

**APLIKASI PENGKLASIFIKASI SEMANTIK TEKS MENGGUNAKAN  
TENSORFLOW LITE PADA RINGKASAN KARYA ILMIAH**

**TUGAS AKHIR**

*Diajukan Sebagai Salah Satu Syarat Untuk Menyelesaikan Program Strata 1 (S1) Pada  
Jurusan Teknik Elektronika Program Studi Pendidikan Teknik Informatika  
Universitas Negeri Padang*



**Oleh :**

**ADAM PERMANA  
NIM : 16076001/2016**

**PROGRAM STUDI PENDIDIKAN TEKNIK INFORMATIKA  
JURUSAN TEKNIK ELEKTRONIKA  
FAKULTAS TEKNIK  
UNIVERSITAS NEGERI PADANG  
2020**

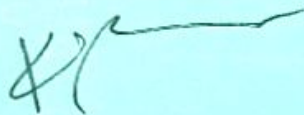
**BALAMAN PERSETUJUAN TUGAS AKHIR**

**APLIKASI PENGKLASIFIKASI SEMANTIK TEKS  
MENGUNAKAN TENSORFLOW LITE  
PADA RINGKASAN KARYA ILMIAH**

Nama : Adam Permana  
NIM / TM : 16076001 / 2016  
Program Studi : Pendidikan Teknik Informatika  
Jurusan : Teknik Elektronika  
Fakultas : Teknik

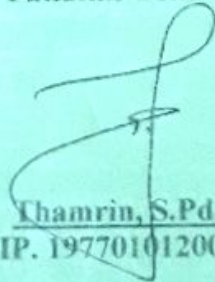
Padang, November 2020

Disetujui Oleh :  
Pembimbing



Khairi Budayawan, S.Pd., M.Kom.  
NIP. 197608102003121002

Mengetahui,  
Ketua Jurusan Teknik Elektronika  
Fakultas Teknik UNP



Chamrin, S.Pd., MT.  
NIP. 197701012008121001

## HALAMAN PENGESAHAN TUGAS AKHIR

*Dinyatakan Lulus Setelah Mempertahankan Tugas Akhir Di Depan Tim Penguji Program Studi Pendidikan Teknik Informatika Jurusan Teknik Elektronika Fakultas Teknik Universitas Negeri Padang*

Judul : Aplikasi Pengklasifikasi Semantik Teks  
Menggunakan Tensorflow Lite Pada  
Ringkasan Karya Ilmiah

Nama : Adam Permana

TM/NIM : 2016 / 16076001

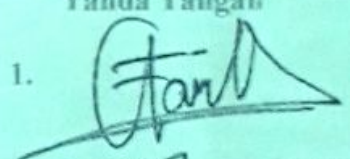
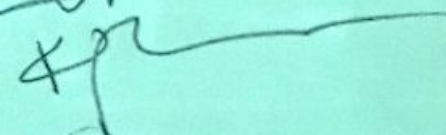
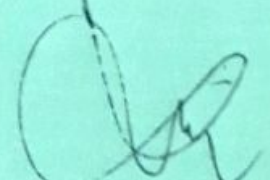
Program Studi : Pendidikan Teknik Informatika

Jurusan : Teknik Elektronika

Fakultas : Teknik

Padang, November 2020

### Tim Penguji

	Nama	Tanda Tangan
1. Ketua	: Geovanne Farell, S.Pd., M.Pd.T.	1. 
2. Anggota	: 1. Khairi Budayawan, S.Pd., M.Kom.	2. 
	2. Drs. Denny Kurniadi, M.Kom.	3. 

## SURAT PERNYATAAN

Dengan ini saya menyatakan tugas akhir saya yang berjudul "**Aplikasi Pengklasifikasi Semantik Teks Menggunakan Tensorflow Lite Pada Ringkasan Karya Tulis Ilmiah**" ini benar-benar karya saya sendiri. Sepanjang Pengetahuan saya tidak terdapat karya atau pendapat yang ditulis atau diterbitkan orang lain kecuali sebagai acuan atau kutipan dengan mengikuti tata penulisan karya tulis ilmiah yang lazim.

Padang, November 2020

Saya yang menyatakan,



**Adam Permana**  
**NIM. 16076001**

## ABSTRAK

**Adam Permana : Aplikasi Pengklasifikasi Semantik Teks Menggunakan Tensorflow Lite Pada Ringkasan Karya Ilmiah**

Perkembangan *neural network* di era industri 4.0 semestinya dapat membantu berbagai bidang pekerjaan, salah satunya literatur karya ilmiah. Permasalahannya adalah karya tulis ilmiah masih menggunakan penyortiran tema/semantik secara manual. Tujuan dari studi ini adalah untuk membangun aplikasi klasifikasi semantik teks yang memungkinkan pengguna melakukan penyortiran berdasarkan tema/semantik menggunakan model *neural network* yang disematkan pada ponsel cerdas. Pengembangan aplikasi menggunakan metode *waterfall* yang didalamnya terdapat analisis dan perancangan sistem. Aplikasi mengimplementasi fitur *text recognition* dari Firebase ML-Kit. Pengembangan model menggunakan metode atau pendekatan umum siklus *machine learning* yang terdiri dari identifikasi data, persiapan data, pemilihan algoritma, pelatihan model, evaluasi model dan deploy model. Model dibangun menggunakan data abstrak karya ilmiah dari Perpustakaan Teknik Elektronika Universitas Negeri Padang. Dari total data didapatkan 84 data latih dan 21 data uji menggunakan perbandingan 80:20 persen untuk uji validasi. Model *neural network* menggunakan spesifikasi AverageWordVec yang disediakan oleh Tensorflow Lite Model Maker dengan 3 output klasifikasi. Uji validasi model mencapai 0.7619 nilai akurasi dengan 0.7782 nilai loss. Model di eksekusi menggunakan *interpreter* Tensorflow Lite yang tertanam pada aplikasi. Hasil aplikasi memenuhi keseluruhan analisis kebutuhan fungsional sistem.

***Kata kunci*** : Tensorflow Lite, ML-Kit, Neural Network, Semantik Teks.

## KATA PENGANTAR



Alhamdulillah puji syukur kehadiran Allah SWT atas berkat rahmat dan karunia-Nya sehingga penyusunan Tugas Akhir dalam rangka memenuhi sebagian persyaratan untuk mendapatkan gelar Sarjana Pendidikan dengan judul “**Aplikasi Pengklasifikasi Semantik Teks Menggunakan Tensorflow Lite Pada Ringkasan Karya Ilmiah**” dapat disusun sesuai dengan harapan. Tugas akhir ini dapat diselesaikan tidak lepas dari bantuan dan kerjasama dengan pihak lain. Berkenaan dengan hal tersebut, penulis menyampaikan ucapan terima kasih kepada yang terhormat:

1. Kepada Kedua Orang Tua yang telah membesarkan, mendidik, memberi perhatian dan dorongan dan do'a yang tiada hentinya. Serta kedua saudara kandung penulis yang tidak henti untuk mengingatkan dan memberi semangat sehingga penulis mampu menyelesaikan jenjang studi Strata 1.
2. Bapak Dr. Fahmi Rizal, M.Pd, M.T, selaku Dekan Fakultas Teknik Universitas Negeri Padang.
3. Bapak Thamrin, S.Pd, M.T, selaku ketua jurusan Teknik Elektronika Fakultas Teknik UNP.

4. Bapak Khairi Budayawan, S.Pd, M.Kom, selaku Ketua Program Studi Pendidikan Teknik Informatika dan Komputer Universitas Negeri Padang yang juga selaku Dosen Pembimbing yang telah membimbing dalam perencanaan, pelaksanaan dan pelaporan tugas akhir ini.
5. Bapak Dr. Elfi Tasrif, M.T selaku dosen Pembimbing Akademik
6. Bapak Drs. Denny Kurniadi, M.Kom. selaku dosen penguji yang telah meluangkan waktu untuk memberi arahan dan membimbing dalam pembuatan Tugas Akhir.
7. Bapak Geovanne Farell, S.Pd, M.Pd.T. selaku penguji yang telah meluangkan waktu untuk memberi arahan dan membimbing dalam pembuatan Tugas Akhir.
8. Rekan-rekan senasib dan seperjuangan yang telah memotivasi dan mendukung penulis dalam menyelesaikan tugas akhir ini

Semoga kebaikan yang diberikan oleh semua pihak kepada penulis menjadi amal shaleh yang senantiasa mendapat balasan dan kebaikan yang berlipat ganda dari Allah SWT. Akhir kata, penulis menyadari bahwa masih terdapat kekurangan dalam penulisan Tugas Akhir ini, untuk itu saran dan kritik yang sifatnya membangun sangat penulis harapkan.

Padang, 25 Agustus 2020

Penulis

## DAFTAR ISI

<b>ABSTRAK .....</b>	<b>i</b>
<b>KATA PENGANTAR .....</b>	<b>ii</b>
<b>DAFTAR ISI .....</b>	<b>iii</b>
<b>DAFTAR GAMBAR .....</b>	<b>x</b>
<b>DAFTAR TABEL .....</b>	<b>xii</b>
<b>DAFTAR LAMPIRAN .....</b>	<b>xiv</b>
<b>BAB I PENDAHULUAN .....</b>	<b>1</b>
A. Latar Belakang Masalah .....	1
B. Identifikasi Masalah .....	5
C. Batasan Masalah .....	6
D. Rumusan Masalah .....	6
E. Tujuan .....	6
F. Manfaat Tugas Akhir .....	7
<b>BAB II LANDASAN TEORI .....</b>	<b>8</b>
A. Ringkasan Karya Ilmiah .....	8
B. Word Embedding & Semantik Teks .....	9
1. Unsupervised Word Embedding.....	11
2. Supervised Word Embedding.....	12
C. Model Machine Learning.....	13
1. Neural Network.....	14
2. Tensorflow & Keras.....	17
D. AverageWordVec .....	18

1. Layer Embedding.....	19
2. Layer Average Pooling 1D.....	20
3. Layer Dense.....	21
4. Layer Drop-Out.....	24
5. Sparse Categorical Cross Entropy.....	25
6. RMSprop.....	25
E. Aplikasi Android .....	26
1. Android SDK.....	26
2. Google Firebase & ML Kit.....	27
3. Tensorflow Lite.....	29
F. Rekayasa Perangkat Lunak .....	31
1. Analisis Sistem.....	31
2. Perancangan Sistem dan Aplikasi.....	33
3. Perangkat Pemodelan Sistem.....	35
4. Perancangan Antarmuka.....	42
5. Perancangan Model Neural Network.....	43
G. Perangkat Pengembangan .....	44
1. Jupyter Notebook & Google Colab.....	44
2. Bahasa Pemrograman Python.....	45
3. IDE Android Studio.....	46
4. Bahasa Pemrograman Java.....	47
5. Bahasa Pemrograman Kotlin.....	60
<b>BAB III ANALISIS PERANCANGAN SISTEM .....</b>	<b>61</b>

A. Analisis Sistem .....	61
1. Analisis Kebutuhan Sistem .....	61
2. Analisis Keamanan Sistem .....	63
3. Analisis Dokumen I/O .....	64
4. Analisis Prosedur .....	64
B. Perancangan Sistem .....	64
1. Use Case Diagram.....	65
2. Activity Diagram.....	66
3. Class Diagram.....	68
4. Object Diagram.....	77
5. Component Diagram.....	77
C. Perancangan Antarmuka .....	78
1. Rancangan Antarmuka Tampilan Depan .....	78
2. Rancangan Antarmuka CameraView.....	79
3. Rancangna Antarmuka Resultview & Mode Data Collection.....	80
D. Perancangan Model Neural Network .....	81
1. Identifikasi Data.....	81
2. Persiapan Data.....	82
3. Pemilihan Algoritma.....	83
<b>BAB IV IMPLEMENTASI DAN PEMBAHASAN .....</b>	<b>84</b>
A. Implementasi dan Pembahasan Antarmuka .....	84
1. Antarmuka Activity Menu Utama .....	84
2. Antarmuka Activity Fitur Prediksi .....	87

3. Antarmuka Activity Fitur Pengumpulan Data .....	89
4. Antarmuka Activity About/Tentang Aplikasi.....	92
5. Antarmuka Activity Help/Bantuan.....	94
<b>B. Implementasi dan Pembahasan Sistem .....</b>	<b>95</b>
1. Spesifikasi Proses Activity Prediksi .....	95
2. Spesifikasi Proses Activity Pengumpulan Data .....	98
<b>C. Hasil Pembahasan Model Neural Network.....</b>	<b>98</b>
1. Training / Pelatihan Data .....	98
2. Evaluasi Model .....	99
3. Deploy Model .....	100
<b>BAB V PENUTUP .....</b>	<b>101</b>
A. Kesimpulan .....	101
B. Saran .....	101
<b>DAFTAR PUSTAKA .....</b>	<b>102</b>
<b>LAMPIRAN .....</b>	<b>104</b>

## DAFTAR GAMBAR

Gambar 1. Ilustrasi One Hot Encoding .....	10
Gambar 2. Ilustrasi Embedding Weight Matrix .....	10
Gambar 3. Word2vec dengan Skip-gram.....	11
Gambar 4. CBOW dan Skip-gram .....	12
Gambar 5. Weight Embedding Matrix berupa vocab file .....	13
Gambar 6. Istilah “model” dalam <i>machine learning</i> .....	14
Gambar 7. Konsep jaringan syaraf tiruan / <i>neural network</i> .....	15
Gambar 8. Ragam <i>loss function</i> .....	16
Gambar 9. Keras & Tensorflow .....	17
Gambar 10. Contoh <i>embedding space</i> pada kasus klasifikasi buku .....	20
Gambar 11. Ilustrasi proses average pooling 1 dimensi .....	21
Gambar 12. Ilustrasi 2 buah dense layer .....	22
Gambar 13. Fungsi aktivasi Rectified Linear Unit .....	23
Gambar 14. Fungsi aktivasi Softmax .....	23
Gambar 15. Ilustrasi penerapan teknik Drop-Out .....	24
Gambar 16. Ilustrasi penerapan Categorical Crossentropy .....	25
Gambar 17. Rumus RMSprop dalam mengatur <i>weight</i> .....	26
Gambar 18. Proses konversi model menggunakan TFLite Converter .....	30
Gambar 19. Contoh <i>class diagram</i> .....	38
Gambar 20. Contoh <i>object diagram</i> .....	39
Gambar 21. Contoh <i>use case diagram</i> .....	40
Gambar 22. Contoh <i>activity diagram</i> .....	42
Gambar 23. Rancangan <i>use case diagram</i> .....	65
Gambar 24. Rancangan <i>activity diagram</i> untuk MainActivity .....	67
Gambar 25. Rancangan <i>activity diagram</i> DataCollectingActivity .....	68

Gambar 26. Rancangan <i>class diagram</i> .....	69
Gambar 27. Flowchart mulaiListener .....	70
Gambar 28. Flowchart decodeBitmap .....	71
Gambar 29. Flowchart method tampilkanHasil .....	72
Gambar 30. Flowchart method Classify .....	74
Gambar 31. Flowchart method tokenizeInputText .....	75
Gambar 32. Rancangan <i>object diagram</i> .....	77
Gambar 33. Rancangan <i>component diagram</i> .....	77
Gambar 34. Rancangan homescreen .....	78
Gambar 35. Rancangan CameraView .....	79
Gambar 36. ResultView & data collection .....	80
Gambar 37. Identifikasi data mentah .....	81
Gambar 38. Persiapan data ( <i>data preprocessing</i> ) .....	82
Gambar 39. Ilustrasi spesifikasi AverageWordVec .....	84
Gambar 40. Antarmuka Menu Utama.....	84
Gambar 41. Tampilan fitur prediksi .....	87
Gambar 42. Tampilan fitur data collecting .....	90
Gambar 43. Tentang aplikasi .....	92
Gambar 44. Tampilan halaman help/bantuan .....	94
Gambar 45. Nilai loss dan akurasi di epoch ke-100 .....	99
Gambar 46. Hasil evaluasi model .....	99
Gambar 47. Deploy model .....	100

## DAFTAR TABEL

Tabel 1. Tanggapan kuisisioner kinerja perpustakaan.....	4
Tabel 2. Spesifikasi model sekuensial AverageWordVec .....	18
Tabel 3. Analisis kebutuhan fungsional .....	61
Tabel 4. Analisis kebutuhan perangkat .....	62
Tabel 5. Analisis keamanan sistem .....	63
Tabel 6. Informasi file persiapan data .....	73

## **DAFTAR LAMPIRAN**

Lampiran 1. Tabel pengujian fungsional

Lampiran 2. Kode sumber (Source Code)

## **DAFTAR SOURCE CODE**

Source code 1. Menampilkan menu utama .....	77
Source code 2. Menampilkan fitur prediksi .....	80
Source code 3. Menampilkan fitur data collecting .....	82
Source code 4. Menampilkan halaman about .....	85
Source code 5. Menampilkan halaman help .....	86

# BAB I PENDAHULUAN

## A. Latar Belakang

Revolusi industri yang akan datang didasarkan pada teknologi revolusioner yang memiliki dampak kuat pada *input* dan *output* produksi industri namun juga seperti percetakan 3D, rekayasa genetika, dan khususnya kecerdasan buatan (Garrett, 2013). Salah satu alasan kecerdasan buatan menjadi berperan pada revolusi industri 4.0 dikarenakan perkembangan infrastruktur yang memadai ditambah dorongan kebutuhan akibat semakin besarnya data yang dikumpulkan oleh industri layanan internet. Data yang terkumpul dengan jumlah besar ini menjadi sulit untuk ditangani oleh metode *database* tradisional sehingga memunculkan istilah *Big Data*. Dalam perjalanannya ilmuwan terus melakukan riset bagaimana menangani *big data* dengan metode-metode statistik yang menjadi cikal bakal *Machine Learning*.

*Machine Learning* adalah cabang dari algoritma komputasi yang dirancang untuk meniru kecerdasan manusia dengan belajar dari lingkungan sekitarnya (El Naqa dan Murphy, 2015 : 3). Ada berbagai macam cabang algoritma dari *machine learning* mulai dari yang paling sederhana seperti *Regression* hingga yang canggih seperti *Deep Neural Network*. JST (Jaringan Saraf Tiruan) atau *Neural Network* menjadi algoritma paling canggih karena terinspirasi dan didasarkan seperti struktur otak biologis. Ini karena JST memungkinkan untuk didesain dengan lapisan (*layer*) jamak sehingga menjadi berbagai macam bentuk arsitektur, JST berlapis ini menjadi fondasi untuk sub-

cabang algoritma komputasi *Machine Learning* mutakhir yang disebut dengan *Deep Learning*.

Pada 9 November 2015, Perusahaan Google mengembangkan *library open-source* yang diberi nama Tensorflow dibawah lisensi Apache 2.0 yang hanya dipakai pada *internal* Google. Tensorflow adalah *library* untuk *dataflow* matematika simbolik yang populer dengan kasus-kasus *machine learning* terutama *neural network* bahkan *deep neural network*. Kemudian pada bulan Februari 2017 versi pertama (1.0.0) Tensorflow dirilis, yang kemudian di bulan Mei 2017, Google merilis *software stack* khusus untuk pengembangan perangkat lunak seluler yang diberi nama Tensorflow Lite. Tensorflow Lite adalah versi ringan dari Tensorflow yang mampu dieksekusi lebih cepat pada perangkat seluler bahkan perangkat *embedded* seperti mikrokontroler (Alsing, 2018 : 31). *Library* Tensorflow juga menyediakan *model maker* khusus untuk Tensorflow Lite yang di beri nama TFLite Model Maker yang bertujuan untuk menyederhanakan proses adaptasi dan mengubah model *neural network* TensorFlow ke *input data* tertentu.

Pada contoh kode yang disediakan TFLite Model Maker, terdapat salah satu algoritma studi NLP (*Natural Language Processing*) yakni *word embedding*. *Word embedding* memungkinkan data masukkan diklasifikasi berdasarkan konteksnya sehingga mampu memisahkan *dataset* berdasarkan semantiknya. Studi *word embedding* sering dipakai pada proses *Semantic Analysis*. Ini memungkinkan pengembang perangkat lunak membangun model yang mampu mengklasifikasi teks berdasarkan semantik atau makna-nya.

Mendapatkan makna dari sebuah teks merupakan kemampuan canggih yang hanya dimiliki otak manusia dari sekian banyak makhluk bumi. Neuron pada otak manusia mampu menafsirkan kalimat bahkan paragraf baik teks dan konteks. Namun otak manusia memiliki keterbatasan dalam daya tahan. Selain itu, memaknai sebuah teks juga bukanlah pekerjaan yang mudah. Dibutuhkan analisis dan pengetahuan dasar sehingga menghabiskan cukup energi terlebih bila dilakukan secara intens dalam waktu yang lama. Masalah ini berpotensi dipermudah oleh keberadaan Tensorflow sebagai *library* pendukung pengembangan kecerdasan buatan.

Seiring dengan pengembangan Tensorflow Lite, pada platform Firebase tim dari Google juga merilis SDK (*Software Development Kit*) khusus *machine learning* yang bisa langsung digunakan pada sistem operasi Android (ponsel dan tablet) dengan nama ML Kit. Adanya ML Kit memungkinkan pengembang perangkat lunak Android membangun fitur-fitur *machine learning* yang diantaranya adalah *Text Recognition*.

Maka dari itu, kedua fitur diatas yakni Tensorflow Lite dan ML Kit (khususnya *Text Recognition*) berpotensi dikembangkan menjadi alat untuk mengklasifikasi teks apapun sesuai kebutuhan. Hal ini tentunya akan mempermudah pekerjaan manusia terutama di bidang administrasi yang berhubungan dengan hardcopy. Beberapa pekerjaan yang berpotensi untuk dipermudah adalah di bidang pendidikan, administrasi perkantoran, psikologi dan perpustakaan. Contoh di bidang pendidikan adalah pada kasus guru yang mengevaluasi nilai dari ujian esai yang banyak. Pada bidang administrasi

perkantoran yakni ketika staf harus membaca satu per satu surat yang bertumpuk untuk di kategorikan. Pada bidang psikologi menurut Pennebaker (1999 : 1296) memungkinkan psikolog mengeksplorasi kepribadian melalui gaya linguistik tulisan klien. Pada bidang arsip dan perpustakaan yakni ketika pustakawan bingung memposisikan buku di rak yang sesuai dengan kategori seharusnya. Bidang perpustakaan dan kearsipan menjadi perhatian khusus pada penelitian ini, oleh karena itu dilakukan survey untuk mengetahui apakah pekerjaan di bidang tersebut perlu ditingkatkan dengan *neural network*.

Pada *survey* yang di lakukan terhadap mahasiswa umum yang terdiri dari 27 responden (16 Wanita dan 11 Pria) didapati bahwa pengunjung perpustakaan cenderung sering menemukan buku dan karya tulis ilmiah yang tidak sesuai dengan kategori atau rak-nya. Detail survey dapat dilihat pada tabel berikut:

Tabel 1. Tanggapan kuisisioner kinerja perpustakaan

No	Pertanyaan	Tanggapan
1	Seberapa sering anda ke perpustakaan (sebelum pandemi) ?	<ul style="list-style-type: none"> <li>• Jarang (22.2%)</li> <li>• Cukup Jarang (22.2%)</li> <li>• Cukup Sering (33.3%)</li> <li>• Sering (22.2%)</li> </ul>
2	Apakah anda kesulitan mencari artikel ilmiah yang sesuai dengan tema yang anda inginkan di perpustakaan?	<ul style="list-style-type: none"> <li>• Ya (63%)</li> <li>• Tidak (29.6%)</li> <li>• Mungkin (7.4%)</li> </ul>
3	Apakah pencarian dengan “kata kunci/keyword” cukup membantu anda mencari tema yang sesuai?	<ul style="list-style-type: none"> <li>• Cukup Membantu (55.6%)</li> <li>• Membantu (37%)</li> <li>• Kurang Membantu (3.7%)</li> </ul>
4	Seberapa sering anda melakukan buku dan karya tulis ilmiah yang tak sesuai dengan kategori / rak-nya	<ul style="list-style-type: none"> <li>• C1/Jarang (11.4%)</li> <li>• C2/Cukup Sering (50%)</li> <li>• C3/Sering (38.4%)</li> </ul>

Sebelum mengimplementasikan aplikasi ke bidang perpustakaan dan kearsipan atau bidang lainnya, tentu perlu di desain prototipe awal dan pengujian agar peneliti selanjutnya mendapat gambaran dalam mengembangkan prototipe ini. Prototipe aplikasi didesain secara modular agar aplikasi tidak hanya dipakai pada satu bidang, namun dapat dipakai pada bidang-bidang lainnya. Sehingga pengembangan selanjutnya hanya perlu mengubah model neural network dengan sedikit atau bahkan tanpa mengubah desain aplikasi.

Maka dari itu perlu dilakukan pengembangan dengan mendesain dan menguji prototipe aplikasi tersebut pada ringkasan karya ilmiah di perpustakaan jurusan elektronika Universitas Negeri Padang. Berdasarkan penjelasan tersebut diangkatlah judul "*Aplikasi Pengklasifikasi Semantik Teks Menggunakan Tensorflow Lite Pada Ringkasan Karya Ilmiah*".

## **B. Identifikasi Masalah**

Berdasarkan latar belakang masalah yang telah dikemukakan diatas, maka dapat diidentifikasi beberapa masalah dalam penelitian diantaranya adalah :

1. Otak manusia memiliki durabilitas untuk menjadi aus sehingga membutuhkan alat bantu.
2. Diperlukan kemampuan menganalisis untuk memaknai sebuah teks.
3. Pengunjung perpustakaan sering menemukan buku dan karya tulis ilmiah tidak sesuai dengan kategori/rak nya.

### C. Batasan Masalah

Berdasarkan identifikasi masalah, maka masalah dalam penelitian ini dibatasi sebagai berikut:

1. Fokus penelitian hanya pada bidang perpustakaan dan kearsipan.
2. Aplikasi ini hanya dapat berjalan di perangkat dan *emulator* berbasis Android (Ponsel dan Tablet).
3. Model *neural network* pada aplikasi ini dibangun menggunakan data *training* dari hasil *scan* ringkasan/abstrak skripsi dimana *class* yang ditentukan adalah ‘Multimedia’, ‘Komputer Jaringan’ dan ‘Rekayasa Perangkat Lunak’.
4. Algoritma yang dipakai untuk membangun model *neural network* prototipe ini adalah *supervised word embedding* dengan class `AverageWordVecModelSpec` atau dikenal `AverageWordVec` pada *library* `Tensorflow Lite Model Maker`.

### D. Rumusan Masalah

Berdasarkan identifikasi dan batasan masalah tersebut, masalah yang dirumuskan adalah **“Bagaimana mengembangkan aplikasi pengklasifikasi semantik teks menggunakan tensorflow lite dengan data ringkasan karya ilmiah?”**

### E. Tujuan

Adapun tujuan yang ingin dicapai pada tugas akhir ini adalah:

1. Merancang dan mengujicoba kombinasi teknologi Firebase ML Kit dengan Tensorflow Lite dengan model *neural network* yang didesain khusus.

2. Menghasilkan prototipe aplikasi pengklasifikasi jenis teks berbasis android.

#### **F. Manfaat Tugas Akhir**

Manfaat dari pembuatan aplikasi pengklasifikasi semantik teks ini adalah sebagai berikut :

1. Manfaat dalam arsip dan perpustakaan jurusan elektronika Universitas Negeri Padang, dapat dijadikan alat bantu pengkategorian jenis karya ilmiah.
2. Bagi pengembang perangkat lunak, sebagai gambaran dasar untuk menyusun proposal litbang produk sejenis di industri.
3. Bagi pelajar dan peneliti, sebagai rujukan untuk penelitian serupa atau penelitian lanjutan mengenai studi *natural language processing* yakni *word embedding*, *machine learning* dan *neural network*.
4. Bagi penulis, untuk menerapkan pengetahuan yang diperoleh selama diperkuliahan, Serta sebagai tugas akhir untuk syarat kelulusan Strata 1.

## **BAB V**

### **PENUTUP**

#### **A. Kesimpulan**

Perancangan dan pembuatan Aplikasi Pengklasifikasi Semantik Teks Menggunakan Tensorflow Lite Pada Ringkasan Karya Ilmiah ini menggunakan metode pengembangan perangkat lunak *waterfall*, maka didapatkan kesimpulan sebagai berikut :

1. Teknologi Firebase ML Kit dengan Tensorflow Lite dapat diandalkan dalam membangun aplikasi pengklasifikasi semantik teks. Namun model *neural network* yang didesain khusus memiliki kekurangan yakni data latih yang terlalu sedikit. Sehingga nilai akurasi belum optimal.
2. Hasil pengujian prototipe sesuai dengan analisis sistem secara keseluruhan berhasil direalisasikan dari rancangan.

#### **B. Saran**

Adapun saran-saran dalam merancang aplikasi pengklasifikasi semantik teks kedepan-nya adala. Karena data yang dibutuhkan untuk membangun model deep neural network sangat banyak, disarankan untuk menggunakan data abstrak berupa softcopy dari database jurusan. Untuk pengembangan model berikutnya, ada baiknya membangun spesifikasi model baru dengan fitur drop-out lebih kecil, dengan input sequence lebih besar dari 256 input. Mengingat beberapa abstrak/ringkasan karya tulis ilmiah memiliki maksimal kata yakni 300-500 kata.



## DAFTAR PUSTAKA

- Alsing, O. (2018). Mobile Object Detection using TensorFlow Lite and Transfer Learning.
- Candanedo, I. S., Nieves, E. H., González, S. R., Martín, M. T. S., & Briones, A. G. (2018, August). Machine learning predictive model for industry 4.0. In *International Conference on Knowledge Management in Organizations* (pp. 501-510). Springer, Cham.
- Cross, S. S., Harrison, R. F., & Kennedy, R. L. (1995). Introduction to neural networks. *The Lancet*, 346(8982), 1075-1079.
- El Naqa, I., & Murphy, M. J. (2015). What is machine learning?. In *machine learning in radiation oncology* (pp. 3-11). Springer, Cham.
- Garrett, B. (2013). *An Emerging Third Industrial Revolution*. Atlantic Council, December, 4.
- Gary Blake and Robert W. Bly, *The Elements of Technical Writing*, hal. 117. New York: Macmillan Publishers, 1993. ISBN 0020130856
- Hansun, S., Kristanda, M. B., & Saputra, M. W. (2018). *Pemrograman android dengan android studio IDE*. Yogyakarta: Andi.
- Hurwitz, J., & Kirsch, D. (2018). *Machine learning for dummies*. IBM Limited Edition, 75.
- Kadir, Abdul, (2005), "Dasar Pemrograman Python", Yogyakarta, Andi Offset.
- Ketkar, N. (2017). Introduction to keras. In *Deep learning with Python* (pp. 97-111). Apress, Berkeley, CA.
- Lai, S., Liu, K., He, S., & Zhao, J. (2016). How to generate a good word embedding. *IEEE Intelligent Systems*, 31(6), 5-14.
- Nugroho, Adi. 2009. *Rekayasa Perangkat Lunak Menggunakan UML dan Java*. Yogyakarta : Andi.
- Pennebaker, J. W., & King, L. A. (1999). Linguistic styles: Language use as an individual difference. *Journal of personality and social psychology*, 77(6), 1296.
- Pradesh, A. (2020). Language Identification Using ML Kit. *International Journal of Engineering Technology Research & Management*.