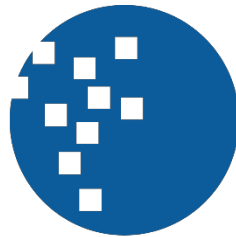


**IMPLEMENTASI ON-DEVICE OFFLINE LEARNING  
MENGUNAKAN ALGORITMA LONG-SHORT TERM  
MEMORY UNTUK NEXT-WORD PREDICTION DALAM  
BAHASA INDONESIA**



**UMN**

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

TUGAS AKHIR

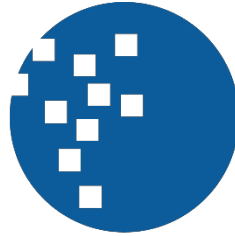
**Jovanko Kenshian**

**0000032025**

**PROGRAM STUDI TEKNIK KOMPUTER  
FAKULTAS TEKNIK DAN INFORMATIKA  
UNIVERSITAS MULTIMEDIA NUSANTARA  
TANGERANG**

**2022**

**IMPLEMENTASI ON-DEVICE OFFLINE LEARNING  
MENGUNAKAN ALGORITMA LONG-SHORT TERM  
MEMORY UNTUK NEXT-WORD PREDICTION DALAM  
BAHASA INDONESIA**



**UMN**  
UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

**TUGAS AKHIR**

Diajukan sebagai Salah Satu Syarat untuk Memperoleh  
Gelar Sarjana Teknik Komputer

**Jovanko Kenshian**

**0000032025**

**PROGRAM STUDI TEKNIK KOMPUTER  
TEKNIK DAN INFORMATIKA  
UNIVERSITAS MULTIMEDIA NUSANTARA  
TANGERANG**

**2022**

i

Implementasi On-Device Offline Learning Menggunakan Algoritma Long-Short Term Memory untuk Next-Word Prediction dalam Bahasa Indonesia, Jovanko Kenshian, Universitas Multimedia Nusantara

## PERNYATAAN TIDAK MELAKUKAN PLAGIAT

Dengan ini saya:

Nama : Jovanko Kenshian

NIM : 00000032025

Program Studi : Teknik Komputer

Tugas Akhir dengan judul:

IMPLEMENTASI ON-DEVICE OFFLINE LEARNING MENGGUNAKAN ALGORITMA LONG-SHORT TERM MEMORY UNTUK NEXT-WORD PREDICTION DALAM BAHASA INDONESIA

merupakan hasil karya saya sendiri bukan plagiat dari karya ilmiah yang ditulis oleh orang lain, dan semua sumber, baik yang dikutip maupun dirujuk, telah saya nyatakan dengan benar serta dicantumkan di Daftar Pustaka.

Jika di kemudian hari terbukti ditemukan kecurangan/penyimpangan, baik dalam pelaksanaan skripsi maupun dalam penulisan laporan skripsi, saya bersedia menerima konsekuensi dinyatakan TIDAK LULUS untuk Tugas Akhir yang telah saya tempuh.

Tangerang, 23 Juni 2022



(Jovanko Kenshian)

# HALAMAN PENGESAHAN TUGAS AKHIR

Tugas Akhir dengan judul

IMPLEMENTASI ON-DEVICE OFFLINE LEARNING MENGGUNAKAN  
ALGORITMA LONG-SHORT TERM MEMORY UNTUK NEXT-WORD  
PREDICTION DALAM BAHASA INDONESIA

Oleh

Nama : Jovanko Kenshian  
NIM : 00000032025  
Program Studi : Teknik Komputer  
Fakultas : Teknik dan Informatika

Telah diujikan pada hari Kamis, 23 Juni 2022

Pukul 08.00 s.d 10.00 dan dinyatakan

LULUS

Dengan susunan pengujian sebagai berikut.

Ketua Sidang



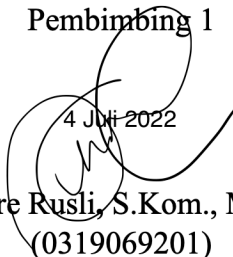
(Samuel, M.T.I.)  
(0304038902)

Penguji



(Nabila Husna Shabrina, S.T., M.T.)  
(0321099301)

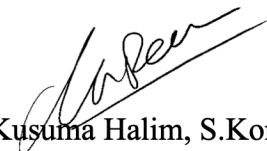
Pembimbing 1



4 Juni 2022

(Andre Rusli, S.Kom., M.Sc.)  
(0319069201)

Pembimbing 2



(Dareen Kusuma Halim, S.Kom., M.Eng.Sc.)  
(0317129202)

Ketua Program Studi Teknik Komputer



(Samuel, M.T.I.)

iii

## HALAMAN PERSETUJUAN PUBLIKASI KARYA ILMIAH UNTUK KEPENTINGAN AKADEMIS

Sebagai civitas academica Universitas Multimedia Nusantara, saya yang bertanda tangan di bawah ini:

Nama : Jovanko Kenshian  
NIM : 00000032025  
Program Studi : Teknik Komputer  
Fakultas : Teknik dan Informatika  
Jenis Karya : ~~Tesis/Skripsi~~ **Tugas Akhir** (\*coret salah satu)

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Multimedia Nusantara Hak Bebas Royalti Noneksklusif (*Non-exclusive Royalty-Free Right*) atas karya ilmiah saya yang berjudul.

IMPLEMENTASI ON-DEVICE OFFLINE LEARNING MENGGUNAKAN ALGORITMA LONG-SHORT TERM MEMORY UNTUK NEXT-WORD PREDICTION DALAM BAHASA INDONESIA

Beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini, Universitas Multimedia Nusantara berhak menyimpan, mengalihmediakan/mengalihformatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan memublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta. Demikian pernyataan ini saya buat dengan sebenarnya.

Tangerang, 15 Juni 2022

Yang menyatakan,



(Jovanko Kenshian)

## KATA PENGANTAR

Puji syukur kepada Tuhan Yang Maha Esa yang menyertai penulis dalam melakukan penelitian dan pembuatan laporan skripsi dengan judul “Implementasi On-device Offline Learning Menggunakan Algoritma Long-short Term Memory untuk Next-word Prediction dalam Bahasa Indonesia” sehingga dapat diselesaikan dengan baik dan benar.

Penulis juga mengucapkan terima kasih kepada:

1. Dr. Ninok Leksono, Rektor Universitas Multimedia Nusantara, yang memberi inspirasi bagi penulis untuk berprestasi,
2. Dr. Eng. Niki Prastomo, Dekan Fakultas Teknik dan Informatika Universitas Multimedia Nusantara,
3. Samuel, M.T.I., selaku Ketua Program Studi Teknik Komputer Universitas Multimedia Nusantara.
4. Andre Rusli, S.Kom., M.Sc., sebagai Pembimbing pertama yang telah memberikan bimbingan, arahan, dan motivasi atas terselesainya tugas akhir ini.
5. Daren Kusuma Halim, S.Kom., M.Eng.Sc., sebagai Pembimbing kedua yang telah memberikan bimbingan, arahan, dan motivasi atas terselesainya tugas akhir ini.
6. Keluarga saya yang telah memberikan bantuan dukungan material dan moral, sehingga penulis dapat menyelesaikan tugas akhir ini.
7. Seluruh teman-teman Teknik Komputer 2018 yang terus berjuang bersama penulis selama empat tahun terakhir.

Semoga tugas akhir ini dapat bermanfaat dan berguna bagi para pembaca

Tangerang, 15 Juni 2022



(Jovanko Kenshian)

# IMPLEMENTASI ON-DEVICE OFFLINE LEARNING MENGUNAKAN ALGORITMA LONG-SHORT TERM MEMORY UNTUK NEXT-WORD PREDICTION DALAM BAHASA INDONESIA

Jovanko Kenshian

## ABSTRAK

Agar sebuah sistem *next-word prediction* dapat memberikan prediksi yang sesuai dengan pengguna, model harus dilakukan personalisasi. Akan tetapi, melakukan personalisasi berarti pengguna harus mengorbankan privasi karena beberapa data personal harus dikirimkan ke server. Untuk mengatasi masalah tersebut, penelitian ini membuat sistem *next-word prediction* secara *on-device* dan dilatih dengan menggunakan *offline learning* dengan menggunakan algoritma LSTM dengan tujuan untuk menjaga privasi pengguna. Model LSTM dibuat dengan menggunakan Python yang kemudian dilakukan pruning dan diubah menjadi bentuk tflite. Aplikasi keyboard dirancang dengan bahasa pemrograman Kotlin untuk memuat model yang telah dibuat. Aplikasi tersebut dirancang agar dapat melakukan inferensi dan pelatihan ulang model untuk melakukan personalisasi. Untuk mengetahui dampak apa saja yang terjadi pada model sewaktu model dilakukan pruning, penulis membandingkan akurasi model top-3 pada model yang dilakukan dan tidak dilakukan pruning dengan menggunakan dataset pengetesan. Didapatkan bahwa model yang dilakukan pruning mempunyai akurasi lebih besar dibandingkan dengan model yang tidak dilakukan pruning. Sewaktu model diubah ke bentuk tflite agar inferensi dapat dilakukan pada perangkat telepon genggam, performa kedua model meningkat. Akan tetapi, hal yang sama juga terlihat dimana model yang dilakukan pruning mempunyai akurasi lebih tinggi daripada model yang tidak dilakukan pruning. Penelitian mendapatkan bahwa dalam pengetesan model dengan menggunakan dataset asing, model yang tidak dilakukan pruning mempunyai performa yang lebih baik daripada model yang dilakukan pruning jika model tidak dilakukan pelatihan ulang. Akan tetapi, model yang dilakukan pruning mempunyai performa yang lebih baik daripada model yang tidak dilakukan pruning jika kedua model dilakukan pelatihan ulang. Penggunaan cache juga terlihat dapat meningkatkan nilai top-3 eff secara keseluruhan. Sewaktu diimplementasi pada perangkat telepon genggam, model yang dilakukan pruning berhasil menghemat lebih banyak memori dan baterai, baik dalam melakukan inferensi ataupun pelatihan ulang, daripada model yang tidak dilakukan pruning.

**Kata kunci:** Next-Word Prediction, On-Device Offline Learning, Long Short-Term Memory

**IMPLEMENTATION OF ON-DEVICE OFFLINE LEARNING  
USING LONG-SHORT TERM MEMORY ALGORITHM FOR  
NEXT-WORD PREDICTION IN INDONESIAN**

Jovanko Kenshian

**ABSTRACT (English)**

*For a next-word prediction system to be able to provide relevant prediction for a user, the model must be personalized. However, personalizing means the user has to sacrifice privacy, as some personal data has to be sent to the server. To solve this problem, this research creates an on-device next word prediction system and asks questions using offline learning using the LSTM algorithm with the aim of maintaining user privacy. The LSTM model was created using Python which was then trimmed and converted into tflite format. The application for the keyboard is created by using the programming language Kotlin. The application is used to load the created model, perform inference, and retrain the model for personalization. To find out what impact occurred to the model when the model was pruned, the author decided to compare the top-3 accuracy of the model on the model that was and was not pruned using the test dataset. It was found that the pruned model had greater accuracy than the model that was not pruned. When the model is converted to tflite format so that inference could be done on mobile devices, the performance of both models improves. However, the same thing can also be seen where the pruning model has a higher accuracy than the model that is not pruned. The study found that in testing the model using foreign datasets, the model that was not pruned had better performance than the model that was pruned if the model was not retrained. However, the model that was pruned had a better performance than the model that was not pruned when both models were retrained. The use of cache is also seen to increase the overall top-3 eff value. When implemented on a mobile device, the pruned model managed to save more memory and battery in both inference and retraining than the unpruned model.*

**Keywords:** *Next-Word Prediction, On-Device Offline Learning, Long Short-Term Memory*

U N I V E R S I T A S  
M U L T I M E D I A  
N U S A N T A R A



## DAFTAR ISI

<b>PERNYATAAN TIDAK MELAKUKAN PLAGIAT .....</b>	<b>ii</b>
<b>HALAMAN PENGESAHAN TUGAS AKHIR .....</b>	<b>iii</b>
<b>HALAMAN PERSETUJUAN PUBLIKASI KARYA ILMIAH UNTUK KEPENTINGAN AKADEMIS .....</b>	<b>iv</b>
<b>KATA PENGANTAR.....</b>	<b>v</b>
<b>ABSTRAK .....</b>	<b>vi</b>
<b><i>ABSTRACT (English)</i>.....</b>	<b>vii</b>
<b>DAFTAR ISI.....</b>	<b>viii</b>
<b>DAFTAR TABEL .....</b>	<b>xi</b>
<b>DAFTAR GAMBAR.....</b>	<b>xii</b>
<b>DAFTAR LAMPIRAN .....</b>	<b>xv</b>
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
<b>1.1 Latar Belakang .....</b>	<b>1</b>
<b>1.2 Identifikasi Masalah .....</b>	<b>3</b>
<b>1.3 Batasan Penelitian .....</b>	<b>4</b>
<b>1.4 Tujuan Penelitian .....</b>	<b>4</b>
<b>1.5 Manfaat Penelitian .....</b>	<b>4</b>
<b>BAB II TINJAUAN PUSTAKA.....</b>	<b>4</b>
<b>2.1 Penelitian Terdahulu.....</b>	<b>4</b>
<b>2.1.1 DeepType: On-Device Deep Learning for Input Personalization Service with Minimal Privacy Concern[6] .....</b>	<b>4</b>
<b>2.1.2 Predictive Text System for Bahasa with Frequency, n-gram, Probability Table, and Syntatic using Grammar[7].....</b>	<b>4</b>
<b>2.1.3 A RNN based Approach for next word prediction in Assamese Phonetic Transcription[9] .....</b>	<b>5</b>
<b>2.1.4 Next Words Prediction Using Recurrent Neural Network[10]... </b>	<b>5</b>
<b>2.1.5 A comparison of LSTM and GRU networks for learning symbolic sequences[2].....</b>	<b>6</b>
<b>2.1.6 Kesimpulan Penelitian.....</b>	<b>7</b>
<b>2.2 Tinjauan Teori.....</b>	<b>7</b>

2.2.1	Next-Word Prediction.....	7
2.2.2	Long Short Term Memory .....	8
2.2.3	On-Device Machine Learning .....	10
<b>BAB III</b>	<b>ANALISIS DAN PERANCANGAN SISTEM.....</b>	<b>11</b>
3.1	Metode Penelitian .....	11
3.2	Studi Literatur .....	11
3.3	Perancangan Model LSTM .....	12
3.3.1	Perancangan Model .....	12
3.3.2	Diagram Alir .....	13
3.4	Perancangan Sistem Prediksi Kata Selanjutnya pada Keyboard....	24
3.4.1	Perancangan Sistem .....	24
3.4.2	Diagram Alir .....	27
3.5	Metrik Pengukuran Performa untuk Evaluasi.....	33
<b>BAB IV</b>	<b>IMPLEMENTASI DAN PENGUJIAN SISTEM .....</b>	<b>35</b>
4.1	Implementasi Pembuatan Model LSTM.....	35
4.1.1	Implementasi Pengambilan data Twitter.....	35
4.1.2	Implementasi Pembersihan data Twitter.....	37
4.1.3	Implementasi Feature Engineering .....	40
4.1.4	Implementasi Pelatihan Word2Vec.....	43
4.1.5	Implementasi Pelatihan Model LSTM.....	44
4.1.6	Implementasi Pruning Model.....	47
4.1.7	Implementasi Perubahan Model menjadi bentuk tflite.....	49
4.2	Implementasi Model pada Keyboard Android.....	51
4.2.1	Implementasi Perancangan Layout dan View Keyboard .....	51
4.2.2	Implementasi Inisialisasi Keyboard .....	52
4.2.3	Implementasi Inferensi pada Keyboard .....	52
4.2.4	Implementasi Penyimpanan Dataset Pelatihan.....	55
4.2.5	Implementasi Pelatihan Ulang pada Perangkat.....	56
4.3	Evaluasi Model dan Aplikasi.....	59
4.3.1	Evaluasi Performa Model LSTM Setelah Pelatihan.....	59
4.3.2	Evaluasi Performa Model LSTM Setelah Pruning.....	60

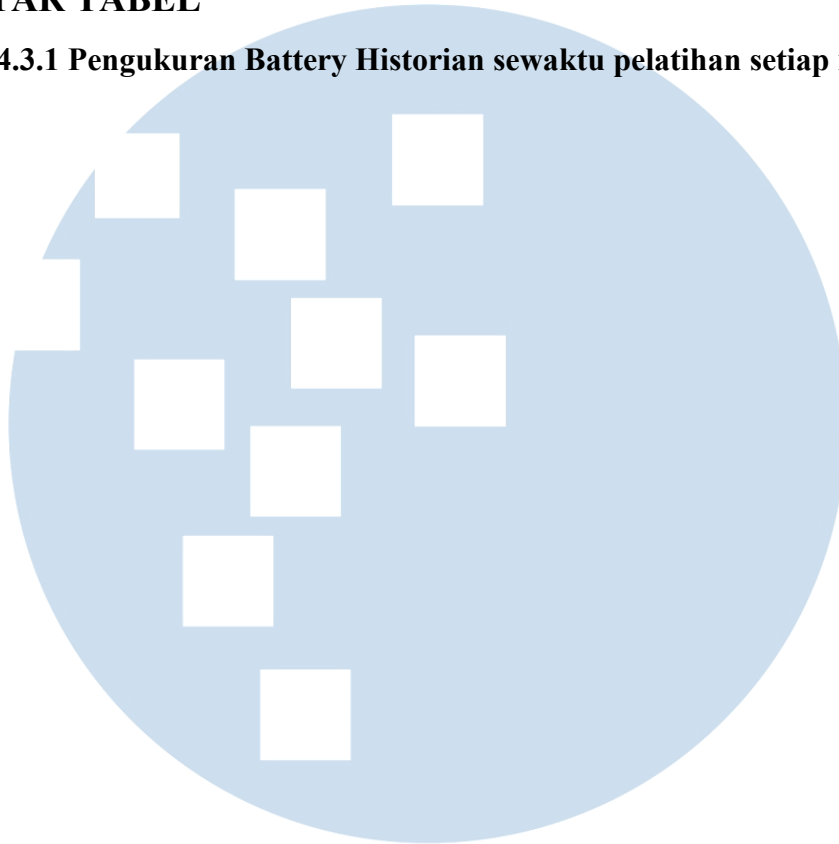
4.3.3	Evaluasi Performa Model tflite dengan Model h5 .....	60
4.3.4	Evaluasi Learning Rate Model tflite .....	63
4.3.5	Evaluasi Performa Model tflite sebelum dan sesudah pruning .....	65
4.3.6	Evaluasi Performa Penerapan Cache pada Model tflite .....	67
4.3.7	Evaluasi View Aplikasi Keyboard .....	69
4.3.8	Evaluasi Performa Inferensi pada Perangkat .....	70
4.3.9	Evaluasi Performa Pelatihan Ulang pada Perangkat .....	73
4.4	Analisis Hasil Evaluasi .....	76
<b>BAB V</b>	<b>SIMPULAN DAN SARAN</b> .....	<b>78</b>
5.1	Simpulan .....	78
5.2	Saran .....	79
<b>DAFTAR PUSTAKA</b>	.....	<b>81</b>
<b>LAMPIRAN</b>	.....	<b>83</b>

UMMN

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

## DAFTAR TABEL

**Tabel 4.3.1 Pengukuran Battery Historian sewaktu pelatihan setiap model 75**



# UMMN

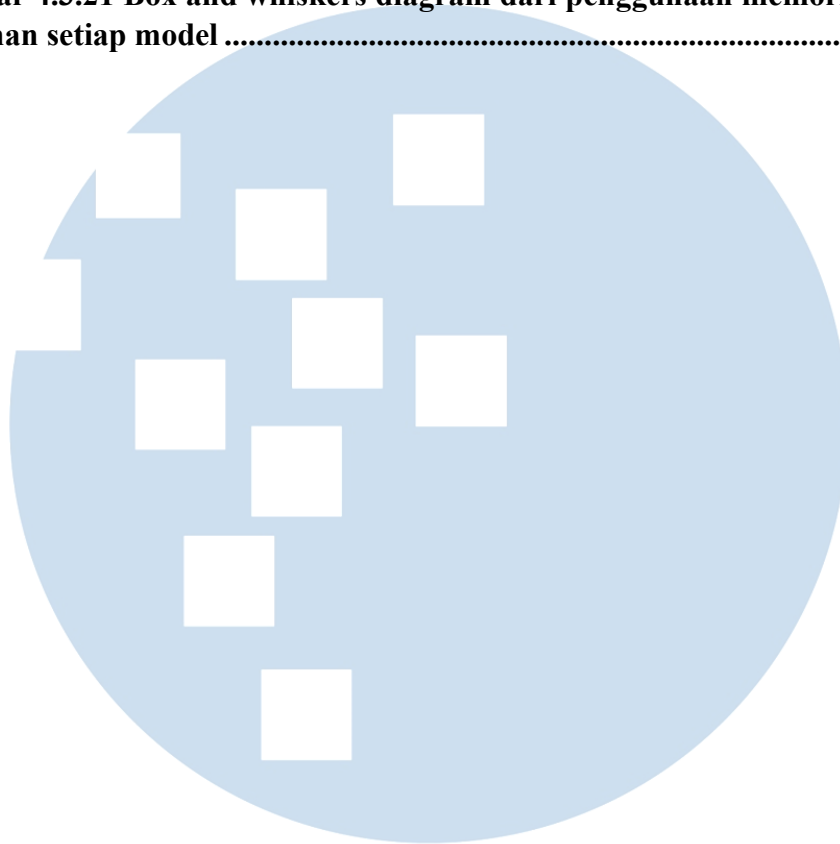
UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

## DAFTAR GAMBAR

Gambar 2.2.1 Struktur LSTM.....	9
Gambar 3.1.1 Diagram Alir sistem secara keseluruhan.....	11
Gambar 3.3.1 Diagram alir perancangan model LSTM .....	13
Gambar 3.3.2 Diagram alir perancangan pengambil data dengan API Twitter .....	14
Gambar 3.3.3 Diagram alir perancangan pembersihan data .....	15
Gambar 3.3.4 Diagram alir perancangan pembersihan kata dengan huruf berulang .....	16
Gambar 3.3.5 Diagram alir perancangan feature engineering.....	17
Gambar 3.3.6 Diagram alir perancangan tokenisasi .....	18
Gambar 3.3.7 Diagram alir perancangan Word2Vec .....	20
Gambar 3.3.8 Diagram alir perancangan pelatihan model .....	21
Gambar 3.3.8 Diagram alir perancangan pruning model.....	22
Gambar 3.3.10 Diagram alir perancangan konversi model ke bentuk tflite. 23	
Gambar 3.4.1 contoh pemberian prediksi sewaktu spasi sudah diketik.....	24
Gambar 3.4.2 contoh sewaktu salah satu kata pada view sugesti dipilih .....	25
Gambar 3.4.3 contoh sewaktu kata pada sugesti dipilih sewaktu di tengah kalimat.....	26
Gambar 3.4.4 contoh sewaktu penulis mengetik kata dengan huruf berulang .....	26
Gambar 3.4.5 Diagram alir sewaktu keyboard dibuka oleh pengguna .....	27
Gambar 3.4.6 Diagram alir sewaktu inferensi dilakukan .....	28
Gambar 3.4.7 Diagram alir penyimpanan dataset.....	30
Gambar 3.4.8 Diagram alir pelatihan model.....	31
Gambar 4.1.1 Rule yang diberikan untuk API Twitter .....	35
Gambar 4.1.2 Contoh data yang diproses untuk mendapatkan tipe bahasa 39	
Gambar 4.1.3 Jumlah data sebelum dan setelah pembersihan .....	39
Gambar 4.1.4 Distribusi dataset yang akan digunakan .....	40
Gambar 4.1.5 Jumlah data pada setiap dataset .....	43
Gambar 4.1.6 Ringkasan model LSTM .....	46
Gambar 4.1.7 Ringkasan model pruning LSTM .....	48

<b>Gambar 4.1.8 List fungsi signature .....</b>	<b>50</b>
<b>Gambar 4.2.1 Layout keyboard.....</b>	<b>51</b>
<b>Gambar 4.3.1 Plot akurasi dan loss model .....</b>	<b>59</b>
<b>Gambar 4.3.2 Plot akurasi dan loss model setelah di-pruning .....</b>	<b>60</b>
<b>Gambar 4.3.3 Bar Chart terhadap akurasi setiap model.....</b>	<b>61</b>
<b>Gambar 4.3.4 Bar Chart terhadap ukuran model.....</b>	<b>62</b>
<b>Gambar 4.3.5 Plot dan line of best fit kata diketahui setiap model learning rate .....</b>	<b>63</b>
<b>Gambar 4.3.6 Plot dan line of best fit kata tidak diketahui setiap model learning rate.....</b>	<b>64</b>
<b>Gambar 4.3.7 Plot dan line of best fit kata diketahui model tflite sebelum &amp; sesudah pruning .....</b>	<b>65</b>
<b>Gambar 4.3.8 Plot dan line of best fit kata tidak diketahui model tflite sebelum &amp; sesudah pruning .....</b>	<b>66</b>
<b>Gambar 4.3.9 Plot dan line of best fit semua kata muncul 1+ model tflite sebelum &amp; sesudah pruning .....</b>	<b>66</b>
<b>Gambar 4.3.10 Plot dan line of best fit kata diketahui model dengan &amp; tanpa cache .....</b>	<b>67</b>
<b>Gambar 4.3.11 Plot dan line of best fit kata tidak diketahui model dengan &amp; tanpa cache .....</b>	<b>68</b>
<b>Gambar 4.3.12 Plot dan line of best fit semua kata muncul 1+ model dengan &amp; tanpa cache .....</b>	<b>68</b>
<b>Gambar 4.3.13 Tampilan sugesti sewaktu diberikan pemisah .....</b>	<b>69</b>
<b>Gambar 4.3.14 Pelengkapan huruf dengan menggunakan model pada perangkat .....</b>	<b>69</b>
<b>Gambar 4.3.15 Bar Chart terhadap waktu inferensi setiap model.....</b>	<b>70</b>
<b>Gambar 4.3.16 Contoh hasil profiler sewaktu inferensi model .....</b>	<b>71</b>
<b>Gambar 4.3.17 Bar Chart penggunaan CPU sewaktu inferensi setiap model .....</b>	<b>72</b>
<b>Gambar 4.3.18 Box and whiskers diagram dari penggunaan memori sewaktu inferensi setiap model .....</b>	<b>72</b>
<b>Gambar 4.3.19 Contoh hasil profiler sewaktu pelatihan model.....</b>	<b>73</b>
<b>Gambar 4.3.20 Bar Chart penggunaan CPU sewaktu pelatihan setiap model .....</b>	<b>74</b>

**Gambar 4.3.21 Box and whiskers diagram dari penggunaan memori sewaktu pelatihan setiap model ..... 75**



UMMN

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

## DAFTAR LAMPIRAN

Lampiran 1 Akses API Twitter pada Google Colab .....	83
Lampiran 2 Pembersihan emoji dari data tweet.....	83
Lampiran 3 Pembersihan emoticon dari data tweet.....	83
Lampiran 4 Pembersihan angka dan tanda baca dari data tweet .....	83
Lampiran 5 Pembersihan variasi kata berbeda dari data tweet .....	84
Lampiran 6 Pembersihan huruf berulang dari data tweet .....	84
Lampiran 7 Pembersihan data yang tidak berbahasa Indonesia.....	84
Lampiran 8 Pembuatan tokenizer .....	85
Lampiran 9 Tokenisasi dataset tweet.....	85
Lampiran 10 Pemrosesan dataset pelatihan.....	85
Lampiran 11 Pemrosesan dataset pengujian.....	86
Lampiran 12 Pembuatan model Word2Vec.....	86
Lampiran 13 Pengambilan weight Word2Vec .....	86
Lampiran 14 Pengambilan class_weight.....	86
Lampiran 15 Inisialisasi model.....	87
Lampiran 16 Fitting model .....	87
Lampiran 17 Inisialisasi pruning model .....	87
Lampiran 18 Fungsi untuk signature train .....	88
Lampiran 19 Fungsi untuk signature infer .....	88
Lampiran 20 Fungsi untuk signature restore.....	88
Lampiran 21 Fungsi untuk signature restore.....	88
Lampiran 22 Menyiapkan model untuk konversi .....	89
Lampiran 23 Konversi model ke tflite .....	89
Lampiran 24 Pemindahan file ke internal storage.....	89
Lampiran 25 Pemindahan file & inisialisasi tokenizer.....	90
Lampiran 26 Pengecekan apakah inferensi akan dilakukan.....	90
Lampiran 27 Pembersihan input dan me-load weight ke model.....	90
Lampiran 28 Memasukkan input ke buffer .....	90
Lampiran 29 Melakukan Inferensi .....	91
Lampiran 30 Membandingkan input dengan isi cache .....	91



Lampiran 31 Memproses input untuk view sugesti .....	91
Lampiran 32 Memproses dan menyimpan data untuk pelatihan model.....	92
Lampiran 33 Counter untuk berapa banyak kalimat sudah diproses.....	92
Lampiran 34 Memproses dataset temporary .....	92
Lampiran 35 Menjalankan pelatihan data pada background.....	92
Lampiran 36 Memproses kata dalam tokenizer yang dapat diganti .....	93
Lampiran 37 Memproses dataset dengan kata yang tidak diketahui .....	93
Lampiran 38 Memproses kelayakan dataset.....	94
Lampiran 39 Penambahan data tambahan ke dataset pelatihan.....	95
Lampiran 40 Pelatihan ulang model .....	96
Lampiran 41 Mengeluarkan notifikasi sewaktu pelatihan selesai .....	96
Lampiran 42 Hasil Turnitin 1 .....	97
Lampiran 43 Hasil Turnitin 2 .....	98
Lampiran 44 Hasil Turnitin 3 .....	99
Lampiran 45 Hasil Turnitin 4.....	100
Lampiran 46 Hasil Turnitin 5.....	101
Lampiran 47 Hasil Turnitin 6.....	102

