.2021.26.05.039>
- Peng, H., Huang, S., Zhou, + Tong, Luo, Y., Wang, C., Wang, Z., Zhao, J., Xie, X., Li, A., Geng, T., Mahmood, K., Wen, W., Xu, X., & Ding, C. (2023). AutoReP: Automatic ReLU Replacement for Fast Private Network Inference. Computer Vision Foundation. <https://github.com/HarveyP123/AutoReP>
- Pratheebha, T., Indhumathi, V., & Megala, S. S. (2021). An Empirical Study On Data Mining Techniques And Its Applications. *International Journal of Software & Hardware Research in Engineering*, 9(4). <https://doi.org/10.26821/ijshre.9.4.2021.9410>
- Rasamoelina, A. D., Adjailia, F., & Sin'cak, P. (2020). A Review of Activation Function for Artificial Neural Network. *IEEE*.
- Reddy Prasad, B., & Deepa, N. (2021). Classification of Analyzed Text in Speech Recognition Using RNN-LSTM in Comparison with Convolutional Neural Network to Improve Precision for Identification of Keywords. *Geintec*, 11(2). <https://doi.org/10.03.2021>
- Rodrigues, J. C., Rodrigues, J. T., Gonsalves, V. L. K., Naik, A. U., Shetgaonkar, P., & Aswale, S. (2020). Machine Deep Learning Techniques for Detection of Fake Reviews: A Survey. *International Conference on Emerging Trends in Information Technology and Engineering, Ic-ETITE 2020*. <https://doi.org/10.1109/ic-ETITE47903.2020.063>
- Roodschild, M., Gotay Sardiñas, J., & Will, A. (2020). A new approach for the vanishing gradient problem on Sigmoid activation. *Progress in Artificial Intelligence*, 9(4), 351–360. <https://doi.org/10.1007/s13748-020-00218-y>
- Salih, A. A., & Abdulazeez, A. M. (2021). Evaluation of Classification Algorithms for Intrusion Detection System: A Review. *Journal of Soft Computing and Data Mining*, 02(01). <https://doi.org/10.30880/jscdm.2021.02.01.004>

- Taky-eddine, O., & Madaoui, R. (2024). Cognitive Overload in the Hypertext Reading Environment. *International Journal of English Language Studies*. <https://doi.org/10.32996/ijels>
- Taunk, K., De, S., Verma, S., & Swetapadma, A. (2019). A Brief Review of Nearest Neighbor Algorithm for Learning and Classification. *ICICCS*.
- Tharwat, A. (2018). Classification assessment methods. *Applied Computing and Informatics*, 17(1), 168–192. <https://doi.org/10.1016/j.aci.2018.08.003>
- Tri Hermanto, D., Setyanto, A., & Luthfi, E. T. (2021). Algoritma LSTM-CNN untuk Sentimen Klasifikasi dengan Word2vec pada Media Online LSTM-CNN Algorithm for Sentiment Clasification with Word2vec On Online Media. *Citec Journal*, 8.
- Van Houdt, G., Mosquera, C., & Nápoles, G. (2020). A review on the long short-term memory model. *Artificial Intelligence Review*, 53(8), 5929–5955. <https://doi.org/10.1007/s10462-020-09838-1>
- Villanueva, A., & Moreno, L. G. (2018). Data mining techniques applied in educational environments: Literature review Andrés Villanueva Manjarres Data mining techniques applied in educational environments: Literature review. In *Salinas Digital Education Review-Number (Vol. 33)*. <http://greav.ub.edu/der/>
- Wachter, C. (2022). REPRESENTING SPATIAL CONCEPTS: MODERN EAST ASIAN HISTORY IN A DIGITAL PUBLICATION FORMAT. *History and Theory*, 61(4), 178–190. <https://doi.org/10.1111/hith.12285>
- Wira Prakasa, A. (2022). Esensi Prosedur Dismissal Dalam Tahap Pra-Persidangan Sengketa Tata Usaha Negara. *Borneo Law Review Universitas Borneo Tarakan*, 6(2).
- Xing, S., & Wu, C. (2020). Implementation of A Neuron Using Sigmoid Activation Function with CMOS. *2020 5th International Conference on Integrated Circuits and Microsystems, ICICM 2020*, 201–204. <https://doi.org/10.1109/ICICM50929.2020.9292239>
- Yahya, A. A., & Osman, A. (2019). Using Data Mining Techniques to Guide Academic Programs Design and Assessment. *Procedia Computer Science*, 163, 472–481. <https://doi.org/10.1016/j.procs.2019.12.130>
- Yusro, A. Y. (2023). Implementation Of Long Short Term Memory (LSTM) Algorithm For Time Series Forecasting Of Instagram Account Engagement. In *DIGITAL REPOSITORY UNILA*. University Of Lampung.
- Yu, Y., Adu, K., Tashi, N., Anokye, P., Wang, X., & Ayidzoe, M. A. (2020). RMAF: ReLU-Memristor-Like Activation Function for Deep Learning. *IEEE Access*, 8, 72727–72741. <https://doi.org/10.1109/ACCESS.2020.2987829>

Zaini. (2019). Tinjauan Konseptual Tentang Pidana dan Pemidanaan Conceptual Review of Criminal and Criminal. Voice Justika Jurnal Hukum Dan Keadilan, 3.



## LAMPIRAN

```

import pandas as pd
import numpy as np
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, LSTM, Embedding, Dropout
from tensorflow.keras.preprocessing.text import Tokenizer
from tensorflow.keras.preprocessing.sequence import pad_sequences
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder
from sklearn.metrics import classification_report, confusion_matrix
import matplotlib.pyplot as plt
import seaborn as sns
from imblearn.over_sampling import SMOTE

# Unggah file CSV
from google.colab import files
uploaded = files.upload()

# Nama file input yang diunggah
input_file = 'data terbaru.csv'
output_file = 'cleaned_dataset.csv'

# Fungsi untuk memeriksa dan membersihkan baris yang tidak sesuai
def clean_csv(input_file, output_file, encoding='latin1'):
    with open(input_file, 'r', encoding=encoding) as file:
        header = file.readline()
        expected_columns = len(header.split(','))

    valid_lines = []
    with open(input_file, 'r', encoding=encoding) as file:
        for i, line in enumerate(file):
            if len(line.split(',')) == expected_columns:
                valid_lines.append(line)
            else:
                print(f"Baris {i+1} memiliki jumlah kolom yang tidak sesuai")

    with open(output_file, 'w', encoding=encoding) as file:
        file.writelines(valid_lines)

# Membersihkan file CSV
clean_csv(input_file, output_file)

# Memuat data yang sudah dibersihkan
data = pd.read_csv(output_file, encoding='latin1')
print("Data telah dibersihkan dan dimuat ke dalam DataFrame.")

```

```

# Menghapus baris dengan nilai kosong di kolom label
data_cleaned = data.dropna(subset=['label'])

# Menghapus baris duplikat berdasarkan kolom content
data_cleaned = data_cleaned.drop_duplicates(subset=['content'])

# Daftar label yang valid
valid_labels = ['Tun', 'Agama', 'Perdata', 'Pidana']

# Fungsi untuk membersihkan label yang tidak konsisten
def clean_label(label):
    label = label.strip() # Menghilangkan spasi di awal dan akhir
    label = label.replace("'", "") # Menghilangkan tanda kutip
    label = label.capitalize() # Mengoreksi kapitalisasi
    if label == 'Promo ridana':
        label = 'Pidana' # Mengoreksi typo
    return label

# Menerapkan fungsi clean_label ke kolom label
data_cleaned['label'] = data_cleaned['label'].apply(clean_label)

# Memfilter hanya label yang valid
data_cleaned = data_cleaned[data_cleaned['label'].isin(valid_labels)]

# Menampilkan distribusi label setelah pembersihan
print(data_cleaned['label'].value_counts())

# Menyimpan DataFrame yang sudah dibersihkan ke file baru
data_cleaned.to_csv('final_cleaned_dataset.csv', index=False)

# Menampilkan informasi dasar tentang data yang sudah dibersihkan
print(data_cleaned.info())
print(data_cleaned.head())

# Memuat data yang sudah dibersihkan
data_cleaned = pd.read_csv('final_cleaned_dataset.csv')

# Menampilkan jumlah data untuk setiap label
print("Jumlah data untuk setiap label:")
print(data_cleaned['label'].value_counts())
print("\n")

# Mendefinisikan fungsi untuk menampilkan beberapa entri untuk setiap label
def show_samples_for_each_label(data, labels, num_samples=5):
    for label in labels:
        print(f"Label: {label}")
        print(data[data['label'] == label].head(num_samples))
        print("\n")

```

```

# Menampilkan beberapa contoh entri untuk setiap label
show_samples_for_each_label(data_cleaned, valid_labels, num_samples=5)

# Tokenisasi dan padding sequence
tokenizer = Tokenizer(num_words=5000)
tokenizer.fit_on_texts(data_cleaned['content'])
sequences = tokenizer.texts_to_sequences(data_cleaned['content'])
max_sequence_length = max([len(seq) for seq in sequences])
data_padded = pad_sequences(sequences, maxlen=max_sequence_length)

# Encode labels
label_encoder = LabelEncoder()
labels_encoded = label_encoder.fit_transform(data_cleaned['label'])

# Oversampling using SMOTE
smote = SMOTE(random_state=42)
X_resampled, y_resampled = smote.fit_resample(data_padded, labels_encoded)

# Print hasil tokenisasi
word_index = tokenizer.word_index
print("Hasil Tokenisasi (Word Index):")
print(word_index)
print("\nHasil Tokenisasi (Sequences):")
print(sequences[:5]) # Menampilkan 5 contoh sequence yang telah ditokenisasi
print("\nHasil Padding (Padded Sequences):")
print(data_padded[:5]) # Menampilkan 5 contoh padded sequences

# Build and train the models

# Split the data
X_train, X_test, y_train, y_test = train_test_split(X_resampled, y_resampled,
test_size=0.2, random_state=42)

# Model LSTM murni
def build_lstm_model():
    model = Sequential()
    model.add(Embedding(input_dim=5000, output_dim=128,
input_length=max_sequence_length))
    model.add(LSTM(150))
    model.add(Dropout(0.2))
    model.add(Dense(4, activation='softmax'))
    model.compile(loss='sparse_categorical_crossentropy', optimizer='adam',
metrics=['accuracy'])
    return model

# Training model LSTM murni
print("Training LSTM model...")

```

```

lstm_model = build_lstm_model()
lstm_history = lstm_model.fit(X_train, y_train, epochs=20, batch_size=64,
validation_data=(X_test, y_test), verbose=2)

# Predict labels on test data
lstm_y_pred_prob = lstm_model.predict(X_test)
lstm_y_pred = np.argmax(lstm_y_pred_prob, axis=1)

# Evaluation: Confusion matrix
lstm_cm = confusion_matrix(y_test, lstm_y_pred)
plt.figure(figsize=(8, 6))
sns.heatmap(lstm_cm, annot=True, fmt='d', cmap='Blues',
xticklabels=label_encoder.classes_, yticklabels=label_encoder.classes_)
plt.xlabel('Predicted Labels')
plt.ylabel('True Labels')
plt.title('Confusion Matrix - LSTM')
plt.show()

# Classification Report
print("Classification Report - LSTM:")
print(classification_report(y_test, lstm_y_pred,
target_names=label_encoder.classes_))

# Save test accuracy results
lstm_test_loss, lstm_test_accuracy = lstm_model.evaluate(X_test, y_test,
verbose=0)
print(f'Test accuracy - LSTM: {lstm_test_accuracy}')

# Split the data
X_train, X_test, y_train, y_test = train_test_split(X_resampled, y_resampled,
test_size=0.2, random_state=42)

# Model LSTM dengan sigmoid
def build_lstm_sigmoid_model():
    model = Sequential()
    model.add(Embedding(input_dim=5000, output_dim=128,
input_length=max_sequence_length))
    model.add(LSTM(150))
    model.add(Dropout(0.2))
    model.add(Dense(4, activation='sigmoid'))
    model.compile(loss='sparse_categorical_crossentropy', optimizer='adam',
metrics=['accuracy'])
    return model
# Training model LSTM dengan sigmoid
print("Training LSTM with Sigmoid model...")
lstm_sigmoid_model = build_lstm_sigmoid_model()

```

```

lstm_sigmoid_history = lstm_sigmoid_model.fit(X_train, y_train, epochs=20,
batch_size=64, validation_data=(X_test, y_test), verbose=2)

# Predict labels on test data
lstm_sigmoid_y_pred_prob = lstm_sigmoid_model.predict(X_test)
lstm_sigmoid_y_pred = np.argmax(lstm_sigmoid_y_pred_prob, axis=1)

# Evaluation: Confusion matrix
lstm_sigmoid_cm = confusion_matrix(y_test, lstm_sigmoid_y_pred)
plt.figure(figsize=(8, 6))
sns.heatmap(lstm_sigmoid_cm, annot=True, fmt='d', cmap='Blues',
xticklabels=label_encoder.classes_, yticklabels=label_encoder.classes_)
plt.xlabel('Predicted Labels')
plt.ylabel('True Labels')
plt.title('Confusion Matrix - LSTM with Sigmoid')
plt.show()

# Classification Report
print("Classification Report - LSTM with Sigmoid:")
print(classification_report(y_test, lstm_sigmoid_y_pred,
target_names=label_encoder.classes_))

# Save test accuracy results
lstm_sigmoid_test_loss, lstm_sigmoid_test_accuracy =
lstm_sigmoid_model.evaluate(X_test, y_test, verbose=0)
print(f'Test accuracy - LSTM with Sigmoid: {lstm_sigmoid_test_accuracy}')

# Split the data
X_train, X_test, y_train, y_test = train_test_split(X_resampled, y_resampled,
test_size=0.2, random_state=42)

# Model LSTM dengan ReLU
def build_lstm_relu_model():
    model = Sequential()
    model.add(Embedding(input_dim=5000, output_dim=128,
input_length=max_sequence_length))
    model.add(LSTM(150, dropout=0.2, recurrent_dropout=0.2))
    model.add(Dense(100, activation='relu'))
    model.add(Dense(4, activation='softmax')) # Output layer dengan softmax
untuk klasifikasi
    model.compile(loss='sparse_categorical_crossentropy', optimizer='adam',
metrics=['accuracy'])
    return model

# Training model LSTM dengan ReLU
print('Training LSTM with ReLU model...')
lstm_relu_model = build_lstm_relu_model()

```

```
lstm_relu_history = lstm_relu_model.fit(X_train, y_train, epochs=20,
batch_size=64, validation_data=(X_test, y_test), verbose=2)

# Predict labels on test data
lstm_relu_y_pred_prob = lstm_relu_model.predict(X_test)
lstm_relu_y_pred = np.argmax(lstm_relu_y_pred_prob, axis=1)

# Evaluation: Confusion matrix
lstm_relu_cm = confusion_matrix(y_test, lstm_relu_y_pred)
plt.figure(figsize=(8, 6))
sns.heatmap(lstm_relu_cm, annot=True, fmt='d', cmap='Blues',
xticklabels=label_encoder.classes_, yticklabels=label_encoder.classes_)
plt.xlabel('Predicted Labels')
plt.ylabel('True Labels')
plt.title('Confusion Matrix - LSTM with ReLU')
plt.show()

# Classification Report
print("Classification Report - LSTM with ReLU:")
print(classification_report(y_test, lstm_relu_y_pred,
target_names=label_encoder.classes_))

# Save test accuracy results
lstm_relu_test_loss, lstm_relu_test_accuracy =
lstm_relu_model.evaluate(X_test, y_test, verbose=0)
print(f'Test accuracy - LSTM with ReLU: {lstm_relu_test_accuracy}')
```



**UNIVERSITAS MEDAN AREA**  
**FAKULTAS TEKNIK**

Kampus I : Jalan Kolan Nomor 1 Medan Estate/Jalan PBSI Nomor 1 (061) 7366878, 7360168, 7364348, 7366781, Fax (061) 7366998 Medan 20223  
Kampus II : Jalan Setebud Nomor 79 / Jalan Sei Serayu Nomor 70 A (061) 8225902, Fax (061) 8225331 Medan 20122  
Website: www.teknik.uma.ac.id E-mail: univ\_medanarea@uma.ac.id

---

Nomor : 281/FT.6/01.10/VI/2024  
Lamp :  
Hal : **Perubahan Judul Tugas Akhir** 24 Juni 2024

Yth, Pembimbing Tugas Akhir  
**Susilawati, S.Kom, M.Kom**  
di  
Tempat

Dengan hormat, Sehubungan dengan adanya perubahan judul tugas akhir maka perlu diterbitkan kembali SK Pembimbing Skripsi baru atas nama mahasiswa tersebut :

Nama : Daniel Royhan Suhega Harahap  
N P M : 198160050  
Jurusan : Teknik Informatika

Maka dengan hormat kami mengharapkan kesediaan saudara :

**Susilawati, S.Kom, M.Kom** ( Sebagai Pembimbing )

Adapun Tugas Akhir Skripsi berjudul :

**"Analisis Fungsi Aktivasi pada Algoritma Convolutional Long Short-Term Memory untuk Klasifikasi Teks".**

SK Pembimbing ini berlaku selama enam bulan terhitung sejak SK ini diterbitkan. Jika proses pembimbing melebihi batas waktu yang telah ditetapkan, SK ini dapat ditinjau ulang.

Demikian kami sampaikan, atas kesediaan saudara diucapkan terima kasih.

  
Dekan,  
**Dr. Eng. Supriatno, ST, MT**

 **UNIVERSITAS MEDAN AREA**  
**FAKULTAS TEKNIK**

Kampus I : Jalan Kolam Nomor 1 Medan Estate/Jalan PBSI Nomor 1 (061) 7366678, 7360168, 7364348, 7366781, Fax: (061) 7369968 Medan 20223  
Kampus II : Jalan Setiabudi Nomor 79 / Jalan Sei Serayu Nomor 70 A, (061) 8225602, Fax: (061) 8226331 Medan 20122  
Website: www.teknik.uma.ac.id E-mail: univ\_medanama@uma.ac.id

Nomor : 51 /FT.6/01.10/I/2024  
Lamp : -  
Hal : -

16 Januari 2024

**: Penelitian Dan Pengambilan Data Tugas Akhir**

Yth. Pimpinan Kantor Hukum Perjuangan Consellor At Law  
Jln. Pahlawan No.61  
Di  
Medan

Dengan hormat,  
Kami mohon kesediaan Bapak/Ibu berkenan untuk memberikan izin dan kesempatan kepada mahasiswa kami tersebut dibawah ini :

NO	NAMA	NPM	PRODI
1	Daniel Royhan Suhega Harahap	198160050	Teknik Informatika

Untuk melaksanakan Penelitian dan Pengambilan Data Tugas Akhir pada perusahaan/Instansi yang Bapak/Ibu Pimpin.

Perlu kami jelaskan bahwa Pengambilan Data tersebut adalah semata-mata untuk tujuan ilmiah dan Skripsi yang merupakan salah satu syarat bagi mahasiswa tersebut untuk mengikuti ujian sarjana pada Fakultas Teknik Universitas Medan Area dan tidak untuk dipublikasikan, dengan judul penelitian :

**Penerapan Metode Convolutional-Long Short Term Memory untuk Klasifikasi Jenis Perkara Berdasarkan Kronologi Perkara**

Atas perhatian dan kerja sama yang baik diucapkan terima kasih.

  
Dekan.  
Dr. Eng. Supriano, ST, MT

Tembusan :  
1. Ka. BAMAI  
2. Mahasiswa  
3. File

