

**IMPLEMENTASI ARSITEKTUR *TRANSFORMER* UNTUK
PENERJEMAHAN BAHASA LAMPUNG - INDONESIA**

(Skripsi)

Oleh

**IRFAN YADI
NPM 1917051032**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2024**

**IMPLEMENTASI ARSITEKTUR *TRANSFORMER* UNTUK
PENERJEMAHAN BAHASA LAMPUNG - INDONESIA**

Oleh

IRFAN YADI

Skripsi

**Sebagai Salah Satu Syarat untuk Mencapai Gelar
SARJANA KOMPUTER**

Pada

**Jurusan Ilmu Komputer
Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2024**

ABSTRAK

IMPLEMENTASI ARSITEKTUR *TRANSFORMER* UNTUK PENERJEMAHAN BAHASA LAMPUNG – INDONESIA

Oleh

IRFAN YADI

Indonesia memiliki beraneka ragam bahasa daerah, salah satunya adalah bahasa Lampung. Bahasa Lampung memiliki dua macam dialek, yaitu dialek Api atau A dan dialek Nyo atau O. Bahasa Lampung kini termasuk ke dalam bahasa daerah yang terancam punah dalam berapa puluh tahun mendatang. Oleh sebab itu, untuk mencegah terjadinya kepunahan pada bahasa Lampung, perlu dilakukan upaya guna menjaga serta melestarikan bahasa Lampung. Upaya yang dapat dilakukan, salah satunya adalah dengan memanfaatkan mesin penerjemah untuk memudahkan dalam proses penerjemahan bahasa. Mesin penerjemah modern masa kini banyak menggunakan arsitektur *transformer*. Arsitektur *Transformer* adalah sebuah model terancang dalam pemrosesan bahasa alami yang memanfaatkan mekanisme *attention* dalam proses pengolahan data. Penelitian ini akan mengimplementasikan arsitektur *transformer* dalam penerjemahan bahasa Lampung dialek A ke bahasa Indonesia serta mengetahui kualitas terjemahan dengan metrik *Bilingual Evaluation Understudy* (BLEU). Penelitian ini menggunakan 11.933 data pasangan kata dan kalimat dalam bahasa Lampung dialek A – Indonesia yang akan dibagi menjadi tiga bagian dengan perbandingan 70%:15%:15% untuk masing-masing data *training*, data *validation*, dan data *testing*. Penelitian ini melalui beberapa tahapan, yaitu *dataset*, normalisasi teks, tokenisasi teks, vektorisasi teks, *embedding*, *positional encoding*, *encoding*, *decoding*, dan evaluasi. Hasil pengujian dari penelitian ini memperoleh skor BLEU untuk masing-masing n-gram, yaitu BLEU 1-gram sebesar 91,66%, BLEU 2-gram sebesar 81,81%, BLEU 3-gram sebesar 70%, dan BLEU 4-gram sebesar 55,55%.

Kata kunci: Arsitektur *Transformer*, Mesin Penerjemah, Bahasa Lampung, Pemrosesan Bahasa Alami.

ABSTRACT

IMPLEMENTATION OF TRANSFORMER ARCHITECTURE FOR LAMPUNG – INDONESIAN LANGUAGE TRANSLATION

By

IRFAN YADI

Indonesia has a variety of regional languages, one of which is Lampung. Lampung has two kinds of dialects, namely the Api dialect or A and the Nyo dialect or O. Lampung is now included in the regional languages that are threatened with extinction in the next few decades. Therefore, to prevent the extinction of Lampung language, efforts need to be made to maintain and preserve Lampung language. One of the efforts that can be made is by utilizing machine translators to facilitate the language translation process. Today's modern machine translators use transformer architecture. Transformer architecture is a state-of-the-art model in natural language processing that utilizes attention mechanism in data processing. This research will implement the transformer architecture in the translation of Lampung dialect A into Indonesian and determine the quality of translation with the Bilingual Evaluation Understudy (BLEU) metric. This research uses 11,933 data pairs of words and sentences in Lampung language dialect A - Indonesia which will be divided into three parts with a ratio of 70%:15%:15% for each training data, validation data, and testing data. This research goes through several stages, namely dataset, text normalization, text tokenization, text vectorization, embedding, positional encoding, encoding, decoding, and evaluation. The test results of this research obtained BLEU scores for each n-gram, namely BLEU 1-gram of 91.66%, BLEU 2-gram of 81.81%, BLEU 3-gram of 70%, and BLEU 4-gram of 55.55%.

Keywords: Transformer Architecture, Machine Translation, Lampung Language, Natural Language Processing.

Judul Skripsi : **IMPLEMENTASI ARSITEKTUR
TRANSFORMER UNTUK PENERJEMAHAN
BAHASA LAMPUNG – INDONESIA**

Nama Mahasiswa : **Irfan Yadi**

Nomor Pokok Mahasiswa : 1917051032

Program Studi : S1 Ilmu Komputer

Fakultas : Matematika dan Ilmu Pengetahuan Alam



2. Ketua Jurusan Ilmu Komputer

[Signature]

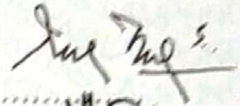
Dwi Sakethi, S.Si. M.Kom.

NIP. 19680611 199802 1 001

MENGESAHKAN

1. Tim Penguji

Ketua : **Dr. Ir. Kurnia Muludi, M.S.Sc.**



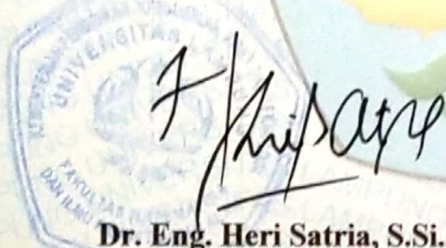
Penguji Pembahas : **Dr. rer. nat. Akmal Junaidi, M.Sc**



Penguji Pembahas : **Favorisen R. Lumbanraja, Ph.D.**



2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam



Dr. Eng. Heri Satria, S.Si., M.Si.

NIP. 19711001 200501 1 002

Tanggal Lulus Ujian Skripsi: **7 Agustus 2024**

PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Irfan Yadi

NPM : 1917051072

Dengan ini menyatakan bahwa skripsi saya yang berjudul “Implementasi Arsitektur *Transformer* Untuk Penerjemahan Bahasa Lampung - Indonesia” merupakan karya saya sendiri, bukan karya orang lain. Semua tulisan yang tertulis dalam skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Jika di kemudian hari terbukti bahwa karya tulis ilmiah saya terbukti hasil menjiplak karya orang lain, maka saya siap menerima sanksi berupa pencabutan gelar yang saya peroleh.

Bandar Lampung, 3 September 2024

Penulis,



Irfan Yadi

NPM. 1917051032

RIWAYAT HIDUP



Penulis dilahirkan di Bandar Lampung, pada tanggal 17 Oktober 2001, sebagai anak kedua dari dua bersaudara dari pasangan Bapak Hendri Yadi, S.E. dan Ibu Eka Yuliantina, S.E., M.M.. Penulis menyelesaikan pendidikan Sekolah Dasar (SD) di SDN 2 Jatimulyo dan selesai pada tahun 2014. Kemudian penulis melanjutkan pendidikan Sekolah Menengah Pertama (SMP) di MTSN 2 Bandar Lampung yang diselesaikan pada tahun 2017, lalu melanjutkan ke pendidikan Sekolah Menengah Atas (SMA) di SMAN 9 Bandar Lampung yang diselesaikan pada tahun 2019.

Pada tahun 2019 penulis terdaftar sebagai mahasiswa Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung melalui jalur Seleksi Bersama Masuk Perguruan Tinggi Negeri (SBMPTN). Selama menjadi mahasiswa, penulis melakukan beberapa kegiatan antara lain.

1. Mengikuti kegiatan Latihan Keterampilan Manajemen Mahasiswa Jurusan Ilmu Komputer Tingkat Dasar (LKMMIK-TD) yang diselenggarakan oleh Himpunan Mahasiswa Jurusan Ilmu Komputer Universitas Lampung pada tahun 2019.
2. Mengikuti kegiatan Latihan Keterampilan Manajemen Mahasiswa Tingkat Menengah (LKMM-TM) yang diselenggarakan oleh Badan Eksekutif Mahasiswa Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung pada tahun 2021.
3. Melaksanakan Kerja Praktik (KP) periode 2021/2022 di Badan Pengelolaan Keuangan dan Aset Daerah Provinsi Lampung.
4. Mengikuti kursus Aplikasi Android dengan Flutter program Kredensial Mikro Mahasiswa Indonesia (KMMI) pada tahun 2021.

5. Mengikuti program Dicoding *Academy* pada kelas Back-End pada tahun 2022.
6. Mengikuti program Merdeka Belajar Kampus Merdeka (MBKM) Studi Independen Bersertifikat pada kelas *Backend Engineering* di PT. Yayasan Anak Bangsa Bisa pada tahun 2022.
7. Melaksanakan Kuliah Kerja Nyata (KKN) periode II tahun 2022 di Desa Sinar Semendo, Kecamatan Talang Padang, Kabupaten Tanggamus, Provinsi Lampung.

MOTTO

“Sesungguhnya bersama kesulitan ada kemudahan.”

(Q.S. Al-Insyirah : 6)

“Janganlah engkau bersedih, sesungguhnya Allah bersama kita.”

(Q.S. At-Taubah : 40)

*“There is only one way to happiness and that is to cease worrying about things
which are beyond the power of our will.”*

(Epictetus)

PERSEMBAHAN

Alhamdulillahirobbilalamin

Puji syukur tercurahkan kepada Allah SWT atas segala Rahmat dan Karunia-Nya sehingga penulis dapat menyelesaikan skripsi ini. Shalawat serta salam selalu tercurahkan kepada baginda Nabi Muhammad SAW.

Aku persembahkan karya ini kepada:

Kedua Orang Tuaku Tercinta

Yang senantiasa mendoakan, menasihati, memotivasi, mendukung, serta mendidik menjadi orang yang berguna dan bermanfaat untuk orang-orang disekitar.

Dosen Pembimbing dan Pembahas

Yang senantiasa membimbing, mengarahkan, serta memberi masukan sejak awal penulisan hingga terselesaikan skripsi ini.

Seluruh Keluarga Besar Ilmu Komputer 2019

Yang selalu memberikan semangat, dukungan, dan doa.

Almamater Tercinta, Universitas Lampung dan Jurusan Ilmu Komputer

Tempat belajar serta mengemban seluruh ilmu yang bermanfaat untuk menjadi bekal hidup di masa mendatang.

SANWACANA

Puji syukur kehadiran Allah SWT atas rahmat dan hidayah-Nya, shalawat serta salam selalu tercurahkan kepada baginda Rasulullah Nabi Muhammad SAW, sehingga penulis dapat menyelesaikan skripsi yang berjudul “Implementasi Arsitektur *Transformer* Untuk Penerjemahan Bahasa Lampung - Indonesia” dengan baik.

Terima kasih penulis ucapkan kepada semua pihak yang telah mendoakan, membantu, mendukung dan berperan besar dalam proses penyusunan skripsi ini hingga selesai, antara lain:

1. Ibu Prof. Dr. Ir. Lusmeilia Afriani, D.E.A.IPM, ASEAN Eng., selaku Rektor Universitas Lampung.
2. Bapak Dr. Eng. Heri Satria, S.Si., M.Si., selaku Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.
3. Bapak Dwi Sakethi, S.Si., M.Kom., selaku Ketua Jurusan Ilmu Komputer Universitas Lampung.
4. Ibu Anie Rose Irawati, S.T., M.Cs., selaku Sekretaris Jurusan Ilmu Komputer Universitas Lampung.
5. Bapak Dr. Ir. Kurnia Muludi, M.S.Sc. selaku Dosen Pembimbing Skripsi yang telah memberikan ide, arahan, serta masukan kepada Penulis sehingga dapat menyelesaikan skripsi ini dengan baik.
6. Bapak Dr. rer. nat. Akmal Junaidi, M.Sc., selaku Dosen Pembahas I yang telah memberikan saran dan masukan yang bermanfaat dalam proses perbaikan skripsi ini.
7. Bapak Favorisen R. Lumbanraja, Ph.D. selaku Dosen Pembahas II yang telah memberikan saran dan masukan yang bermanfaat dalam proses perbaikan skripsi ini.

8. Ibu Nurul Fitrianisa, S.Pd., selaku validator data pada penelitian ini yang telah membantu dalam memeriksa dan memvalidasi data penelitian yang digunakan oleh penulis.
9. Bapak Bambang Hermanto, S.Kom., M.Cs., selaku Dosen Pembimbing Akademik yang telah membantu dalam hal akademik selama proses perkuliahan di Jurusan Ilmu Komputer Universitas Lampung.
10. Ibu Ade Noera Maela, selaku Admin Jurusan Ilmu Komputer yang telah membantu penulis dalam mengurus segala administrasi di Jurusan Ilmu Komputer Universitas Lampung.
11. Bang Zainudin, Mas Sam, dan Mas Nofal yang telah membantu penulis dalam mengurus segala jenis perizinan di Jurusan Ilmu Komputer Universitas Lampung.
12. Bapak dan Ibu Dosen Jurusan Ilmu Komputer Universitas Lampung yang telah membantu penulis dalam proses menimba ilmu yang bermanfaat untuk penulis selama perkuliahan di Jurusan Ilmu Komputer Universitas Lampung.
13. Kedua orang tua dan kakak penulis yang selalu mendoakan, menyemangati, menasihati, membiayai serta mendukung penulis baik secara moral maupun material.
14. Teman satu bimbingan skripsi, yaitu Mohammed Raihan Akbar yang selalu mendukung, menyemangati, dan membantu selama proses penyusunan skripsi ini hingga selesai.
15. Teman dekat di jurusan, yaitu Abbie Syeh Nahri dan Mohammed Raihan Akbar yang telah menjadi teman yang baik dan selalu memberikan banyak dukungan dan bantuan dalam hal apapun.
16. Teman-teman Ilmu Komputer 2019, yang tidak dapat disebut satu persatu yang telah mendukung dan membantu selama proses perkuliahan di Jurusan Ilmu Komputer Universitas Lampung.
17. *.Support System*, yaitu Rika Septiani yang selalu mendoakan, menyemangati, membantu, dan mendukung untuk menyelesaikan skripsi ini.
18. *Last but not least, I wanna thank me. I wanna thank me for believing in me. I wanna thank me for all doing this hard work. I wanna thank me for having no*

days off. I wanna thank me for never quitting. I wanna thank me for just being me at all times.

Penulis menyadari bahwa dalam penulisan skripsi ini masih terdapat banyak kekurangan dikarenakan keterbatasan pengalaman, kemampuan dan pengetahuan penulis, serta skripsi ini juga masih jauh dari kata sempurna. Oleh sebab itu, penulis mengharapkan kritik dan saran yang membangun dari para pembaca untuk bahan evaluasi dan pembelajaran penulis di masa mendatang. Semoga skripsi ini dapat membawa manfaat dan keberkahan bagi para pembaca.

Bandar Lampung, 3 September 2024

Irfan Yadi

NPM 1917051032

DAFTAR ISI

Halaman

DAFTAR ISI.....	xvi
DAFTAR GAMBAR.....	xix
DAFTAR POTONGAN KODE	xx
DAFTAR TABEL	xxi
I. PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	4
1.3 Batasan Masalah.....	5
1.4 Tujuan Penelitian	5
1.5 Manfaat Penelitian	5
II. TINJAUAN PUSTAKA	6
2.1 Penelitian Terdahulu	6
2.2 Uraian Tinjauan Pustaka	9
2.2.1 Penerjemahan	9
2.2.2 Bahasa Lampung.....	9
2.2.3 Python	9
2.2.4 Google Colaboratory	10
2.2.5 <i>Deep Learning</i>	10
2.2.6 <i>Natural Language Processing</i>	11
2.2.7 <i>Transformer</i>	11
2.2.8 <i>Overfitting</i>	15
2.2.9 <i>Dropout</i>	16
2.2.10 <i>Accuracy dan Loss</i>	16

2.2.11	<i>Bilingual Evaluation Understudy (BLEU)</i>	17
III.	METODOLOGI PENELITIAN	19
3.1	Waktu dan Tempat Penelitian	19
3.1.1	Waktu Penelitian	19
3.1.2	Tempat Penelitian.....	19
3.2	Perangkat Penelitian.....	19
3.2.1	Perangkat Keras	20
3.2.2	Perangkat Lunak.....	20
3.3	Tahapan Penelitian	20
3.3.1	<i>Dataset</i>	21
3.3.2	Normalisasi	22
3.3.3	Tokenisasi	23
3.3.4	Vektorisasi	23
3.3.5	<i>Embedding</i>	23
3.3.6	<i>Positional Encoding</i>	23
3.3.7	<i>Encoding</i>	24
3.3.8	<i>Decoding</i>	24
3.3.9	Evaluasi	25
IV.	HASIL DAN PEMBAHASAN	27
4.1	Implementasi Arsitektur <i>Transformer</i>	27
4.1.1	Persiapan Data.....	27
4.1.2	Normalisasi Teks.....	28
4.1.3	Tokenisasi Teks.....	29
4.1.4	Vektorisasi Teks.....	30
4.1.5	<i>Embedding</i>	31
4.1.6	<i>Positional Encoding</i>	32
4.1.7	<i>Encoding</i>	33
4.1.8	<i>Decoding</i>	38
4.2	Analisis Hasil	43
4.2.1	Hasil Nilai <i>Accuracy</i> dan <i>Loss</i>	43
4.2.2	Hasil Skor BLEU	45

4.2.3 Hasil Penerjemahan Teks	50
V. KESIMPULAN DAN SARAN	54
5.1 Kesimpulan	54
5.2 Saran.....	54
DAFTAR PUSTAKA	55

DAFTAR GAMBAR

Gambar	Halaman
1. Arsitektur <i>Transformer</i> (Vaswani et al., 2017).	12
2. <i>Multi-Head Attention</i> (Vaswani et al., 2017).	13
3. <i>Encoder Transformer</i> (Vaswani et al., 2017).	14
4. <i>Decoder Transformer</i> (Vaswani et al., 2017).	15
5. Tahapan Penelitian.	20
6. Distribusi Panjang Token.	30
7. <i>Plot Model Encoder</i> .	33
8. <i>Plot Model Multi-Head Self-Attention</i> .	35
9. <i>Plot Model Feed-Forward</i> .	38
10. <i>Plot Model Decoder</i> .	39
11. <i>Plot Model Cross-Attention</i> .	42
12. Grafik <i>Accuracy</i> dan <i>Loss</i> .	45

DAFTAR POTONGAN KODE

Potongan Kode	Halaman
1. Proses Normalisasi Teks.	28
2. Proses Tokenisasi Teks.	29
3. Proses Vektorisasi Teks.	31
4. Proses <i>Embedding</i>	32
5. Proses <i>Positional Encoding</i>	32
6. Proses <i>Multi-Head Self-Attention</i>	35
7. Proses <i>Feed-Forward</i>	37
8. Proses <i>Masked Multi-Head Self-Attention</i>	40
9. Proses <i>Cross-Attention</i>	41
10. Proses Penerjemahan Teks.	51

DAFTAR TABEL

Tabel	Halaman
1. Penelitian Terdahulu	6
2. <i>Dataset</i> bahasa Lampung – Indonesia.....	21
3. <i>Hyperparameters</i> Arsitektur <i>Transformer</i>	43
4. Hasil Nilai <i>Accuracy</i> dan <i>Loss</i>	44
5. Hasil Skor BLEU	49
6. Hasil Penerjemahan Teks.....	52

I. PENDAHULUAN

1.1 Latar Belakang

Indonesia merupakan sebuah negara kepulauan yang memiliki ribuan pulau yang terletak di kawasan Asia Tenggara. Hal itulah yang membuat negara Indonesia terkenal dengan keanekaragaman budaya dan bahasanya. Indonesia merupakan negara yang memiliki bahasa daerah terbanyak di dunia yang berjumlah kurang lebih 750 bahasa daerah. Namun, 160 bahasa daerah hampir punah atau tidak digunakan lagi dalam berkegiatan sehari-hari. Hal tersebut disebabkan oleh kurangnya pengajaran tentang berbahasa daerah di sekolah dan di lingkungan sekitar (Putri, 2021).

Bahasa daerah merupakan bagian penting dari keanekaragaman bahasa yang ada di Indonesia yang mencerminkan kekayaan akan budaya dan sejarah bangsa. Masyarakat setempat sering menggunakan bahasa daerah sebagai bahasa pergaulan dalam berkegiatan sehari-hari, mulai dari berkegiatan sosial, pendidikan, kebudayaan serta keagamaan. Bahasa daerah juga merupakan aspek penting dari identitas budaya bangsa yang harus dilestarikan agar tidak punah. Indonesia memiliki beberapa bahasa daerah yang populer, salah satunya adalah bahasa daerah yang digunakan di Provinsi Lampung, yaitu bahasa Lampung.

Bahasa Lampung merupakan bahasa yang tergolong dalam rumpun bahasa Austronesia dari bahasa Melayu-Polinesia. Bahasa Lampung mempunyai jumlah penutur jati yang relatif besar, yaitu sebesar 5,19 juta jiwa. Bahasa Lampung terdiri dari dua macam dialek, yaitu dialek Api atau dialek A yang digunakan oleh

masyarakat Pesisir dan dialek Nyo atau dialek O yang digunakan oleh masyarakat Pepadun. Meskipun bahasa Lampung mempunyai jumlah penutur jati yang relatif besar, namun bahasa Lampung sendiri merupakan bahasa minoritas di Provinsi Lampung.

Bahasa Lampung merupakan aspek penting yang dimiliki oleh masyarakat Lampung. Jika tidak adanya upaya yang dilakukan untuk melestarikannya, maka akan mengakibatkan dampak yang merugikan, yaitu punahnya warisan budaya yang sangat bermakna bagi masyarakat Lampung. Upaya untuk menjaga serta melestarikan bahasa daerah tersebut bukan hanya tanggung jawab dari pemerintah pusat saja, tetapi juga memerlukan peran aktif dan partisipasi dari berbagai pihak. Pemerintah daerah, masyarakat, dan institusi pendidikan atau lembaga pendidikan harus turut berpartisipasi untuk menjaga serta melestarikan bahasa daerah (Abidin, 2017).

Diperlukan upaya yang konkret untuk menjaga eksistensi dan melindungi bahasa Lampung agar tetap digunakan oleh masyarakat Lampung (Prihantika *et al.*, 2019). Salah satu cara yang dapat kita lakukan untuk mencapai tujuan melestarikan bahasa Lampung yaitu dengan memanfaatkan mesin penerjemah untuk mempermudah masyarakat untuk mempelajari atau menggunakan bahasa Lampung dalam berkegiatan sehari-hari. Mesin penerjemah yang paling populer atau banyak orang ketahui saat ini yaitu google *translate*. Namun mesin penerjemah ini hanya terdapat bahasa Jawa dan Sunda, sedangkan bahasa Lampung masih belum tersedia pada mesin penerjemah tersebut.

Penerjemahan bahasa yang menggunakan mesin penerjemah termasuk dalam kategori *Natural Language Processing* (NLP) atau biasa disebut dengan pemrosesan bahasa alami. NLP sudah menjadi salah satu bidang yang menarik dalam ilmu komputer dan kecerdasan buatan. NLP ini memungkinkan komputer untuk bisa memahami serta mengolah teks yang hanya dapat dipahami manusia ke dalam berbagai bahasa (Amien, 2023). NLP ini memiliki tujuan utama yaitu untuk

memungkinkan sebuah komputer untuk bisa memahami, memproses, dan menghasilkan bahasa manusia dengan cara yang lebih efektif.

Transformer merupakan arsitektur yang digunakan pada *deep learning* untuk melakukan pemrosesan teks, seperti penerjemahan bahasa. Arsitektur ini pertama kali dikembangkan oleh Google dalam sebuah jurnal yang dipublikasi pada tahun 2017. Sejak saat itulah arsitektur ini sudah menjadi standar pada bidang *natural language processing*. Arsitektur *Transformer* ini memanfaatkan teknik *self-attention* untuk dapat memecahkan masalah dependensi jarak jauh pada teks, yang sebelumnya dianggap sebagai masalah yang menantang dalam pemrosesan teks. Dengan demikian model tersebut dapat menangani input dalam jumlah besar serta menghasilkan hasil yang lebih baik dibandingkan dengan arsitektur sebelumnya.

Arsitektur *transformer* pertama kali diperkenalkan oleh Ashish Vaswani dkk. pada jurnal yang berjudul “*Attention Is All You Need*” pada tahun 2017. Pada jurnal tersebut menunjukkan bahwa arsitektur *transformer* ini dapat mengatasi masalah pemrosesan bahasa tanpa menggunakan *Recurrent Neural Network* (RNN) atau *Convolutional Neural Network* (CNN) (Vaswani *et al.*, 2017), sehingga arsitektur ini dianggap sebagai arsitektur yang paling efektif untuk menangani masalah tersebut. Arsitektur *transformer* menggunakan metode *self-attention* untuk dapat mengatasi masalah pemrosesan bahasa yang memerlukan konteks yang luas, seperti penerjemahan bahasa. Pada jurnal tersebut juga menunjukkan bahwa *transformer* memperoleh nilai atau skor BLEU yang lebih baik dibandingkan dengan arsitektur canggih sebelumnya untuk penerjemahan bahasa Inggris - Jerman dengan perolehan nilai BLEU sebesar 28,4 dan bahasa Inggris - Prancis diperoleh nilai BLEU sebesar 41,8 (Vaswani *et al.*, 2017).

Arsitektur *transformer* ini juga telah digunakan pada beberapa mesin penerjemahan bahasa yang ada di dunia. Salah satunya pada sebuah penelitian penerjemahan bahasa yang ditulis oleh (Tian *et al.*, 2021) yang menggunakan *dataset* bahasa Prancis – bahasa Inggris. Pada penelitian tersebut dijelaskan bahwa hasil dari terjemahan bahasa Prancis – Inggris berdasarkan model *transformer* yang

dirancang dalam penelitian ini memperoleh tingkat akurasi terjemahan yang lumayan besar, yaitu sebesar 80% setelah melakukan pelatihan aplikasi (Tian *et al.*, 2021).

Pada penelitian lainnya yang ditulis oleh Wismoyo pada tahun 2018, juga menggunakan arsitektur *transformer* untuk diimplementasikan pada mesin penerjemah bahasa Inggris – Indonesia dengan berbasis jaringan saraf tiruan. Penelitian ini dilakukan untuk mengetahui penilaian BLEU dari hasil terjemahan bahasa Inggris ke bahasa Indonesia. Hasil pengujian menggunakan korpus uji dev2010, tst2010, tst2017plus, dan test.go.id menunjukkan bahwa model mesin penerjemah mendapatkan skor BLEU 24,83% (dev2010) 24,14% (tst2010), 27,31% (tst2017plus), dan 34,18% (test.go.id) (Wismoyo, 2018).

Penelitian yang berkaitan dengan penerjemahan bahasa sudah lumayan banyak dilakukan. Walaupun begitu, penelitian yang berkaitan dengan penerjemahan bahasa daerah masih jarang ditemukan terutama untuk penerjemahan bahasa Lampung – Indonesia. Pada penelitian ini akan menggunakan 11.933 data pasangan kata dan kalimat bahasa Lampung berdialek A yang sebelumnya sudah diterjemahkan ke dalam bahasa Indonesia. Berdasarkan beberapa penelitian terdahulu yang mencapai perolehan nilai BLEU atau akurasi terjemahan terbaik atau terbesar diantara metode lainnya. Maka, penelitian ini akan menggunakan arsitektur *transformer* untuk penerjemahan bahasa Lampung – Indonesia.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang ada maka dapat disimpulkan bahwa rumusan masalah yang akan diteliti pada penelitian ini yaitu bagaimana mengimplementasikan arsitektur *transformer* untuk penerjemahan bahasa Lampung – Indonesia.

1.3 Batasan Masalah

Untuk memperoleh gambaran yang lebih jelas dalam penelitian ini, maka diberikan batasan-batasan sebagai berikut:

1. Menggunakan arsitektur *transformer* generasi pertama dengan mekanisme *attention* untuk penerjemahan bahasa Lampung - Indonesia.
2. Menggunakan *dataset* terjemahan bahasa Lampung – Indonesia dengan fokus pada dialek Api.
3. Menggunakan metrik *Bilingual Evaluation Understudy* (BLEU) untuk mengevaluasi hasil dari penerjemahan bahasa Lampung – Indonesia.

1.4 Tujuan Penelitian

Tujuan yang ingin dicapai dalam penelitian ini adalah sebagai berikut:

1. Mengimplementasikan arsitektur *transformer* untuk penerjemahan bahasa Lampung – Indonesia.
2. Mengukur hasil terjemahan dari arsitektur *transformer* untuk penerjemahan bahasa Lampung – Indonesia.

1.5 Manfaat Penelitian

Adapun manfaat dalam penelitian ini adalah sebagai berikut:

1. Mengetahui performa dari arsitektur *transformer* untuk penerjemahan bahasa Lampung – Indonesia.
2. Dapat digunakan sebagai referensi atau acuan dalam penelitian yang akan datang yang memiliki kesamaan dengan topik penelitian yang sama.

II. TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Penelitian terdahulu bertujuan untuk membandingkan penelitian yang sudah ada dengan penelitian yang akan dilakukan. Selain itu, penelitian terdahulu akan dijadikan bahan pembelajaran. Penelitian tersebut dapat dilihat pada Tabel 1.

Tabel 1. Penelitian Terdahulu

No.	Judul	Metode	Data	Hasil
Penelitian				
1	<i>Attention Is All You Need</i> (Vaswani et al., 2017).	<i>Transformer</i>	<i>Dataset</i> WMT 2014 <i>English-German</i> dan WMT 2014 <i>English-French</i> .	Model baru ini memperoleh nilai BLEU sebesar 28,4 dari bahasa Inggris - Jerman dan juga memperoleh nilai BLEU sebesar 41,8 dari bahasa Inggris - Prancis.
2	<i>A French-to-English Machine Translation Model Using</i>	<i>Transformer</i>	Data korpus yang memuat bahasa Prancis sebagai bahasa sumber dan	Penelitian ini merancang mesin penerjemah bahasa Prancis – Inggris berbasis <i>transformer</i>

- Transformer Network* (Tian et al., 2021). bahasa Inggris dengan menggunakan ssbagai bahasa mekanisme *attention* target. yang menghasilkan tingkat akurasi terjemahan sebesar 80%.
- 3 Mesin Penerjemah Bahasa Inggris – Indonesia Berbasis Jaringan Saraf Tiruan Dengan Mekanisme *Attention* Menggunakan Arsitektur *Transformer* (Wismoyo, 2018). *Transformer* Data korpus uji dev2010, tst2010, tst2017plus, dan test.go.id. Penelitian ini menguji model *transformer* pada penerjemahan bahasa Inggris ke Indonesia dengan menggunakan data korpus uji dev2010, tst2010, tst2017plus, dan test.go.id menunjukkan bahwa model mesin penerjemah mendapatkan skor evaluasi BLEU sebesar 24,83% (dev2010), 24,14% (tst2010), 27,31% (tst2017plus), dan 34,18% (test.go.id).
- 4 *Exploring Text-to-Text Transformers for English to Hinglish Machine Translation* mT5 dan mBART *Dataset PHINC* yang memuat korpus paralel sebanyak 13.738 kalimat bahasa *Hinglish* sebagai monolingual ke bahasa sumber multilingual, yaitu dari

<p><i>with Synthetic Code-Mixing</i> (Jawahar et al., 2021).</p>	<p>dan bahasa Inggris ke Inggris sebagai bahasa Hinglish (<i>Hindi & English</i>). Penelitian ini menunjukkan bahwa model mT5 memperoleh kinerja model penerjemahan terbaik, yaitu sebesar 12,67 BLEU, sedangkan model mBART sebesar 11,00 BLEU.</p>
<p>5 <i>Experiment on a Transformer Model Indonesian-to-Sundanese Neural Machine Translation with Sundanese Speech Level Evaluation</i> (Primandhika et al., 2021).</p>	<p>Data yang digunakan dari berbagai sumber, mulai dari <i>subtitle film</i> di <i>subscene</i>, alkitab online dwibahasa dan website berbahasa Sunda-Indonesia. Penelitian ini berfokus pada masalah dalam penerjemahan bahasa Indonesia ke bahasa Sunda dan juga mengevaluasi tingkat tutur bahasa Sunda dalam teks yang sudah diterjemahkan dengan menggunakan model <i>transformer</i>. Penelitian ini memperoleh nilai BLEU terbesar dibandingkan dengan model lainnya, yaitu sebesar 42,72 %.</p>

2.2 Uraian Tinjauan Pustaka

2.2.1 Penerjemahan

Penerjemahan merupakan suatu keterampilan yang dipakai dalam upaya untuk mengubah pesan atau pernyataan tertulis dalam satu bahasa ke dalam bahasa lainnya (Wuryantoro, 2018). Meskipun penerjemahan sangat penting serta sudah lama digunakan baik di negara lain maupun di Indonesia, penerjemahan bukanlah suatu hal yang mudah untuk dilakukan. Penerjemahan bukan hanya semata-mata kegiatan dalam mengartikan suatu kata dengan mencari pasangan kata tersebut dalam bahasa sasarannya. Budaya penerjemah memengaruhi dalam proses pengalihan suatu pesan teks dari bahasa sumber ke bahasa target, yang tercermin dari cara seseorang memahami, memandang, dan mengungkapkan pesan melalui bahasa yang mereka gunakan (Ardi, 2015).

2.2.2 Bahasa Lampung

Bahasa Lampung adalah salah satu bahasa daerah yang terdapat di Indonesia. Bahasa Lampung juga menjadi identitas daerah yang dimiliki oleh masyarakat Lampung. Lampung merupakan salah satu provinsi di Indonesia yang letaknya sangat strategis di Pulau Sumatera. Provinsi Lampung terletak di ujung Selatan Pulau Sumatera, menjadikan Provinsi Lampung dihuni oleh dua golongan, yaitu masyarakat asli dan masyarakat pendatang (Hartono, 2016). Provinsi Lampung adalah salah satu provinsi yang berada di Pulau Sumatera. Pada Provinsi Lampung masyarakat menggunakan dua bahasa daerah yang berbeda dialek yaitu bahasa Lampung dialek Api atau dialek A dan bahasa Lampung dialek Nyo atau dialek O (Abidin & Permata, 2021).

2.2.3 Python

Python adalah bahasa pemrograman tingkat tinggi yang dibangun oleh Guido Van Rossum dan pertama kali diperkenalkan pada tahun 1991. Selain itu juga python merupakan bahasa pemrograman yang dapat digunakan untuk berbagai tujuan contohnya digunakan untuk *Machine Learning* dan *Deep Learning*. Python mempunyai penulisan sintaksis yang sederhana. Bahasa pemrograman ini juga

memiliki banyak *library* serta memiliki dukungan komunitas yang kuat karena python bersifat *open source* (Alfarizi *et al.*, 2023). Bahasa pemrograman ini dapat dipakai untuk berbagai macam tujuan, termasuk pengembangan web, komputasi ilmiah, dan kecerdasan buatan (Lutz, 2021).

2.2.4 Google Colaboratory

Google Colaboratory atau dikenal juga dengan "Google Colab" merupakan sebuah proyek penelitian yang bertujuan untuk membuat prototipe model pembelajaran mesin yang menggunakan perangkat keras yang kuat seperti *Graphics Processing Unit* (GPU) dan *Tensor Processing Unit* (TPU). Proyek ini menyediakan *environment* jupyter notebook tanpa server guna pengembangan yang lebih interaktif. Google Colab ini gratis untuk digunakan seperti produk Google Suite yang lain (Bisong, 2019). Google Colab menyediakan *runtime* python 2 dan 3 yang sudah dikonfigurasi sebelumnya yang dilengkapi dengan *library* pembelajaran mesin dan kecerdasan buatan utama, seperti TensorFlow, Matplotlib, dan Keras (Carneiro *et al.*, 2018).

2.2.5 Deep Learning

Deep learning merupakan subbidang dari *machine learning* dengan algoritma yang terinspirasi dari struktur otak manusia, struktur ini dikenal dengan *Artificial Neural Networks* (ANN). Pada prinsipnya, *deep learning* merupakan jaringan saraf yang mempunyai tiga atau lebih lapisan ANN. *Deep Learning* dapat belajar serta beradaptasi dengan sejumlah data yang besar serta dapat menyelesaikan beragam permasalahan yang sulit dipecahkan menggunakan algoritma *machine learning* yang lain (Raup *et al.*, 2022). Semakin banyak data yang dipelajari oleh *deep learning* maka membuat *deep learning* semakin meningkatkan akurasi dan kinerja model dalam membuat suatu prediksi atau klasifikasi.

Deep learning adalah konsep dari *machine learning* yang didasarkan pada jaringan saraf tiruan. Pada beberapa tujuan, model *deep learning* ini memperoleh hasil yang lebih baik dibandingkan dengan model *machine learning* yang sederhana dan metode analisis data tradisional (Sharma & Chaudhary, 2023). *Deep learning*

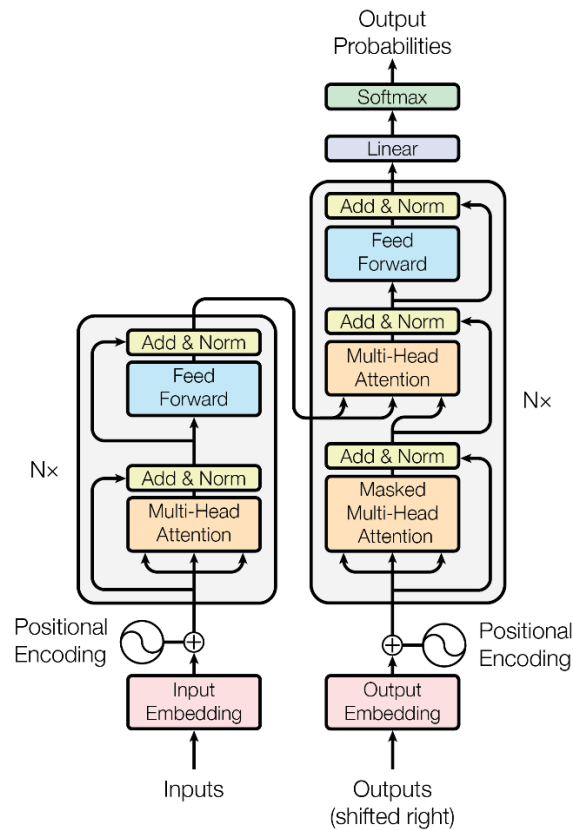
merupakan pendekatan baru pada analisis data yang memiliki tujuan untuk mengenali pola dengan data bervolume besar serta bertambah secara cepat pada beberapa bidang seperti *computer vision*, *natural language processing*, dan *speech recognition*.

2.2.6 Natural Language Processing

Natural Language Processing (NLP) atau pemrosesan bahasa alami merupakan bagian dari kecerdasan buatan yang memiliki tujuan untuk memastikan komputer dapat memahami, menafsirkan, dan memanipulasi bahasa manusia. Proses pengolahan bahasa natural (NLP) melibatkan penggunaan korpus teks dan teknik pembelajaran yang mendalam untuk dapat menyelesaikan tugas statistik seperti klasifikasi teks, analisis sentimen, dan penerjemahan mesin, kedua teknik ini berhasil dalam banyak aplikasi (Glaz et al., 2021). Kinerja pada tugas-tugas NLP semakin meningkat karena terdapat kemajuan pada *machine learning* terutama pada bidang *deep learning*. Peningkatan dalam tugas-tugas NLP ini dapat terjadi karena terdapat sebuah arsitektur baru yang memiliki kinerja lebih baik dalam *deep learning* yaitu arsitektur *transformer*.

2.2.7 Transformer

Arsitektur *Transformer* pertama kali diperkenalkan oleh Ashish Vaswani dalam penelitiannya yang berjudul “*Attention Is All You Need*”. *Transformer* sepenuhnya memanfaatkan *attention mechanisms* dan tidak menggunakan rekurensi atau konvolusi seperti arsitektur sebelumnya (Vaswani et al., 2017). *Transformer* merupakan model terbaru dan terancang pada mesin penerjemah akhir-akhir ini (Wang et al., 2020). *Transformer* terdiri dari beberapa bagian, yaitu *self-attention*, *encoder*, *decoder*, dan *feed forward neural network* (Tian et al., 2021). Arsitektur *Transformer* ini sangat bergantung dengan jaringan *self-attention* (Tang et al., 2018). Model *transformer* dapat dilihat pada Gambar 1.



Gambar 1. Arsitektur *Transformer* (Vaswani et al., 2017).

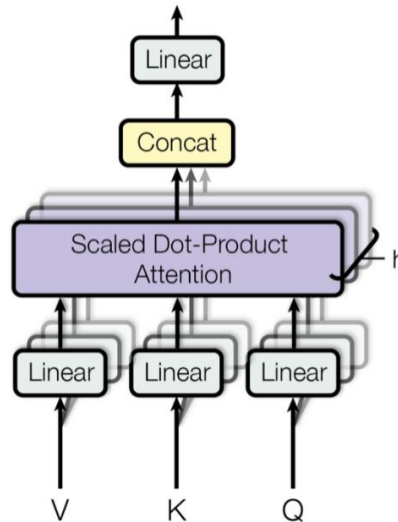
2.2.7.1 *Positional Encoding*

Arsitektur *transformer* tidak menggunakan rekurensi dan konvolusi seperti arsitektur sebelumnya, ini berarti model *transformer* tidak dapat memahami urutan dari suatu urutan token secara alami. Oleh karena itu, pada model *transformer* perlu ditambahkan informasi posisi token atau kata dalam suatu urutan tersebut. Sehingga model dapat memproses urutan token secara tepat sesuai dengan urutan aslinya (Vaswani et al., 2017).

2.2.7.2 *Multi-Head Attention*

Multi-Head Attention merupakan suatu mekanisme *attention* yang menjalankan *scaled dot-product attention* beberapa kali secara paralel. Dengan menggunakan beberapa *Multi-Head Attention*, model dapat memfokuskan *attention* pada bagian-bagian yang berbeda dari urutan token. Hal tersebut memungkinkan model untuk dapat memahami konteks secara lebih tepat serta dapat meningkatkan kualitas terjemahan atau melakukan tugas-tugas NLP lainnya (Weng, 2018). *Multi-Head*

Attention terdiri dari empat bagian, yaitu *linear layer*, *contact all heads*, *scaled dot-product attention*, dan *final linear layer* (Tian *et al.*, 2021). Gambar *Multi-Head Attention* pada arsitektur *transformer* dapat dilihat pada Gambar 2 berikut.



Gambar 2. Multi-Head Attention (Vaswani et al., 2017).

Scaled Dot-Product Attention adalah jenis *attention* yang digunakan dalam *transformer* untuk menghitung bobot *attention* antara *query* dan *key*. Pada *Scaled Dot-Product Attention*, setiap *query*, *key* dan *value* direpresentasikan sebagai vektor, dan kemudian bobot *attention* dihitung dengan mengalikan *query* dengan *key* dan kemudian dinormalisasi dengan faktor skala yang bergantung pada dimensi vektor. Setelah itu, bobot *attention* ini digunakan untuk menghitung *output* sebagai jumlah tertimbang dari *value*. Skala ini digunakan untuk mencegah hasil perkalian *dot-product* menjadi terlalu besar, yang dapat menghambat pembelajaran model.

2.2.7.3 Add & Normalization

Pada setiap lapisan atau blok dari model arsitektur *transformer* terdapat dua komponen utama, yaitu *Add & Normalization*. Kedua komponen ini bekerja sama untuk meningkatkan pembelajaran serta stabilitas dari model.

1. Addition

Komponen *addition* atau disebut juga dengan *residual connection* atau *skip connection*, memungkinkan input untuk melewati beberapa lapisan transformasi tambahan, yaitu dengan menambahkan input ke *output* lapisan

tersebut (Vaswani et al., 2017). Misal x adalah input ke lapisan dan $\text{layer}(x)$ adalah *output* dari lapisan tersebut, penjumlahan dilakukan seperti berikut:

$$\text{Output} = x + \text{layer}(x).$$

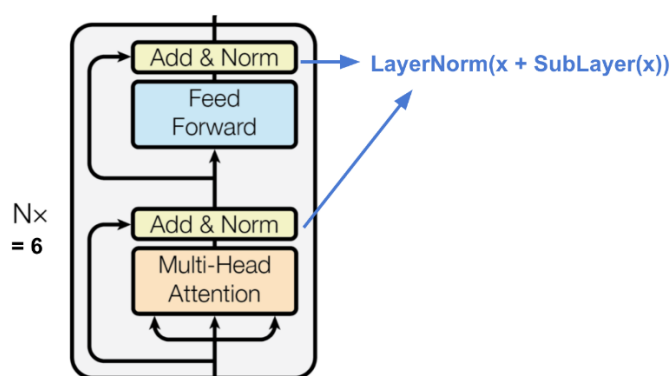
Penjumlahan ini dapat mengatasi masalah gradien yang menghilang dalam *very deep neural networks*.

2. Normalization

Lapisan *normalization* merupakan komponen yang memiliki tujuan untuk menjaga distribusi data dalam lapisan agar tetap stabil selama proses *training*. *Layer Normalization* menghitung statistik seperti *mean* (rata-rata) dan variansi dari data dalam lapisan, dan kemudian menggunakan statistik ini untuk menormalisasi data dalam lapisan. Normalisasi ini membantu untuk mencegah perubahan yang besar-besaran pada distribusi data, yang dapat menyebabkan masalah pada proses *training*.

2.2.7.4 Encoder

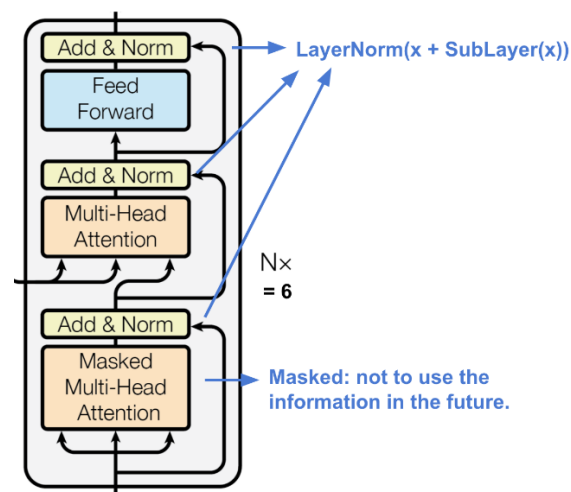
Encoder terdiri dari tumpukan $N = 6$ lapisan yang identik. Setiap lapisan memiliki dua sub-lapisan. Lapisan yang pertama, yaitu lapisan *multi-head self-attention* dan yang kedua, yaitu lapisan *fully connected feed-forward network*. Setiap sub-lapisan menggunakan koneksi residual dan juga diikuti dengan lapisan *normalization* (Vaswani et al., 2017). Gambar *encoder* pada arsitektur *transformer* dapat dilihat pada Gambar 3.



Gambar 3. Encoder Transformer (Vaswani et al., 2017).

2.2.7.5 Decoder

Decoder juga terdiri dari tumpukan $N = 6$ lapisan yang identik. Selain dua sub-lapisan di setiap lapisan *encoder*, *decoder* menambahkan sub-lapisan ketiga, yaitu lapisan *masked multi-head self-attention*. Sama seperti dengan *encoder*, setiap sub-lapisan pada *decoder* juga menggunakan koneksi residual dan juga diikuti dengan lapisan *normalization*. Sublapisan *multi-head attention* pertama dimodifikasi untuk mencegah posisi atau kata dalam kalimat dapat memperhatikan kata-kata berikutnya dalam konteks *self-attention* (Vaswani *et al.*, 2017). Gambar *decoder* pada arsitektur *transformer* dapat dilihat pada Gambar 4.



Gambar 4. *Decoder Transformer* (Vaswani *et al.*, 2017).

2.2.8 Overfitting

Overfitting merupakan masalah yang sering muncul pada *machine learning* dan tidak dapat dihindari sepenuhnya. Masalah ini muncul dikarenakan adanya keterbatasan pada data pelatihan, seperti jumlah data yang sedikit, atau bisa juga karena terdapat banyaknya *noise* dalam data pelatihan yang membuat keakuratan model terganggu. Selain itu, terdapat kendala pada algoritma yang terlalu kompleks, yang memerlukan banyak parameter dan membuat model menjadi terlalu teliti terhadap data pelatihan. Sehingga mengakibatkan model tersebut menjadi gagal dalam melakukan generalisasi terhadap data baru yang belum pernah dilihat sebelumnya oleh model, sehingga pada saat model melakukan prediksi terhadap data baru mengakibatkan performa model menjadi buruk (Ying, 2019).

2.2.9 Dropout

Dropout adalah salah satu teknik regularisasi untuk mengatasi masalah *overfitting* dan meningkatkan keahlian model untuk generalisasi data. Teknik ini mengeluarkan unit-unit sekaligus koneksinya dari jaringan saraf secara acak selama pelatihan model. Hal tersebut mencegah agar unit-unit tidak terlalu terikat pada data pelatihan. Selama pengujian, teknik ini dapat memperkirakan hasil akhir dengan menggunakan satu jaringan penuh dengan bobot yang lebih kecil, yang dimana itu mewakili rata-rata dari semua jaringan yang lebih kecil tadi. Teknik ini sangat efektif untuk mengurangi *overfitting* serta meningkatkan performa model dibandingkan dengan metode regularisasi lainnya. *Dropout* telah teruji untuk meningkatkan hasil model jaringan saraf dalam bermacam tugas, seperti klasifikasi dokumen, pengenalan gambar dan suara dengan memperoleh hasil yang lebih baik (Srivastava et al., 2014).

2.2.10 Accuracy dan Loss

Accuracy merupakan sebuah metrik yang digunakan untuk mengetahui seberapa besar prediksi yang benar yang dihasilkan oleh model dibandingkan dengan total prediksi yang dilakukan. Secara umum, *accuracy* diartikan sebagai perbandingan antara jumlah prediksi yang benar dengan total jumlah data. *Accuracy* pada penerjemahan bahasa merupakan perbandingan antara jumlah terjemahan yang benar baik secara semantik maupun sintaksis dengan total jumlah terjemahan yang dihasilkan (Lee et al., 2023). *Loss* merupakan suatu fungsi yang sering ditemukan dalam *deep learning*. Terdapat beberapa jenis fungsi *loss* yang sering digunakan pada *deep learning*, salah satunya adalah *sparse categorical cross-entropy*. *Sparse categorical cross-entropy* adalah fungsi *loss* yang digunakan untuk menentukan kinerja model pada masalah klasifikasi di mana label targetnya bersifat kategorikal dalam bentuk indeks (Dhiman et al., 2022). Fungsi *loss* ini sering digunakan pada model penerjemahan bahasa yang memanfaatkan jaringan saraf, seperti model *Transformer* atau *Recurrent Neural Networks* (RNN).

2.2.11 Bilingual Evaluation Understudy (BLEU)

Metrik BLEU atau dikenal dengan skor BLEU berguna untuk mengukur kualitas dari terjemahan berdasarkan akurasi penerjemahan frasa teks ke referensi teks terjemahan dengan panjang frasa yang beragam (Tran *et al.*, 2019). *Bilingual Evaluation Understudy* (BLEU) merupakan algoritma pencocokan *string* yang menyediakan metrik kualitas dasar untuk peneliti dan pengembang *machine translation* (MT). BLEU merupakan metrik penilaian kualitas *machine translation* yang sering digunakan dalam kurun waktu 15 tahun terakhir. Walaupun sebagian besar memahami bahwa BLEU ini terdapat beberapa kekurangan, tetapi metrik ini selalu menjadi metrik utama dan populer dalam mengukur *output* sistem *machine translation* sampai saat ini, pada masa kejayaan *Neural Machine Translation* (Vashee, 2019).

Adapun rumus untuk menghitung skor BLEU adalah sebagai berikut:

$$BP = \begin{cases} 1, & \text{if } c > r \\ \exp^{(1-\frac{r}{c})}, & \text{if } c \leq r \end{cases}$$

$$BLEU = BP \cdot \exp(\sum_{n=1}^n w_n \log p_n)$$

Keterangan:

BP = *Brevity Penalty*

C = Jumlah kata dari hasil terjemahan otomatis (candidate)

r = Jumlah kata rujukan (reference)

w_n = $1/N$ (standar nilai N untuk BLEU adalah 4)

p_n = Jumlah n-gram hasil terjemahan yang sesuai dengan rujukan dibagi jumlah n-gram hasil terjemahan

Skor BLEU adalah salah satu metrik yang sering digunakan untuk mengevaluasi kualitas dari terjemahan mesin. Skor BLEU mengukur akurasi terjemahan dengan cara membandingkan n-gram (kelompok kata berurutan dari n kata) dalam hasil terjemahan mesin dengan n-gram pada terjemahan referensi. Skor BLEU memiliki rentang nilai dari 0 hingga 1, yang dimana nilai 1 menunjukkan kesamaan yang

sempurna dengan terjemahan referensi sedangkan 0 menunjukkan tidak ada kesamaan. Skor BLEU yang tinggi menunjukkan bahwa terjemahan mesin sangat mirip dengan terjemahan referensi manusia, yang berarti kualitas terjemahan yang baik, sedangkan skor BLEU yang rendah menunjukkan kualitas terjemahan buruk. Skor BLEU dalam praktiknya biasanya dinyatakan dalam persentase, misalnya 0,85 atau 85%.

III. METODOLOGI PENELITIAN

3.1 Waktu dan Tempat Penelitian

Adapun waktu dan tempat penelitian adalah sebagai berikut:

3.1.1 Waktu Penelitian

Penelitian ini akan dilakukan pada semester genap tahun ajaran 2022/2023, yang direncanakan berlangsung mulai dari bulan Juni 2023 dan berlanjut hingga bulan September 2023. Periode penelitian ini akan mencakup berbagai kegiatan, yaitu pengumpulan data, pemrograman arsitektur *transformer*, evaluasi kinerja arsitektur *transformer*, dan penulisan laporan.

3.1.2 Tempat Penelitian

Penelitian ini akan dilakukan di Jurusan Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Lampung yang berada di Jalan Prof. Dr. Ir. Sumantri Brojonegoro No.1, Gedong Meneng, Kec. Rajabasa, Kota Bandar Lampung, Lampung dan dikediaman Lampung Selatan yaitu di Jatimulyo, Kec. Jati Agung, Kab. Lampung Selatan, Lampung.

3.2 Perangkat Penelitian

Penelitian ini menggunakan perangkat pendukung baik perangkat keras maupun perangkat lunak dengan masing-masing spesifikasinya adalah sebagai berikut.

3.2.1 Perangkat Keras

Perangkat Keras yang digunakan adalah laptop dengan spesifikasi sebagai berikut:

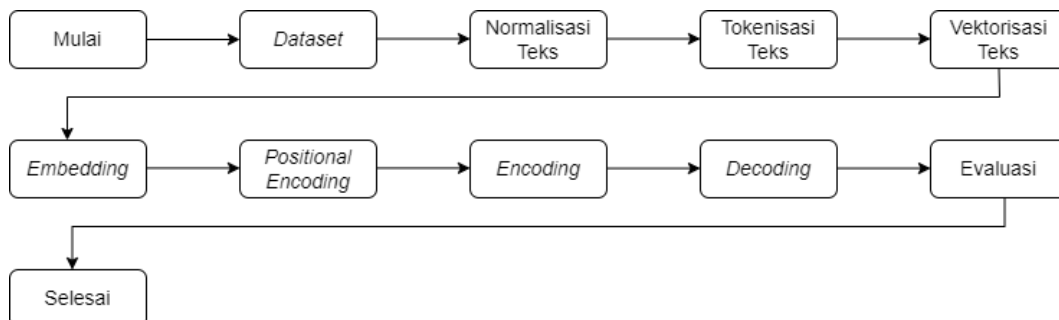
- *Processor* : *Intel® Core™ i3-6006U*
- *Ram* : 4 GB
- *Gpu* : *Intel(R) HD Graphics 520*
- *Penyimpanan* : HDD 1 TB

3.2.2 Perangkat Lunak

Perangkat Lunak yang digunakan adalah sebagai berikut:

- Windows 10 (64 bit)
- Python 3 digunakan sebagai bahasa pemrograman
- Google Colab digunakan untuk membuat *script* program, adapun spesifikasi google colab yang digunakan sebagai berikut:
 - ❖ *Runtime Type* : Python 3
 - ❖ *Hardware Accelerator* : T4 GPU

3.3 Tahapan Penelitian



Gambar 5. Tahapan Penelitian.

Berdasarkan pada Gambar 5, tahapan penelitian ini terdapat sembilan tahapan, yaitu *Dataset*, *Normalisasi*, *Tokenisasi*, *Vektorisasi*, *Embedding*, *Positional Encoding*, *Encoding*, *Decoding* dan *Evaluasi*.

3.3.1 Dataset

Tahapan pertama dalam penelitian ini adalah mengumpulkan bahan utama penelitian yaitu berupa *dataset*. *Dataset* yang digunakan diambil dari situs web manythings (<https://www.manythings.org/anki/>). Jumlah data yang digunakan yaitu berjumlah 11.933 data. *Dataset* yang digunakan terdiri dari pasangan kata dan kalimat yang terdiri dari kata-kata atau kalimat-kalimat dalam bahasa sumber yaitu bahasa Lampung dialek A dan bahasa target yaitu bahasa Indonesia yang sudah diterjemahkan secara manual sebelumnya. *Dataset* ini juga nantinya akan dibagi menjadi tiga bagian dengan perbandingan 70%:15%:15%, yaitu:

a) Data Training

Data *Training* merupakan data yang digunakan untuk melatih model *Transformer*, agar model dapat belajar menerjemahkan bahasa sumber yaitu bahasa Lampung ke bahasa target yaitu bahasa Indonesia. Jumlah data *training* yang digunakan, yaitu 8.353,1 data pasangan kata dan kalimat.

b) Data Testing

Data *Testing* merupakan data yang digunakan oleh model untuk menguji seberapa baik kinerja model *Transformer* untuk dapat melakukan penerjemahan dari bahasa sumber ke bahasa target pada data yang belum pernah dilihat oleh model sebelumnya. Jumlah data *testing* yang digunakan, yaitu 1.789,95 data pasangan kata dan kalimat.

c) Data Validation

Data *Validation* merupakan data yang digunakan untuk mengoptimalkan performa model selama pelatihan dengan mengukur metrik-metrik seperti *accuracy*, *loss*, atau skor BLEU. Jumlah data *validation* yang digunakan, yaitu 1.789,95 data pasangan kata dan kalimat.

Berikut adalah contoh *dataset* yang akan digunakan pada penelitian ini yang dapat dilihat pada Tabel 2.

Tabel 2. Dataset bahasa Lampung – Indonesia

	Bahasa Lampung	Bahasa Indonesia
1	Hai	Hai
2	Tegagh	Lari!
3	Tegar!	Lari!
....		
49	Ngapi haghush nyak?	Kenapa harus saya?

50	Kughuk pai	Masuklah.
...		
454	Sapa sai nyawako heno?	Siapa yang mengatakan itu?
455	Heno jak sapa?	Itu punya siapa?
...		
4479	Nyak ghadu mak ngeghintokni.	Aku sudah tidak menginginkannya.
4480	Nyak pagun kung haga mati.	Aku masih belum mau mati.
...		
5190	Keghitani dijadwalko guwai tigoh pada pukul nom.	Keretanya dijadwalkan untuk tiba pada pukul enam.
5191	Tiyan dibayagh per hari.	Mereka dibayar per hari.
...		
8237	Teghjemahko buku sinji mit lom bahasa Pghancis.	Terjemahkan buku ini ke dalam bahasa Prancis.
8238	Lamun mak teghai, nyak bakal iluk lapah.	Kalau tidak hujan, aku akan ikut pergi.
...		
11931	Januaghi, Pebghuaghi, Maghet, Apghil, Mei, Juni, Juli, Agustus, Septembegh, Oktobegh, Nopembegh, Desembegh iyulah wa belas bulan lom setahun.	Januari, Februari, Maret, April, Mei, Juni, Juli, Agustus, September, Oktober, November, dan Desember adalah dua belas bulan dalam setahun.
11932	Irene Pepperberg, seulun peneliti di Univeghsitas Northwestern ngepehaluko bahwa putik beo mak Cuma dapok ngenighuko pecawaan manusiya kidang juga dapok ngemahhomi aghti anjak pecawaan tesebut.	Irene Pepperberg, seorang peneliti di Universitas Northwestern menemukan bahwa burung beo tidak hanya bisa menirukan perkataan manusia tetapi juga bisa memahami arti dari perkataan tersebut.
11933	Ki seseulun mak bekesempatan guwai nguasai bahasa sai di hagakoni waktu ngilik dewasa, mula lunik kehalokan ia bakal dapok tigoh tingkatan penutugh asli lom bahasa tesebut.	Jika seseorang tidak berkesempatan untuk menguasai bahasa yang diinginkannya ketika menginjak dewasa, maka kecil kemungkinan ia akan bisa mencapai tingkatan penutur asli dalam bahasa tersebut.

3.3.2 Normalisasi

Normalisasi teks merupakan proses untuk mengubah teks yang masih mentah menjadi bentuk teks yang seragam agar dapat diproses oleh model dengan mudah. Normalisasi teks mencakup berbagai langkah yang bertujuan untuk mengurangi variasi dalam teks yang tidak sesuai, sehingga memudahkan proses pemodelan dan

analisis. Tahapan normalisasi teks ini dapat meliputi penghapusan spasi kosong, mengubah semua teks menjadi huruf kecil dan normalisasi lainnya. Proses normalisasi teks ini dapat membuat data yang akan digunakan menjadi lebih bersih dan konsisten, sehingga dapat meningkatkan akurasi model.

3.3.3 Tokenisasi

Tokenisasi adalah tahapan dimana teks dipecah menjadi unit-unit kecil yang dinamakan dengan token. Proses ini bertujuan untuk memudahkan model dalam memahami dan mengolah teks. Token ini dapat berupa kata, frasa, atau karakter sesuai dengan kebutuhan analisis. Melalui proses tokenisasi ini, model dapat mengenali pola dan hubungan pada teks dengan lebih mudah.

3.3.4 Vektorisasi

Vektorisasi adalah proses untuk mengubah teks menjadi format numerik yang dapat dipahami dan diolah oleh model. Vektorisasi teks memungkinkan representasi kata-kata, kalimat, atau dokumen dalam bentuk vektor sehingga dapat dianalisis oleh model karena model memerlukan input dalam bentuk angka. Proses ini sangat penting dalam *deep learning* dan *natural language processing*, karena dapat membuat model memungkinkan untuk mendeteksi pola serta hubungan dalam teks atau kalimat.

3.3.5 *Embedding*

Pada tahap ini, token-token yang telah dihasilkan dari tahapan sebelumnya akan diubah menjadi representasi numerik dalam bentuk vektor. Representasi ini akan membantu model untuk memahami makna dari teks yang telah diinputkan dan menghasilkan terjemahan yang lebih akurat. Proses *embedding* ini bertujuan agar model dapat memproses teks dengan lebih efektif serta menghasilkan *output* yang lebih akurat.

3.3.6 *Positional Encoding*

Pada tahap ini, vektor-vektor yang telah dihasilkan dalam tahapan *embedding* akan digabungkan dengan vektor posisi. Tahapan ini memiliki tujuan untuk memberikan

informasi posisi pada setiap kata dalam kalimat yang berguna untuk mengatasi masalah dalam menghitung hubungan antara setiap kata dalam kalimat. Proses menambahkan vektor posisi ini dapat membantu model dalam memahami urutan kata, sehingga dapat memahami makna kontekstual dengan lebih tepat. Hal tersebut sangat penting pada arsitektur *transformer* yang memerlukan informasi posisi atau urutan untuk pemrosesan teks yang lebih efektif.

3.3.7 *Encoding*

Pada tahap ini, vektor-vektor yang telah dihasilkan dari tahapan *embedding* akan melewati blok *encoder*. Blok *encoder* terdiri dari dua sub-blok, yaitu *multi-head attention* dan *feed-forward*. Gabungan dari kedua sub-blok ini memungkinkan model untuk memahami teks dengan lebih baik serta komprehensif.

3.3.7.1 *Multi-Head Attention*

Pada tahap ini, model akan melakukan *multi-head attention* untuk setiap vektor *embedding* untuk menghasilkan representasi konteks dari seluruh teks yang telah diinputkan. Kemudian, representasi konteks ini akan memungkinkan model untuk mempertimbangkan seluruh konteks dalam teks sumber, sehingga dapat menghasilkan representasi yang lebih baik.

3.3.7.2 *Feed-Forward*

Setelah dilakukan *multi-head attention*, model akan memproses representasi konteks tersebut melalui jaringan *feed-forward* untuk menghasilkan representasi akhir dari teks sumber. Representasi akhir ini akan digunakan pada tahap *decoding* untuk menghasilkan terjemahan.

3.3.8 *Decoding*

Setelah melewati tahap *encoding*, vektor hasil representasi dijadikan input bagi blok decoder. Tahapan *decoding* terdiri dari tiga bagian sub-blok *decoder*, yaitu *masked multi-head attention*, *multi-head attention*, dan *feed-forward*.

3.3.8.1 *Masked Multi-Head Attention*

Pada tahap ini, model akan melakukan *masked multi-head attention* pada representasi vektor teks sumber dan representasi vektor token awal teks terjemahan. *Masked multi-head attention* diperlukan untuk memastikan bahwa model hanya melihat token yang telah dihasilkan pada saat itu saja dan tidak melihat token yang belum dihasilkan pada tahap *decoding*.

3.3.8.2 *Multi-Head Attention*

Setelah *masked multi-head attention*, model akan melanjutkan proses selanjutnya, yaitu *multi-head attention* untuk representasi konteks dari teks sumber dan representasi vektor dari token awal terjemahan yang telah diproses melalui jaringan *feed-forward* pada tahap *encoding*. Proses ini juga bertujuan untuk mengintegrasikan informasi dari blok *encoder*.

3.3.8.3 *Feed-Forward*

Setelah dilakukan *multi-head attention*, model akan memproses representasi konteks tersebut melalui jaringan *feed-forward* untuk menghasilkan representasi akhir dari token terjemahan pada tahap tersebut. Token-token tersebut kemudian diubah kembali menjadi kalimat-kalimat dalam bahasa target.

3.3.9 Evaluasi

Evaluasi merupakan tahapan terakhir yang dilakukan untuk menilai atau mengevaluasi model, setelah melakukan pelatihan dan pengujian pada model. Pada penelitian ini, evaluasi model akan dilakukan dengan menggunakan metrik evaluasi, yaitu metrik skor *Bilingual Evaluation Understudy* (BLEU). Skor BLEU adalah salah satu metrik yang sering digunakan untuk mengevaluasi kualitas dari terjemahan mesin. Skor BLEU mengukur akurasi terjemahan dengan cara membandingkan n-gram (kelompok kata berurutan dari n kata) dalam hasil terjemahan mesin dengan n-gram pada terjemahan referensi. Metrik ini dipilih karena metrik ini merupakan metrik paling populer dalam *machine translation* yang sudah digunakan sejak 15 tahun terakhir.

Adapun rumus untuk menghitung skor BLEU adalah sebagai berikut:

$$BP = \begin{cases} 1, & \text{if } c > r \\ \exp^{(1-\frac{r}{c})}, & \text{if } c \leq r \end{cases}$$

$$BLEU = BP \cdot \exp(\sum_{n=1}^n w_n \log p_n)$$

Keterangan:

BP = *Brevity Penalty*

C = Jumlah kata dari hasil terjemahan otomatis (*candidate*)

r = Jumlah kata rujukan (*reference*)

w_n = 1/N (standar nilai N untuk BLEU adalah 4)

p_n = Jumlah n-gram hasil terjemahan yang sesuai dengan rujukan dibagi jumlah n-gram hasil terjemahan

V. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Kesimpulan yang didapatkan berdasarkan hasil penelitian yang telah dilakukan adalah sebagai berikut:

1. Arsitektur *Transformer* dapat diimplementasikan dalam penerjemahan bahasa Lampung ke bahasa Indonesia melalui beberapa proses, yaitu persiapan *dataset*, normalisasi, tokenisasi, vektorisasi, *embedding*, *positional encoding*, *encoding*, *decoding* dan evaluasi.
2. Hasil terjemahan yang dihasilkan oleh Arsitektur *Transformer* ini diukur dengan menggunakan metrik evaluasi *Bilingual Evaluation Understudy* (BLEU). Skor BLEU {1, 2, 3, 4} masing-masing nilainya yaitu {91,66%, 81,81%, 70%, 55,55%}, yang berarti terjemahan mesin masih cukup baik dalam menjaga konteks dalam teks yang cukup panjang.

5.2 Saran

Adapun saran yang diberikan oleh penulis yang dapat dilakukan untuk penelitian selanjutnya, yaitu:

1. Mengimplementasikan model dengan menggunakan data dari bahasa daerah lainnya yang ada di Indonesia.
2. Mengatasi *overfitting* dengan menambahkan data *training* lebih banyak lagi atau dengan cara lainnya agar performa model lebih baik.
3. Membangun *user interface* untuk mengembangkan model dalam bentuk web atau aplikasi.

DAFTAR PUSTAKA

DAFTAR PUSTAKA

- Abidin, Z. (2017). Penerapan Neural Machine Translation untuk Eksperimen Penerjemahan secara Otomatis pada Bahasa Lampung – Indonesia. *Prosiding Seminar Nasional Metode Kuantitatif*, 978, 53–68.
- Abidin, Z., & Permata, P. (2021). Pengaruh Penambahan Korpus Paralel Pada Mesin Penerjemah Statistik Bahasa Indonesia Ke Bahasa Lampung Dialek Nyo. *Jurnal Teknoinfo*, 15(1), 13. <https://doi.org/10.33365/jti.v15i1.889>
- Alfarizi, M. R. S., Al-farish, M. Z., Taufiqurrahman, M., Ardiansah, G., & Elgar, M. (2023). Penggunaan Python Sebagai Bahasa Pemrograman Untuk Machine Learning Dan Deep Learning. *Karimah Tauhid*, 2(1), 1–6.
- Amien, M. (2023). *Sejarah dan Perkembangan Teknik Natural Language Processing (NLP) Bahasa Indonesia: Tinjauan tentang sejarah, perkembangan teknologi, dan aplikasi NLP dalam bahasa Indonesia*. 2007, 1–7. <http://arxiv.org/abs/2304.02746>
- Ardi, H. (2015). *Pengantar penerjemahan (Introduction to translation)*. SUKABINA PRESS.
- Bisong, E. (2019). *Building Machine Learning and Deep Learning Models on Google Cloud Platform*. Apress, Berkeley, CA. https://doi.org/https://doi.org/10.1007/978-1-4842-4470-8_7
- Carneiro, T., Da Nobrega, R. V. M., Nepomuceno, T., Bian, G. Bin, De Albuquerque, V. H. C., & Filho, P. P. R. (2018). Performance Analysis of Google Colaboratory as a Tool for Accelerating Deep Learning Applications. *IEEE Access*, 6, 61677–61685. <https://doi.org/10.1109/ACCESS.2018.2874767>
- Dhiman, P., Kukreja, V., Manoharan, P., Kaur, A., & Kamruzzaman, M. M. (2022). *A Novel Deep Learning Model for Detection of Severity Level of the Disease*

in Citrus Fruits. 1–14.

- Glaz, A. Le, Haralambous, Y., Lenca, P., & Billot, R. (2021). *Machine Learning and Natural Language Processing in Mental Health : Systematic Review Corresponding Author : 23*. <https://doi.org/10.2196/15708>
- Hartono, H. (2016). *PERANAN MULOK BAHASA LAMPUNG DALAM UPAYA PELESTARIAN BAHASA DAN BUDAYA LAMPUNG (Studi Kasus di SMP Negeri 20 Bandar Lampung Tahun Pelajaran 2015/2016)*.
- Jawahar, G., Nagoudi, E. M. B., Abdul-Mageed, M., & Lakshmanan, L. V. S. (2021). Exploring Text-to-Text Transformers for English to Hinglish Machine Translation with Synthetic Code-Mixing. *Computational Approaches to Linguistic Code-Switching, CALCS 2021 - Proceedings of the 5th Workshop*, 5, 36–46. https://doi.org/10.26615/978-954-452-056-4_006
- Lee, S., Lee, J., Moon, H., Park, C., Seo, J., Eo, S., Koo, S., & Lim, H. (2023). A Survey on Evaluation Metrics for Machine Translation. *Mathematics*, 11(4), 1–22. <https://doi.org/10.3390/math11041006>
- Lutz, M. (2021). *Learning Python* (J. Steele (ed.)). O'Reilly Media, Inc.
- Prihantika, I., Utoyob, B., Rahmadhanic, T. P., & Sutiyo. (2019). Seminar nasional fisp unila (sefila) - 3 tahun 2019. *Partisipasi Masyarakat Dalam Tata Kelola Industri Pariwisata a (Studi Terhadap Partisipasi Masyarakat Dalam Pengembangan Objek Wisata Di Lampung Selatan)*.
- Primandhika, R. B., Munawar, M. N., & Saifullah, A. R. (2021). Experiment on a Transformer Model Indonesian-to-Sundanese Neural Machine Translation with Sundanese Speech Level Evaluation. *Proceedings of the Thirteenth Conference on Applied Linguistics (CONAPLIN 2020)*, 546(Conaplin 2020), 452–459. <https://doi.org/10.2991/assehr.k.210427.069>
- Project, T. (2023). *Tab-delimited Bilingual Sentence Pairs*. <https://www.manythings.org/anki/>, diakses pada tanggal 15 Juli 2023
- Putri, N. W. (2021). Fenomena Keberagaman Bahasa Di Kota Bandar Lampung. *PRASASTI: Journal of Linguistics*, 6(1), 106. <https://doi.org/10.20961/prasasti.v6i1.48742>
- Raup, A., Ridwan, W., Khoeriyah, Y., Supiana, S., & Zaqiah, Q. Y. (2022). Deep Learning dan Penerapannya dalam Pembelajaran. *JlIP - Jurnal Ilmiah Ilmu*

- Pendidikan*, 5(9), 3258–3267. <https://doi.org/10.54371/jiip.v5i9.805>
- Sharma, S., & Chaudhary, P. (2023). Machine learning and deep learning. *Quantum Computing and Artificial Intelligence: Training Machine and Deep Learning Algorithms on Quantum Computers*, 71–84. <https://doi.org/10.1515/9783110791402-004>
- Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15, 1929–1958.
- Tang, G., Müller, M., Rios, A., & Sennrich, R. (2018). Why self-attention? A targeted evaluation of neural machine translation architectures. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018*, 4263–4272. <https://doi.org/10.18653/v1/d18-1458>
- Tian, T., Song, C., Ting, J., & Huang, H. (2021). A French-to-English Machine Translation Model Using Transformer Network. *Procedia Computer Science*, 199, 1438–1443. <https://doi.org/10.1016/j.procs.2022.01.182>
- Tran, N., Tran, H., Nguyen, S., Nguyen, H., & Nguyen, T. (2019). Does BLEU score work for code migration? *IEEE International Conference on Program Comprehension*, 2019-May, 165–176. <https://doi.org/10.1109/ICPC.2019.00034>
- Vashee, K. (2019). *Understanding machine translation Quality: BLEU scores*. RWS. <https://www.rws.com/blog/understanding-mt-quality-bleu-scores/>, diakses pada tanggal 20 April 2023
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems, 2017-Decem(Nips)*, 5999–6009.
- Wang, Q., Li, B., Xiao, T., Zhu, J., Li, C., Wong, D. F., & Chao, L. S. (2020). Learning deep transformer models for machine translation. *ACL 2019 - 57th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference*, 1810–1822. <https://doi.org/10.18653/v1/p19-1176>
- Weng, L. (2018). *Attention? Attention!* <https://lilianweng.github.io/posts/2018-06-24-attention/>, diakses pada tanggal 20 April 2023

- Wismoyo, P. A. (2018). Mesin penerjemah bahasa inggris – indonesia berbasis jaringan saraf tiruan dengan mekanisme attention menggunakan arsitektur transformer skripsi. *Undip E-Journal System*.
- Wuryantoro, A. (2018). *Pengantar Penerjemahan*. Deepublish.
- Yin, K. (2020). Sign Language Translation With Transformers. *ArXiv Preprint ArXiv*.
- Ying, X. (2019). An Overview of Overfitting and its Solutions. *Journal of Physics: Conference Series*, 1168(2). <https://doi.org/10.1088/1742-6596/1168/2/022022>