PENGARUH *NAMED ENTITY RECOGNITION* DAN AUGMENTASI DATA TERHADAP KINERJA MODEL *DEEP LEARNING* DALAM KLASIFIKASI TEKS

Skripsi

Oleh

NUR ANNISA PUTRI REZKIA
NPM. 2217031173

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2026

ABSTRACT

THE IMPACT OF NAMED ENTITY RECOGNITION AND DATA AUGMENTATION ON THE PERFORMANCE OF DEEP LEARNING MODELS IN TEXT CLASSIFICATION

By

Nur Annisa Putri Rezkia

This study aims to classify news texts from the InaCOVED dataset into event and non-event categories using MLP, CNN, and LSTM models. Synonym-based data augmentation through the Kateglo API was applied to improve class balance, while NER based on BERT and an LLM with a prompting approach was used to extract key entities (Person, Organization, Location, and Disease). Text representations were generated using Word2Vec and the Keras Embedding Layer and evaluated using accuracy, precision, recall, F1-score, and ROC-AUC metrics. The best performance was achieved by the CNN model with the Keras Embedding Layer and LLM-based NER using 100-dimensional vectors, obtaining an accuracy of 0.9597, precision of 0.9442, recall of 0.9772, F1-score of 0.9604, and ROC-AUC of 0.9888, indicating that entity extraction methods, vector representation, and data augmentation significantly influence classification performance.

Keywords: NER, LLM, BERT, Data Augmentation, Word Embedding, Text Classification.

ABSTRAK

PENGARUH *NAMED ENTITY RECOGNITION* DAN AUGMENTASI DATA TERHADAP KINERJA MODEL *DEEP LEARNING* DALAM KLASIFIKASI TEKS

Oleh

Nur Annisa Putri Rezkia

Penelitian ini bertujuan mengklasifikasikan teks berita pada dataset InaCOVED ke dalam kategori *event* dan *non-event* menggunakan model MLP, CNN, dan LSTM. Augmentasi data berbasis penggantian sinonim melalui API Kateglo diterapkan untuk meningkatkan keseimbangan kelas, sementara NER berbasis BERT dan LLM dengan pendekatan *prompting* digunakan untuk mengekstraksi entitas penting (*Person*, *Organization*, *Location*, dan *Disease*). Representasi teks dibentuk menggunakan Word2Vec dan Keras *Embedding* Layer, kemudian dievaluasi dengan metrik *accuracy*, *precision*, *recall*, *F1-score*, dan ROC-AUC. Hasil terbaik diperoleh pada model CNN dengan Keras *Embedding* Layer dan NER berbasis LLM pada dimensi vektor 100 dengan *accuracy* 0,9597, *precision* 0,9442, *recall* 0,9772, *F1-score* 0,9604, dan ROC-AUC 0,9888, yang menunjukkan bahwa metode ekstraksi entitas, representasi vektor, dan augmentasi data berpengaruh signifikan terhadap kinerja klasifikasi.

Kata-kata kunci: NER, LLM, BERT, Augmentasi Data, *Word Embedding*, Klasifikasi Teks.

PENGARUH *NAMED ENTITY RECOGNITION* DAN AUGMENTASI DATA TERHADAP KINERJA MODEL *DEEP LEARNING* DALAM KLASIFIKASI TEKS

NUR ANNISA PUTRI REZKIA

Skripsi

Sebagai Salah Satu Syarat untuk Memperoleh Gelar
SARJANA MATEMATIKA

Pada

Jurusan Matematika

Fakultas Matematika dan Ilmu Pengetahuan Alam

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2026

Judul Skripsi	:	**PENGARUH *NAMED ENTITY RECOGNITION* DAN AUGMENTASI DATA TERHADAP KINERJA MODEL *DEEP LEARNING* DALAM KLASIFIKASI TEKS**
Nama Mahasiswa	:	Nur Annisa Putri Rezkia
Nomor Pokok Mahasiswa	:	**2217031173**
Program Studi	:	**Matematika**
Fakultas	:	**Matematika dan Ilmu Pengetahuan Alam**

MENYETUJUI

1. Komisi Pembimbing

Dr.Aang Nuryaman, S.Si., M.Si.
NIP.197403162005011001

Dr.Purnomo Husnul Khotimah, M.T.
NIP 198003232005022002

2. Ketua Jurusan Matematika

Dr.Aang Nuryaman, S.Si., M.Si.
NIP. 197403162005011001

MENGESAHKAN

1. Tim Penguji

Ketua : **Dr.Aang Nuryaman, S.Si., M.Si.** .....

Sekretaris : **Dr. Purnomo Husnul Khotimah, M.T.** .....

Penguji

Bukan Pembimbing : **Dr. Khoirin Nisa, S.Si., M.Si.** .....

2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam

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

NIP. 197110012005011002

Tanggal Lulus Ujian Skripsi: **13 Maret 2026**

PERNYATAAN SKRIPSI MAHASISWA

Yang bertanda tangan di bawah ini:

Nama	:	**Nur Annisa Putri Rezkia**
Nomor Pokok Mahasiswa	:	**2217031173**
Jurusan	:	**Matematika**
Judul Skripsi	:	**Pengaruh *Named Entity Recognition* dan Augmentasi Data terhadap Kinerja Model *Deep Learning* dalam Klasifikasi Teks**

Dengan ini menyatakan bahwa skripsi ini adalah hasil pekerjaan saya sendiri. Apabila kemudian hari terbukti bahwa skripsi ini merupakan hasil salinan atau dibuat oleh orang lain, maka saya bersedia menerima sanksi sesuai dengan ketentuan akademik yang berlaku.

Bandar Lampung,

Penulis,

Nur Annisa Putri Rezkia

RIWAYAT HIDUP

Penulis bernama Nur Annisa Putri Rezkia, lahir di Kasui pada tanggal 9 Desember 2003. Penulis merupakan anak ketiga dari empat bersaudara yang tumbuh dalam lingkungan keluarga yang menanamkan nilai kerja keras, tanggung jawab, dan semangat untuk terus belajar.

Pendidikan formal penulis diawali di TK Melati Penumangan Baru, kemudian melanjutkan pendidikan dasar di SD Negeri 1 Penumangan Baru, pendidikan menengah pertama di SMP Bina Desa, dan pendidikan menengah atas di SMA Negeri 1 Tulang Bawang Tengah. Setelah menyelesaikan pendidikan menengah atas, penulis melanjutkan studi di Jurusan Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam (FMIPA), Universitas Lampung melalui jalur Seleksi Bersama Masuk Perguruan Tinggi Negeri (SBMPTN).

Selama menempuh pendidikan di perguruan tinggi, penulis tidak hanya berfokus pada kegiatan akademik, tetapi juga aktif dalam berbagai kegiatan organisasi dan sosial. Penulis tergabung dalam Himpunan Mahasiswa Matematika (HIMATIKA) sebagai bentuk kontribusi dalam pengembangan organisasi kemahasiswaan. Selain itu, penulis turut serta dalam kegiatan kerelawanan di Labuan Bajo, Pulau Boleng, serta aktif dalam program berbagi melalui kegiatan Hibahin sebagai wujud kepedulian terhadap sesama.

Dalam bidang pengembangan diri, penulis mengikuti ajang Putri Hijab Lampung 2024 dan berhasil terpilih sebagai Top 25 Putri Hijab Lampung 2024, yang menjadi pengalaman berharga dalam meningkatkan kepercayaan diri, kemampuan komunikasi, dan kepemimpinan.

Sebagai bentuk penguatan kompetensi akademik dan penelitian, penulis melaksanakan Kerja Praktik di Badan Riset dan Inovasi Nasional (BRIN) Bandung, KST Samaun Samadikun selama 40 hari. Selanjutnya, penulis mengikuti Program Penelitian MBKM di BRIN Bandung, KST Samaun Samadikun selama 6 bulan, yang memberikan pengalaman langsung dalam kegiatan riset serta memperluas wawasan keilmuan di bidang statistika dan analisis data.

Selama masa studi, penulis senantiasa berusaha menunjukkan ketekunan, disiplin, dan dedikasi dalam menyelesaikan berbagai tanggung jawab akademik. Penulis berharap hasil penelitian yang telah dilakukan dapat memberikan kontribusi nyata bagi pengembangan ilmu pengetahuan, khususnya di bidang Statistika, serta menjadi langkah awal untuk terus berkembang dan memberikan manfaat bagi masyarakat.

KATA INSPIRASI

”Pendidikan adalah senjata paling ampuh yang dapat kamu gunakan untuk mengubah dunia”
-Nelson Mandela

”Live as if you were to die tomorrow. Learn as if you were to live forever”
-Mahatma Gandhi

”Ilmu tanpa amal adalah kesia-siaan, dan amal tanpa ilmu adalah kesesatan”
-Imam Al-Ghazali

”Don’t watch the clock; do what it does. Keep going”
-Sam Levenson

”Kesuksesan adalah hasil dari persiapan, kerja keras, dan belajar dari kegagalan”
-Colin Powell

”It always seems impossible until it’s done”
-Nelson Mandela

”Success is not final, failure is not fatal: it is the courage to continue that counts”
-Winston Churchill

”Masa depan adalah milik mereka yang percaya pada keindahan mimpi-mimpinya”
-Eleanor Roosevelt

”Dream big and dare to fail”
-Norman Vaughan

”Sebaik-baik manusia adalah yang paling bermanfaat bagi manusia lainnya”
-HR. Ahmad

”Hidup bukan tentang menemukan diri sendiri, tetapi tentang membangun diri sendiri”
-George Bernard Shaw

”Maka sesungguhnya bersama kesulitan ada kemudahan”
-QS. Al-Insyirah: 6

”Tawakal bukan berarti berhenti berusaha, tetapi percaya bahwa usaha terbaik akan diberi jalan terbaik”
-Ali bin Abi Thalib

”The beautiful thing about learning is that no one can take it away from you”
-B.B. King

PERSEMBAHAN

Alhadulillahirobbil”alamin
Dengan mengucapkan puji dan syukur atas kehadirat Allah Subhanahu Wata”ala karena limpahan rahmat dan karunia-Nya sehingga skripsi ini dapat terselesaikan dengan baik dan tepat pada waktunya.
Tak lupa shalawat beserta salam selalu tercurah kepada junjungan kita Nabi Muhammad Shallallahu Alaihi Wasallam.
Dengan rasa syukur dan Bahagia, saya persembahkan rasa terimakasih saya kepada:

Bapak dan Ibuku Tercinta

Terimakasih kepada orang tuaku atas segala pengorbanan, motivasi, doa dan ridho serta dukungannya selama ini. Terimakasih telah memberikan pelajaran berharga kepada anakmu ini tentang makna perjalanan hidup yang sebenarnya sehingga kelak bisa menjadi orang yang bermanfaat bagi banyak orang.

Dosen Pembimbing dan Pembahas

Terimakasih kepada dosen pembimbing dan pembahas yang sudah sangat membantu, memberikan motivasi, memberikan arahan serta ilmu yang berharga.

Sahabat-sahabatku

Terimakasih kepada semua orang-orang baik yang telah memberikan pengalaman, semangat, motivasinya, serta doa-doanya dan senantiasa memberikan dukungan dalam hal apapun.

Almamater Tercinta
Universitas Lampung

SANWACANA

Alhamdulillah, puji dan syukur penulis panjatkan kepada Allah SWT atas limpahan rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan skripsi ini yang berjudul ”Pengaruh *Named Entity Recognition* dan Augmentasi Data Terhadap Kinerja Model *Deep Learning* dalam Klasifikasi Teks” dengan baik dan lancar serta tepat pada waktu yang telah ditentukan. Shalawat serta salam semoga senantiasa tercurahkan kepada Nabi Muhammad SAW.

Dalam proses penyusunan skripsi ini, banyak pihak yang telah membantu memberikan bimbingan, dukungan, arahan, motivasi serta saran sehingga skripsi ini dapat terselesaikan. Oleh karena itu, dalam kesempatan ini penulis mengucapkan terimakasih kepada:

1. Diri sendiri, atas segala usaha, ketekunan, dan perjuangan yang telah dilakukan untuk menyelesaikan skripsi ini tepat pada waktunya. Terima kasih telah bertahan di tengah tantangan dan terus maju meski dihadapkan pada berbagai rintangan.
2. Bapak Dr. Aang Nuryaman, S.Si.,M.Si. selaku Pembimbing 1 yang telah banyak meluangkan waktunya untuk memberikan arahan, bimbingan, motivasi, saran serta dukungan kepada penulis sepanjang proses penyusunan skripsi ini.
3. Ibu Dr. Purnomo Husnul Khotimah, M.T. selaku Pembimbing 2 yang telah memberikan arahan, dukungan, serta doa sehingga penulis dapat menyelesaikan skripsi ini.
4. Ibu Dr. Khoirin Nisa, S.Si., M.Si. selaku Penguji yang telah bersedia memberikan saran, kritik, serta evaluasi yang membangun sehingga penulis dapat menyelesaikan skripsi ini.
5. Ibu Dina Eka Nurvazly, S.Pd., M.Si. selaku dosen Pembimbing Akademik.

6. Bapak Dr. Aang Nuryaman, S.Si., M.Si. selaku Ketua Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.

7. Bapak Andri Fachrur Rozie S.Kom., M.Eng. selaku salah satu pembimbing magang MBKM dari Pusat Riset Sains Data dan Informasi.

8. Seluruh dosen, staff dan karyawan Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.

9. Penulis menyampaikan rasa syukur yang mendalam serta terima kasih yang sebesar-besarnya kepada kedua orang tua tercinta, Mama dan Papa, atas segala doa, kasih sayang, dukungan moral maupun material, serta pengorbanan yang tiada henti diberikan kepada penulis. Setiap langkah dan pencapaian penulis hingga berada di titik ini tidak terlepas dari restu, bimbingan, dan ketulusan cinta yang selalu mengiringi. Ucapan terima kasih juga penulis sampaikan kepada abang, kakak, ayuk, teteh, dan ayi yang senantiasa memberikan semangat, dukungan, serta motivasi dalam setiap proses yang penulis jalani. Kehadiran dan dukungan keluarga menjadi sumber kekuatan terbesar bagi penulis dalam menyelesaikan pendidikan ini.

10. Penulis juga menyampaikan terima kasih kepada Sony Afandi, yang senantiasa memberikan doa, dukungan, dan semangat selama proses penyusunan skripsi ini. Terima kasih atas kehadiran yang selalu membersamai, terutama di saat penulis merasa lelah dan berada pada titik terendah. Dukungan, perhatian, dan motivasi yang diberikan menjadi penguat bagi penulis untuk bangkit dan menyelesaikan penelitian ini dengan sebaik-baiknya.

11. Penulis juga menyampaikan terima kasih yang tulus kepada sahabat tercinta, Ira, yang senantiasa hadir memberikan dukungan, semangat, dan doa dalam setiap proses yang dilalui penulis. Terima kasih atas kebersamaan, perhatian, serta motivasi yang diberikan, terutama di saat-saat penulis merasa lelah dan hampir menyerah. Kehadiran dan dukunganmu menjadi penguat bagi penulis untuk terus melangkah hingga akhirnya dapat menyelesaikan skripsi ini.

12. Penulis juga mengucapkan terima kasih kepada sahabat-sahabat semasa SMA, yaitu Mellysa, Ayu, Nindi, Tera, Fella, Dhena, Rina, dan Yudis, yang senantiasa memberikan dukungan, semangat, serta doa kepada penulis. Terima kasih atas kebersamaan, perhatian, dan motivasi yang terus diberikan meskipun jarak dan waktu memisahkan. Kehadiran kalian menjadi salah satu penyemangat bagi penulis dalam menyelesaikan pendidikan ini.

13. Penulis juga menyampaikan terima kasih kepada teman-teman perkuliahan, yaitu Erin, Nazla, Anita, Fatur, dan Oja, atas kebersamaan, dukungan, serta semangat yang diberikan selama menjalani masa perkuliahan. Terima kasih atas kerja sama, diskusi, dan motivasi yang turut membantu penulis dalam menyelesaikan studi ini. Ucapan terima kasih juga penulis sampaikan kepada teman-teman kader yang telah menjadi bagian dari perjalanan dan proses pembelajaran selama di bangku perkuliahan. Kebersamaan, pengalaman, serta dukungan yang diberikan menjadi kenangan dan pembelajaran berharga bagi penulis.

Semoga skripsi ini dapat bermanfaat bagi kita semua. Penulis menyadari bahwa skripsi ini masih jauh dari sempurna, sehingga penulis mengharapkan kritik dan saran yang membangun untuk menjadikan skripsi ini lebih baik lagi.

Bandar Lampung,

Nur Annisa Putri Rezkia

DAFTAR ISI

DAFTAR TABEL

DAFTAR GAMBAR

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

COVID-19 telah menciptakan lonjakan informasi yang masif melalui berbagai kanal seperti berita daring, media sosial, hingga publikasi ilmiah. Informasi ini mencakup laporan kasus, kebijakan pemerintah, hingga opini masyarakat yang bercampur antara fakta dan *noise*. Kondisi tersebut menimbulkan tantangan dalam memilah data yang relevan untuk mendukung pemantauan situasi serta pengambilan keputusan. Dalam konteks ini, *Natural Language Processing* (NLP) memainkan peran vital sebagai teknologi untuk mengolah data teks tidak terstruktur agar dapat diubah menjadi pengetahuan yang bermanfaat (Cambria and White, 2014).

Klasifikasi teks menjadi salah satu aspek mendasar dalam NLP, yang berfungsi untuk memisahkan informasi ke dalam kategori tertentu. Pada domain kesehatan, klasifikasi teks dapat digunakan untuk membedakan antara informasi yang terkait dengan peristiwa penting (*event*) dan yang tidak relevan (*non-event*). Dengan adanya sistem klasifikasi ini, informasi dapat lebih mudah dimanfaatkan oleh lembaga kesehatan maupun pemerintah dalam menyusun strategi respons pandemi. Penelitian sebelumnya menunjukkan bahwa klasifikasi otomatis dapat membantu moderator komunitas kesehatan daring untuk mengidentifikasi diskusi yang penting (Huh et al., 2013) dan mempercepat identifikasi gejala atau alasan tes COVID-19 dari catatan medis tidak terstruktur (Abu Lekham et al., 2022).

Named Entity Recognition (NER) menempati posisi krusial sebagai teknik untuk mengekstraksi entitas penting dari teks, seperti *Person*, *Organization*, *Location*, dan *Disease*. NER telah banyak digunakan dalam analisis dokumen medis, berita pandemi, hingga monitoring media sosial untuk mengidentifikasi pola penyebaran penyakit (Ma et al., 2021). Dalam beberapa tahun terakhir, perkembangan model transformer seperti (*Bidirectional Encoder Representations from Transformers*)

BERT serta model berbasis *Large Language Model* (LLM) seperti Gemini atau ChatGPT semakin meningkatkan akurasi dalam tugas NER, khususnya karena kemampuannya memahami konteks semantik yang lebih dalam. Penelitian terdahulu menunjukkan bahwa model BERT mampu mencapai akurasi sebesar 89,31% pada tugas NER, yang menjadi tolok ukur penting dalam kemajuan teknologi ini (Devlin et al., 2019) (Achiam et al., 2023).

Ketidakseimbangan data masih menjadi tantangan signifikan dalam klasifikasi *event* dan *non-event*. Pada kasus nyata, jumlah data *non-event* jauh lebih banyak dibandingkan data *event*, sehingga model cenderung bias memprediksi kelas mayoritas. Hal ini menyebabkan penurunan performa terutama pada recall kelas minoritas (He and Garcia, 2009). Untuk mengatasi hal ini, berbagai pendekatan dikembangkan, seperti *oversampling*, *undersampling*, hingga data *augmentation* dengan teknik sinonim. Dalam konteks bahasa Indonesia, augmentasi berbasis API Kateglo yang menghasilkan variasi kalimat melalui sinonim merupakan strategi relevan untuk memperkaya data *event* tanpa mengubah makna inti kalimat.

Augmentasi berbasis sinonim terbukti sebagai strategi efektif untuk memperkaya data sekaligus menjaga makna inti teks. *Studi Easy Data Augmentation* (EDA) memperkenalkan substitusi sinonim untuk menghasilkan variasi kalimat baru (Ningsih et al., 2022) (Ma et al., 2021). Tinjauan lain menegaskan efektivitas metode augmentasi dalam meningkatkan robustnes model NLP (Tseng et al., 2021) meninjau secara komprehensif efektivitas berbagai metode augmentasi dalam meningkatkan robustnes model NLP. Dengan memanfaatkan strategi ini, dataset *event* yang terbatas dapat diperluas sehingga distribusinya lebih seimbang dengan kelas *non-event*.

Sejalan dengan hal tersebut, penelitian ini berfokus pada upaya menganalisis dan membedakan teks berita yang berkaitan dengan peristiwa (*event*) dan yang tidak berkaitan (*non-event*). Proses klasifikasi dilakukan dengan memanfaatkan tiga arsitektur model pembelajaran mendalam, yaitu *Multi-Layer Perceptron* (MLP), *Convolutional Neural Network* (CNN), dan *Long Short-Term Memory* (LSTM), guna memperoleh pemahaman yang lebih komprehensif mengenai kemampuan masing-masing model dalam mengenali pola semantik pada teks berita berbahasa Indonesia.

Penelitian ini dirancang dengan tiga skenario utama. Skenario pertama menggunakan data mentah dari dataset InaCOVED tanpa penerapan NER maupun augmentasi data. Tahapan dimulai dari *preprocessing*, setelah teks bersih, data langsung diubah menjadi representasi vektor menggunakan dua metode *embedding*, yaitu Word2Vec

dan Keras *Embedding Layer*, dengan variasi dimensi (17, 22, 38, 50, dan 100). Hasil representasi vektor ini kemudian dilatih dan diuji menggunakan tiga model *deep learning* MLP, CNN, dan LSTM.

Skenario kedua menggunakan NER berbasis BERT untuk mengekstraksi entitas penting, yaitu *Person*, *Organization*, *Location*, dan *Disease*. Model BERT diterapkan secara otomatis tanpa *prompting*, melalui mekanisme token *classification* untuk mendeteksi posisi dan tipe entitas pada setiap token dalam teks. Hasil ekstraksi entitas dari BERT kemudian dijadikan dasar untuk proses augmentasi data menggunakan API Kateglo, di mana token dengan label "O" diganti dengan sinonim agar makna utama kalimat tetap terjaga. Dataset hasil augmentasi kemudian direpresentasikan dengan Word2Vec dan Keras *Embedding Layer*, sebelum akhirnya diklasifikasikan menggunakan tiga model *deep learning* (MLP, CNN, dan LSTM).

Skenario ketiga memiliki struktur alur yang serupa dengan skenario kedua, tetapi tahap NER dilakukan menggunakan model berbasis LLM, yaitu Gemini 1.5 dan Gemini 2.0 Flash. Proses NER dilakukan melalui pendekatan *prompt-based learning* dengan tiga jenis *prompting*: *zero-shot, one-shot, dan five-shot*, yang memungkinkan model mengenali entitas *Person*, *Organization*, *Location*, dan *Disease*. Setelah entitas berhasil diekstraksi, dilakukan augmentasi data berbasis sinonim menggunakan API Kateglo agar distribusi data *event* dan *non-event* menjadi seimbang. Data hasil augmentasi kemudian dikonversi menjadi representasi vektor menggunakan Word2Vec dan Keras *Embedding Layer*, lalu diklasifikasikan menggunakan MLP, CNN, dan LSTM.

Dimensi vektor representasi menjadi aspek lain yang turut dieksplorasi. Representasi vektor berperan penting dalam memetakan teks ke bentuk numerik sebelum masuk ke model klasifikasi. Beberapa studi melaporkan bahwa variasi *embedding* dapat mempengaruhi performa model karena terkait langsung dengan kapasitas representasi (Liu and Zhang, 2018)(Mikolov et al., 2013). Dengan membandingkan berbagai dimensi *embedding*, penelitian ini dapat mengevaluasi *trade-off* antara kompleksitas model dan akurasi.

Kontribusi penelitian ini mencakup peningkatan performa klasifikasi teks di bidang kesehatan, sekaligus menyediakan perbandingan menyeluruh antara pendekatan berbasis BERT dan LLM pada konteks bahasa Indonesia. Hasil penelitian diharapkan mampu memberikan dasar ilmiah bagi pengembangan sistem monitoring informasi COVID-19 dan diaplikasikan pada kasus serupa di masa depan. Implikasi praktis dari penelitian ini menegaskan pentingnya dukungan teknologi NLP dalam sistem

peringatan dini dan pengambilan keputusan strategis. Dengan klasifikasi *event* dan *non-event* yang lebih akurat, pihak berwenang dapat mengidentifikasi isu kritis secara cepat serta mengalokasikan sumber daya secara lebih efektif. Kontribusi ini memperlihatkan peran penting NLP dalam menjawab tantangan era digital, khususnya menghadapi krisis kesehatan global (Cambria and White, 2014) (He and Garcia, 2009) (Sulieman et al., 2020).

1.2 Tujuan Penelitian

Tujuan dari penelitian ini yaitu: Tujuan penelitian dalam skripsi sebagai berikut:

1) Mengklasifikasikan teks dari dataset InaCOVED menjadi dua kategori, yaitu *event* dan *non-event*, dengan menggunakan tiga model utama yaitu MLP, CNN, dan LSTM.

2) Menerapkan dan mengevaluasi teknik augmentasi data berbasis sinonim menggunakan API Kateglo untuk menyeimbangkan distribusi data *event* dan *non-event* serta meningkatkan variasi teks hasil ekstraksi entitas.

3) Mengembangkan metode klasifikasi teks dengan memanfaatkan NER berbasis BERT dan LLM untuk mengekstraksi entitas penting *Person*, *Organization*, *Location*, dan *Disease* yang relevan dengan konteks kesehatan.

1.3 Manfaat Penelitian

Adapun manfaat dari penelitian ini adalah sebagai berikut:

1) Menghasilkan sistem klasifikasi teks yang efektif untuk membedakan berita *event* dan *non-event* dengan menerapkan tiga arsitektur utama, yaitu MLP, CNN, dan LSTM, sehingga dapat meningkatkan ketepatan analisis teks di bidang kesehatan.

2) Memberikan kontribusi terhadap pengelolaan data tidak seimbang melalui penerapan teknik augmentasi data berbasis sinonim menggunakan API Kateglo, agar model *deep learning* dapat belajar secara optimal dan menghasilkan prediksi yang lebih seimbang.

3) Meningkatkan kualitas analisis teks kesehatan melalui pemanfaatan NER berbasis BERT dan LLM, untuk mengekstraksi entitas penting seperti *Person*,

Organization, *Location*, dan *Disease*, yang berperan dalam identifikasi pola informasi pada teks berita.

BAB II

TINJAUAN PUSTAKA

2.1 *Natural Language Processing* (NLP)

Natural Language Processing (NLP) merupakan bidang interdisipliner yang menggabungkan linguistik, ilmu komputer, dan kecerdasan buatan untuk memungkinkan mesin memahami bahasa manusia secara alami. Sejak awal perkembangannya, NLP berkembang dari pendekatan berbasis aturan menuju metode berbasis statistik dan kini beralih pada *deep learning* (Baltrušaitis et al., 2019). Perubahan paradigma ini membuat NLP lebih adaptif dalam menangani teks tidak terstruktur dibandingkan pendekatan lama yang kaku (Orson, 2017).

Peran NLP semakin penting karena volume data teks digital meningkat drastis. Data ini muncul dari berbagai sumber seperti berita daring, media sosial, hingga publikasi ilmiah. Dalam ranah kesehatan masyarakat, NLP digunakan untuk mengekstraksi gejala penyakit dari catatan medis tidak terstruktur serta memantau laporan epidemiologi secara otomatis (Alsentzer et al., 2019). Pada kasus pandemi COVID-19, NLP juga dimanfaatkan untuk menyaring berita dan postingan media sosial, guna membedakan informasi valid dengan misinformasi yang dapat memengaruhi persepsi publik (Marcel et al., 2020).

Kemajuan signifikan dalam NLP dimulai dengan representasi kata berbasis vektor. Model klasik seperti Word2Vec dan GloVe mampu menangkap hubungan semantik antar kata sehingga lebih unggul dibanding representasi berbasis frekuensi kata (Mikolov et al., 2013) (Pennington et al., 2014). *Word embedding* ini menjadi fondasi penting bagi hampir seluruh model NLP modern, termasuk klasifikasi teks, sentiment *analysis*, hingga NER (Abdul-Mageed and Ungar, 2017). Selain representasi kata, arsitektur *deep learning* juga menjadi tonggak dalam NLP. Model CNN terbukti efektif menangkap pola lokal dalam teks (Kim, 2014), sedangkan *Recurrent Neural Networks* (RNN) dan LSTM unggul dalam mempelajari ketergantungan jangka panjang antar kata (Hochreiter and Schmidhuber, 1997). Kombinasi kedua

pendekatan ini telah diaplikasikan luas untuk tugas klasifikasi teks dalam berbagai bahasa, termasuk bahasa Indonesia (Persiani and Hellström, 2019).

Terobosan besar berikutnya datang dari transformer yang diperkenalkan oleh (Vaswani et al., 2017) dengan mekanisme *self-attention*, transformer memungkinkan model memahami konteks global dalam sebuah kalimat, model seperti BERT dan RoBERTa secara konsisten melampaui performa CNN dan LSTM dalam hampir semua benchmark NLP. Tidak hanya itu, transformer juga menjadi basis bagi model besar generatif seperti GPT (Devlin et al., 2019) (Liu et al., 2019). Kebangkitan LLM seperti GPT-3 dan GPT-4 serta model multimodal seperti Gemini memperluas kemampuan NLP. LLM dapat melakukan *zero-shot* dan *few-shot learning*, artinya model mampu menyelesaikan tugas baru tanpa perlu pelatihan khusus. Teknologi ini semakin mempermudah implementasi NLP pada domain medis, analisis berita, hingga sistem monitoring kebencanaan (Team et al., 2023) (Brown et al., 2020).

Teknologi NLP masih menghadapi ketidakseimbangan data sebagai tantangan utama. Pada teks kesehatan, jumlah data *non-event* biasanya jauh lebih besar daripada *event*, sehingga model sering bias pada kelas mayoritas (He and Garcia, 2009), untuk mengatasi masalah ini, berbagai metode seperti *oversampling*, undersampling, dan augmentasi data terus dikembangkan (Wei and Zou, 2019a). Dalam konteks bahasa Indonesia, penerapan NLP menghadapi keterbatasan dataset besar yang representatif.

Penelitian (Persiani and Hellström, 2019) menunjukkan pentingnya pembangunan dataset khusus bahasa lokal untuk mendukung klasifikasi teks. Oleh karena itu, teknik augmentasi berbasis sinonim seperti pemanfaatan API Kateglo menjadi salah satu pendekatan yang efektif untuk memperkaya data berlabel *event*, sehingga model lebih seimbang dalam memprediksi (Tseng et al., 2021). Dengan beragam kemajuan ini, NLP kini menjadi teknologi yang sangat strategis untuk pemrosesan teks di berbagai domain. Tidak hanya berkontribusi dalam bidang akademik, NLP juga berperan penting dalam aplikasi nyata seperti sistem monitoring pandemi, klasifikasi informasi kritis, dan deteksi berita palsu. Penelitian terbaru menegaskan bahwa kombinasi antara model *deep learning*, representasi kata yang kuat, serta strategi augmentasi data merupakan kunci dalam mencapai NLP yang lebih andal (Qiu et al., 2020).

2.2 *Named Entity Recognition* (NER)

Named Entity Recognition (NER) merupakan salah satu tugas penting dalam NLP yang bertujuan untuk mengidentifikasi entitas tertentu dari teks tidak terstruktur, seperti *Person*, *Organization*, *Location*, hingga *Disease*. Dengan adanya NER, informasi yang semula tidak terstruktur dapat diubah menjadi representasi terorganisir, sehingga lebih mudah dianalisis untuk berbagai aplikasi seperti analisis berita, sistem pencarian informasi, hingga pemantauan kesehatan masyarakat (Yadav and Bethard, 2018). Awalnya, pendekatan NER berbasis aturan (*rule-based systems*) dan metode statistik klasik seperti *Hidden Markov Models* (HMM) serta *Conditional Random Fields* (CRF) menjadi fondasi dalam ekstraksi entitas. Namun, metode ini sangat bergantung pada *hand-crafted features* dan kurang fleksibel dalam menangani variasi bahasa alami. Untuk bahasa dengan morfologi kompleks seperti bahasa Indonesia, keterbatasan ini semakin terlihat karena fitur yang dibutuhkan menjadi sangat beragam (Lample et al., 2016) (Finkel et al., 2005).

Seiring kemajuan *deep learning*, banyak penelitian beralih ke model-model neural network untuk NER. Dalam model ini, fitur tidak dirancang secara manual, melainkan dipelajari oleh jaringan berbasis representasi kata (*embedding*) dan arsitektur sekuensial seperti RNN, LSTM, atau CNN. Misalnya (Li et al., 2020) menjelaskan bagaimana *deep learning* techniques memungkinkan pemodelan representasi yang lebih abstrak dan fleksibel daripada metode tradisional.

Model-model awal dalam arsitektur neural menyatukan *embedding* kata dengan context encoder seperti BiLSTM, kemudian diteruskan ke *tag decoder* seperti CRF untuk memodelkan dependensi antar label (contoh: “B-ORG” tidak akan diikuti langsung oleh “I-PER”). Kombinasi BiLSTM + CRF menjadi salah satu arsitektur populer dalam NER sebelum munculnya model berbasis transformer. (Seow et al., 2025) menyebut bahwa pendekatan-pendekatan ini membentuk dasar untuk berbagai varian model modern. Transformasi besar dalam NER terjadi dengan munculnya model berbasis transformer, terutama BERT. BERT memanfaatkan *self-attention* dan konteks dua arah (*bidirectional*) untuk menghasilkan representasi token yang sangat kontekstual. Dalam banyak penelitian, fine-tuning BERT pada dataset NER telah memberikan performa unggul dibandingkan metode sebelumnya (Devlin et al., 2019). Variannya seperti RoBERTa, ALBERT, dan model domain spesifik (misalnya BioBERT) juga banyak diuji dalam domain medis atau biomedis (Lee et al., 2020).

Belakangan, riset terbaru mulai mengeksplorasi integrasi LLM ke dalam tugas NER, terutama dalam skenario *few-shot* atau *zero-shot*. Model seperti GPT-3 / GPT-4 atau

model generatif besar lainnya dapat digunakan untuk mengekstrak entitas tanpa pelatihan khusus, hanya melalui perintah *prompt* (*prompt engineering*). (Alqaaidi et al., 2023) meninjau metode NER dan Relasi Klasifikasi yang fokus pada *few-shot learning*, termasuk pendekatan dengan LLM. (Keraghel et al., 2024) Dalam survey terbaru juga menyoroti bahwa tren integrasi LLM ke tugas NER semakin meningkat, meskipun terdapat tantangan seperti kebutuhan data *prompt*, interpretabilitas, dan ukuran model besar.

2.3 Augmentasi Data Teks

Augmentasi data merupakan salah satu teknik penting dalam *machine learning* yang bertujuan untuk memperbanyak jumlah data pelatihan tanpa harus benar-benar mengumpulkan data baru. Pada teks, augmentasi dilakukan dengan cara memodifikasi kalimat asli menjadi variasi baru yang tetap mempertahankan makna utama (Burns et al., 2021). Teknik ini sangat krusial pada kasus *imbalanced* dataset, di mana jumlah data antar kelas tidak seimbang dan dapat menurunkan kinerja model (He and Garcia, 2009). Beberapa pendekatan augmentasi teks yang populer antara lain *back-translation*, *random insertion/deletion*, dan *synonym replacement*. Di antara teknik tersebut, *synonym replacement* sering digunakan karena relatif sederhana dan tetap menjaga konteks semantik. (Wei and Zou, 2019b) memperkenalkan *Easy Data Augmentation* (EDA) yang memanfaatkan penggantian kata dengan sinonim untuk menghasilkan variasi data yang lebih beragam. Hasil penelitian mereka menunjukkan bahwa augmentasi berbasis sinonim dapat meningkatkan performa klasifikasi teks secara signifikan, terutama pada dataset berukuran kecil.

Dalam konteks bahasa Indonesia, augmentasi berbasis sinonim dapat diimplementasikan menggunakan sumber daya leksikal seperti Kateglo API. API ini menyediakan informasi sinonim, antonim, serta relasi leksikal lainnya yang memungkinkan pembentukan variasi kalimat baru. Pendekatan ini sangat relevan untuk teks kesehatan, seperti data InaCOVED, di mana hanya entitas dengan label ”O” (*non-entity*) yang disinonimkan agar tidak mengubah arti dari entitas utama (misalnya *Person*, *Organization*, *Location*, *Disease*). Strategi ini memastikan bahwa informasi inti tetap terjaga, sekaligus memperkaya variasi data pada kelas *event* yang minoritas. Selain itu, penelitian terbaru juga menunjukkan bahwa augmentasi dapat meningkatkan robustness model terhadap variasi bahasa alami, misalnya (Shao and Nakashole, 2020) menunjukkan bahwa augmentasi berbasis

sinonim dapat mengurangi *overfitting* dan membuat model lebih generalisasi. (Tseng et al., 2021) juga menekankan bahwa kombinasi berbagai teknik augmentasi mampu menghasilkan performa yang lebih konsisten pada model NLP modern dengan demikian, dalam penelitian ini augmentasi digunakan tidak hanya untuk menyeimbangkan distribusi kelas *event* dan *non-event*, tetapi juga untuk mengevaluasi dampaknya terhadap representasi vektor (Word2Vec dan Keras *Embedding Layer*) serta performa tiga arsitektur model (MLP, CNN, dan LSTM).

Augmentasi data sangat relevan dalam domain kesehatan karena data berlabel seringkali terbatas, sementara kebutuhan analisis teks semakin meningkat. Studi oleh (Holmgren et al., 2020) menunjukkan bahwa augmentasi berbasis sinonim dan parafrasa dapat membantu meningkatkan performa model dalam mengekstraksi informasi medis dari catatan pasien. Teknik ini juga mampu memperbaiki recall pada kelas minoritas, misalnya dalam mendeteksi gejala penyakit langka yang jarang muncul dalam data pelatihan. Dalam konteks pandemi COVID-19, augmentasi digunakan untuk memperkaya data terkait laporan kasus dan berita agar sistem klasifikasi lebih tangguh terhadap variasi istilah medis dan kebijakan (Stawska et al., 2021).

Pada bahasa dengan sumber daya terbatas (*low-resource language*) seperti bahasa Indonesia, augmentasi data memiliki peran ganda: menyeimbangkan distribusi kelas dan menutupi keterbatasan korpus yang tersedia. Penelitian oleh (Hedderich et al., 2021) menekankan bahwa augmentasi menjadi salah satu solusi utama untuk NLP di *low-resource language* karena proses anotasi manual sangat mahal dan memakan waktu. Dengan adanya augmentasi berbasis sinonim melalui API seperti Kateglo, peneliti dapat memperluas variasi teks tanpa kehilangan struktur linguistik khas bahasa Indonesia. Hal ini sejalan dengan penelitian (Ningsih et al., 2022) yang membuktikan bahwa augmentasi sinonim meningkatkan akurasi klasifikasi teks pendek berbahasa Indonesia, dengan demikian, penelitian ini tidak hanya menyumbang pada pengembangan klasifikasi *event*/*non-event*, tetapi juga pada pemanfaatan teknik augmentasi dalam bahasa Indonesia yang masih jarang dieksplorasi dalam bidang kesehatan digital.

2.4 *Deep Learning*

Deep Learning merupakan subbidang dari kecerdasan buatan (AI) yang mengembangkan algoritma berbasis *artificial neural networks* dengan banyak lapisan (*deep architecture*). Tujuan utama pendekatan ini adalah meniru cara kerja otak

manusia dalam mengenali pola dan membuat keputusan berdasarkan data yang kompleks. Berbeda dengan *machine learning* tradisional, *deep learning* dapat secara otomatis melakukan *feature learning* dari data mentah tanpa perlu rekayasa fitur manual yang intensif (LeCun et al., 2015) (Schmidhuber, 2015). Dalam ranah NLP, *deep learning* telah merevolusi banyak tugas seperti sentiment *analysis*, *machine translation*, dan klasifikasi teks. Hal ini dimungkinkan karena representasi kata (*word representation*) berbasis *embedding* yang memungkinkan teks diubah menjadi vektor numerik. Dengan representasi tersebut, hubungan semantik antar kata dapat dimodelkan dengan lebih baik dibandingkan metode tradisional seperti *Bag-of-Words* (BoW) atau TF-IDF yang hanya menghitung frekuensi kata (Davies et al., 2018).

Salah satu fondasi penting dalam *deep learning* untuk NLP adalah *word embedding*, yaitu representasi vektor padat (*dense vector*) yang memetakan kata ke ruang multidimensi. Word2Vec (Mikolov et al., 2013) menjadi salah satu model populer yang mampu menangkap hubungan semantik antar kata melalui pendekatan skip-gram dan continuous *bag-of-words* (CBOW). Selain itu, Keras menyediakan *Embedding Layer* yang melatih representasi kata secara *end-to-end* dalam model neural network. Kedua pendekatan ini digunakan dalam penelitian untuk membandingkan performa embedding terhadap klasifikasi teks *event* dan *non-event*. Penggunaan *deep learning* dalam klasifikasi teks menawarkan kemampuan untuk memahami konteks urutan kata dan pola semantik. Model *Multi-Layer Perceptron* (MLP) dapat mempelajari relasi *non-linear* antar fitur, *Convolutional Neural Network* (CNN) efektif menangkap pola lokal seperti n-gram, sementara *Long Short-Term Memory* (LSTM) unggul dalam memproses urutan kata dan menangani masalah long-term dependencies. Dengan membandingkan ketiga model ini, penelitian dapat memberikan gambaran yang lebih komprehensif mengenai keunggulan masing-masing arsitektur dalam konteks data kesehatan (Carr and Silk, 2018) (Kim, 2014).

Masalah umum dalam klasifikasi teks adalah *imbalanced* dataset, di mana jumlah data kelas mayoritas jauh lebih besar daripada kelas minoritas. Pada penelitian ini, jumlah data *non-event* jauh lebih besar dibandingkan data *event*, yang dapat menyebabkan model bias memprediksi kelas mayoritas. *Deep learning* sering mengalami penurunan performa pada *recall* kelas minoritas dalam kondisi ini. Oleh karena itu, teknik data augmentation berbasis sinonim digunakan untuk menyeimbangkan distribusi data agar model tidak kehilangan sensitivitas pada kelas *event* (He and Garcia, 2009) (Wei and Zou, 2019a). Augmentasi data dalam konteks *deep learning* terbukti mampu meningkatkan robustness dan generalisasi model. Penelitian (Shao and Nakashole, 2020) menunjukkan bahwa augmentasi berbasis

sinonim dapat mengurangi *overfitting*, sedangkan (Tseng et al., 2021) menekankan bahwa kombinasi berbagai metode augmentasi menghasilkan kinerja yang lebih konsisten pada model modern. Dalam penelitian ini, augmentasi dilakukan pada kata berlabel O saja (*non-entity*), sehingga entitas utama seperti *Person*, *Organization*, *Location*, dan *Disease* tetap terjaga. Hal ini memastikan informasi kritis dalam teks tidak berubah, namun variasi kalimat tetap bertambah untuk memperkaya data *event*.

Selain augmentasi, pemilihan dimensi *embedding* juga merupakan faktor penting dalam *deep learning*. Dimensi *embedding* yang terlalu rendah dapat mengurangi kapasitas representasi semantik, sementara dimensi yang terlalu tinggi meningkatkan risiko *overfitting* dan beban komputasi. Beberapa studi (Yin and Shen, 2018) (Wang, 2019) menemukan bahwa variasi dimensi *embedding* mempengaruhi performa model, sehingga diperlukan eksperimen untuk menemukan ukuran optimal. Pada penelitian ini, diuji lima variasi dimensi embedding (17, 22, 38, 50, 100) untuk mengevaluasi *trade-off* antara akurasi dan efisiensi. Lebih jauh, integrasi NER dengan *deep learning* memberikan nilai tambah dalam analisis teks kesehatan. NER memungkinkan model mengekstraksi entitas penting sebelum proses klasifikasi, sehingga representasi data menjadi lebih kaya.

Dengan memanfaatkan NER berbasis BERT maupun LLM (Gemini), penelitian ini menguji bagaimana hasil ekstraksi entitas dapat memengaruhi kualitas augmentasi dan klasifikasi. Hal ini sejalan dengan tren penelitian terbaru yang menggabungkan NER dengan klasifikasi teks berbasis *deep learning* untuk meningkatkan akurasi sistem informasi kesehatan (Devlin et al., 2019) (Ma et al., 2021). Dengan kerangka tersebut, *deep learning* bukan hanya sekadar alat klasifikasi, tetapi juga menjadi pendekatan komprehensif yang menggabungkan representasi teks (*embedding*/word2vec), NER untuk ekstraksi entitas, augmentasi data untuk keseimbangan kelas, serta perbandingan arsitektur (MLP, CNN, LSTM). Evaluasi menyeluruh ini diharapkan menghasilkan model klasifikasi teks *event* dan *non-event* yang lebih akurat, robust, dan relevan untuk mendukung pemantauan kesehatan masyarakat.

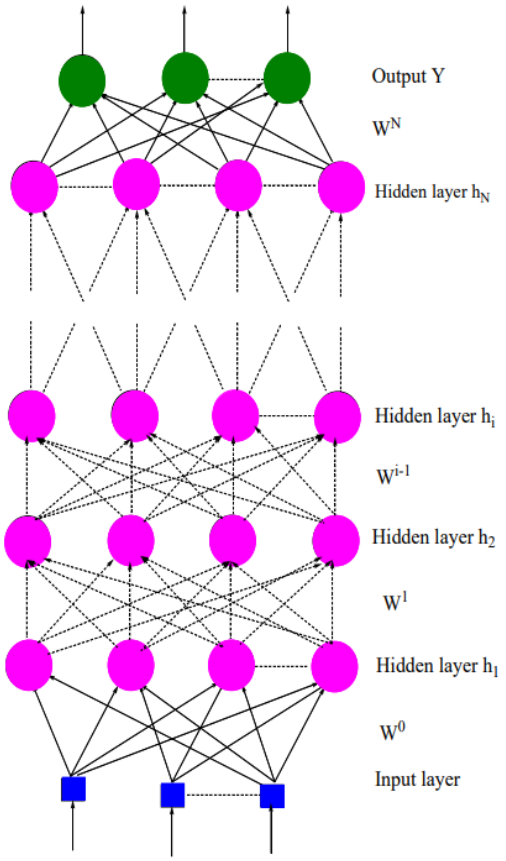
2.4.1 *Multi-Layer Perceptron* (MLP)

Multi-Layer Perceptron (MLP) merupakan salah satu arsitektur paling fundamental dalam *deep learning* yang menjadi dasar bagi banyak model jaringan saraf modern. MLP termasuk dalam kategori *feed-forward neural network*, di mana informasi mengalir secara searah dari lapisan input menuju lapisan output tanpa adanya umpan

balik (*recurrent connection*) (Goodfellow et al., 2016).

Arsitektur dasar MLP terdiri dari tiga komponen utama, yaitu *input layer*, satu atau lebih *hidden layer*, dan *output layer*. Setiap neuron pada lapisan tersembunyi terhubung penuh (*fully connected*) dengan neuron pada lapisan berikutnya, menghasilkan struktur yang mampu mempelajari relasi kompleks antar fitur. Proses pembelajaran dilakukan dengan menyesuaikan bobot jaringan melalui algoritma *backpropagation* dan optimisasi gradien, seperti Adam optimizer atau *stochastic gradient descent* (SGD) (Adam et al., 2014).

Dalam konteks pemrosesan bahasa alami (NLP), MLP berperan penting dalam tahap klasifikasi setelah representasi teks diubah menjadi bentuk numerik. Proses transformasi teks menjadi vektor dilakukan melalui teknik *embedding* seperti Word2Vec, Glove atau Keras *Embedding Layer* (Mikolov et al., 2013) (Pennington et al., 2014). Representasi ini memungkinkan model untuk memahami hubungan semantik antar kata dan memanfaatkan pola statistik dalam korpus bahasa. MLP kemudian memproses vektor tersebut melalui beberapa lapisan tersembunyi dengan fungsi aktivasi *non-linear* seperti ReLU atau tanh, yang membantu jaringan mempelajari hubungan *non-linear* antar fitur teks (LeCun et al., 2015).



Gambar 2.1 Struktur Jaringan *Multilayer Perceptron* (Ramchoun et al., 2016)

Dengan asumsi kita menggunakan lapisan input dengan n_0 neuron

$$X = (x_0, x_1, \ldots, x_{n_0})$$

dan fungsi aktivasi sigmoid

$$f(x) = \frac{1}{1 + e^{-x}} \tag{2.4.1}$$

Untuk menghasilkan keluaran dari jaringan, setiap unit pada lapisan tersembunyi dihitung berdasarkan bobot dan fungsi aktivasi. Misalkan terdapat sejumlah *hidden layers* $(h_1, h_2, \ldots, h_N)$, dengan n_i sebagai jumlah neuron pada lapisan tersembunyi ke-i. Maka, keluaran pada lapisan tersembunyi pertama dapat didefinisikan sebagai:

$$h_j^1 = f\left(\sum_{k=1}^{n_0} w_{k,j}^0 x_k\right), \quad j = 1, \ldots, n_1 \tag{2.4.2}$$

Sedangkan untuk lapisan tersembunyi berikutnya:

$$h_j^i = f\left(\sum_{k=1}^{n_{i-1}} w_{k,j}^{i-1} h_k^{i-1}\right), \quad i = 2, \ldots, N,\ j = 1, \ldots, n_i \tag{2.4.3}$$

Keluaran dari seluruh neuron pada lapisan tersembunyi ke-i kemudian dapat dinyatakan sebagai:

$$h_i = (h_1^i, h_2^i, \ldots, h_{n_i}^i) \tag{2.4.4}$$

Selanjutnya, lapisan keluaran (*output layer*) dihitung dengan menjumlahkan kontribusi dari lapisan tersembunyi terakhir:

$$y_j = f\left(\sum_{k=1}^{n_N} w_{k,j}^N h_k^N\right), \quad Y = (y_1, y_2, \ldots, y_{N+1}) = F(W, X) \tag{2.4.5}$$

Dengan $w_{k,j}^N$ adalah bobot antara neuron ke-k pada lapisan tersembunyi terakhir dan neuron ke-j pada lapisan keluaran. Fungsi F adalah fungsi transformasi dari model, sementara W merepresentasikan seluruh matriks bobot:

$$W = [W^0, W^1, \ldots, W^N], \quad W^i = (w_{j,k}^i), \quad w_{j,k}^i \in \mathbb{R} \tag{2.4.6}$$

Untuk penyederhanaan, dapat diasumsikan bahwa setiap lapisan memiliki jumlah neuron yang sama ($n_i = n$), sehingga perhitungan dapat dilakukan secara berulang untuk setiap lapisan tersembunyi. Lapisan aktivasi f berperan penting dalam menambahkan *non-linearitas* agar jaringan dapat mempelajari hubungan kompleks antar fitur masukan.

Penggunaan MLP pada klasifikasi teks telah banyak diteliti karena kesederhanaan strukturnya dan efisiensi komputasinya. (Zhang et al., 2015) menunjukkan bahwa MLP dapat memberikan performa kompetitif dibanding model sekuensial seperti LSTM pada teks pendek, asalkan fitur representasi yang digunakan sudah cukup informatif. Keunggulan MLP terletak pada kemampuannya dalam melakukan *feature abstraction* yakni mengubah fitur mentah menjadi representasi laten yang lebih padat dan bermakna (Lu et al., 2022). Pada tugas klasifikasi biner, seperti pemisahan teks *event* dan *non-event* dalam dataset InaCOVED, lapisan keluaran MLP biasanya menggunakan satu neuron dengan fungsi aktivasi sigmoid untuk menghasilkan probabilitas antara 0 dan 1. Selain itu, MLP juga banyak digunakan sebagai baseline model dalam penelitian NLP modern. Hal ini karena MLP memberikan gambaran awal seberapa baik representasi teks bekerja sebelum dibandingkan dengan model

yang lebih kompleks seperti CNN atau LSTM.

Penelitian oleh (Suneera and Prakash, 2020) menunjukkan bahwa MLP dapat digunakan sebagai model referensi untuk mengevaluasi pengaruh ukuran embedding terhadap akurasi klasifikasi. Semakin besar dimensi *embedding*, semakin kaya pula representasi semantik yang dihasilkan, namun juga berpotensi menambah kompleksitas komputasi dan risiko *overfitting*. Oleh karena itu, penelitian ini membandingkan performa MLP dengan variasi dimensi *embedding* (17, 22, 38, 50, dan 100) untuk menemukan konfigurasi paling optimal.

Dalam studi-studi terkini, MLP sering dikombinasikan dengan teknik regularization seperti *dropout* (Srivastava et al., 2014) dan L2 *regularization* untuk mencegah *overfitting*. Pendekatan ini terbukti efektif terutama ketika dataset memiliki ketidakseimbangan kelas seperti pada data *event* dan *non-event* yang digunakan dalam penelitian ini. Beberapa penelitian sebelumnya, seperti oleh (Elhassan et al., 2023), menunjukkan bahwa penerapan MLP dengan dropout layer pada klasifikasi teks emosi berbahasa Arab meningkatkan generalisasi model secara signifikan. Dengan demikian, MLP menjadi pilihan tepat sebagai salah satu arsitektur untuk mengevaluasi kinerja *embedding* dalam konteks data berimbangan maupun tidak seimbang. Kelebihan lain dari MLP adalah fleksibilitasnya dalam menangani berbagai representasi fitur. Ketika digunakan bersama *embedding* seperti Word2Vec atau Keras *Embedding Layer*, MLP dapat mengekstraksi pola makna dari teks meskipun tanpa memperhatikan urutan kata secara eksplisit. Hal ini membuat MLP cocok sebagai model awal untuk menganalisis sejauh mana representasi vektor mampu mempengaruhi performa klasifikasi teks. Dalam penelitian ini, MLP tidak hanya digunakan pada skenario data mentah, tetapi juga pada data hasil proses NER dan augmentasi menggunakan Kateglo API, sehingga dapat dibandingkan bagaimana pengaruh kedua tahap tersebut terhadap akurasi model.

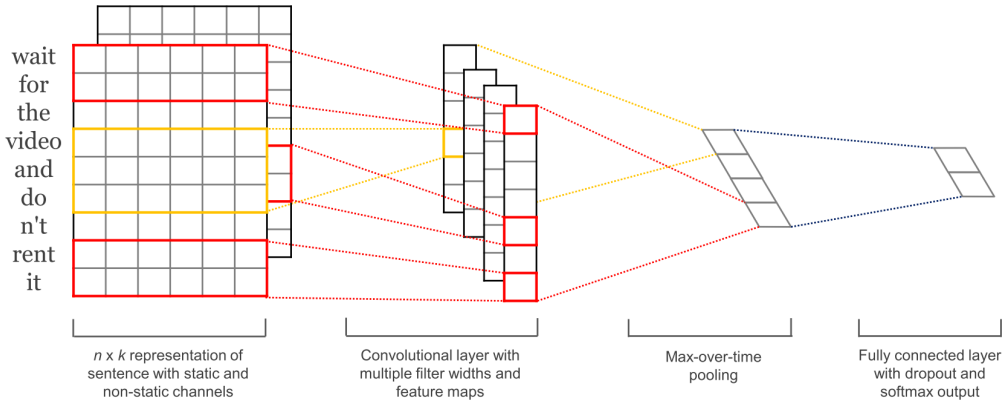
Dalam konteks penelitian terkait COVID-19, penerapan MLP juga telah memberikan hasil menjanjikan. Contohnya (Khanam et al., 2021) menggunakan MLP untuk mendeteksi berita palsu seputar pandemi COVID-19 dengan hasil akurasi mencapai 92%, menunjukkan bahwa model ini efektif ketika dipadukan dengan teknik embedding yang baik. MLP juga digunakan oleh (Aryal and Bhattarai, 2021) dalam analisis opini publik mengenai vaksinasi COVID-19, di mana model ini berhasil mengklasifikasikan sentimen dengan tingkat kesalahan yang rendah. Studi-studi ini menunjukkan bahwa meskipun sederhana, MLP tetap relevan dan kuat dalam tugas-tugas NLP modern, terutama dalam skenario dengan data berlabel terbatas atau tidak seimbang. Dalam keseluruhan arsitektur sistem penelitian ini, MLP

berfungsi sebagai salah satu model pembanding utama untuk mengukur pengaruh penggunaan NER dan augmentasi data terhadap klasifikasi teks. Hasil dari MLP akan dibandingkan dengan CNN dan LSTM untuk menilai sejauh mana informasi spasial (*spatial features*) atau temporal (*sequential dependencies*) memberikan kontribusi pada peningkatan akurasi. Dengan pendekatan eksperimental yang sistematis, hasil dari model MLP diharapkan dapat memberikan pemahaman empiris mengenai keterkaitan antara representasi *embedding*, teknik augmentasi, dan performa klasifikasi teks dalam bahasa Indonesia, khususnya dalam domain kesehatan publik seperti pandemi COVID-19.

2.4.2 *Convolutional Neural Network* (CNN)

Convolutional Neural Network (CNN) merupakan salah satu arsitektur *deep learning* yang dirancang untuk mengenali pola spasial dan lokal dalam data. Model ini diperkenalkan pertama kali oleh (Lecun et al., 1998) melalui arsitektur LeNet-5, yang digunakan untuk pengenalan tulisan tangan pada dataset MNIST. CNN bekerja dengan mengandalkan operasi matematis yang disebut *convolution*, di mana sekelompok filter atau kernel secara sistematis digeser ke seluruh area input untuk mengekstraksi fitur penting. Konsep dasar ini kemudian berkembang pesat dan digunakan tidak hanya untuk pengolahan citra, tetapi juga dalam bidang pemrosesan bahasa alami (NLP), termasuk klasifikasi teks dan ekstraksi entitas (LeCun et al., 2015).

Dalam konteks NLP, CNN berperan penting karena kemampuannya mendeteksi pola lokal antar kata yang memiliki keterkaitan semantik, misalnya urutan kata yang membentuk makna tertentu dalam kalimat. Representasi teks terlebih dahulu diubah menjadi bentuk vektor menggunakan *word embedding* seperti Word2Vec atau Keras *Embedding Layer*. Kemudian, setiap vektor kata akan menjadi input ke lapisan konvolusi (*convolution layer*) yang memiliki filter dengan ukuran kernel tertentu. Setiap filter bertanggung jawab untuk mengenali pola lokal, seperti kombinasi kata yang sering muncul bersama dalam konteks yang bermakna. Misalnya, pada kalimat terkait COVID-19, CNN dapat mengenali pola seperti “peningkatan kasus harian” atau “vaksinasi massal” sebagai indikasi dari suatu peristiwa (*event*) penting (Kim, 2014).



Gambar 2.2 Struktur Jaringan *Convolutional Neural Network* (Kim, 2014)

Lapisan konvolusi (*convolution layer*) berfungsi untuk mengekstraksi pola-pola lokal dari urutan kata dalam teks, seperti hubungan *n-gram*, yang mewakili makna semantik pada konteks tertentu (Yin et al., 2017). Misalnya, dengan menggunakan jendela konvolusi berukuran tiga, model CNN akan memproses tiga kata secara bersamaan untuk mendeteksi keterkaitan antar kata. Untuk setiap jendela kata w, representasi vektor dihasilkan melalui operasi konvolusi dengan bobot W dan bias b, kemudian diteruskan ke fungsi aktivasi *non-linear* seperti *tanh* atau *ReLU*. Persamaan operasinya dituliskan sebagai berikut:

$$p_i = \tanh(W \cdot c_i + b) \tag{2.4.7}$$

di mana c_i merupakan hasil penggabungan *embedding* dari kata-kata dalam jendela konvolusi ke-i. Proses ini memungkinkan CNN mengenali fitur linguistik lokal yang relevan terhadap tugas klasifikasi teks. Setelah proses konvolusi, CNN menerapkan *max pooling layer* untuk mengurangi dimensi keluaran dengan cara memilih nilai maksimum dari setiap fitur hasil konvolusi (LeCun et al., 2015). Operasi ini dapat ditulis sebagai berikut:

$$x_j = \max(p_{1,j}, p_{2,j}, \ldots, p_{n,j}) \tag{2.4.8}$$

dengan x_j merupakan fitur hasil agregasi untuk dimensi ke-j. Lapisan *pooling* ini berfungsi mempertahankan fitur yang paling dominan dan membuang informasi yang kurang penting, sehingga menghasilkan representasi vektor tetap yang efisien untuk tahap klasifikasi selanjutnya.

Dalam penelitian ini, CNN digunakan sebagai salah satu arsitektur pembanding untuk mengklasifikasikan teks berbahasa Indonesia dari dataset InaCOVED menjadi dua kelas, yaitu *event* dan *non-event*. Arsitektur CNN yang diterapkan terdiri atas lapisan Embedding dengan dimensi vektor 17, 22, 38, 50, dan 100, diikuti dengan lapisan Conv1D yang memiliki 128 filter dan ukuran kernel 5. Setelah itu digunakan lapisan GlobalMaxPooling1D untuk memilih fitur paling menonjol, lalu diikuti oleh lapisan Dense dengan 64 neuron dan fungsi aktivasi ReLU. Tahap terakhir adalah lapisan keluaran dengan fungsi aktivasi sigmoid yang bertugas untuk mengklasifikasikan teks menjadi kelas *event* atau *non-event*. Model ini dipilih karena CNN terbukti mampu menangani data teks dengan panjang kalimat bervariasi serta memiliki robustnes tinggi terhadap data yang tidak seimbang (Kurniawan and Mustikasari, 2021). Dalam konteks penelitian ini, CNN diharapkan mampu mendeteksi pola linguistik yang menandakan terjadinya peristiwa COVID-19, seperti peningkatan kasus, perubahan kebijakan pemerintah, atau pelaksanaan vaksinasi massal.

Selain digunakan untuk klasifikasi langsung, CNN juga memiliki potensi besar bila dikombinasikan dengan pendekatan lain seperti LSTM atau BERT. Studi oleh (Zhou et al., 2015) memperkenalkan model C-LSTM yang menggabungkan CNN untuk ekstraksi fitur lokal dan LSTM untuk memahami konteks sekuensial, menghasilkan performa unggul dalam klasifikasi teks. Namun, penelitian ini secara khusus menggunakan CNN tunggal untuk menganalisis pengaruh embedding dan augmentasi data terhadap performa klasifikasi. Dengan struktur seperti ini, CNN menjadi model yang efisien, kuat, dan mudah ditafsirkan. CNN tidak hanya dievaluasi berdasarkan akurasi, tetapi juga menggunakan metrik seperti ROC-AUC untuk melihat sensitivitas model terhadap kelas minoritas. Perbandingan dengan model MLP dan LSTM diharapkan dapat memberikan gambaran mendalam tentang bagaimana CNN bekerja pada teks bahasa Indonesia yang telah melalui proses NER dan augmentasi berbasis sinonim, terutama dalam domain kesehatan yang kompleks dan dinamis.

2.4.3 *Long Short-Term Memory* (LSTM)

Long Short-Term Memory (LSTM) merupakan salah satu jenis *arsitektur Recurrent Neural Network* (RNN) yang dikembangkan oleh (Hochreiter and Schmidhuber, 1997) untuk mengatasi kelemahan utama RNN konvensional dalam mempelajari ketergantungan jangka panjang (*long-term dependency problem*). Model RNN

tradisional cenderung mengalami *vanishing gradient*, yaitu kondisi di mana nilai gradien yang sangat kecil menyebabkan jaringan gagal mempelajari hubungan antar kata yang jauh dalam urutan teks (Bengio et al., 1994). LSTM memperbaiki hal ini dengan menambahkan struktur memori internal yang disebut cell state, yang berfungsi sebagai jalur informasi jangka panjang dengan aliran gradien yang stabil. Dengan mekanisme ini, LSTM mampu mengingat konteks dari kalimat yang panjang tanpa kehilangan makna penting.

Arsitektur LSTM terdiri dari tiga gerbang utama yaitu: *input gate*, *forget gate*, dan *output gate* yang berfungsi mengatur arus informasi di dalam jaringan. Forget gate bertugas menentukan informasi dari memori sebelumnya yang akan dihapus, input gate menambahkan informasi baru yang relevan ke dalam cell state, sementara *output gate* menghasilkan keluaran yang akan digunakan pada langkah selanjutnya (Gers et al., 2000). Ketiga gerbang ini beroperasi secara sinergis untuk memastikan bahwa model dapat menyimpan informasi penting dan membuang data yang tidak relevan. Secara umum, mekanisme kerja LSTM dapat dijelaskan melalui formulasi matematis sebagai berikut:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{2.4.9}$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{2.4.10}$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{2.4.11}$$

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t \tag{2.4.12}$$

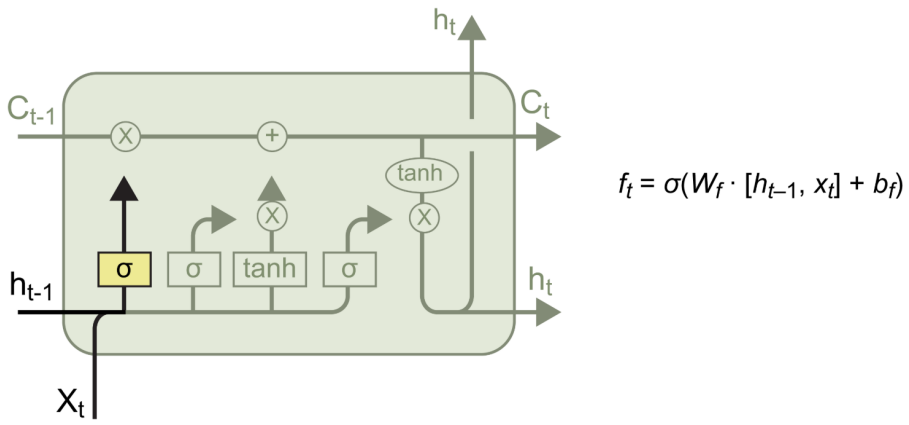
$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{2.4.13}$$

$$h_t = o_t \cdot \tanh(C_t) \tag{2.4.14}$$

Keterangan:

- f_t : *Forget gate*, menentukan informasi dari memori sebelumnya yang akan dilupakan.
- i_t : *Input gate*, mengatur seberapa banyak informasi baru yang akan disimpan dalam memori.
- $\tilde{C}_t$: Kandidat nilai baru untuk *cell state*.
- C_t : *Cell state*, menyimpan informasi jangka panjang.
- o_t : *Output gate*, mengatur keluaran dari unit LSTM pada waktu t.

- h_t : *Hidden state*, keluaran yang diteruskan ke langkah waktu berikutnya.
- σ : Fungsi aktivasi *sigmoid*.
- $\tanh$: Fungsi aktivasi hiperbolik *tangent*.



Gambar 2.3 Struktur Jaringan *Long Short-Term Memory* (Liu, 2024)

Keunggulan utama LSTM terletak pada kemampuannya dalam memahami konteks temporal dan relasi semantik antar kata. Dalam analisis teks, urutan kata sangat berpengaruh terhadap makna, misalnya perbedaan antara "positif COVID-19" dan "tidak positif COVID-19". LSTM dapat membedakan pola seperti ini karena mempertimbangkan hubungan antar token dalam urutan waktu (Zaremba et al., 2014). Selain itu, LSTM lebih stabil dibandingkan RNN standar dalam menangani urutan panjang, menjadikannya pilihan utama untuk berbagai tugas NLP seperti *text classification*, *language modeling*, *machine translation*, dan NER (Young et al., 2018). Dalam penelitian berbasis teks berbahasa Indonesia, LSTM terbukti unggul dalam menangani bahasa alami yang bersifat kompleks dan kontekstual. Studi oleh (Yana et al., 2020) menunjukkan bahwa LSTM dengan *word embedding* memberikan hasil klasifikasi yang lebih baik dibandingkan pendekatan berbasis *Bag of Words* (BoW) dan TF-IDF. Model ini mampu memahami konteks semantik antar kata meskipun struktur kalimat dalam bahasa Indonesia sering kali fleksibel. Hasil serupa juga ditemukan oleh (Nayoga et al., 2021) yang melaporkan bahwa LSTM mencapai akurasi hingga 92% dalam klasifikasi berita bahasa Indonesia, melebihi performa CNN dan MLP pada dataset serupa.

LSTM juga dikenal memiliki kemampuan yang baik dalam menangani data teks yang tidak seimbang (*imbalanced*). Melalui penggunaan lapisan *dropout* dan regularisasi, model ini mampu menahan *overfitting* pada kelas minoritas, terutama ketika

digunakan bersama dengan teknik augmentasi data seperti synonym replacement. (Liu, 2024) menyebutkan bahwa penerapan *dropout rate* sebesar 0.5 meningkatkan kemampuan generalisasi model hingga 7% dibandingkan model tanpa *dropout*. Dalam konteks penelitian ini, penggunaan dropout membantu menjaga stabilitas model ketika berhadapan dengan data hasil augmentasi dari API Kateglo. Selain arsitektur tunggal, beberapa penelitian menggabungkan LSTM dengan model lain untuk meningkatkan performa. Misalnya, (Zhou et al., 2015) mengusulkan CNN-LSTM *hybrid*, di mana CNN digunakan untuk mengekstraksi fitur spasial dari teks dan LSTM menangkap hubungan temporalnya. Pendekatan ini terbukti memberikan hasil yang lebih akurat pada klasifikasi teks dibandingkan penggunaan model tunggal. Walaupun penelitian ini hanya menggunakan LSTM murni, konsep *hybrid* ini menjadi dasar penting dalam memahami potensi pengembangan model di masa depan untuk data sekuensial seperti teks berita COVID-19.

Lebih lanjut, integrasi LSTM dengan *word embedding* modern seperti Word2Vec dan Keras *Embedding Layer* menjadi fokus penting dalam penelitian ini. (Mikolov et al., 2013) menunjukkan bahwa representasi vektor yang baik dapat meningkatkan pemahaman semantik model, karena *embedding* mampu memetakan kata ke ruang numerik berdasarkan kemiripan konteks. Oleh karena itu, penelitian ini menguji lima variasi dimensi *embedding* (17, 22, 38, 50, dan 100) untuk menemukan ukuran optimal yang memberikan keseimbangan antara akurasi dan efisiensi komputasi. Dalam konteks klasifikasi teks *event* dan *non-event* pada dataset InaCOVED, LSTM digunakan sebagai model untuk mengevaluasi dampak tahap NER dan augmentasi data terhadap kinerja klasifikasi. Melalui NER, model dapat fokus pada entitas penting seperti *person*, *organization*, *location*, dan *disease*, sementara augmentasi berbasis Kateglo memperkaya variasi kalimat agar data lebih seimbang.

Kombinasi kedua proses ini diharapkan mampu meningkatkan kemampuan LSTM dalam mengenali pola semantik yang kompleks pada teks berbahasa Indonesia (Sulieman et al., 2020). Dengan arsitektur yang dirancang untuk memahami hubungan temporal dan mekanisme regulasi memori yang efisien, LSTM menjadi komponen kunci dalam penelitian ini. Penggunaannya tidak hanya berfokus pada peningkatan akurasi, tetapi juga pada kemampuan generalisasi model terhadap variasi bahasa dan struktur kalimat. Melalui eksperimen yang dilakukan pada berbagai dimensi *embedding* dan skenario data (mentah, NER-LLM, NER-BERT), penelitian ini diharapkan dapat memberikan kontribusi nyata dalam pengembangan sistem klasifikasi teks berbasis *deep learning* untuk domain kesehatan berbahasa Indonesia.

2.5 Representasi Teks

Representasi teks merupakan salah satu tahapan fundamental dalam pemrosesan bahasa alami (NLP). Tujuan utama dari tahap ini adalah mengubah data berbentuk teks yang bersifat simbolik dan tidak terstruktur menjadi bentuk numerik yang dapat dipahami oleh model pembelajaran mesin. Dalam konteks ini, representasi teks berperan sebagai jembatan antara bahasa manusia dan sistem komputasi, karena algoritma *machine learning* dan *deep learning* hanya dapat memproses input dalam bentuk angka (Parsing, 2009). Pendekatan awal dalam representasi teks adalah metode berbasis frekuensi seperti *Bag of Words* (BoW) dan *Term Frequency–Inverse Document Frequency* (TF-IDF). Kedua teknik ini menghitung seberapa sering kata muncul dalam dokumen dan seberapa penting kata tersebut dalam keseluruhan korpus. Meskipun sederhana, metode ini memiliki kelemahan utama, yaitu kehilangan konteks urutan kata dan hubungan semantik antar kata. Sebagai contoh, kata "positif" dalam kalimat "hasil tes positif COVID-19" memiliki makna yang sangat berbeda dari "sikap positif terhadap vaksin", namun BoW dan TF-IDF tidak dapat membedakan perbedaan semantik tersebut (Schütze et al., 2008).

Untuk mengatasi keterbatasan tersebut, muncul pendekatan berbasis *word embedding* yang merepresentasikan kata sebagai vektor kontinu dalam ruang berdimensi tetap. Representasi ini memungkinkan kata-kata dengan makna serupa memiliki posisi berdekatan dalam ruang vektor, sehingga model dapat menangkap informasi semantik dan sintaksis sekaligus (Mikolov et al., 2013). Teknik ini menjadi titik balik penting dalam NLP modern, karena berhasil menggabungkan makna linguistik dengan efisiensi komputasi.

Word embedding tidak hanya mempertimbangkan kehadiran kata, tetapi juga konteks kemunculannya dalam kalimat. Dalam penelitian ini, representasi teks difokuskan pada dua pendekatan utama, yaitu Word2Vec dan Keras *Embedding Layer*. Pemilihan ini dilakukan karena kedua metode memiliki karakteristik yang saling melengkapi. Word2Vec digunakan sebagai *embedding pre-trained* yang memanfaatkan pembelajaran berbasis konteks untuk menghasilkan representasi semantik yang stabil. Sementara itu, Keras *Embedding Layer* digunakan sebagai *embedding trainable*, di mana bobot representasi kata dipelajari secara langsung selama proses pelatihan model. Dengan demikian, *embedding* ini bersifat dinamis dan dapat menyesuaikan dengan domain spesifik, yaitu teks COVID-19 dalam dataset InaCOVED.

Pendekatan Word2Vec bekerja berdasarkan prinsip distributional semantics, yang

menyatakan bahwa kata dengan konteks serupa memiliki makna yang mirip. Dalam implementasinya, Word2Vec memiliki dua arsitektur utama yaitu *Continuous Bag of Words* (CBOW) dan Skip-Gram. CBOW memprediksi kata target berdasarkan konteksnya, sedangkan Skip-Gram memprediksi konteks dari kata target. Penelitian sebelumnya menunjukkan bahwa Word2Vec efektif untuk menangkap hubungan semantik antar entitas seperti *Person*, *Organization*, dan *Location*, sehingga relevan untuk penelitian yang melibatkan NER (Ma et al., 2022).

Sementara itu, Keras *Embedding Layer* merupakan lapisan bawaan dalam pustaka Keras yang berfungsi untuk mengonversi indeks kata menjadi vektor berdimensi tetap. Setiap kata dipetakan ke dalam matriks *embedding* $W \in \mathbb{R}^{V \times d}$, di mana V adalah ukuran kosakata dan d adalah dimensi *embedding*. Nilai-nilai vektor ini diperbarui selama proses pelatihan model menggunakan algoritma *backpropagation* agar *embedding* yang dihasilkan sesuai dengan pola data (Goodfellow et al., 2016). Pendekatan ini sangat fleksibel karena *embedding* dapat menyesuaikan terhadap konteks dataset spesifik tanpa bergantung pada data pra-latih eksternal.

Dalam konteks penelitian ini, penggunaan dua pendekatan *embedding* tersebut (Word2Vec dan Keras *Embedding Layer*) memungkinkan analisis komparatif yang mendalam terhadap kualitas representasi teks. Dengan menggunakan lima variasi dimensi vektor (17, 22, 38, 50, dan 100), penelitian ini bertujuan untuk mengevaluasi pengaruh ukuran *embedding* terhadap performa tiga model klasifikasi *deep learning*, yaitu MLP, CNN, dan LSTM. Studi oleh (Trnecka and Trneckova, 2021) menunjukkan bahwa pemilihan dimensi *embedding* yang tepat memiliki pengaruh signifikan terhadap kinerja model, karena berkaitan dengan kemampuan model menangkap struktur semantik dan kompleksitas bahasa. Representasi teks berbasis *embedding* juga memiliki implikasi penting dalam konteks ketidakseimbangan data (*imbalanced* dataset). Pada penelitian ini, data *event* jumlahnya lebih sedikit dibandingkan *non-event*, sehingga augmentasi data dilakukan untuk memperkaya kelas minoritas. *Embedding* berperan penting dalam tahap ini karena memastikan bahwa variasi hasil augmentasi tetap memiliki kesamaan semantik dengan kalimat aslinya. Dengan demikian, model tetap dapat belajar konteks yang benar tanpa terganggu oleh sinonim yang tidak relevan (Wei and Zou, 2019a).

Secara keseluruhan, representasi teks menggunakan Word2Vec dan Keras *Embedding Layer* menjadi landasan penting dalam penelitian ini. Kedua pendekatan tersebut memberikan kombinasi ideal antara stabilitas semantik dan kemampuan adaptasi terhadap domain spesifik. Melalui pembandingan kinerja di berbagai dimensi *embedding*, penelitian ini berupaya menemukan konfigurasi representasi

terbaik yang mampu meningkatkan akurasi klasifikasi teks *event* dan *non-event* dalam konteks informasi COVID-19. Pendekatan ini diharapkan dapat memberikan kontribusi terhadap pengembangan model NLP yang lebih efektif untuk Bahasa Indonesia dan domain kesehatan publik.

2.5.1 Word2Vec

Representasi kata berbasis *Word Embedding* merupakan lompatan besar dalam pengolahan bahasa alami (NLP), terutama sejak diperkenalkannya Word2Vec oleh (Mikolov et al., 2013). Model ini menggantikan pendekatan konvensional seperti *Bag of Words* (BoW) dan TF-IDF yang hanya menghitung frekuensi kemunculan kata tanpa mempertimbangkan konteks. Word2Vec memungkinkan setiap kata direpresentasikan sebagai vektor kontinu berdimensi tetap yang belajar dari distribusi konteks kata dalam korpus. Dengan demikian, kata yang memiliki makna serupa akan berada berdekatan dalam ruang vektor (Mikolov et al., 2013). Terdapat dua arsitektur utama dalam Word2Vec, yaitu *Continuous Bag of Words* (CBOW) dan Skip-Gram. Arsitektur CBOW berfungsi untuk memprediksi kata target berdasarkan kata-kata konteks di sekitarnya, sedangkan Skip-Gram melakukan kebalikannya, yaitu memprediksi konteks dari kata target. Kedua pendekatan ini didasarkan pada prinsip distribusional linguistik: kata yang muncul dalam konteks serupa cenderung memiliki makna yang mirip. Model CBOW umumnya lebih cepat dan stabil pada dataset besar, sedangkan Skip-Gram lebih baik untuk menangkap hubungan semantik pada dataset kecil atau kata-kata langka (Rong, 2014). Secara matematis, Word2Vec memaksimalkan probabilitas bersyarat antara kata target dan konteksnya. Untuk arsitektur Skip-Gram, fungsi objektifnya dapat diformulasikan sebagai:

$$J = \frac{1}{T} \sum_{t=1}^{T} \sum_{-c \leq j \leq c, j \neq 0} \log P(w_{t+j} | w_t) \tag{2.5.15}$$

di mana T adalah jumlah total kata dalam korpus dan c merupakan ukuran jendela konteks. Probabilitas bersyarat $P(w_{t+j}|w_t)$ dihitung menggunakan fungsi *softmax* sebagai berikut:

$$P(w_O | w_I) = \frac{\exp({v'_{w_O}}^{\top} v_{w_I})}{\sum_{w=1}^{W} \exp({v'_w}^{\top} v_{w_I})} \tag{2.5.16}$$

Word2Vec tidak hanya merepresentasikan kata sebagai angka, tetapi juga mampu

menangkap hubungan semantik dan sintaksis. Contohnya, hasil vektor dari operasi aljabar seperti:

$$\text{king} - \text{man} + \text{woman} \approx \text{queen} \tag{2.5.17}$$

menunjukkan bahwa *embedding* ini mempelajari pola hubungan antar kata secara konseptual. Hal ini menjadikan Word2Vec unggul dibanding metode tradisional karena mampu menggeneralisasi hubungan antar kata meski tidak muncul langsung dalam korpus pelatihan.

Dalam konteks bahasa Indonesia, penerapan Word2Vec juga menunjukkan hasil menjanjikan. Penelitian oleh (Mayo et al., 2018) menggunakan Word2Vec untuk NER Bahasa Indonesia dan memperoleh peningkatan signifikan dibanding representasi BoW. *Word embedding* berperan penting dalam menangkap makna morfologis yang kompleks pada bahasa Indonesia yang bersifat aglutinatif. Implementasi Word2Vec lokal seperti IndoNLU Word2Vec juga menunjukkan potensi tinggi dalam klasifikasi berita dan deteksi hoaks (Wilie et al., 2020). Dalam penelitian ini, Word2Vec digunakan untuk mengonversi teks dari dataset InaCOVED yang berisi berita dan laporan terkait COVID-19 ke dalam bentuk numerik sebelum dimasukkan ke model MLP, CNN, dan LSTM. Setiap kata diubah menjadi vektor berdimensi 17, 22, 38, 50, dan 100. Tujuannya adalah untuk mengukur pengaruh variasi dimensi *embedding* terhadap kinerja model klasifikasi *event* dan *non-event*. Studi oleh (Yin and Shen, 2018) menunjukkan bahwa dimensi *embedding* yang terlalu kecil gagal menangkap konteks semantik, sedangkan dimensi terlalu besar meningkatkan kompleksitas komputasi tanpa keuntungan performa signifikan.

Selain itu, *embedding* Word2Vec dapat dimanfaatkan sebagai bobot awal (*pre-trained embedding*) pada lapisan *embedding* model *deep learning*. (Reimers and Gurevych, 2019) menunjukkan bahwa inisialisasi bobot dengan *embedding* pra-latih mempercepat konvergensi pelatihan dan menghasilkan stabilitas yang lebih baik, terutama pada dataset berukuran kecil. Dalam konteks penelitian ini, pendekatan tersebut membantu model memahami pola semantik berita COVID-19 tanpa memerlukan data pelatihan besar. Dengan demikian, Word2Vec dalam penelitian ini tidak hanya berfungsi sebagai tahap pra-pemrosesan, tetapi juga sebagai mekanisme untuk memperkaya representasi semantik sebelum klasifikasi. Pendekatan ini penting dalam domain kesehatan publik, di mana istilah medis, lokasi wabah, atau nama organisasi sering kali memiliki pola linguistik yang khas. Evaluasi terhadap berbagai dimensi *embedding* akan memberikan pemahaman empiris mengenai sejauh mana representasi kata mempengaruhi performa klasifikasi teks.

2.5.2 Keras *Embedding Layer*

Berbeda dengan Word2Vec yang bersifat *pre-trained*, Keras *Embedding Layer* merupakan *embedding Layer* yang trainable artinya bobot representasi kata dipelajari langsung selama proses pelatihan model *deep learning*. Lapisan ini merupakan bagian dari pustaka Keras (Chollet, 2021) dan secara umum digunakan untuk mengonversi urutan kata (token) menjadi representasi vektor berdimensi tetap. Keras *Embedding Layer* bekerja dengan menginisialisasi matriks bobot $W \in \mathbb{R}^{V \times d}$, di mana V adalah ukuran kosakata (*vocabulary size*) dan d adalah dimensi *embedding*. Setiap kata w_i direpresentasikan sebagai vektor *embedding* $e_i = W[i]$. Matriks bobot W ini akan diperbarui selama proses pelatihan menggunakan algoritma *backpropagation*, sehingga *embedding* yang dihasilkan dapat menyesuaikan dengan pola linguistik yang relevan terhadap tugas klasifikasi. Pendekatan ini memberikan fleksibilitas tinggi karena *embedding* yang dihasilkan bersifat spesifik terhadap domain dataset yang digunakan (Goodfellow et al., 2016).

Parameter input dim menunjukkan jumlah kata unik, output dim adalah ukuran *embedding* (misalnya 17, 22, 38, 50, 100), dan *input length* adalah panjang maksimum teks. Output dari lapisan *embedding* akan diteruskan ke lapisan jaringan seperti CNN, LSTM, atau MLP untuk proses klasifikasi. Pendekatan ini memiliki keunggulan dalam adaptivitas. Karena bobot *embedding* diperbarui selama pelatihan, representasi kata dapat berubah sesuai konteks dan kebutuhan model. Dalam penelitian NLP terkini, *contextual fine-tuning* seperti ini terbukti efektif dalam domain-domain khusus, termasuk kesehatan dan biomedis (Lee et al., 2020).

Dalam penelitian ini, Keras *Embedding Layer* digunakan untuk membandingkan performa *embedding* trainable dengan *embedding* static Word2Vec. Dengan dimensi *embedding* yang sama (17–100), perbandingan ini membantu menganalisis sejauh mana model dapat belajar representasi sendiri dibanding menggunakan *embedding* yang sudah dilatih sebelumnya. Pendekatan ini relevan untuk Bahasa Indonesia, yang memiliki kekayaan morfologis tinggi dan masih relatif kurang sumber daya pra-latih (Koto et al., 2020). Keras *Embedding Layer* juga memungkinkan eksplorasi efek dimensionality terhadap kinerja model. (Trnecka and Trneckova, 2021) menemukan bahwa *embedding* berukuran menengah (sekitar 50–100 dimensi) memberikan keseimbangan terbaik antara representasi semantik dan efisiensi pelatihan. Namun, *embedding* yang terlalu tinggi dapat menyebabkan *overfitting* terutama pada dataset yang tidak terlalu besar.

Dari sisi integrasi, *embedding layer* Keras juga dapat digunakan bersama teknik

regularization seperti *Dropout* untuk mencegah *overfitting* dan memperkuat generalisasi model (Srivastava et al., 2014). Hal ini sangat relevan dalam penelitian ini, mengingat model CNN dan LSTM akan dilatih pada data augmentasi yang bervariasi hasil sinonim dari Kateglo API. Dengan menggunakan *embedding* yang dilatih langsung, model mampu menangkap nuansa linguistik yang unik dari teks COVID-19 dalam Bahasa Indonesia. Proses ini diharapkan meningkatkan sensitivitas model terhadap konteks semantik kalimat, terutama dalam membedakan kategori *event* dan *non-event*. Selain itu, hasil perbandingan dengan Word2Vec dapat memberikan pemahaman empiris terhadap kelebihan *embedding* dinamis dalam aplikasi domain kesehatan publik.

2.6 Metrik Evaluasi

Dalam penelitian ini, evaluasi kinerja model dilakukan menggunakan beberapa metrik utama, yaitu *Accuracy*, *Precision*, *Recall*, *F1-Score*, dan *ROC-AUC*. Setiap metrik digunakan untuk memberikan gambaran menyeluruh mengenai kemampuan model dalam mengklasifikasikan data dengan benar serta mengukur keseimbangan antara prediksi positif dan negatif.

2.6.1 *Accuracy*

Akurasi merupakan ukuran yang menunjukkan seberapa besar proporsi prediksi yang benar terhadap keseluruhan prediksi yang dilakukan model. Nilai akurasi memberikan informasi umum tentang performa model dalam mengenali pola data secara keseluruhan (Powers, 2020). Rumus akurasi ditunjukkan sebagai berikut:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Di mana:

- **TP (True Positive)**: jumlah prediksi positif yang benar,
- **TN (True Negative)**: jumlah prediksi negatif yang benar,
- **FP (False Positive)**: jumlah prediksi positif yang salah,
- **FN (False Negative)**: jumlah prediksi negatif yang salah.

Semakin tinggi nilai akurasi, semakin baik kemampuan model dalam mengklasifikasikan data secara benar.

2.6.2 *Precision*

Presisi mengukur ketepatan model dalam memprediksi kelas positif. Nilai presisi tinggi menunjukkan bahwa sebagian besar data yang diprediksi positif memang benar-benar positif. Rumusnya dituliskan sebagai berikut:

$$Precision = \frac{TP}{TP + FP}$$

Presisi penting terutama pada kasus di mana kesalahan dalam memprediksi positif (*false positive*) harus diminimalkan, misalnya dalam diagnosis medis atau deteksi spam (Bishop and Nasrabadi, 2006).

2.6.3 *Recall*

Recall atau sensitivitas mengukur sejauh mana model mampu mengenali seluruh data positif yang ada di dataset. Nilai *recall* yang tinggi menandakan kemampuan model untuk menghindari kesalahan tipe *False Negative*. Rumus recall dinyatakan sebagai:

$$Recall = \frac{TP}{TP + FN}$$

Semakin besar nilai *recall*, semakin baik model dalam mendeteksi semua instance positif dalam dataset (Fawcett, 2006).

2.6.4 *F1-Score*

F1-Score merupakan rata-rata harmonik dari presisi dan *recall*, yang berguna untuk menilai keseimbangan antara keduanya. Metrik ini sangat penting ketika terdapat ketidakseimbangan data antar kelas. Rumus F1-*Score* adalah sebagai berikut:

$$F1\text{-}Score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

F1-Score memberikan gambaran yang lebih adil terhadap performa model ketika jumlah data positif dan negatif tidak seimbang (Bishop and Nasrabadi, 2006).

2.6.5 ROC-AUC

Receiver Operating Characteristic - Area Under Curve (ROC-AUC) digunakan untuk menilai kemampuan model dalam membedakan antara kelas positif dan negatif. ROC menggambarkan hubungan antara *True Positive Rate (TPR)* dan *False Positive Rate (FPR)*, sedangkan AUC menunjukkan luas area di bawah kurva tersebut. Rumus perhitungan dasarnya adalah:

$$TPR = \frac{TP}{TP + FN}$$

$$FPR = \frac{FP}{FP + TN}$$

Nilai AUC yang mendekati 1 menunjukkan bahwa model memiliki performa klasifikasi yang sangat baik, sedangkan nilai mendekati 0.5 menunjukkan performa yang sebanding dengan tebakan acak (Fawcett, 2006).

BAB III

METODE PENELITIAN

3.1 Waktu dan Tempat Penelitian

Penelitian ini dilakukan pada semester genap pada tahun pelajaran 2024/2025 dan semester ganjil pada tahun pelajaran 2025/2026 di BRIN KST Samaun Samadi kun Bandung dan Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.

3.2 Data Penelitian

Dataset yang digunakan dalam penelitian ini berasal dari Corpus InaCOVED (*Indonesian Corpus for COVID-19 Event Detection*), yaitu korpus teks berbahasa Indonesia yang dikembangkan untuk mendeteksi *event* atau kejadian terkait pandemi COVID-19 pada berita daring. Dataset ini diperoleh melalui Badan Riset dan Inovasi Nasional (BRIN) yang mengembangkan korpus InaCOVED, yang dikumpulkan dari berbagai portal berita nasional Indonesia seperti Tirto, Tempo, Republika, Merdeka, Kompas, Detik, dan Antara dengan periode publikasi Januari hingga Mei 2020.

Korpus InaCOVED berisi kumpulan teks berita yang telah diberi label ke dalam dua kategori utama, yaitu *event* dan *non-event*.

- Kategori *event* mencakup berita yang menggambarkan kejadian nyata yang terjadi selama masa pandemi COVID-19, seperti laporan kasus positif, kebijakan pemerintah, kegiatan vaksinasi, atau peristiwa sosial yang benar-benar terjadi.
- Kategori *non-event* berisi berita atau informasi yang tidak merepresentasikan kejadian nyata.

Pemilihan dataset InaCOVED didasarkan pada beberapa pertimbangan utama. Pertama, korpus ini memiliki cakupan linguistik yang luas dengan variasi gaya bahasa jurnalistik, sehingga mampu merepresentasikan karakteristik teks berita Indonesia secara autentik. Kedua, dataset ini telah melalui proses pelabelan manual menjadi dua kategori utama, yaitu *event* dan *non-event*, yang memudahkan dalam penerapan model klasifikasi berbasis *deep learning*. Ketiga, konteks pandemi COVID-19 menjadikan dataset ini relevan secara tematik terhadap isu kesehatan publik, sejalan dengan fokus penelitian ini dalam mengidentifikasi peristiwa yang berkaitan dengan domain kesehatan.

Selain itu, InaCOVED telah digunakan dalam sejumlah penelitian sebelumnya yang berfokus pada deteksi peristiwa dan analisis semantik, sehingga validitasnya sebagai korpus penelitian dapat dipertanggungjawabkan. Dengan karakteristik tersebut, dataset InaCOVED dinilai representatif untuk pengujian model berbasis NER dan augmentasi data dalam tugas klasifikasi teks.

3.3 Metode Penelitian

Penelitian ini menggunakan metode studi literatur dan eksperimen komputasional yang berfokus pada penerapan NER dan augmentasi data terhadap peningkatan kinerja model *deep learning* untuk klasifikasi teks. Sumber literatur yang digunakan berasal dari jurnal ilmiah, skripsi terdahulu, serta buku-buku yang relevan. Penelitian ini juga memanfaatkan berbagai sumber daring terpercaya sebagai referensi tambahan untuk memperkuat landasan teoritis dan metodologis penelitian.

Adapun langkah-langkah yang digunakan dalam penyusunan dan pelaksanaan penelitian ini antara lain:

1) Mempelajari arsitektur dan konsep algoritma yang digunakan dalam penelitian, termasuk MLP, CNN, dan LSTM sebagai model deep learning, serta metode representasi teks seperti Word2Vec dan Keras *Embedding Layer*.

2) Mengidentifikasi permasalahan dan menentukan tujuan penelitian, yaitu untuk menganalisis pengaruh penerapan NER dan augmentasi data terhadap kinerja model *deep learning* dalam klasifikasi teks.

3) Merancang algoritma dan alur penelitian berdasarkan konsep yang telah dipelajari, yang meliputi tiga skenario utama:

- Skenario 1: Skenario pertama menggunakan data mentah dari dataset InaCOVED tanpa penerapan NER maupun augmentasi data. Tahapan dimulai dari *preprocessing*, setelah teks bersih, data langsung diubah menjadi representasi vektor menggunakan dua metode *embedding*, yaitu Word2Vec dan Keras *Embedding Layer*, dengan variasi dimensi (17, 22, 38, 50, dan 100). Hasil representasi vektor ini kemudian dilatih dan diuji menggunakan tiga model *deep learning* MLP, CNN, dan LSTM.
- Skenario 2: Skenario kedua menggunakan NER berbasis BERT untuk mengekstraksi entitas penting, yaitu *Person*, *Organization*, *Location*, dan *Disease*. Model BERT diterapkan secara otomatis tanpa *prompting*, melalui mekanisme token *classification* untuk mendeteksi posisi dan tipe entitas pada setiap token dalam teks. Hasil ekstraksi entitas dari BERT kemudian dijadikan dasar untuk proses augmentasi data menggunakan API Kateglo, di mana token dengan label "O" diganti dengan sinonim agar makna utama kalimat tetap terjaga. Dataset hasil augmentasi kemudian direpresentasikan dengan Word2Vec dan Keras *Embedding Layer*, sebelum akhirnya diklasifikasikan menggunakan tiga model *deep learning* (MLP, CNN, dan LSTM).
- Skenario 3:Skenario ketiga memiliki struktur alur yang serupa dengan skenario kedua, tetapi tahap NER dilakukan menggunakan model berbasis LLM, yaitu Gemini 1.5 dan Gemini 2.0 Flash. Proses NER dilakukan melalui pendekatan *prompt-based learning* dengan tiga jenis *prompting*: *zero-shot*, *one-shot*, dan *five-shot*, yang memungkinkan model mengenali entitas *Person*, *Organization*, *Location*, dan *Disease*. Setelah entitas berhasil diekstraksi, dilakukan augmentasi data berbasis sinonim menggunakan API Kateglo agar distribusi data *event* dan *non-event* menjadi seimbang. Data hasil augmentasi kemudian dikonversi menjadi representasi vektor menggunakan Word2Vec dan Keras *Embedding Layer*, lalu diklasifikasikan menggunakan MLP, CNN, dan LSTM.

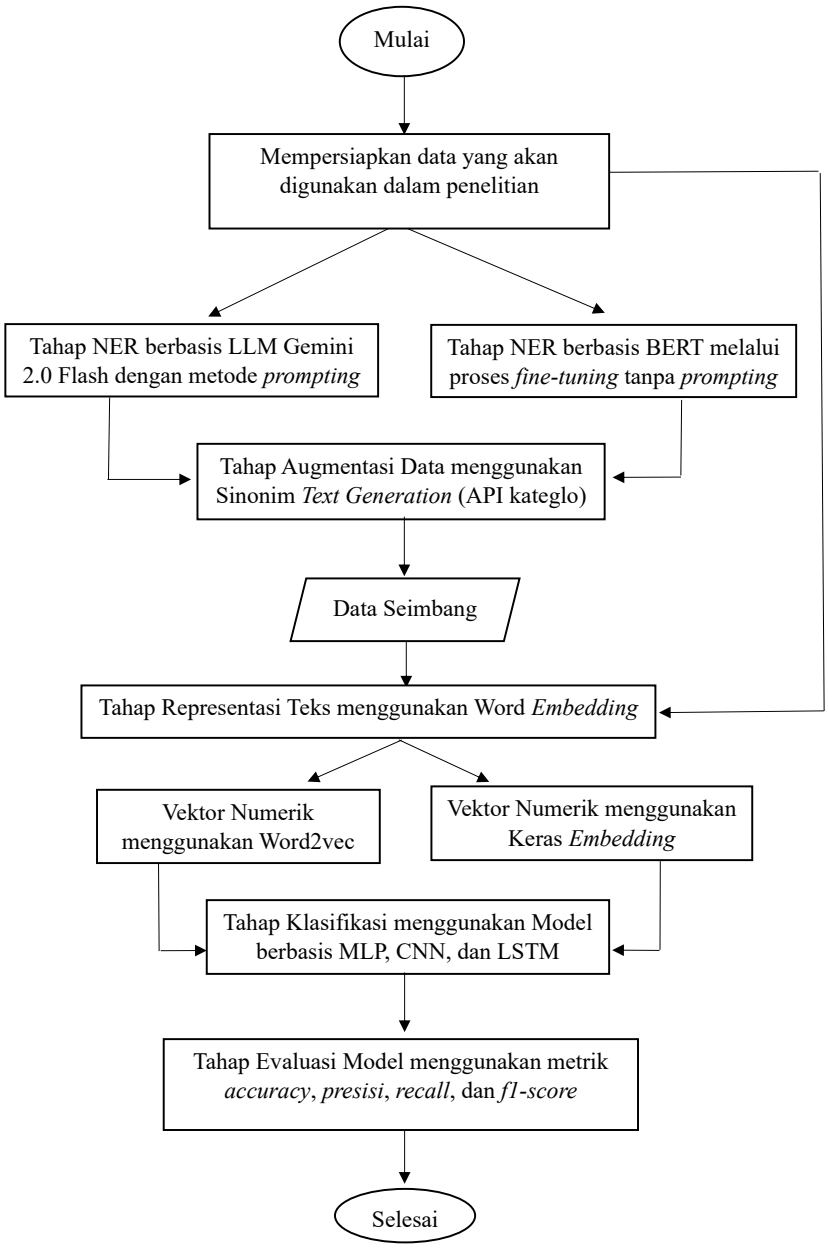
4) Menentukan parameter eksperimen, termasuk ukuran vektor (17, 22, 38, 50, dan 100) yang digunakan untuk membandingkan performa antara Word2Vec dan Keras *Embedding Layer* pada tiap model MLP, CNN, dan LSTM.

5) Melakukan proses pelatihan dan evaluasi model menggunakan metrik *accuracy*, *precision*, *recall*, *F1-score*, dan ROC-AUC untuk menilai efektivitas model pada masing-masing skenario.

6) Menganalisis dan membandingkan hasil eksperimen guna mengetahui sejauh mana pengaruh penerapan NER dan augmentasi data terhadap performa model *deep learning*.

7) Menyimpulkan hasil penelitian berdasarkan analisis performa model serta memberikan rekomendasi untuk penelitian selanjutnya.

Penelitian ini dilakukan menggunakan perangkat keras prosesor AMD Ryzen 5 7520U (2.8 GHz), memori 8 GB DDR5, dan penyimpanan SSD 256 GB. Perangkat lunak yang digunakan mencakup Google Colab dan pustaka Python pendukung analisis data serta pemodelan *deep learning*. Supaya memberikan gambaran yang lebih jelas mengenai proses penelitian ini, maka disiapkan flowchart yang menggambarkan keseluruhan alur penelitian, yang ditunjukkan pada Gambar 3.1.



Gambar 3.1 *Flowchart* alur proses klasifikasi

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan seluruh tahapan penelitian yang telah dilakukan, dapat disimpulkan bahwa penelitian ini berhasil mencapai tujuan yang telah dirumuskan, yaitu melakukan klasifikasi teks pada dataset InaCOVED ke dalam dua kategori, yaitu *event* dan *non-event*, menggunakan arsitektur MLP, CNN, dan LSTM. Hasil evaluasi menunjukkan bahwa ketiga model mampu melakukan klasifikasi dengan baik, namun kinerjanya dipengaruhi oleh perbedaan arsitektur, metode representasi teks, serta strategi augmentasi data yang diterapkan.

Penerapan augmentasi data berbasis sinonim menggunakan API Kateglo memberikan kontribusi dalam menyeimbangkan distribusi data antara kelas *event* dan *non-event* sekaligus meningkatkan variasi teks pada data latih. Variasi teks yang dihasilkan tetap mempertahankan makna utama kalimat sehingga membantu model mempelajari pola yang lebih beragam dan meningkatkan kemampuan generalisasi. Efektivitas augmentasi ini semakin terlihat ketika dikombinasikan dengan proses ekstraksi entitas, karena informasi semantik penting tetap terjaga.

Pemanfaatan NER berbasis BERT dan LLM terbukti mampu mengekstraksi entitas penting seperti *Person*, *Organization*, *Location*, dan *Disease* yang relevan dengan konteks berita kesehatan. Hasil penelitian menunjukkan bahwa NER berbasis LLM memberikan peningkatan performa yang lebih signifikan dibandingkan NER berbasis BERT maupun data mentah. Hal ini disebabkan oleh kemampuan LLM dalam menghasilkan representasi entitas dan variasi teks yang lebih kaya serta kontekstual, sehingga dapat memperkuat kualitas data hasil augmentasi dan meningkatkan performa model klasifikasi.

Dari seluruh kombinasi model, metode *embedding*, dan dimensi vektor yang diuji, diperoleh bahwa model CNN dengan Keras *Embedding Layer* pada data hasil NER berbasis LLM dengan dimensi vektor 100 merupakan konfigurasi terbaik dalam

penelitian ini. Model tersebut menghasilkan *accuracy* sebesar 0.9638, *precision* sebesar 0.9497, *recall* sebesar 0.9811, *F1-score* sebesar 0.9651, serta ROC-AUC sebesar 0.9756, yang menunjukkan performa klasifikasi paling optimal dibandingkan konfigurasi lainnya. Hal ini menunjukkan bahwa kombinasi Keras *Embedding Layer*, ekstraksi entitas menggunakan NER berbasis LLM, serta arsitektur CNN mampu menghasilkan representasi teks yang lebih informatif dan kontekstual.

Selain performa evaluasi, penelitian ini juga menganalisis efisiensi waktu *running* dari masing-masing model. Hasil pengujian menunjukkan bahwa model MLP memiliki waktu komputasi paling cepat karena arsitekturnya relatif sederhana dan tidak memproses urutan kata secara mendalam. Model CNN memiliki waktu komputasi yang lebih tinggi dibandingkan MLP, namun masih tergolong efisien karena mampu mengekstraksi fitur lokal teks secara paralel. Sementara itu, model LSTM memiliki waktu *running* paling lama karena harus memproses urutan kata secara berurutan sehingga membutuhkan komputasi yang lebih kompleks.

Secara keseluruhan, penelitian ini menunjukkan bahwa integrasi augmentasi data berbasis sinonim, ekstraksi entitas menggunakan NER khususnya NER berbasis LLM, serta pemilihan arsitektur dan metode representasi teks yang tepat mampu meningkatkan kinerja model klasifikasi teks secara signifikan. Dengan demikian, pendekatan yang diusulkan dalam penelitian ini dapat menjadi acuan dalam pengembangan sistem klasifikasi teks berbasis augmentasi data dan NER, khususnya pada domain berita kesehatan.

5.2 Saran

Penelitian selanjutnya disarankan untuk mengeksplorasi teknik augmentasi data yang lebih beragam, seperti parafrase otomatis atau pendekatan berbasis model bahasa generatif, guna memperkaya variasi teks tanpa mengurangi makna konteks. Selain itu, penggunaan model ekstraksi entitas yang lebih spesifik terhadap domain kesehatan dapat dipertimbangkan agar informasi yang dihasilkan semakin relevan. Pengembangan arsitektur model berbasis mekanisme perhatian atau pemanfaatan sumber daya komputasi yang lebih optimal juga berpotensi meningkatkan kinerja dan efisiensi sistem klasifikasi teks pada penelitian berikutnya.

Selain itu, penelitian selanjutnya juga dapat melakukan eksplorasi lebih lanjut terhadap variasi dimensi vektor embedding, khususnya pada rentang 100 hingga 300 dimensi, yang dalam berbagai penelitian sebelumnya sering digunakan untuk

memperoleh representasi semantik yang lebih kaya. Pengujian variasi dimensi yang lebih luas berpotensi menghasilkan konfigurasi model yang lebih optimal dalam meningkatkan performa klasifikasi teks.

DAFTAR PUSTAKA

Abdul-Mageed, M. and Ungar, L. (2017). Emonet: Fine-grained emotion detection with gated recurrent neural networks. In *Proceedings of the 55th annual meeting of the association for computational linguistics (volume 1: Long papers)*, pages 718–728.

Abu Lekham, L., Wang, Y., Hey, E., and Khasawneh, M. T. (2022). Multi-criteria text mining model for covid-19 testing reasons and symptoms and temporal predictive model for covid-19 test results in rural communities. *Neural Computing and Applications*, 34(10):7523–7536.

Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al. (2023). Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Adam, K. D. B. J. et al. (2014). A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 1412(6).

Alqaaidi, S. K., Bozorgi, E., Shams, A., and Kochut, K. (2023). A few-shot learning focused survey on recent named entity recognition and relation classification methods. *arXiv preprint arXiv:2310.19055*.

Alsentzer, E., Murphy, J., Boag, W., Weng, W.-H., Jindi, D., Naumann, T., and McDermott, M. (2019). Publicly available clinical BERT embeddings. In Rumshisky, A., Roberts, K., Bethard, S., and Naumann, T., editors, *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 72–78, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Aryal, R. R. and Bhattarai, A. (2021). Sentiment analysis on covid-19 vaccination tweets using naïve bayes and lstm. *Advances in Engineering and Technology: An International Journal*, 1(1):57–70.

Baltrušaitis, T., Ahuja, C., and Morency, L.-P. (2019). Multimodal machine learning: A survey and taxonomy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(2):423–443.

Bengio, Y., Simard, P., and Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5(2):157–166.

Bishop, C. M. and Nasrabadi, N. M. (2006). *Pattern recognition and machine learning*, volume 4. Springer.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. (2020). Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Burns, P. J., Brofos, J. A., Li, K., Chaudhuri, P., and Dexter, J. P. (2021). Profiling of intertextuality in Latin literature using word embeddings. In Toutanova, K., Rumshisky, A., Zettlemoyer, L., Hakkani-Tur, D., Beltagy, I., Bethard, S., Cotterell, R., Chakraborty, T., and Zhou, Y., editors, *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4900–4907, Online. Association for Computational Linguistics.

Cambria, E. and White, B. (2014). Jumping nlp curves: A review of natural language processing research [review article]. *IEEE Computational Intelligence Magazine*, 9(2):48–57.

Carr, B. and Silk, J. (2018). Primordial black holes as generators of cosmic structures. *Monthly Notices of the Royal Astronomical Society*, 478(3):3756–3775.

Chollet, F. (2021). *Deep learning with Python*. simon and schuster.

Davies, M., Srinivasa, N., Lin, T.-H., Chinya, G., Cao, Y., Choday, S. H., Dimou, G., Joshi, P., Imam, N., Jain, S., et al. (2018). Loihi: A neuromorphic manycore processor with on-chip learning. *Ieee Micro*, 38(1):82–99.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186.

Elhassan, N., Varone, G., Ahmed, R., Gogate, M., Dashtipour, K., Almoamari, H., El-Affendi, M. A., Al-Tamimi, B. N., Albalwy, F., and Hussain, A. (2023). Arabic

sentiment analysis based on word embeddings and deep learning. *Computers*, 12(6):126.

Fawcett, T. (2006). An introduction to roc analysis. *Pattern recognition letters*, 27(8):861–874.

Finkel, J. R., Grenager, T., and Manning, C. (2005). Incorporating non-local information into information extraction systems by Gibbs sampling. In Knight, K., Ng, H. T., and Oflazer, K., editors, *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 363–370, Ann Arbor, Michigan. Association for Computational Linguistics.

Gers, F. A., Schmidhuber, J., and Cummins, F. (2000). Learning to forget: Continual prediction with lstm. *Neural computation*, 12(10):2451–2471.

Goodfellow, I., Bengio, Y., Courville, A., and Bengio, Y. (2016). *Deep learning*, volume 1. MIT press Cambridge.

He, H. and Garcia, E. A. (2009). Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, 21(9):1263–1284.

Hedderich, M. A., Lange, L., Adel, H., Strötgen, J., and Klakow, D. (2021). A survey on recent approaches for natural language processing in low-resource scenarios. In Toutanova, K., Rumshisky, A., Zettlemoyer, L., Hakkani-Tur, D., Beltagy, I., Bethard, S., Cotterell, R., Chakraborty, T., and Zhou, Y., editors, *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2545–2568, Online. Association for Computational Linguistics.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

Holmgren, A. J., Apathy, N. C., and Adler-Milstein, J. (2020). Barriers to hospital electronic public health reporting and implications for the covid-19 pandemic. *Journal of the American Medical Informatics Association*, 27(8):1306–1309.

Huh, J., Yetisgen-Yildiz, M., and Pratt, W. (2013). Text classification for assisting moderators in online health communities. *Journal of biomedical informatics*, 46(6):998–1005.

Keraghel, I., Morbieu, S., and Nadif, M. (2024). A survey on recent advances in named entity recognition. *arXiv preprint arXiv:2401.10825*.

Khanam, Z., Alwasel, B., Sirafi, H., and Rashid, M. (2021). Fake news detection using machine learning approaches. In *IOP conference series: materials science and engineering*, volume 1099, page 012040. IOP Publishing.

Kim, Y. (2014). Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*.

Koto, F., Rahimi, A., Lau, J. H., and Baldwin, T. (2020). Indolem and indobert: A benchmark dataset and pre-trained language model for indonesian nlp. *arXiv preprint arXiv:2011.00677*.

Kurniawan, A. A. and Mustikasari, M. (2021). Implementasi deep learning menggunakan metode cnn dan lstm untuk menentukan berita palsu dalam bahasa indonesia. *Jurnal Informatika Universitas Pamulang*, 5(4):544–552.

Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., and Dyer, C. (2016). Neural architectures for named entity recognition. In Knight, K., Nenkova, A., and Rambow, O., editors, *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 260–270, San Diego, California. Association for Computational Linguistics.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *nature*, 521(7553):436–444.

Lecun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C. H., and Kang, J. (2020). Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Li, J., Sun, A., Han, J., and Li, C. (2020). A survey on deep learning for named entity recognition. *IEEE transactions on knowledge and data engineering*, 34(1):50–70.

Liu, C. (2024). Long short-term memory (lstm)-based news classification model. *Plos one*, 19(5):e0301835.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Liu, Y. and Zhang, M. (2018). Neural network methods for natural language processing.

Lu, H., Ehwerhemuepha, L., and Rakovski, C. (2022). A comparative study on deep learning models for text classification of unstructured medical notes with various levels of class imbalance. *BMC medical research methodology*, 22(1):181.

Ma, C., Zhang, W. E., Guo, M., Wang, H., and Sheng, Q. Z. (2022). Multi-document summarization via deep learning techniques: A survey. *ACM Computing Surveys*, 55(5):1–37.

Ma, H., Shen, L., Sun, H., Xu, Z., Hou, L., Wu, S., Fang, A., Li, J., and Qian, Q. (2021). Covid term: a bilingual terminology for covid-19. *BMC Medical Informatics and Decision Making*, 21(1):231.

Marcel, G., Cangemi, F., Rodriguez, J., Neilsen, J., Ferreira, J., Petrucci, P.-O., Malzac, J., Barnier, S., and Clavel, M. (2020). A unified accretion-ejection paradigm for black hole x-ray binaries-v. low-frequency quasi-periodic oscillations. *Astronomy & Astrophysics*, 640:A18.

Mayo, M., Wakes, S., and Anderson, C. (2018). Neural networks for predicting the output of wind flow simulations over complex topographies. In *2018 IEEE International Conference on Big Knowledge (ICBK)*, pages 184–191.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Nayoga, B. P., Adipradana, R., Suryadi, R., and Suhartono, D. (2021). Hoax analyzer for indonesian news using deep learning models. *Procedia Computer Science*, 179:704–712.

Ningsih, F. S. S., Khotimah, P. H., Arisal, A., Rozie, A. F., Munandar, D., Riswantini, D., Nugraheni, E., Suwarningsih, W., and Kurniasari, D. (2022). Synonym-based text generation in restructuring imbalanced dataset for deep learning models. In *2022 5th International Conference on Networking, Information Systems and Security: Envisage Intelligent Systems in 5g//6G-based Interconnected Digital Worlds (NISS)*, pages 1–6. IEEE.

Orson, P. (2017). Double l–groups and doubly slice knots. *Algebraic & Geometric Topology*, 17(1):273–329.

Parsing, C. (2009). Speech and language processing. *Power Point Slides*, page 20.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Persiani, M. and Hellström, T. (2019). Unsupervised inference of object affordance from text corpora. In Hartmann, M. and Plank, B., editors, *Proceedings of the 22nd Nordic Conference on Computational Linguistics*, pages 115–120, Turku, Finland. Linköping University Electronic Press.

Powers, D. M. (2020). Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation. *arXiv preprint arXiv:2010.16061*.

Qiu, X., Sun, T., Xu, Y., Shao, Y., Dai, N., and Huang, X. (2020). Pre-trained models for natural language processing: A survey. *Science China technological sciences*, 63(10):1872–1897.

Ramchoun, H., Ghanou, Y., Ettaouil, M., and Janati Idrissi, M. A. (2016). Multilayer perceptron: Architecture optimization and training.

Reimers, N. and Gurevych, I. (2019). Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Rong, X. (2014). word2vec parameter learning explained. *arXiv preprint arXiv:1411.2738*.

Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural networks*, 61:85–117.

Schütze, H., Manning, C. D., and Raghavan, P. (2008). *Introduction to information retrieval*, volume 39. Cambridge University Press Cambridge.

Seow, W. L., Chaturvedi, I., Hogarth, A., Mao, R., and Cambria, E. (2025). A review of named entity recognition: from learning methods to modelling paradigms and tasks. *Artificial Intelligence Review*, 58(10):1–87.

Shao, Y. and Nakashole, N. (2020). ChartDialogs: Plotting from Natural Language Instructions. In Jurafsky, D., Chai, J., Schluter, N., and Tetreault, J., editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3559–3574, Online. Association for Computational Linguistics.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958.

Stawska, S., Chmielewski, J., Bacharz, M., Bacharz, K., and Nowak, A. (2021). Comparative accuracy analysis of truck weight measurement techniques. *Applied Sciences*, 11(2):745.

Sulieman, L., Robinson, J. R., and Jackson, G. P. (2020). Automating the classification of complexity of medical decision-making in patient-provider messaging in a patient portal. *Journal of surgical research*, 255:224–232.

Suneera, C. and Prakash, J. (2020). Performance analysis of machine learning and deep learning models for text classification. In *2020 IEEE 17th India council international conference (INDICON)*, pages 1–6. IEEE.

Team, G., Anil, R., Borgeaud, S., Alayrac, J.-B., Yu, J., Soricut, R., Schalkwyk, J., Dai, A. M., Hauth, A., Millican, K., et al. (2023). Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Trnecka, M. and Trneckova, M. (2021). Model order selection for approximate boolean matrix factorization problem. *Knowledge-Based Systems*, 227:107184.

Tseng, B.-H., Bhargava, S., Lu, J., Moniz, J. R. A., Piraviperumal, D., Li, L., and Yu, H. (2021). CREAD: Combined resolution of ellipses and anaphora in dialogues. In Toutanova, K., Rumshisky, A., Zettlemoyer, L., Hakkani-Tur, D., Beltagy, I., Bethard, S., Cotterell, R., Chakraborty, T., and Zhou, Y., editors, *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3390–3406, Online. Association for Computational Linguistics.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.

Wang, Y. (2019). Single training dimension selection for word embedding with pca. *arXiv preprint arXiv:1909.01761*.

Wei, J. and Zou, K. (2019a). Eda: Easy data augmentation techniques for boosting performance on text classification tasks. *arXiv preprint arXiv:1901.11196*.

Wei, J. and Zou, K. (2019b). EDA: Easy data augmentation techniques for boosting performance on text classification tasks. In Inui, K., Jiang, J., Ng, V., and Wan, X., editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural*

Language Processing (EMNLP-IJCNLP), pages 6382–6388, Hong Kong, China. Association for Computational Linguistics.

Wilie, B., Vincentio, K., Winata, G. I., Cahyawijaya, S., Li, X., Lim, Z. Y., Soleman, S., Mahendra, R., Fung, P., Bahar, S., et al. (2020). Indonlu: Benchmark and resources for evaluating indonesian natural language understanding. *arXiv preprint arXiv:2009.05387*.

Yadav, V. and Bethard, S. (2018). A survey on recent advances in named entity recognition from deep learning models. In Bender, E. M., Derczynski, L., and Isabelle, P., editors, *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2145–2158, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Yana, A., Santoso, T., et al. (2020). Sentiment analysis of facebook comments on indonesian presidential candidates using the naïve bayes method. In *Journal of Physics: Conference Series*, volume 1641, page 012012. IOP Publishing.

Yin, W., Kann, K., Yu, M., and Schütze, H. (2017). Comparative study of cnn and rnn for natural language processing. *arXiv preprint arXiv:1702.01923*.

Yin, Z. and Shen, Y. (2018). On the dimensionality of word embedding. *Advances in neural information processing systems*, 31.

Young, T., Hazarika, D., Poria, S., and Cambria, E. (2018). Recent trends in deep learning based natural language processing [review article]. *IEEE Computational Intelligence Magazine*, 13(3):55–75.

Zaremba, W., Sutskever, I., and Vinyals, O. (2014). Recurrent neural network regularization. *arXiv preprint arXiv:1409.2329*.

Zhang, X., Zhao, J., and LeCun, Y. (2015). Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28.

Zhou, C., Sun, C., Liu, Z., and Lau, F. (2015). A c-lstm neural network for text classification. *arXiv preprint arXiv:1511.08630*.