

KLASIFIKASI PERTANYAAN KONSULTASI KESEHATAN ONLINE MENGGUNAKAN INDOBERT DAN WORD EMBEDDING FASTTEXT SEBAGAI VEKTOR INPUT PADA ARSITEKTUR BILSTM

TUGAS AKHIR

Diajukan untuk melengkapi tugas-tugas dan memenuhi syarat-syarat guna pelaksanaan penelitian Tugas Akhir

Oleh:

RAUDINA LUTHFIANI
1708107010030



**PROGRAM STUDI INFORMATIKA JURUSAN INFORMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN
ALAM
UNIVERSITAS SYIAH KUALA, BANDA ACEH
MEI, 2024**

PENGESAHAN

KLASIFIKASI PERTANYAAN KONSULTASI KESEHATAN ONLINE MENGGUNAKAN INDOBERT DAN WORD EMBEDDING FASTTEXT SEBAGAI VECTOR INPUT PADA ASRITEKTUR BILSTM

QUESTION CLASSIFICATION FOR ONLINE HEALTH CONSULTATION USING INDOBERT AND WORD EMBEDDING FASTTEXT AS INPUT VECTOR ON BILSTM ARCHITECTURE

Oleh:

Nama : Raudina Luthfiani
NPM : 1708107010030
Jurusan : Informatika

Menyetujui:

Pembimbing I,

Pembimbing II,

Prof. Dr. Taufik Fuadi Abidin, S.Si., M.Tech.
NIP. 197010081994031002

Nazaruddin, S.Si., M.Eng.Sc.
NIP. 197202061997021001

Mengetahui:

Dekan Fakultas MIPA
Universitas Syiah Kuala,

Ketua Program Studi Informatika
FMIPA Universitas Syiah Kuala,

Prof. Dr. Taufik Fuadi Abidin, S.Si., M.Tech.
NIP. 197010081994031002

Alim Misbullah, S.Si., M.S.
NIP. 1988806032019031011

Lulus seminar hasil pada hari ..., ... 2024

ABSTRAK

Internet telah menjadi media komunikasi utama untuk mengakses informasi atau berita yang digunakan oleh banyak orang saat ini. Salah satu informasi yang banyak diakses oleh pengguna internet adalah informasi terkait kesehatan. Kemudahan akses untuk mendapatkan informasi kesehatan mendorong pengguna internet untuk memilih mengandalkan informasi kesehatan yang ada di internet sebelum mencari bantuan medis ketika terdapat kekhawatiran terhadap kondisi kesehatan mereka. Di tambah lagi, perkembangan Teknologi Informasi Komunikasi (TIK) secara pesat membuat pelayanan kesehatan secara jarak jauh semakin mungkin dilakukan sehingga meningkatkan efisiensi dan memperluas akses layanan kesehatan kepada masyarakat. Salah satu layanan kesehatan jarak jauh yang mengalami peningkatan penggunaan bagi masyarakat saat ini adalah layanan konsultasi kesehatan dengan tenaga medis dalam bentuk forum *online* yang tersedia di situs web kesehatan. Namun, layanan konsultasi kesehatan *online* yang ada umumnya menggunakan *annotator* manual untuk mengelompokkan pertanyaan pengguna ke dalam subtopik yang sesuai dengan spesialisasi medis sehingga membutuhkan waktu untuk pertanyaan sampai kepada dokter spesialis atau tenaga medis yang berkeahlian khusus untuk menangani pertanyaan tersebut. Akibatnya, pengguna membutuhkan waktu yang tidak sebentar untuk menerima jawaban dari pertanyaan konsultasi mereka, apalagi jika pertanyaan yang masuk ke dalam forum layanan tersebut tidak sedikit. Oleh karena itu, pengklasifikasian pertanyaan konsultasi kesehatan secara otomatis dapat membantu meningkatkan efisiensi waktu untuk pertanyaan sampai kepada tenaga medis dan jawaban dari pertanyaan tersebut sampai kepada pengguna. Penelitian ini melakukan klasifikasi pertanyaan pengguna konsultasi kesehatan menggunakan model berbasis *deep learning*, yaitu arsitektur BiLSTM. Penelitian ini menguji dua pendekatan representasi kata, yaitu *word embedding* fastText dan *contextual embedding* IndoBERT sebagai vektor *input* BiLSTM untuk menentukan pendekatan mana yang berkinerja tinggi dalam hal pembelajaran untuk tugas klasifikasi teks pertanyaan dalam domain kesehatan. Pada pelatihan model, penelitian ini menggunakan tiga skenario untuk penetapan nilai *learning rate*, yaitu 0.001, 0.0001, dan 0.00001 dan dua skenario *batch size*, yaitu 16 dan 32. Hasil penelitian menunjukkan bahwa model BiLSTM+IndoBERT mencapai kinerja tertinggi dengan akurasi dan *F1 score* masing-masing sebesar, 90.67% dan 90.71% dengan parameter terbaik untuk kedua model adalah *learning rate* 0.001 dan *batch size* 32.

Kata Kunci: klasifikasi teks, representasi teks, *natural language processing*, *word embedding*, *contextual embedding*, fastText, IndoBERT, *neural networks*, BiLSTM

ABSTRACT

The Internet has become the primary media for accessing information and news for many people today. Health-related information is one of the most frequently accessed types of content by internet users. The ease of access to health information encourages individuals to rely on it before seeking medical help when they have health concerns. Furthermore, the rapid development of Information and Communication Technology (ICT) has made telemedicine increasingly feasible, enhancing efficiency and broadening access to healthcare services. One telemedicine service that has seen increased usage is *online* health consultations with medical professionals through forums available on health websites. However, existing *online* health consultation services typically use manual annotators to categorize user questions into subtopics corresponding to medical specialties, causing delays in routing questions to the appropriate specialists. Consequently, users may experience significant wait times for responses, especially when the volume of questions is high. Therefore, automating the classification of health consultation questions can improve the efficiency of directing questions to medical professionals and reduce response times. This study employs a deep learning-based model, specifically the BiLSTM architecture, to classify user health consultation questions. We investigate two word representation approaches—fastText word embeddings and IndoBERT contextual embeddings—as input vectors for the BiLSTM to determine which approach yields higher performance for text classification in the health domain. The training process explores three learning rate scenarios (0.001, 0.0001, and 0.00001) and two batch size scenarios (16 and 32). The results indicate that the BiLSTM+IndoBERT model achieves the highest performance with an accuracy and F1 score of 90.67% and 90.71%, respectively. The optimal parameters for both models are a learning rate of 0.001 and a batch size of 32.

Keywords: text classification, text representations, natural language processing, word embedding, contextual embedding, fastText, IndoBERT, neural networks, BiLSTM

KATA PENGANTAR

Dengan menyebut nama Allah Yang Maha Pemurah lagi Maha Penyayang. Segala puji bagi Allah Tuhan Semesta Alam atas segala rahmat, rida, berkah, dan karunia-Nya sehingga penulis dapat menyelesaikan Tugas Akhir yang berjudul **“Klasifikasi Pertanyaan Konsultasi Kesehatan Online Menggunakan IndoBERT dan Word Embedding FastText sebagai Vektor Input pada Arsitektur BiLSTM”**. Shalawat serta salam tidak lupa penulis hanturkan kepada junjungan Nabi Muhammad SAW beserta keluarga beliau yang telah menjadi suri tauladan yang baik bagi umat manusia karena berkat beliau kondisi umat kala itu yang penuh dengan kebodohan telah berubah menjadi umat yang memiliki prinsip, akhlak, dan ilmu pengetahuan seperti kondisi yang dirasakan saat ini.

Tugas Akhir ini dilakukan dalam rangka memenuhi salah satu syarat untuk memperoleh gelar Sarjana Komputer di Jurusan Informatika Fakultas Matematika dan Ilmu Pengetahuan alam Universitas Syiah Kuala. Penulis menyadari bahwa dalam menyelesaikan penulisan Tugas Akhir ini tidak lepas dari doa, dukungan, semangat, dan bimbingan baik secara moril maupun materil sehingga Tugas Akhir ini dapat terselesaikan. Melalui kesempatan ini, penulis ingin menyampaikan rasa hormat dan terima kasih yang tidak terhingga kepada semua yang telah mendukung penulis dalam menyelesaikan Tugas Akhir ini, yaitu kepada:

1. Ayah dr. Indra Lutfi dan Mamak Fitriani Umar S.Ag. selaku orang tua penulis yang selalu tulus, sabar, dan tanpa pamrih mendukung penulis dalam bentuk apapun. Terima kasih atas rida, doa, kesabaran, kepercayaan, dan kekuatan yang diberikan kepada penulis sehingga penulis bisa berada pada kondisi yang sangat baik sampai hari ini. Pengorbanan Ayah dan Mamak tidak bisa penulis balas dalam bentuk apapun dan akan selalu penulis ingat sampai kapanpun.
2. Bapak Prof. Dr. Taufik Fuadi Abidin, S.Si., M.Tech. selaku Dekan Fakultas MIPA Universitas Syiah Kuala dan selaku Pembimbing I yang telah meluangkan waktu, tenaga, dan pikiran dalam membimbing dan mengarahkan penulis dalam penyelesaian Tugas Akhir.

3. Bapak Alim Misbullah, S.Si., M.Sc. selaku Ketua Program Studi Informatika Fakultas MIPA Universitas Syiah Kuala yang telah banyak membantu dan memberikan dorongan kepada penulis dalam penyelesaian Tugas Akhir.
4. Bapak Nazaruddin, S.Si., M.Eng.Sc. selaku Pembimbing II yang telah meluangkan waktu, tenaga, dan pikiran dalam membimbing dan mengarahkan penulis dalam penyelesaian Tugas Akhir.
5. Bapak Razief Perucha Fauzie Afidh, S.Si., M.Sc. dan Ibu Laina Farsiah, S.Si., M.S. selaku Dosen Penguji telah meluangkan waktu, tenaga, dan pikiran dalam membimbing dan mengarahkan penulis dalam penyelesaian Tugas Akhir.
6. Bapak Rasudin M.Si., M.Info.Tech. selaku Dosen Wali dan Pembimbing Akademik yang telah mendidik dan memberikan dorongan akademik kepada penulis selama perkuliahan.
7. Ibu Sri Azizah Nazhifah, S.Kom., M.Sc. selaku Koordinator Tugas Akhir Program Studi Informatika yang telah banyak membantu dan memberikan dorongan kepada penulis dalam penyelesaian Tugas Akhir.
8. Seluruh Dosen Program Studi Informatika yang telah sabar mendidik, memimbing, dan memberikan ilmu pengetahuan serta inspirasi bagi penulis selama perkuliahan.
9. Seluruh Staf Administrasi Jurusan Informatika dan Laboran yang telah banyak membantu penulis dalam proses administrasi penyelesaian Tugas Akhir.
10. Muhammad Farhan Luthfi, Muhammad Fakhri Luthfi, Muhammad Fariz Luthfi, dan Muhammad Fakhruna Rasyid selaku adik-adik penulis yang senantiasa memberikan semangat dan dukungan kepada penulis.
11. Seluruh sahabat dan teman seperjuangan Informatika Angkatan 2017 yang telah banyak memberikan dukungan dan bantuannya dari awal perkuliahan hingga saat ini, terkhusus untuk Fitri Mauizah, Husna Yulia Suwandi, Naila Suqya, Najli Fadhlina, Natasya Azzahra, Mauliza Ananda, Miftahul Jannah, Raisya Husna Agustin, Qamara Anggini, Thari Annisa, Wirdayani, dan Zahriana.
12. Alna Helmarisa, Arifa Farhani, Bulqisa Putri Gawa, Elsal Sabila Afirda, Mardhatillah, Ratu Ikramah Rizki, dan Reny Octavira selaku sahabat penulis yang selalu setia memberikan dukungan kepada penulis hingga Tugas Akhir ini selesai.

13. Semua pihak yang tidak bisa penulis sebutkan namanya satu per satu yang telah memberikan semangat, dukungan, dan nasihat dalam menyelesaikan Tugas Akhir.

Penulis menyadari atas keterbatasan pengetahuan, pengalaman, dan kemampuan dalam penulisan maupun pemahaman teori sehingga Tugas Akhir ini tidak luput dari kekurangan dan ketidaksempurnaan. Penulis berharap Tugas Akhir ini dapat bermanfaat bagi perkembangan ilmu pengetahuan.

Banda Aceh, 6 Oktober 2023

Raudina Luthfiani
1708107010030

DAFTAR ISI

	Halaman
PENGESAHAN.....	ii
ABSTRAK	iii
ABSTRACT.....	iv
KATA PENGANTAR.....	v
DAFTAR ISI	viii
DAFTAR TABEL	x
DAFTAR GAMBAR.....	xi
DAFTAR LAMPIRAN.....	xiii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	6
1.3 Tujuan Penelitian	6
1.4 Manfaat Penelitian	6
BAB II TINJAUAN KEPUSTAKAAN.....	8
2.1 <i>Natural Language Processing</i>	8
2.2 <i>Machine Learning</i>	10
2.3 <i>Deep Learning</i>	13
2.4 <i>Neural Network</i>	13
2.5 <i>Text Mining</i>	19
2.6 Klasifikasi Teks.....	21
2.7 <i>Word Embedding</i>	23
2.8 FastText.....	24
2.9 <i>Bidirectional Encoder Representations from Transformers (BERT)</i>	26
2.10 IndoBERT	29
2.11 <i>Long Short-Term Memory (LSTM)</i>	31
2.12 Pengukuran Kinerja Model	36
BAB III METODE PENELITIAN.....	39
3.1 Tempat Dan Waktu Penelitian	39
3.1.1 Tempat Penelitian	39
3.2 Alat Dan Bahan	39
3.2.1 Alat	39
3.2.2 Bahan	39
3.3 Metode Penelitian.....	40
3.3.1 Studi Literatur	40
3.3.2 PembangunanKumpulan Data.....	40
3.3.3 <i>Preprocessing</i> Data	43
3.3.4 Pembangunan Model <i>Embedding</i>	44
3.3.5 Pembangunan Model Klasifikasi BiLSTM.....	49
3.3.6 Evaluasi Model.....	52

BAB IV HASIL DAN PEMBAHASAN	53
4.1 Pembangunan Kumpulan Data.....	53
4.1.1 Kumpulan Data Berlabel	53
4.1.2 Kumpulan Data Tidak Berlabel	55
4.2 Pelabelan Kumpulan Data	56
4.3 <i>Text Preprocessing</i> Kumpulan Data	57
4.4 Penerapan Model <i>Word Embedding</i> FastText.....	58
4.5 Pemisahan Kumpulan Data	60
4.6 Pembangunan Model Klasifikasi BiLSTM+FastText.....	62
4.6.1 Lapisan <i>Word Encoder</i>	62
4.6.2 Lapisan BiLSTM	64
4.7 Pelatihan Model BiLSTM+FastText.....	66
4.8 Pengujian Model BiLSTM+FastText.....	72
4.9 Pembangunan Model Klasifikasi BiLSTM+IndoBERT	85
4.9.1 Persiapan Lapisan <i>Word Encoder</i>	85
4.9.2 Lapisan <i>Encoder</i> dan BiLSTM.....	87
4.10 Pelatihan Model BiLSTM+IndoBERT	89
4.11 Pengujian Model BiLSTM+IndoBERT	95
4.12 Perbandingan Kinerja Model Klasifikasi	108
BAB V KESIMPULAN DAN SARAN.....	112
5.1 Kesimpulan	112
5.2 Saran	112
DAFTAR KEPUSTAKAAN	114
LAMPIRAN	118

DAFTAR TABEL

Tabel 3.1. Distribusi pertanyaan dan kategori kumpulan data	41
Tabel 3.2. Contoh kumpulan data berlabel	42
Tabel 3.3. Contoh penerapan case folding.....	44
Tabel 3.4. Contoh penerapan tokenisasi	44
Tabel 3.5. Parameter <i>word embedding</i> fastText	46
Tabel 3.6. Model klasifikasi teks yang diusulkan.....	50
Tabel 3.7. Pengaturan hyperparameter BiLSTM.....	51
Tabel 4.1. Perbandingan label dalam bentuk teks dan numerik.....	64
Tabel 4.2. Kinerja model pelatihan BiLSTM+FastText untuk setiap percobaan	72
Tabel 4.3. Perbandingan metrik kinerja hasil percobaan model BiLSTM+FastText	83
Tabel 4.4. Perbandingan metrik kinerja hasil percobaan model BiLSTM+IndoBERT	107
Tabel 4.5. Perbandingan metrik kinerja <i>baseline</i> dengan model BiLSTM menggunakan fastText dan IndoBERT.....	109
Tabel 4.6. Perbandingan kinerja model berdasarkan kategori spesialisasi kedokteran	110

DAFTAR GAMBAR

Gambar 2.1. Tahapan analisis dalam pemrosesan bahasa alami (Dale, 2010).	10
Gambar 2.2. Analogi neuron biologis dan tiruan.....	14
Gambar 2.3. Tahapan yang dilakukan oleh model neural network (Chollet, 2017)..	18
Gambar 2.4. Arsitektur Transformer (Devlin et al., 2019)	27
Gambar 2.5. Representasi input BERT (Devlin et al., 2019)	29
Gambar 2.6. Hasil performa model pada IndoNLU benchmark (Wilie et al., 2020)	30
Gambar 2.7. Unfolded RNN berdasarkan timesteps (Graves, 2012).....	32
Gambar 2.8. Struktur LSTM.....	33
Gambar 2.9. Bidirectional LSTM (Graves et al., 2013)	36
Gambar 2.10. Confusion Matrix (Lever et al., 2016)	37
Gambar 3.1. Diagram alir tahapan penelitian	40
Gambar 3.2. Proses tokenisasi menggunakan IndoBERT Tokenizer	47
Gambar 3.3 Contoh representasi input IndoBERT	48
Gambar 3.4 Input dan output dalam model IndoBERT	49
Gambar 3.5 Sekuens Input dalam lapisan <i>encoder</i> model IndoBERT	49
Gambar 3.6 Diagram Klasifikasi Teks pada BiLSTM (Xie et al., 2019)	50
Gambar 3.7 Arsitektur BiLSTM untuk tugas klasifikasi teks (Xie et al., 2019)	51
Gambar 4.1. Proses <i>selecting</i> bagian HTML yang berisi pertanyaan	54
Gambar 4.2. Kumpulan data berlabel disimpan dalam bentuk file CSV	54
Gambar 4.3. Proses <i>scraping</i> menggunakan ChromeDriver Selenium	55
Gambar 4.4. Situs web untuk pengumpulan data tidak berlabel	56
Gambar 4.5. Kumpulan data tidak berlabel disimpan dalam bentuk file CSV	56
Gambar 4.6. Visualisasi persebaran data pertanyaan dan kategori kumpulan data ...	57
Gambar 4.7. <i>Text Preprocessing</i> kumpulan data	57
Gambar 4.8. Proses pelatihan <i>word embedding</i> fastText.....	59
Gambar 4.9. Implementasi menampilkan vektor suatu kata.....	60
Gambar 4.10. Implementasi menampilkan kedekatan suatu kata dengan kata lain...	60
Gambar 4.11. Proses pemisahan kumpulan data	61
Gambar 4.12. Statistik kumpulan data <i>training</i> untuk setiap kategori	61
Gambar 4.13. Statistik kumpulan data <i>testing</i> untuk setiap kategori.....	61
Gambar 4.14. Kode program proses tokenisasi berdasarkan data <i>training</i>	63
Gambar 4.15. Kode Program proses menyimpan vektor <i>word embedding</i>	64
Gambar 4.16. Kode program untuk membangun dan melatih model klasifikasi BiLSTM+FastText.....	65
Gambar 4.17. Rangkuman model klasifikasi BiLSTM+FastText	66
Gambar 4.18. Grafik akurasi dan <i>loss</i> model BiLSTM+FastText dengan <i>lr</i> =0.001 dan <i>batch size</i> 16.....	67
Gambar 4.19. Grafik akurasi dan <i>loss</i> model BiLSTM+FastText dengan <i>lr</i> =0.001 dan <i>batch size</i> 32.....	67
Gambar 4.20. Grafik akurasi dan <i>loss</i> model BiLSTM+FastText dengan <i>lr</i> =0.0001 dan <i>batch size</i> 16.....	68
Gambar 4.21. Grafik akurasi dan <i>loss</i> model BiLSTM+FastText dengan <i>lr</i> =0.0001 dan <i>batch size</i> 32.....	68

Gambar 4.22. Grafik akurasi dan <i>loss</i> model BiLSTM+FastText dengan $lr=0.00001$ dan <i>batch size</i> 16.....	69
Gambar 4.23. Grafik akurasi dan <i>loss</i> model BiLSTM+FastText dengan $lr=0.00001$ dan <i>batch size</i> 32.....	69
Gambar 4.24. <i>Confusion matrix</i> model BiLSTM+FastText percobaan pertama.....	73
Gambar 4.25. Metrik kinerja model BiLSTM+FastText percobaan pertama.....	73
Gambar 4.26. <i>Confusion matrix</i> model BiLSTM+FastText percobaan kedua	74
Gambar 4.27. Metrik kinerja model BiLSTM+FastText percobaan kedua	74
Gambar 4.28. <i>Confusion matrix</i> model BiLSTM+FastText percobaan ketiga.....	76
Gambar 4.29. Metrik kinerja model BiLSTM+FastText percobaan ketiga.....	77
Gambar 4.30. <i>Confusion matrix</i> model BiLSTM+FastText percobaan keempat	77
Gambar 4.31. Metrik kinerja model BiLSTM+FastText percobaan keempat.....	78
Gambar 4.32. <i>Confusion matrix</i> model BiLSTM+FastText percobaan kelima.....	80
Gambar 4.33. Metrik kinerja model BiLSTM+FastText percobaan kelima.....	80
Gambar 4.34. <i>Confusion matrix</i> model BiLSTM+FastText percobaan keenam	81
Gambar 4.35. Metrik kinerja model BiLSTM+FastText percobaan keenam	81
Gambar 4.36. Kode Program <i>encoding</i> data <i>training</i> untuk representasi <i>input</i> model IndoBERT.....	86
Gambar 4.37. <i>Output</i> proses <i>encoding</i> data <i>training</i> untuk representasi <i>input</i> model IndoBERT.....	86
Gambar 4.38. Kode program untuk membangun model klasifikasi BiLSTM+IndoBERT	88
Gambar 4.39. Rangkuman model klasifikasi BiLSTM+IndoBERT.....	89
Gambar 4.40. Grafik akurasi dan <i>loss</i> model BiLSTM+FastText dengan $lr=0.001$ dan <i>batch size</i> 16.....	90
Gambar 4.41. Grafik akurasi dan <i>loss</i> model BiLSTM+FastText dengan $lr=0.001$ dan <i>batch size</i> 32.....	90
Gambar 4.42. Grafik akurasi dan <i>loss</i> model BiLSTM+FastText dengan $lr=0.0001$ dan <i>batch size</i> 16.....	92
Gambar 4.43. Grafik akurasi dan <i>loss</i> model BiLSTM+FastText dengan $lr=0.0001$ dan <i>batch size</i> 32.....	92
Gambar 4.44. Grafik akurasi dan <i>loss</i> model BiLSTM+FastText dengan $lr=0.00001$ dan <i>batch size</i> 16.....	94
Gambar 4.45. Grafik akurasi dan <i>loss</i> model BiLSTM+FastText dengan $lr=0.00001$ dan <i>batch size</i> 32.....	94
Gambar 4.46. <i>Confusion matrix</i> model BiLSTM+IndoBERT percobaan pertama ...	96
Gambar 4.47. Metrik kinerja model BiLSTM+IndoBERT percobaan pertama	96
Gambar 4.48. <i>Confusion matrix</i> model BiLSTM+IndoBERT percobaan kedua	97
Gambar 4.49. Metrik kinerja model BiLSTM+IndoBERT percobaan kedua	97
Gambar 4.50. <i>Confusion matrix</i> model BiLSTM+IndoBERT percobaan ketiga	100
Gambar 4.51. Metrik kinerja model BiLSTM+IndoBERT percobaan ketiga	100
Gambar 4.52. <i>Confusion matrix</i> model BiLSTM+IndoBERT percobaan keempat	101
Gambar 4.53. Metrik kinerja model BiLSTM+IndoBERT percobaan keempat	101
Gambar 4.54. <i>Confusion matrix</i> model BiLSTM+IndoBERT percobaan kelima ...	103
Gambar 4.55. Metrik kinerja model BiLSTM+IndoBERT percobaan kelima	104
Gambar 4.56. <i>Confusion matrix</i> model BiLSTM+IndoBERT percobaan keenam..	104
Gambar 4.57. Metrik kinerja model BiLSTM+IndoBERT percobaan keenam	105

DAFTAR LAMPIRAN

Lampiran 1. <i>Word Cloud</i> kumpulan data pada kategori Kandungan	118
Lampiran 2. <i>Word Cloud</i> kumpulan data pada kategori Kulit dan Kelamin	119
Lampiran 3. <i>Word Cloud</i> kumpulan data pada kategori Gigi dan Mulut	120
Lampiran 4. <i>Word Cloud</i> kumpulan data pada kategori Pencernaan	121
Lampiran 5. <i>Word Cloud</i> kumpulan data pada kategori Pernapasan.....	122
Lampiran 6. <i>Word Cloud</i> kumpulan data pada kategori Psikologi.....	123
Lampiran 7. <i>Word Cloud</i> kumpulan data pada kategori Lainnya.....	124
Lampiran 8. Hasil prediksi pertanyaan konsultasi kesehatan menggunakan model BiLSTM+FastText.....	125
Lampiran 9. Hasil prediksi pertanyaan konsultasi kesehatan menggunakan model BiLSTM+IndoBERT	136

BAB I

PENDAHULUAN

1.1 Latar Belakang

Internet telah menjadi media komunikasi utama yang digunakan oleh banyak orang saat ini. Berdasarkan survey yang dilakukan oleh Asosiasi Penyelenggara Jasa Internet Indonesia (APJII) terkait profil internet Indonesia pada tahun 2021-2022, tingkat penetrasi internet di Indonesia adalah sebesar 77,02% dari populasi Indonesia yaitu sebanyak 210.026.769 jiwa yang terkoneksi internet dari total 272.682.600 jiwa penduduk Indonesia di tahun 2021. Tingginya tingkat penetrasi internet di Indonesia berkaitan dengan kemudahan yang diperoleh oleh masyarakat untuk berbagi dan mendapatkan konten atau informasi mengenai berbagai topik. Hal ini ditunjukkan dengan 92,21% dari total jumlah penduduk Indonesia yang terkoneksi internet menggunakan internet sebagai media untuk mengakses informasi atau berita. Salah satu informasi yang dapat diakses melalui internet adalah informasi terkait kesehatan.

Informasi terkait kesehatan merupakan salah satu konten yang mengalami peningkatan minat bagi pengguna internet sekarang ini. Konten mengenai Kesehatan biasanya diperoleh oleh pengguna melalui situs web, blog, dan sosial media. Mudahnya akses untuk mendapatkan informasi terkait dengan kesehatan, seperti daftar penyakit dan obat, mendorong masyarakat memilih untuk mengandalkan informasi *online* tersebut sebelum mencari bantuan medis ketika terdapat kekhawatiran terhadap kondisi kesehatan mereka. Selain itu, Teknologi Informasi dan Komunikasi (TIK) yang berkembang pesat dalam beberapa tahun terakhir membuat pelayanan kesehatan secara jarak jauh semakin mungkin untuk dilakukan sehingga dapat meningkatkan efisiensi dan memperluas akses layanan kesehatan kepada masyarakat, seperti *telemedicine*. Terdapat banyak layanan kesehatan yang dapat dijangkau oleh masyarakat saat ini hanya dengan menggunakan internet melalui teknologi *telemedicine*, salah satu layanan yang sering digunakan oleh masyarakat adalah layanan telekonsultasi atau konsultasi kesehatan secara *online* dengan tenaga kesehatan.

Telekonsultasi adalah sebuah sistem tanya jawab antara dokter dengan pasien tentang berbagai topik kesehatan. Telekonsultasi sering digunakan dengan tujuan

untuk mendapatkan pengetahuan secara mendalam tentang berbagai penyakit dan gejalanya, penanganan terhadap gejala atau pencegahan pada penyakit tersebut, penggunaan obat-obatan beserta efek sampingnya, bahkan termasuk menanyakan tentang hal-hal yang dianggap tabu di masyarakat.

Salah satu layanan konsultasi kesehatan yang umum digunakan oleh masyarakat adalah layanan konsultasi kesehatan yang tersedia di situs web. Layanan konsultasi Kesehatan berbasis situs web memiliki dua model yaitu *community based* dan *patient-doctor based* (Hakim et al., 2017). Model *community based* memungkinkan pengguna untuk mengirimkan pertanyaan ke forum publik, di mana pertanyaan tersebut akan dijawab oleh siapapun yang berada di dalam forum tersebut meskipun mereka tidak memiliki keahlian dalam menangani kondisi atau penyakit tertentu. Sebaliknya, model *patient-doctor based* memungkinkan pertanyaan pengguna di forum *online* terkait kekhawatiran atau masalah kesehatan mereka dapat ditunjukan secara langsung kepada tim dokter sehingga yang akan menjawab pertanyaan di dalam forum adalah dokter yang disesuaikan pada kategori atau spesialisasi kedokteran tertentu berdasarkan pertanyaan pengguna. Pada penelitian ini hanya akan berpusat pada model layanan konsultasi kesehatan berbasis *patient-doctor*.

Situs web kesehatan biasanya memiliki kategori atau *tag* untuk mengelompokkan subtopik pertanyaan yang dikirimkan oleh pengguna ke forum konsultasi. Subtopik tersebut merujuk kepada spesialisasi terhadap suatu bidang atau bagian tubuh tertentu di dalam ilmu kedokteran sehingga pertanyaan yang termasuk ke dalam subtopik tersebut akan ditunjukan kepada dokter spesialis yang memiliki keahlian khusus untuk menangani kondisi atau penyakit tersebut. Saat ini, pengelompokan kumpulan pertanyaan ke dalam beberapa subtopik dilakukan oleh tim editor sebagai anotatornya. Pengklasifikasian yang dilakukan secara manual oleh manusia sangat memakan waktu dan mahal karena membutuhkan banyak tenaga manusia apalagi dengan jumlah data yang tidak sedikit. Selain itu, klasifikasi manual juga rentan terhadap kesalahan manusia yang disebabkan oleh inkonsisten dan ambiguitas. Oleh karena itu, untuk membantu mengurangi pekerjaan manual yang membutuhkan banyak usaha manusia dalam mengidentifikasi subtopik yang tepat dari kumpulan pertanyaan pengguna di forum konsultasi kesehatan *online* sehingga diperlukan sistem klasifikasi otomatis yang efektif dan berkinerja tinggi.

Klasifikasi teks adalah tugas penting dan mendasar dalam aplikasi *natural language processing* (NLP) yang bertujuan untuk memberikan kategori atau *tag* pada data tekstual, seperti kalimat, kueri, paragraf, dan dokumen. Pendekatan yang biasanya digunakan sebagai *baseline* atau dasar pada tugas klasifikasi adalah pengklasifikasi linier, seperti pendekatan berbasis aturan (*ruled based method*) dan pendekatan berbasis *machine learning* tradisional (Minaee et al., 2021). Namun, pendekatan tersebut sangat bergantung pada pemilihan fitur yang tepat untuk menghasilkan model yang berkinerja tinggi. Model klasifikasi linier memiliki ketergantungan pada fitur yang diekstrak secara manual berdasarkan pada kumpulan data sehingga dibutuhkan pengetahuan domain yang mendalam untuk merancang fitur.

Akhir-akhir ini, pendekatan yang didasarkan pada *neural networks* telah mencapai kesuksesan yang luar biasa dan semakin populer digunakan untuk tugas klasifikasi dalam beberapa tahun terakhir ini (Minaee et al., 2021). Pendekatan ini juga menunjukkan kemajuan kinerja yang signifikan dibandingkan dengan pendekatan tradisional. Pendekatan *neural networks* diperkenalkan untuk mengatasi keterbatasan penggunaan fitur yang diekstrak secara manual. Secara umum, semua pendekatan klasifikasi teks didasarkan pada kata-kata sebagai fiturnya sehingga tantangan bagi model klasifikasi teks adalah bagaimana merancang dan menangkap fitur-fitur terbaik dari unit tekstual. Model *word embedding* adalah komponen dasar dalam NLP untuk tugas klasifikasi teks yang dipelajari berdasarkan pendekatan *neural networks* yang memetakan teks ke dalam vektor kontinu berdimensi rendah yang dapat mengelompokkan kata-kata yang mirip secara semantik serta dapat merepresentasikan keteraturan dan pola sintaksis dan semantik kata. *Word embedding* juga dapat meningkatkan kinerja model klasifikasi *neural networks* ketika digunakan sebagai fitur atau vektor *input* (Wang et al., 2016). Model *word embedding* dipelajari menggunakan data pelatihan tidak berlabel dalam jumlah yang besar dan selanjutnya dapat digunakan sebagai vektor *input* pada berbagai tugas, seperti klasifikasi teks.

Pada tahun 2013, Google mengembangkan arsitektur word2vec yang terdiri dua model pembelajaran, yaitu *continuous bag-of-words* (CBOW) dan *skip-gram* yang dipelajari menggunakan enam miliar kata (Mikolov et al., 2013a). Word2vec memetakan kata-kata yang memiliki kesamaan makna atau semantik ke dalam ruang representasi vektor yang berdekatan. Word2vec mengandalkan informasi lokal dari

kata, di mana suatu kata tertentu dipengaruhi oleh kata-kata sekitarnya. Selanjutnya pada tahun 2014, Stanford memperkenalkan pendekatan *word embedding* baru yaitu GloVe yang dilatih menggunakan lima korpora dengan ukuran yang besar (Pennington et al., 2014). GloVe menangkap hubungan semantik dan sintaksis antar kata berdasarkan pola kemunculannya dalam teks menggunakan rasio probabilitas kemunculan kata-kata secara bersamaan berdasarkan statistik korpus secara global. Pada tahun 2017, Facebook memperkenalkan fastText yang merupakan perkembangan dari model *skip-gram word2vec*. FastText merupakan model berbasis karakter atau *subword* yang merepresentasikan setiap kata sebagai sekumpulan karakter n-gram sehingga model ini dapat menangkap informasi terkait morfologi suatu kata (Bojanowski et al., 2017). FastText dikembangkan untuk mengatasi kelemahan dari pendekatan *word2vec*, *gloVe*, dan model *embedding* lainnya yang mengabaikan struktur internal atau morfologi suatu kata sehingga tidak mampu mempelajari kata-kata di luar kosakata atau *out-of-vocabulary* (OOV) dan kata-kata langka (Khattak et al., 2019; Y. Zhang et al., 2019).

Pendekatan *word embedding* yang akan digunakan dalam penelitian ini adalah model fastText karena berdasarkan hasil yang ditunjukkan melalui penelitian yang dilakukan oleh Y. Zhang et al. (2019) pada domain Biomedis bahwa model *embedding* fastText mengungguli model *word embedding* lainnya yang populer saat ini, seperti *word2vec* dan *gloVe*. Penelitian ini menjelaskan bahwa jika model *word embedding* dapat menangkap informasi subkata dan memanfaatkan struktur internal kata untuk memperkaya hasil representasi suatu kata maka model ini dapat meningkatkan kualitas representasi kata tersebut dan juga dapat meningkatkan kinerja suatu model apabila digunakan sebagai vektor *input* untuk tugas ekstrinsik, seperti klasifikasi (Y. Zhang et al., 2019).

Selanjutnya pada tahun 2018, Google mengembangkan arsitektur BERT berdasarkan pendekatan Transformer dua arah. BERT menggunakan model *contextual embedding* yang memetakan setiap kata ke dalam vektor penyematan yang berbeda bergantung pada konteksnya (Devlin et al., 2019). Pada penelitian ini akan menggunakan pendekatan *contextual embedding* IndoBERT yang merupakan jenis dari arsitektur BERT yang telah diadaptasi ke dalam bahasa Indonesia yang dilatih menggunakan empat miliar kata bahasa Indonesia (Wilie et al., 2020).

Pengklasifikasian teks pertanyaan konsultasi kesehatan di forum *online* menghadirkan banyak tantangan karena pertanyaan ditulis oleh pengguna non-profesional, di mana pertanyaan ditulis dalam format teks bebas atau informal sehingga sifatnya tidak terstruktur dan sering kali ditemukan banyak singkatan, kesalahan pada ejaan dan tata bahasa serta teksnya yang ambigu. Terdapat beberapa pendekatan yang sudah pernah diusulkan untuk mengatasi tantangan tersebut, salah satunya adalah penelitian yang dilakukan oleh Hakim et al. (2017). Hakim et al. (2017) mengembangkan skema klasifikasi pertanyaan berdasarkan spesialisasi kedokteran untuk pertanyaan pengguna non-profesional pada konsultasi kesehatan *online*. Pendekatan *dictionary based* dan *machine learning based* digunakan pada penelitian ini untuk mengklasifikasikan teks pertanyaan ke dalam 13 kategori berdasarkan pada spesialisasi kedokteran, di mana spesialisasi yang digunakan mengacu pada daftar departemen klinis Fakultas Kedokteran Universitas Indonesia. Algoritma *machine learning* yang digunakan dalam penelitian ini adalah *Support Vector Machine* (SVM) dan *Naïve Bayes* (NB) dan terdapat tiga teknik ekstraksi fitur yang digunakan dalam penelitian ini, yaitu *n-grams*, *N* kata teratas pada setiap kategori, dan *controlled vocabulary* sebagai vektor *input* pada model klasifikasi teks. Kinerja tertinggi yang didapatkan dalam penelitian Hakim et al. (2017) melalui metrik evaluasi *F1 score* adalah 72,1%. Kinerja tertinggi ini dicapai oleh algoritma SVM dengan menerapkan dua fitur, yaitu 100 kata teratas di setiap kategori dan *controlled vocabulary*, di mana model diuji dengan *5-fold cross validation*. Pendekatan yang diusulkan dalam penelitian ini sangat bergantung pada pemilihan fitur yang tepat untuk menghasilkan model dengan kinerja tinggi. Selain itu, penelitian ini menggunakan data eksternal sebagai tambahan ketika mengekstraksi fitur, seperti istilah-istilah dalam bidang kesehatan sehingga diperlukan pengetahuan mengenai domain kesehatan untuk menghasilkan fitur yang dapat membantu meningkatkan kinerja model.

Berbeda dengan pendekatan yang dilakukan oleh Hakim et al. (2017), penelitian ini akan melakukan tugas klasifikasi teks pertanyaan pengguna konsultasi kesehatan *online* berbasis *deep learning* menggunakan arsitektur BiLSTM. Penelitian ini menerapkan dua pendekatan representasi *embedding*, yaitu IndoBERT sebagai *contextual embedding* dan FastText sebagai *word embedding*, di mana pendekatan ini sebelumnya sudah dilatih menggunakan data tidak berlabel dalam jumlah yang besar.

Representasi kata yang dihasilkan ini akan dijadikan sebagai fitur atau vektor *input* dalam pembangunan model klasifikasi menggunakan arsitektur BiLSTM. Selain itu, pada penelitian ini akan menggunakan data tidak berlabel yang berhubungan dengan domain kesehatan dalam jumlah yang besar. Data ini akan dimanfaatkan untuk membantu model *embedding* untuk menghasilkan vektor kata terlatih atau fitur dalam domain kesehatan yang dapat menangkap sintaktis dan semantik kata sehingga dapat meningkatkan kinerja model klasifikasi BiLSTM karena data pelatihan berlabel cenderung sangat sulit diperoleh terutama untuk domain kesehatan.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dipaparkan di atas, maka rumusan masalah yang didapatkan untuk penelitian ini adalah sebagai berikut:

1. Bagaimana menentukan model klasifikasi teks pertanyaan konsultasi kesehatan *online* berkinerja tinggi menggunakan model IndoBERT dan *pretrained word embedding* FastText sebagai vektor *input* pada arsitektur BiLSTM.
2. Bagaimana menentukan *hyperparameter* BiLSTM terbaik sehingga menghasilkan model klasifikasi yang berkinerja tinggi.

1.3 Tujuan Penelitian

Berdasarkan rumusan masalah yang telah disebutkan sebelumnya, maka maksud dan tujuan dilakukannya penelitian ini adalah sebagai berikut:

1. Menentukan model klasifikasi teks pertanyaan konsultasi kesehatan *online* berkinerja tinggi menggunakan IndoBERT dan *pretrained word embedding* FastText sebagai vektor *input* pada arsitektur BiLSTM.
2. Menentukan *hyperparameter* BiLSTM terbaik sehingga menghasilkan model klasifikasi yang berkinerja tinggi.

1.4 Manfaat Penelitian

Berdasarkan rumusan masalah dan tujuan penelitian yang dipaparkan di atas, manfaat dilakukannya penelitian ini adalah:

1. Menghasilkan model berbasis *deep neural network* BiLSTM untuk klasifikasi teks menggunakan teknik *embedding* yang didasarkan pada dua pendekatan

representasi yaitu, *static representation* FastText dan *contextualized representation* IndoBERT sebagai vektor *input*.

2. Menjadi *baseline* ataupun tolok ukur untuk penelitian selanjutnya dalam bidang penerapan *natural language processing* (NLP) pada domain Kesehatan, terutama dalam tugas klasifikasi pertanyaan pada sistem *question-answering* (QA).

BAB II

TINJAUAN KEPUSTAKAAN

2.1 *Natural Language Processing*

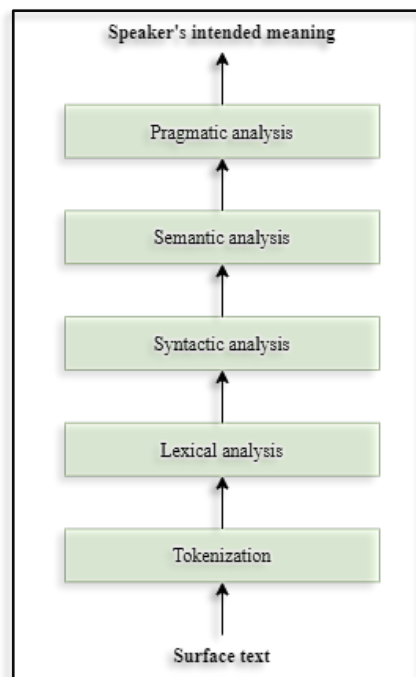
Natural Language Processing atau biasa dikenal sebagai NLP adalah suatu bidang penelitian dalam ilmu komputer dan kecerdasan buatan yang menggabungkan ilmu linguistik, ilmu komputasi, dan ilmu kognitif (Deng & Liu, 2018). NLP meneliti tentang bagaimana penggunaan komputer dalam memahami atau memproses bahasa manusia secara otomatis, seperti tuturan kata (ucapan) atau teks dengan tujuan untuk melakukan tugas-tugas yang berguna. Berdasarkan perspektif ilmiah, NLP bertujuan untuk memodelkan sistem kognitif yang didasarkan pada pengetahuan tentang bagaimana manusia memperoleh, memproduksi, atau memahami suatu bahasa. Sebagian dari NLP berkaitan dengan pemahaman tentang hubungan antara rangkaian linguistik dengan konteks atau wacana tertentu dan sebagian lagi berkaitan dengan bagaimana memahami struktur atau aturan linguistik di dalam bahasa apa yang digunakan oleh manusia dalam berkomunikasi, seperti bahasa Inggris, China, Prancis, dan sebagainya. Pada dasarnya, pemahaman ini meliputi proses penerjemahan bahasa alami ke bentuk numerik sehingga menghasilkan data yang dapat digunakan oleh komputer untuk mempelajari suatu bahasa. NLP dalam bentuk sederhananya didefinisikan sebagai metode yang menggunakan bahasa alami sebagai masukannya (*input*) atau menghasilkan bahasa alami sebagai keluaran (*output*) (Goyal et al., 2018).

Seiring dengan meningkatnya ketersediaan informasi di internet terdapat banyak jenis komunikasi yang digunakan manusia baik itu dalam bentuk tertulis, diucapkan, maupun simbol yang ditangkap dalam bentuk digital. Oleh karena itu, kemunculan bidang penelitian NLP ini salah satunya adalah untuk memudahkan pekerjaan atau tugas pengguna dalam memenuhi keinginannya untuk berkomunikasi dengan komputer atau sistem dengan menggunakan bahasa alami untuk mendapatkan informasi atau pengetahuan (Khurana et al., 2023). Penerapan dalam bidang NLP dapat diharapkan dapat membangun sistem yang benar-benar mampu memahami apa yang dimaksudkan oleh manusia ketika mereka berbicara atau menulis serta menghasilkan rangkaian linguistik yang berhasil mengungkapkan tentang konten apa yang sedang dimaksud (Bender & Lascarides, 2019). Terdapat beberapa penerapan

NLP yang telah dilakukan penelitian, yaitu peringkasan teks otomatis, *coreference resolution*, *discourse analysis*, mesin penerjemah, segmentasi morfologis, *optical character recognition* (OCR), *named entity recognition* (NER), *Part-of-speech tagging* (POS) dan pengenalan suara (Goyal et al., 2018; Khurana et al., 2023).

Bahasa pada dasarnya merupakan susunan simbol yang bersifat diskrit. Unsur dari suatu bahasa tertulis adalah karakter. Susunan atau rangkaian dari karakter akan membentuk kata-kata yang akan merujuk pada suatu objek, konsep, peristiwa, tindakan, ataupun gagasan. Setiap kata tersebut akan memiliki representasi makna yang berbeda (Goldberg & Hirst, 2017). Bahasa alami merujuk pada bagaimana manusia berkomunikasi satu sama lain, baik itu dalam bentuk ucapan maupun teks. Bentuk fisik yang dapat diamati dari bahasa alami adalah teks (Deng & Liu, 2018).

Tahapan analisis dalam NLP tradisional dapat diuraikan ke dalam beberapa tahap diantaranya, yaitu sintaksis, semantik, dan pragmatis (Dale, 2010). Pada dasarnya, teks dimulai dari latar belakang linguistiknya kemudian dilanjutkan ke tingkat kata, yaitu tahap analisis leksikal yang bertujuan untuk menafsirkan makna dari kata-kata secara individu sehingga mendapatkan pemahaman tentang kata tersebut dengan cara menandai atau menentukan sifat morfologis kata, yaitu struktur internal kata atau *part-of-speech* untuk setiap kata. Selanjutnya, kumpulan kata dalam sebuah kalimat diamati untuk menemukan struktur kalimat tersebut menggunakan aturan tata bahasa sehingga menghasilkan representasi kalimat yang mengungkapkan hubungan ketergantungan struktural antar kata, tahap ini dikenal sebagai tahap analisis sintaksis. Tahap selanjutnya adalah tahap semantik yang dapat menentukan kemungkinan makna dari sebuah kalimat melalui hubungan makna di level kata dalam kalimat. Selanjutnya, tahap pragmatis adalah proses yang berkaitan tentang memahami tujuan atau maksud dari bahasa yang tersirat baik itu dalam bentuk ujaran maupun teks dalam konteks yang ditentukan (Khurana et al., 2017). Sesudah itu, tahap terakhir adalah tahap menganalisis domain atau *discourse* dari suatu teks atau bahasa. Tahapan analisis dalam pemrosesan bahasa alami ditunjukkan melalui Gambar 2.1.



Gambar 2.1. Tahapan analisis dalam pemrosesan bahasa alami (Dale, 2010).

2.2 *Machine Learning*

Samuel (1962) mendefinisikan *machine learning* atau pembelajaran mesin sebagai penelitian yang memungkinkan komputer untuk belajar tanpa harus diprogram atau dijelaskan secara eksplisit atau detail. Pada dasarnya, *machine learning* sama seperti pemrograman reguler yaitu sebuah cara bagaimana membuat komputer dapat menyelesaikan suatu tugas tertentu. Namun, *machine learning* membiarkan komputer untuk belajar dari pengalamannya, bukan melalui pengkodean secara manual setiap langkah-langkahnya. Menurut Samuel (1962), memberikan atau memperlihatkan contoh-contoh kepada komputer terkait bagaimana suatu tugas diselesaikan lebih baik daripada memberikan langkah-langkah yang tepat kepada komputer dalam menyelesaikan tugas tersebut (Samuel, 1962).

Algoritma *machine learning* memiliki kemampuan untuk mengekstraksi pola dari data mentah yang dapat digunakan untuk menyelesaikan suatu tugas atau permasalahan tertentu. Tipikalnya, *machine learning* terdiri dari tiga komponen yaitu *representation*, *objective*, dan *optimization*. Terdapat beberapa tahap untuk membangun sebuah sistem *machine learning* yang efektif, yaitu tahap pertama adalah mengekstrak informasi yang berguna dari data mentah ke dalam suatu vektor atau fitur yang digunakan sebagai representasi dari data tersebut. Selanjutnya, perancangan

fungsi objektif adalah tahap yang menggunakan algoritma optimasi untuk menentukan pengaturan parameter yang optimal untuk suatu tugas tertentu, seperti klasifikasi, *clustering*, dsb. Jika terdapat banyak informasi berguna yang direpresentasikan ke dalam bentuk fitur dari data mentah, maka kinerja dari klasifikasi atau prediksi akan cenderung lebih baik. Oleh karena itu, kinerja algoritma *machine learning* yang efektif sangat bergantung pada representasi data yang diberikan. Masing-masing fitur yang merepresentasikan data tersebut akan dipetakan ke *output* dari suatu tugas tertentu. Banyak tugas *artificial intelligence* yang dapat diselesaikan dengan merancang kumpulan fitur yang tepat untuk diekstraksi untuk suatu tugas tertentu. Namun, banyak juga tugas yang sulit untuk menentukan fitur apa yang harus diekstraksi dari data mentahnya. *Machine learning* tidak hanya menemukan pemetaan dari representasi data ke *output* tetapi juga representasi itu sendiri (Goodfellow et al., 2016).

Tom Mitchell mendefinisikan *machine learning* sebagai *well-posed learning problem*, dia menyebutkan suatu program komputer dikatakan belajar dari pengalaman E dalam konteks tugas T dan ukuran kinerja P, kinerja pada tugas T yang diukur oleh P dapat meningkat dengan pengalaman E. Istilah E, T, dan P pada gagasan tersebut merujuk pada *experience*, *task*, dan *performance*. Jenis pembelajaran yang dapat meningkatkan kinerja model *machine learning* pada suatu *task* atau tugas tertentu umumnya dibagi menjadi empat kategori, yaitu *supervised learning*, *unsupervised learning*, *self-supervised learning*, dan *reinforcement learning* (Goodfellow et al., 2016).

Supervised learning adalah pembelajaran yang bertujuan untuk mempelajari hubungan antara input dan output pada saat proses *training*. Pembelajaran ini menangani tugas yang memprediksi label yang diberikan oleh fitur input (A. Zhang et al., 2023). Pada pembelajaran ini telah disediakan suatu kumpulan data yang kategorinya telah dilabeli oleh manusia sehingga terdapat pasangan fitur-label di dalam dataset ini yang biasa disebut sebagai *instance* (Bird et al., 2009; Chollet, 2018). *Supervised learning* dikatakan sebagai pembelajaran yang diawasi karena datanya sudah disertai label yang telah ditentukan sebelumnya untuk setiap *instance*. Secara umum *supervised learning* dikategorikan ke dalam dua kategori, yaitu klasifikasi dan regresi.

Unsupervised learning adalah pembelajaran yang menangani tugas atau permasalahan dengan sedikit label atau tanpa tahu seperti apa label yang diharapkan nantinya. Pembelajaran ini tidak menyediakan bantuan apapun terkait kategori atau label tersebut. Pada pembelajaran ini diberikan suatu kumpulan data yang hanya berisi fitur tanpa diberi nilai label atau target yang sesuai karena pembelajaran *unsupervised* tidak membutuhkan pengetahuan mengenai hubungan atau pengaruh antara fitur tersebut terhadap *output* (Chollet, 2018).

Self-supervised learning adalah pembelajaran tanpa adanya pelabelan oleh manusia atau tidak ada intervensi manusia di dalam pemberian label pada pembelajaran ini. Akan tetapi, tetap masih ada label yang terlibat di dalam prosesnya karena ini masih pembelajaran yang harus diawasi atau dibimbing oleh sesuatu. Label atau targetnya dihasilkan dari data *input*, seperti *autoencoder*. Label atau target yang dihasilkan dari data *input* awal akan dijadikan sebagai data *input* baru untuk diprediksi selanjutnya. Contohnya, ketika memprediksi *next word* atau kata selanjutnya dalam suatu kalimat dengan menggunakan kata-kata sebelumnya dari kalimat tersebut. Pembelajaran ini dikenal juga dengan istilah *temporally supervised learning* yang di mana pembelajarannya menggunakan data temporal atau sekuens (Goodfellow et al., 2016).

Reinforcement learning adalah salah satu algoritma *machine learning* yang memengaruhi dan dipengaruhi oleh lingkungannya, sehingga ada *loop* umpan balik antara tindakan dan pengalamannya (Goodfellow et al., 2016). Pembelajaran ini digunakan untuk membangun sebuah *agent* yang berinteraksi dengan lingkungannya dengan cara menerima informasi mengenai lingkungan tersebut dan belajar untuk mengambil atau memilih suatu tindakan yang akan menghasilkan *reward* yang maksimal (Goodfellow et al., 2016; A. Zhang et al., 2023). *Agent* akan mempelajari suatu hal dengan melakukan tindakan dan melihat hasil dari tindakan tersebut. Tindakan *agent* diatur oleh suatu kebijakan. Kebijakan ini merupakan suatu fungsi yang akan memetakan pengamatan di lingkungan ke tindakan yang diambil. Oleh karena itu, *reinforcement learning* adalah pendekatan yang bertujuan untuk menghasilkan suatu kebijakan yang baik (A. Zhang et al., 2023).

2.3 *Deep Learning*

Deep learning adalah subbidang *machine learning* yang merupakan pendekatan baru terkait *data representation learning* yang menggunakan lapisan representasi berurutan atau dikenal dengan *sequence layers* untuk menghasilkan representasi data yang semakin bermakna. Banyaknya lapisan yang berkontribusi untuk membangun representasi pada kumpulan data disebut sebagai kedalaman atau *depth model*. Nama lain dari pendekatan *deep learning* adalah pembelajaran representasi berlapis atau pembelajaran representasi hierarkis. *Deep learning* modern seringkali melibatkan puluhan atau bahkan ratusan lapisan representasi yang berurutan dan semua representasi ini dipelajari secara otomatis berdasarkan paparan data pelatihan yang diberikan (Chollet, 2018).

Representasi berlapis dalam *deep learning* ini dipelajari melalui model yang disebut sebagai *artificial neural network* (ANN) atau dikenal juga sebagai *neural network*. Pada pendekatan *machine learning*, *input* akan dipetakan ke *target* dengan cara mengamati banyak contoh dari pasangan *input* dan *target*, sedangkan *neural network* melakukan pemetaan *input* ke *target* melalui proses rangkaian urutan mendalam dari lapisan transformasi atau representasi data yang transformasi data sederhana ini dipelajari melalui paparan contoh atau data yang diberikan (Chollet, 2018).

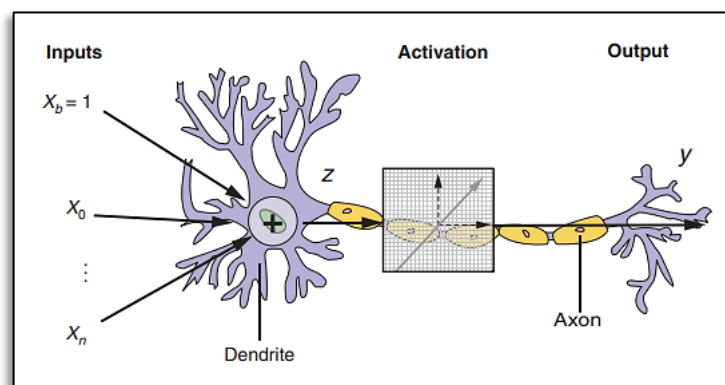
Tujuan dari *deep learning* adalah mempelajari ekstraksi fitur dengan kontribusi manusia yang minimum karena dari awal pelatihan tidak ada diberikan pengetahuan atau informasi mengenai seperti apa bentuk dari *input* yang dimasukkan ke dalam model apakah bentuk teks, gambar, atau bentuk lainnya. Pada metode *deep learning* tidak diperlukan fitur yang tepat, fungsi dasar, atau kernel yang sesuai yang harus dipersiapkan oleh manusia dan tidak perlu khawatir tentang pengoptimalan arsitektur jaringan yang tepat karena algoritma *deep learning* akan menemukan sendiri apa-apa yang diperlukan (Alpaydin, 2020).

2.4 *Neural Network*

Neural network atau dikenal sebagai *artificial neural network* pertama kali diperkenalkan oleh ahli neurofisiologi dan ahli matematika yaitu Warren McCulloch dan Walter Pitts pada tahun 1943 dalam makalah yang berjudul “*A Logical Calculus of Ideas Immanent in Neurons Activity*”. Warren McCulloch dan Walter Pitts

memperkenalkan model komputasi sederhana yang menggambarkan bagaimana neuron biologis dapat bekerja sama di dalam otak hewan untuk melakukan komputasi kompleks menggunakan logika proposional. Penelitian ini merupakan gagasan pertama mengenai arsitektur *artificial neural network* (Geron, 2019).

Artificial neural network adalah kumpulan unit atau *node* yang saling berhubungan yang fungsinya didasarkan pada *neuron* manusia. Kemampuan pemrosesan jaringan disimpan dalam kekuatan koneksi antar *node* yang disebut sebagai *weights* atau bobot yang diperoleh melalui proses adaptasi atau belajar dari serangkaian pelatihan. Koneksi antar *neuron* diperantarai oleh sambungan elektrokimia yang disebut sinapsis. Setiap *neuron* biasanya menerima ribuan koneksi dari neuron lain sehingga *neuron* terus menerus menerima banyak sinyal masuk yang akhirnya akan sampai ke badan sel. Sinyal-sinyal tersebut akan diintegrasikan atau dijumlahkan menggunakan beberapa cara tertentu. Jika sinyal yang dihasilkan melebihi *threshold* atau ambang batas yang ditentukan, maka *neuron* akan aktif dan menghasilkan tegangan impuls sebagai respons. Kemudian impuls ini akan ditransmisikan ke *neuron* lain melalui akson. Sinapsis dimodelkan sebagai satu angka yang merepresentasikan bobot. Setiap *input* dikalikan dengan bobot sebelum dikirim ke badan sel. Selanjutnya, *input* tersebut akan dijumlahkan dengan penjumlahan aritmatika yang sederhana untuk memasuki simpul aktivasi yang dikenal sebagai fungsi aktivasi (Gurney, 1997). Gambar 2.2 menunjukkan visualisasi dari analogi antara sinyal dalam neuron biologis dan neuron tiruan.



Gambar 2.2. Analogi neuron biologis dan tiruan (Lane et al., 2019)

Istilah *network* atau jaringan pada *artificial neural network* mengacu pada susunan lapisan yang saling terhubung. Setiap lapisan menerapkan operasi matematika pada data yang mengalir melaluinya dan memiliki satu set variabel yang dapat

dimodifikasi, yaitu bobot untuk menentukan bagaimana perilaku yang tepat untuk dapat diterapkan pada suatu lapisan tertentu. Data di sini merujuk pada tensor yaitu vektor dengan banyak dimensi. Setiap lapisan jaringan diikuti oleh suatu fungsi aktivasi yang memetakan output dari satu lapisan ke input lapisan berikutnya. Pada beberapa jenis arsitektur, jaringan lengkap dapat memiliki lebih dari 100 lapisan. Lapisan yang terletak di tengah jaringan disebut *hidden layers* atau lapisan tersembunyi dan *output* dari setiap lapisan tersembunyi ini disebut sebagai *hidden units*. Istilah *hidden* atau tersembunyi berasal dari kenyataannya bahwa *node* atau unit ini tidak langsung terlihat dari luar model, apakah dia sebagai *input* atau *output*. Jumlah *output* di lapisan ini tergantung pada model. Meskipun ada beberapa aturan tentang cara menentukan jumlah dan ukuran lapisan tersembunyi ini akan tetapi tidak ada kebijakan umum untuk memilih aturan yang terbaik selain *trial and errors* (Osinga, 2018).

Salah satu arsitektur *artificial neural network* yang paling sederhana adalah *perceptron* (Geron, 2019). *Perceptron* adalah lapisan yang menerima input sebagai sebuah matriks, mengalikan matriks tersebut dengan matriks lain yaitu bobot, dan menjumlahkannya dengan matriks lagi yaitu bias. *Perceptron* merupakan pengklasifikasian linier. *Perceptron* memiliki kemampuan untuk membentuk garis pemisah dan *hyperplane* dalam menyelesaikan permasalahan klasifikasi yang dapat dipisahkan secara linier. Pada tahap pelatihan, *perceptron* menyesuaikan bobotnya menggunakan vektor kesalahan antara titik data dan garis keputusan yang memisahkannya. Penyesuaian ini dilakukan dengan mengubah bobot jaringan sampai tidak ada sampel dataset pelatihan yang menghasilkan kesalahan (Priddy & Keller, 2005).

Perceptron adalah unit pemrosesan dasar dari *artificial neural network*. *Input* pada umumnya bisa berasal dari *environment* atau mungkin *output* dari *perceptron* lain. *Neuron* menerima *input* $x_j \in \mathbb{R}$, di mana $j = 1, \dots, d$. Selanjutnya, x_j dikalikan dengan bobot koneksi atau bobot sinaptik $w_j \in \mathbb{R}$ dan y adalah *output*, dalam kasus sederhana pejumlahan bobot pada input dapat dirumuskan dengan persamaan (2.1).

$$y = \sum_{j=1}^d w_j x_j + w_0 \quad (2.1)$$

w_0 adalah nilai intersep untuk membuat model lebih umum dan biasanya dimodelkan sebagai bobot yang berasal dari unit bias w . *Output* dari *perceptron* dapat dituliskan sebagai *dot product* seperti yang dirumuskan pada persamaan (2.2).

$$y = w^T x \quad (2.2)$$

di mana $w = [w_0, w_1, \dots, w_d]^T$ dan $x = [1, x_1, \dots, x_d]^T$ memperbesar vektor dengan memasukkan bobot bias dan *input*.

Persamaan (2.1) menunjukkan bagaimana *perceptron* mendefinisikan *hyperplane* sehingga dapat digunakan untuk membagi ruang *input* menjadi dua bagian, yaitu setengah ruang positif dan setengahnya lagi adalah negatif. *Perceptron* dapat memisahkan dua kelas dengan memeriksa tanda keluaran dengan mengimplementasikan fungsi diskriminan linier agar tidak hanya dibatasi untuk pemetaan linier saja dalam memeriksa tanda keluaran tersebut dapat menggunakan fungsi aktivasi. Fungsi aktivasi membuat jaringan lebih ekspresif dengan memasukkan persamaan (2.1) ke dalam fungsi aktivasi non linier. Fungsi aktivasi adalah jenis fungsi yang memetakan bilangan real ke bilangan antara 0 dan 1 yang merepresentasikan aktivitas *neuron*, di mana 0 menunjukkan *neuron* yang dinonaktifkan dan 1 menunjukkan *neuron* yang diaktifkan sepenuhnya. Fungsi aktivasi yang baik adalah fungsi aktivasi yang *smooth* yaitu fungsi yang terdiferensiasi secara kontinu dan mudah dihitung untuk meminimalkan kompleksitas komputasi *neural network* (Priddy & Keller, 2005). Salah satu fungsi aktivasi yang sering digunakan untuk mempelajari tugas klasifikasi adalah fungsi *softmax*. *Softmax* menormalisasi *output* dari *neural network* ke dalam rentang [0,1] dan jika dijumlahkan, semua *output* harus berjumlah satu. Jadi, hasil lapisan *output* dengan menggunakan fungsi *softmax* dapat diartikan sebagai probabilitas. Untuk setiap K *node* pada lapisan *output* dapat dihitung menggunakan fungsi eksponensial yang dinormalisasi seperti yang ditunjukkan pada persamaan (2.3):

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}} \quad (2.3)$$

di mana z adalah *neuron* pada lapisan *output* dan j adalah indeks vektor keluaran dari lapisan *output* (Lane et al., 2019).

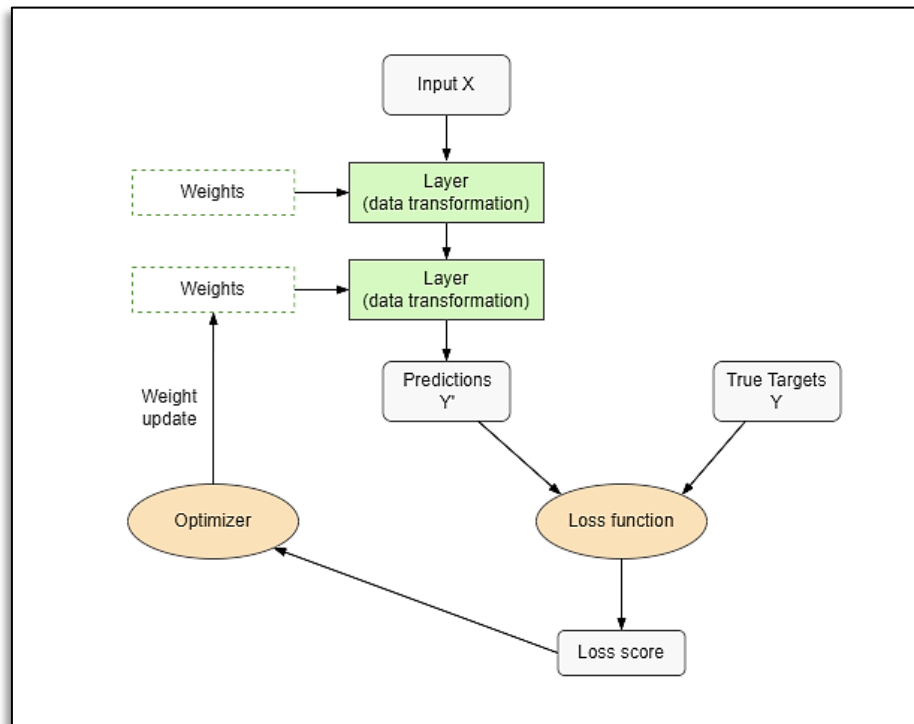
Pembelajaran *online* biasanya digunakan ketika melatih model *neural network*, di mana kumpulan data tidak sekaligus diberikan atau dimasukkan ke dalam model, tetapi setiap *instance* pada kumpulan data dimasukkan satu per satu ke dalam

model, tujuannya adalah agar jaringan memperbarui parameter atau bobot setelah setiap *instance* tersebut menyesuaikan dirinya pada saat proses pelatihan. Pembelajaran *online* dimulai dengan pemberian bobot awal secara acak. Pada setiap iterasi, bobot tersebut akan disesuaikan sedikit demi sedikit untuk meminimalkan kesalahan tanpa melupakan apa yang telah dipelajari sebelumnya. Jika fungsi kesalahan (*loss function*) ini dapat diturunkan maka *gradient descent* dapat digunakan. Pembaruan parameter atau bobot dalam proses pelatihan melibatkan faktor pembelajaran atau biasa dikenal dengan *learning rate*, faktor ini akan menurun secara bertahap pada waktunya untuk sampai ke konvergensi. Besarnya pembaruan bobot tergantung juga pada nilai input. Jika inputnya mendekati 0 maka mengakibatkan output aktualnya kecil dan oleh karenanya bobot akan diperbarui dengan bilangan yang kecil. Semakin besar input maka semakin besar juga pembaruan pada bobot. Sebaiknya pada nilai input dilakukan normalisasi sehingga semuanya terpusat di sekitar 0 dan memiliki skala yang sama. Namun, besar pembaruan bergantung pada faktor pembelajaran. Jika nilai faktor pembelajaran ini terlalu besar, maka pembaruan akan cenderung kepada *instance* terbaru, di mana sistem seolah-olah memiliki memori yang sangat pendek. Jika faktor ini kecil, maka akan membutuhkan banyak pembaruan. Penyesuaian nilai faktor pembelajaran sangat penting terutama dengan jaringan yang memiliki banyak lapisan dan *node* (Alpaydin, 2020).

Multi layer perceptron atau MLP yang merupakan perluasan dari pendekatan *perceptron*. MLP dapat mengekstrak fitur di lapisan tersembunyi tersebut. Keuntungan menggunakan pendekatan ini adalah lapisan pertamanya dapat mengekstraksi fitur dan pada lapisan kedua, fitur-fitur tersebut digabungkan untuk memprediksi hasil akhir menggunakan pembelajaran terawasi (*supervised learning*). Secara empiris, suatu jaringan yang panjang dan tipis tidak hanya memiliki parameter yang lebih sedikit yang mengakibatkan komputasinya lebih kecil daripada jaringan yang pendek dan tebal tetapi juga jaringan tersebut memperoleh generalisasi yang lebih baik. Ini adalah gagasan dari pendekatan *deep neural network*, dimana pendekatan ini terdiri dari jaringan yang memiliki banyak lapisan tersembunyi (Alpaydin, 2020).

Perincian tentang apa yang dilakukan oleh lapisan terhadap inputnya disimpan dalam bobot lapisan yang pada dasarnya adalah sekumpulan angka. Bobot berisi informasi yang dipelajari oleh model dari paparan data pelatihan. Transformasi yang

diterapkan pada lapisan diparameterisasikan oleh bobot. Oleh karena itu, bobot juga disebut sebagai parameter lapisan. Istilah ‘belajar’ pada metode *deep learning* berarti menemukan sekumpulan nilai bobot untuk semua lapisan dalam jaringan, sehingga jaringan akan memetakan *instance* setiap input dengan benar ke target yang diinginkan (Chollet, 2018). Gambaran mengenai apa yang dilakukan oleh model *artificial neural network* dapat dilihat pada Gambar 2.3.



Gambar 2.3. Tahapan yang dilakukan oleh model neural network (Chollet, 2017)

Gambar 2.3 menunjukkan bahwa dalam mengontrol *output neural network* dibutuhkan pengukuran mengenai seberapa jauh perbedaan *output* yang didapatkan dari *output* yang diinginkan. Pengukuran selisih atau beda dari kedua *output* ini adalah tugas dari *loss function* atau disebut juga sebagai *objective function*. *Loss function* mengambil prediksi dan *output* yang diinginkan dari model *neural network* untuk dilakukan perhitungan skor atau nilai jarak yang merepresentasikan seberapa baik kinerja suatu model. Model berbasis *deep learning* menggunakan nilai dari *loss function* ini sebagai sinyal umpan balik untuk menyesuaikan bobot ke arah yang dapat menurunkan nilai *loss*. Penyesuaian nilai bobot ini adalah tugas *optimizer* yang menerapkan algoritma *backpropagation*. *Training loop* dilakukan dengan memasukkan setiap *instance* ke dalam model yang bobotnya disesuaikan ke arah yang

benar sehingga nilai *loss* berkurang. *Training loop* biasanya diulang sebanyak puluhan iterasi dengan ribuan *instances* sampai pada akhirnya menghasilkan nilai bobot yang meminimalkan *loss function*. Model dengan *loss function* yang minimal adalah model yang outputnya sedekat mungkin dengan target (Chollet, 2018).

Back propagation merupakan algoritma yang diterapkan model pada tahap *optimizer* yang berdasarkan pada algoritma *gradient descent* untuk mengoptimalkan parameter dalam model. *Back propagation* adalah pendekatan utama yang digunakan untuk melatih model *deep learning*. Secara singkat, proses *backpropagation* direpresentasikan dalam dua langkah berikut:

- *Forward calculation*, pada langkah ini diberikan satu kumpulan *input* beserta bobotnya kemudian model akan melakukan penjumlahan bobot untuk setiap *neuron* sehingga model akan menghasilkan *output* atau prediksi, di mana model melakukannya dalam urutan maju atau biasanya langkah ini dikenal juga sebagai *forward pass*.
- *Backward propagation*, pada langkah ini model akan menghitung nilai *loss* pada setiap *output* untuk mengoptimalkan parameter lalu model akan memperbarui bobot dengan turunan parsial yang sesuai dalam urutan mundur.

2.5 Text Mining

Secara umum data dapat disusun dalam bentuk yang sangat terstruktur, seperti dalam bentuk basis data relasional atau dapat juga disusun dalam format semi terstruktur yang merupakan penggabungan antara data terstruktur dan tidak terstruktur, seperti file XML, JSON, dan HTML. Menurut (Han et al., 2023) 80% data yang terdapat di dunia merupakan data yang tidak terstruktur, seperti data teks dari berbagai sumber tertulis, seperti buku, email, situs web, artikel, dan ulasan serta media sosial yang merupakan sumber dari berbagai jenis data, seperti data video, gambar, dan audio. Salah satu data tidak terstruktur adalah data tekstual yang sebagian besar datanya diperoleh dari halaman situs web dan situs jejaring sosial. Data teks tidak terstruktur mengandung banyak informasi dan pengetahuan yang berharga, tetapi menangani data tidak terstruktur merupakan pekerjaan yang menantang karena dari jumlah datanya yang sangat besar dan strukturnya yang beragam sehingga membutuhkan metode dan algoritma yang dapat memproses teks dalam jumlah besar secara efektif untuk berbagai macam aplikasi.

Text mining adalah proses mengekstraksi informasi atau pengetahuan yang berguna, bermakna, dan berharga dari data teks tidak terstruktur melalui identifikasi dan eksplorasi pola, struktur, serta interpretasi yang sebelumnya tidak diketahui menjadi sesuatu yang dapat dipahami dari kumpulan teks atau korpus yang besar. Pada dasarnya, *text mining* memiliki kemiripan dengan *data mining*, sehingga terdapat kesamaan arsitektur di antara keduanya, seperti keduanya yang sama-sama mengandalkan metode *preprocessing* data, algoritma untuk penemuan pola, dan visualisasi data. Hanya saja, sumber data dalam *data mining* merupakan data relasional dan disimpan dalam format yang terstruktur atau dalam basis data yang sudah disusun.

Tahap *preprocessing* pada *text mining* melibatkan proses identifikasi dan ekstraksi fitur Bahasa alami. Pada tahap ini, data tidak terstruktur diubah menjadi bentuk yang lebih terstruktur secara eksplisit. Salah satu tugas dalam *text mining* adalah mengubah teks yang tidak terstruktur menjadi data terstruktur. Penambangan informasi dari data tekstual tidak terstruktur menggunakan NLP, pemodelan statistik, dan *machine learning* merupakan tugas yang sulit karena Bahasa alami umumnya tidak konsisten dan bersifat ambiguitas. Secara umum, menurut (Y. Zhang et al., 2015) terdapat tiga langkah dasar dalam *text mining*, yaitu *text preprocessing*, *text mining operations*, dan *postprocessing*.

1. *Text Preprocessing*

Pada langkah ini melibatkan proses *data selection*, klasifikasi, dan ekstraksi fitur yang mengubah dokumen ke dalam format terstruktur yang bermakna sesuai dengan tugas *text mining* tertentu.

2. *Text Mining Operations*

Langkah ini merupakan inti dari sistem *text mining*, di mana data yang sudah diubah menjadi bentuk yang dapat digunakan pada langkah sebelumnya akan diterapkan ke dalam berbagai tugas, seperti *clustering*, klasifikasi, analisis tren, penemuan pola, dan algoritma penemuan pengetahuan lainnya.

3. *Postprocessing*

Pada tahap *postprocessing*, data atau pengetahuan yang didapatkan dari langkah *text mining operations* akan dievaluasi sehingga menghasilkan interpretasi dan visualisasi berdasarkan pada pengetahuan yang diperoleh pada tahap *text mining operations*.

2.6 Klasifikasi Teks

Klasifikasi teks adalah salah satu tugas dari NLP yang mengklasifikasikan unit tekstual, seperti kalimat, kueri, paragraf, dan dokumen ke dalam beberapa kategori atau label yang telah ditentukan sebelumnya, di mana kategori atau label didasarkan pada konteks atau isi dari unit tekstual tersebut. Tugas klasifikasi bisa saja memiliki k kategori yang berbeda dan tidak ada keterkaitan di antara kategori-kategori tersebut. Klasifikasi teks diterapkan ke dalam banyak tugas, seperti deteksi spam, sistem tanya jawab, analisis sentimen, kategorisasi berita, *textual entailment*, dan sebagainya (Minaee et al., 2021).

Klasifikasi disebut sebagai pendekatan *supervised learning* karena prosesnya menyediakan data pelatihan yang digunakan untuk memandu atau membimbing model dalam pengklasifikasian data. Data latih menyediakan *instance* disertai dengan kategori yang mengindikasikan konteks dari *instance* tersebut yang dijadikan sebagai persepsi atau dugaan dasar untuk model dalam mengklasifikasi. Klasifikasi teks juga melibatkan proses pengujian terhadap model yang sudah dibangun dan dilatih sebelumnya menggunakan data pelatihan melalui data pengujian. Data pengujian merupakan sekumpulan *instances* yang labelnya tidak diketahui atau tidak disediakan. Proses mempelajari model klasifikasi dari data pelatihan dan menerapkannya pada data pengujian disebut sebagai generalisasi. Ide dasar dari proses generalisasi adalah menggeneralisasi pemahaman atau hasil pengamatan yang didapatkan selama proses pelatihan ke data pengujian acak dengan label yang tidak diketahui (Aggarwal, 2022).

Pada dasarnya klasifikasi teks dapat dilakukan melalui pelabelan secara manual atau otomatis. Namun semakin meningkatnya skala data tekstual yang tersedia saat ini membuat klasifikasi teks otomatis semakin penting. Model klasifikasi teks otomatis umumnya dibagi ke dalam dua pendekatan, yaitu pendekatan berbasis aturan atau *rule-based* dan pendekatan *machine learning* atau *data-driven based*. Pendekatan *rule-based* mengklasifikasikan data tekstual ke dalam kategori yang berbeda berdasarkan aturan yang secara eksplisit telah ditentukan sebelumnya secara manual sehingga membutuhkan pemahaman yang mendalam terkait suatu domain tertentu. Berbeda dengan pendekatan *rule-based*, pendekatan berbasis *machine learning* mengklasifikasikan teks ke dalam kategori yang berbeda berdasarkan pengamatan modelnya terhadap data pelatihan, di mana pendekatan ini mempelajari hubungan

antara teks dan labelnya. Akhir-akhir ini, banyak peneliti yang menggunakan pendekatan berbasis *neural* atau *deep learning* yang saat ini menjadi semakin populer. Pendekatan *deep learning* membangun model klasifikasi teks berdasarkan pada fitur yang dikenal sebagai *word embedding* yang dipelajari secara otomatis oleh mesin untuk memetakan teks ke dalam vektor kontinu berdimensi rendah, sehingga tidak diperlukan fitur yang dirancang secara manual. Performa model klasifikasi yang baik dalam pendekatan *deep learning* sangat bergantung pada efektivitas *word embedding* (Minaee et al., 2021).

Metode proses yang umum digunakan untuk mengklasifikasi teks terdiri dari tahap *preprocessing*, ekstraksi fitur, pemilihan fitur, dan klasifikasi yang dijelaskan sebagai berikut (Kowsari et al., 2019; Uysal & Gunal, 2014):

1. Tahap *Preprocessing*

Tahap *preprocessing* merupakan tahap yang melibatkan tugas-tugas seperti tokenisasi, penghapusan *stopwords*, pengubahan huruf besar ke huruf kecil, *stemming*, dan normalisasi. Tahap ini memiliki pengaruh yang kuat terhadap kinerja model klasifikasi.

2. Tahap Ekstraksi Fitur

Setelah data dari uni tekstual dibersihkan pada tahap *preprocessing*, selanjutnya data yang tidak terstruktur tersebut harus diubah ke dalam ruang fitur terstruktur menggunakan pendekatan representasi ruang vektor. Representasi fitur yang umum digunakan dalam ekstraksi fitur adalah *bag-of-words* (BOW), *n-gram*, *term frequency-inverse document frequency* (TF-IDF), *Term Frequency* (TF), dan representasi fitur berdimensi rendah yang akhir-akhir ini populer digunakan, seperti Word2vec, GloVe, dan FastText.

3. Tahap Klasifikasi

Tahap yang paling penting dalam klasifikasi teks adalah memilih algoritma klasifikasi terbaik dengan tujuan untuk mendapatkan model klasifikasi yang paling efisien dalam melakukan klasifikasi. Algoritma yang populer digunakan untuk klasifikasi adalah *Naïve Bayes classifier* (NB), *K-nearest neighbor* (KNN), dan *Support Vector Machines* (SVMs) serta algoritma sederhana yang telah banyak menangani sebagian besar domain *data mining* yaitu *Logistic Regression* (LR).

2.7 Word Embedding

Word embedding atau diartikan ke dalam bahasa Indonesia sebagai penyematan kata adalah salah satu pendekatan dalam pembelajaran representasi data yang dipelajari untuk data teks. *Word embedding* merupakan blok dasar dalam pemrosesan bahasa alami atau NLP yang merepresentasikan setiap kata sebagai vektor *real-valued* dalam ruang vektor dimensi R yang sudah ditentukan sebelumnya dan biasanya berupa vektor berdimensi rendah. Di dalam *word embedding*, setiap kata akan dipetakan ke dalam satu vektor numerik dan vektor tersebut akan dipelajari dengan cara yang mirip dengan cara pembelajaran model *neural network*. *Word embedding* dikenal juga dengan istilah *word vector* atau *neural word representation*. Vektor kata yang dihasilkan oleh *word embedding* dapat merepresentasikan semantik atau makna dari suatu kata serta memberikan informasi sintaksis kata tersebut baik itu yang berupa makna literal maupun tersirat sehingga memungkinkan bagi *word embedding* untuk menangkap konotasi dari suatu kata baik itu yang bersifat konseptual ataupun yang berhubungan dengan manusia, hewan, benda, atau lainnya.

Pendekatan *word embedding* termasuk ke dalam *distributed representation* yang memetakan setiap kata di dalam kosakata yang muncul dalam dokumen teks ke dalam vektor *continuous real-valued* sebagai *distributed representation* dari kata-kata (Bengio et al., 2003). Istilah *word embedding* diperkenalkan oleh Bengio et al. (2003) melalui sebuah tulisan yang berjudul “*A Neural Probabilistic Language Model*” yang mempelajari bagaimana vektor fitur terdistribusi pada kata-kata yang dipelajari menggunakan *neural networks* dapat menggambarkan kesamaan antar kata secara semantik. Vektor fitur untuk kata-kata tersebut dipelajari berdasarkan pada kemungkinan mereka untuk muncul secara bersamaan dalam konteks atau urutan yang sama. *Word embedding* menggunakan pembelajaran *unsupervised* yang bertujuan untuk menghasilkan vektor kata yang dimana *word embedding* dari kata-kata yang mirip memiliki vektor yang sama, dikarenakan dalam mendefinisikan kesamaan antar kata-kata itu sulit maka digunakan *distributional hypothesis* untuk mendefinisikan kesamaan antar kata tersebut. Hipotesis ini mendorong munculnya gagasan *distributional semantic* yang merupakan salah satu tujuan yang harus dicapai oleh *word embedding*. *Distributional semantics* mengukur kesamaan semantik antar kata menurut sifat distribusinya dalam kumpulan rangkaian teks yang besar. Makna sebuah

kata dapat diturunkan atau diperoleh dari distribusi kata tersebut di dalam kumpulan konteks di mana kata itu digunakan.

Misalkan sebuah kata w muncul dalam teks, maka konteksnya adalah sekumpulan kata yang muncul di dekatnya sehingga makna suatu kata itu diperoleh dari kata-kata yang sering muncul di sekitarnya, di mana kumpulan konteks tersebut yang akan merepresentasikan kata w . Ide ini sejalan dengan apa yang dinyatakan oleh Bengio et al. (2003) bahwa tujuan dari *distributed representation* adalah untuk mencapai generalisasi dari urutan kata dalam suatu bahasa di mana probabilitas kemunculan suatu kata itu dapat didistribusikan ke dalam urutan kata yang besar yang berada di dekatnya. Generalisasi tersebut dikatakan tercapai jika rangkaian kata yang belum pernah dipelajari sebelumnya oleh model mendapatkan probabilitas yang tinggi yang didapatkan dari kata-kata yang mirip atau vektor kata yang berdekatan dalam kumpulan kata yang sudah terlihat sebelumnya pada tahap pelatihan. Selain itu, keuntungan dari *word embedding* yang dilatih dengan sejumlah data yang besar yang tidak beranotasi memungkinkan model untuk menggeneralisasi kejadian yang tidak terlihat. Oleh karena itu, *word embedding* memungkinkan untuk membangun model bahasa yang dapat digeneralisasikan, alih-alih membiarkan model bersusah payah untuk mempelajari *input* kata-kata baru atau langka. Model tersebut memungkinkan kata-kata yang serupa atau mirip dan yang lebih umum dapat berbagi parameter dan komputasi probabilitasnya kepada kata-kata yang tidak terlihat atau *unseen words*. *Word embedding* dapat mengatasi masalah *unseen words* secara lebih efisien dan optimal dibandingkan representasi *one-hot encoding*.

2.8 FastText

FastText merupakan pendekatan penyematan subkata, di mana subkata mengacu pada karakter n -gram pada suatu kata. Model fastText merepresentasikan suatu kata sebagai sekumpulan karakter n -gram. Representasi setiap kata diperoleh berdasarkan hasil penjumlahan dari kumpulan representasi vektor subkata. Pendekatan fastText adalah pengembangan dari model *skip-gram word2vec*. Tujuannya adalah untuk mempelajari representasi kata dengan tetap memperhitungkan morfologi kata, di mana ini merupakan kelemahan dari model *skip-gram* yang mengabaikan morfologi kata ketika mempelajari representasi suatu kata. FastText memodelkan morfologi kata

dengan memperhitungkan struktur internal atau unit subkata dan merepresentasikan kata tersebut dengan jumlah karakter n-gramnya (Bojanowski et al., 2017).

FastText dapat dianggap sebagai *skip-gram* tingkat subkata, di mana setiap *center word* direpresentasi oleh jumlah vektor subkatanya. Jika diberikan kamus n-gram berukuran G dan diberikan sebuah *center word* w , maka himpunan n-gram atau gabungan semua subkata yang muncul di w dinyatakan sebagai $\mathcal{G}_w \subset \{1, \dots, G\}$. Representasi vektor setiap subkata g dalam himpunan $\mathcal{G}_w$ dinyatakan sebagai z_g , sedangkan untuk mendapatkan representasi vektor kata w adalah dengan menjumlahkan semua vektor subkatanya. Selanjutnya, jika diberikan vektor *context words* sebagai v_c , maka diperoleh fungsi penilaian antara *center word* dan *context word* yang dirumuskan melalui persamaan (2.4).

$$s(w, c) = \sum_{g \in \mathcal{G}_w} z_g^T v_c \quad (2.4)$$

N-gram akan dipetakan menggunakan fungsi *hashing* ke dalam bilangan bulat 1 hingga K sehingga sebuah kata akan direpresentasikan oleh indeksnya dalam kamus kata dan kumpulan *hashed n-gram* di dalamnya. Model FastText sangat bergantung pada karakter n-gram yang digunakan untuk merepresentasikan kata sebagai vektor. Model ini memberi batasan untuk ukuran n-gram yang digunakan, di mana nilai n berkisar antara 3 hingga 6 karakter. Berdasarkan penelitian yang dilakukan oleh Bojanowski et al. (2017), n-gram dengan rentang tersebut sudah memberikan berbagai informasi subkata dalam jumlah yang wajar. Pada saat melakukan ekstraksi karakter n-gram, sebelumnya sebuah kata harus ditambahkan karakter khusus $<$ dan $>$ di awal dan di akhir kata untuk membedakan prefiks dan sufiks dari rangkaian subkata lainnya. Sebagai contoh, jika diberikan sebuah kata yaitu *kesehatan* dengan ukuran $n = 3$, maka kata tersebut akan direpresentasikan oleh karakter n-gram dengan panjang 3 seperti berikut:

$$< ke, kes, ese, seh, eha, hat, ata, tan, an >$$

dan ditambah dengan subkata khusus

$$< kesehatan >.$$

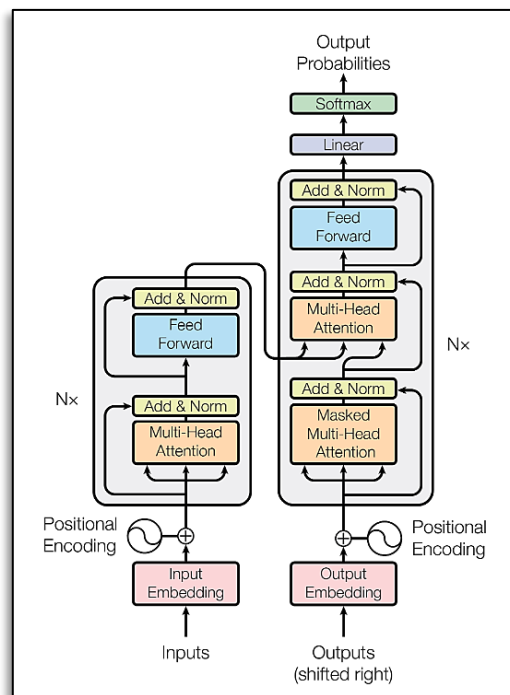
Kata *kesehatan* juga dimasukkan ke dalam kumpulan n-gramnya karena untuk mempelajari representasi kata itu sendiri di samping mempelajari karakter n-gramnya. Dalam praktiknya, kata tersebut akan dilakukan ekstraksi karakter n-gram untuk semua ukuran n , di mana $3 \leq n \leq 6$ (Bojanowski et al., 2017).

Model fastText memungkinkan kata yang memiliki struktur yang sama untuk berbagi parameter atau berbagi representasi dari subkatanya sehingga model ini dapat mempelajari representasi untuk kata-kata langka dan bahkan kata-kata di luar kosakata yang tidak muncul selama pelatihan model atau *out of vocabulary* (OOV) sehingga kata tersebut dipecah menjadi karakter n-gram kemudian menjumlahkan vektor n-gramnya untuk mendapatkan representasi kata tersebut. Ini merupakan keunggulan model fastText yang menggunakan informasi subkata berupa karakter n-gram untuk merepresentasikan suatu kata. FastText dapat melatih model pada kumpulan data yang sangat besar dengan cepat dan dapat memberikan vektor kata yang sangat baik walaupun menggunakan kumpulan data pelatihan yang sangat kecil (Joulin et al., 2016).

2.9 Bidirectional Encoder Representations from Transformers (BERT)

Bidirectional Encoder Representations from Transformers atau biasa dikenal sebagai BERT adalah model representasi terlatih (*pretrained representation*) sekaligus sebagai model representasi kontekstual (*contextual representation*) yang dilatih menggunakan model dua arah dari kumpulan data teks tidak berlabel yang berskala besar berdasarkan pendekatan Transformer. BERT melakukan *pre-training* secara dua arah dengan menggabungkan konteks kiri dan kanan pada lapisan Transformer yang bertujuan agar model mempelajari representasi kontekstual sehingga model dapat memahami bahasa (Devlin et al., 2019).

Transformer adalah pendekatan yang menggunakan mekanisme *self-attention* untuk mempelajari kata atau token dalam sekuens. *Self-attention* adalah mekanisme yang menghitung representasi dari suatu sekuens dengan menghubungkan antar sekuens tunggal dari posisi yang berbeda, di mana mekanisme ini menghitung representasi *input* dan *output* nya tanpa menyelaraskan posisi sekuens dengan *timesteps* dalam komputasi seperti yang dilakukan oleh arsitektur RNN atau LSTM sehingga memungkinkan bagi Transformer melakukan paralelisasi (Vaswani et al., 2023). Arsitektur Transformer terdiri dari dua komponen yaitu encoder dan decoder yang ditunjukkan melalui Gambar 2.4.



Gambar 2.4. Arsitektur Transformer (Devlin et al., 2019)

Berdasarkan Gambar 2.4 struktur dari tiap komponen pada arsitektur Transformer akan dijelaskan sebagai berikut:

a. *Encoder*

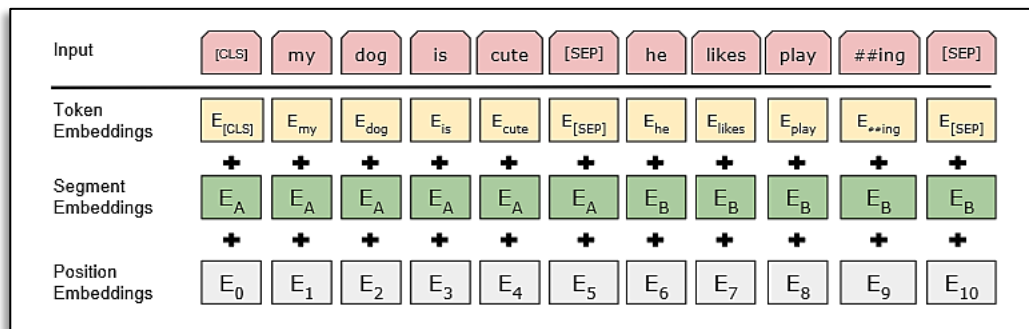
Encoder adalah komponen yang menerima *input* sekaligus komponen yang mempersiapkan representasi untuk model BERT yang berupa rangkaian vektor penyematan. *Encoder* umumnya terdiri dari $N = 6$ tumpukan lapisan yang identik. Walaupun setiap N blok *encoder* memiliki struktur yang identik tetapi mereka tidak menggunakan parameter yang sama, di mana setiap lapisan belajar dari lapisan sebelumnya dan menguji berbagai cara untuk menangkap hubungan antar token dalam sekuens *input*. Setiap lapisan *encoder* memiliki dua sublapisan, yaitu lapisan *self-attention* dan lapisan *feed forward neural network*. Vektor kata yang dihasilkan oleh setiap sublapisan *encoder* dan lapisan *embedding* memiliki dimensi yang sama yaitu 512. Lapisan *self-attention* pada *encoder* membantu model untuk memberikan perhatian khusus pada kata lain saat model sedang merepresentasikan suatu kata tertentu sehingga *encoder* dapat menangkap hubungan suatu kata terhadap kata lainnya.

b. *Decoder*

Decoder menggunakan representasi yang dihasilkan dari komponen *encoder* dan *input* lainnya untuk menghasilkan rangkaian token secara berulang, di mana satu token dihasilkan dalam satu waktu. *Decoder* juga terdiri dari $N = 6$ tumpukan lapisan yang identik. Namun, pada setiap lapisan *decoder* ditambahkan lapisan yang melakukan *multi-head attention* pada *output* dari tumpukan *encoder*. Lapisan *self-attention* pada *decoder* dimodifikasi menjadi lapisan *masked multi-head self-attention* untuk memastikan bahwa token yang dihasilkan pada *timesteps* hanya didasarkan pada *output* masa lalu dan token saat ini yang diprediksi.

Model BERT merupakan arsitektur berdasarkan pada *multi-layer bidirectional encoder* pada arsitektur Transformer. BERT dengan konteks atau mekanisme *self-attention* dua arah pada lapisan *encoder* dapat menangkap pemahaman tentang sekuens dan hubungan saling ketergantungan antar kata. Pada praktiknya, BERT terdiri dari dua ukuran model, yaitu BERT_{BASE} dan BERT_{LARGE}. Model BERT_{BASE} memiliki 12 lapisan *encoder* atau Transformer blok, 768 dimensi vektor *embedding*, 12 *self-attention heads*, dan total parameternya 110 juta. Sebaliknya, Model BERT_{LARGE} memiliki 24 lapisan *encoder* atau Transformer blok, 1024 dimensi vektor *embedding*, 16 *self-attention heads*, dan total parameternya 340 juta.

Urutan token yang dimasukkan ke dalam model BERT dapat berupa satu kalimat atau beberapa kalimat yang digabung menjadi satu. BERT menggunakan penyematan WordPiece dengan 30.000 token untuk menghasilkan *token embeddings* dari suatu kalimat. WordPiece adalah algoritma tokenisasi berbasis *subword*, di mana kata-kata yang langka dalam suatu kalimat akan dipecah menjadi subkata yang lebih kecil sehingga tidak ada kata-kata yang tidak ada dalam kosakata atau biasa dikenal dengan istilah *out of vocabulary* (OOV), sedangkan untuk kata yang sering digunakan tidak dipecah ke dalam bentuk yang lebih kecil. Gambaran mengenai proses representasi *input* pada BERT ditunjukkan pada Gambar 2.5.



Gambar 2.5. Representasi input BERT (Devlin et al., 2019)

Berdasarkan Gambar 2.5 mengenai representasi *input* pada BERT, token pertama dari setiap urutan *input* pada BERT adalah token khusus [CLS]. Sementara itu, kalimat yang digabung menjadi satu dalam urutan token akan dipisah menggunakan token khusus [SEP] untuk membedakan kalimat pertama dan kedua, tahap ini dikenal sebagai *segment embeddings*. Selanjutnya, *positional embedding* ditambahkan pada setiap token untuk menunjukkan posisi token pada kalimat. *Output* dari token *embedding*, *segment embedding*, dan *positional embedding* akan dijumlahkan untuk menghasilkan representasi *input* model BERT. Selanjutnya, *output* akhir dari token [CLS] akan digunakan sebagai representasi keseluruhan untuk tugas klasifikasi.

Terdapat dua tahap dalam pembangunan model terlatih BERT, yaitu tahap *pre-training* dan *fine-tuning*. Tahap *pre-training* adalah tahap yang memungkinkan model untuk memahami bahasa dan mempelajari representasi kontekstual. Model BERT melibatkan dua tugas *unsupervised* dalam tahap *pre-training*, yaitu *masked language modeling* (Masked LM) dan *next sentence prediction* (NSP). Selanjutnya, tahap *fine-tuning* adalah tahap adaptasi atau penyesuaian terhadap model yang sudah dilatih sebelumnya ke dalam *downstream task*, seperti tugas klasifikasi kalimat. Keuntungan dari tahap ini adalah model hanya sedikit menyempurnakan atau memodifikasi parameter yang telah dipelajari sebelumnya ke dalam berbagai *downstream tasks*.

2.10 IndoBERT

IndoBERT adalah *contextualized model* berdasarkan arsitektur BERT yang sudah diadaptasi ke dalam bahasa Indonesia (Wilie et al., 2020). IndoBERT adalah model terlatih yang dilatih secara *self-supervised learning* menggunakan kumpulan data Indo4B yang terdiri dari 4 miliar kata dengan sekitar 250 juta kalimat. Indo4B

merupakan kumpulan data yang dikumpulkan dari 12 dataset yang mencakup kalimat bahasa Indonesia formal dan informal atau bahasa sehari-hari. IndoBERT menggunakan algoritma SentencePiece dengan tokenisasi *byte pair coding* (BPE) dalam pembuatan kosakata.

Terdapat dua model IndoBERT, yaitu IndoBERT yang berdasarkan arsitektur BERT dan IndoBERT-lite yang berdasarkan arsitektur ALBERT. Setiap model IndoBERT memiliki dua ukuran model, yaitu IndoBERT_{BASE}, IndoBERT_{LARGE}, IndoBERT-lite_{BASE}, dan IndoBERT-lite_{LARGE}. Semua model IndoBERT dilatih dalam dua tahap. Pada tahap pertama, model dilatih dengan panjang maksimum sekuens 128 token. Pada tahap kedua, model dilatih dengan panjang maksimum sekuens 512 token. Pada tahap *pre-training* model IndoBERT *base* dan *large* dilatih pada tugas *Masked LM*.

Semua model IndoBERT akan dilakukan *fine-tuning* dan dievaluasi performanya pada semua tugas yang dikelompokkan ke dalam dua kategori, yaitu klasifikasi dan *sequence labeling*. Model akan dilatih ke dalam 12 *downstream tasks* yang diperkenalkan oleh IndoNLU. Hasil pengujian performa model IndoBERT ditunjukkan pada Gambar 2.6.

Model	Classification						Sequence Labeling							
	EmoT	SmSA	CASA	HoASA	WRtE	AVG	POSP	BaPOS	TermA	KEPS	NERGrit	NERP	FacQA	AVG
Scratch	52.84	67.07	69.88	76.83	58.06	64.94	90.33	85.63	68.97	41.00	55.17	36.11	6.42	54.80
fastText-cc-id	67.43	78.84	81.61	85.01	61.13	74.80	95.72	91.51	73.44	50.00	37.61	51.40	15.80	59.35
fastText-indo4b	68.47	83.07	81.96	86.20	60.33	76.00	96.12	91.76	71.04	47.64	41.47	53.80	23.65	60.78
mBERT	67.30	84.14	72.23	84.63	84.40	78.54	96.47	95.10	88.12	57.90	77.56	80.75	58.57	79.21
XLM-MLM	65.75	86.33	82.17	88.89	64.35	77.50	96.87	96.79	89.09	58.35	78.80	80.43	58.61	79.85
XLM-R _{BASE}	71.15	91.39	91.71	91.57	79.95	85.15	97.02	96.54	89.83	63.03	81.79	80.93	64.58	81.96
XLM-R _{LARGE}	78.51	92.35	92.40	94.27	83.82	88.27	97.04	96.85	90.61	64.45	81.57	83.84	73.70	84.01
IndoBERT-lite _{BASE} [†]	71.95	89.87	84.71	87.57	80.30	82.88	94.76	93.08	89.58	59.93	73.84	77.07	55.80	77.72
+ phase two	72.27	90.29	87.63	87.62	83.62	84.29	94.95	93.91	89.52	63.98	72.54	77.44	46.88	77.03
IndoBERT _{BASE} [†]	75.48	87.73	93.23	92.07	78.55	85.41	96.90	96.12	89.38	64.33	76.58	81.71	51.30	79.47
+ phase two	76.28	87.66	93.24	92.70	78.68	85.71	96.87	96.15	90.48	63.47	75.67	80.92	54.76	79.76
IndoBERT-lite _{LARGE}	75.19	88.66	90.99	89.53	78.98	84.67	95.96	95.34	90.12	64.53	75.82	82.12	61.14	80.72
+ phase two	71.67	90.13	88.88	88.80	81.19	84.13	96.14	95.41	89.49	62.72	76.85	81.98	66.76	81.34
IndoBERT _{LARGE}	77.04	93.71	96.64	93.27	84.17	88.97	97.25	96.44	91.58	64.84	79.72	83.19	65.10	82.59
+ phase two	79.47	92.03	94.94	93.38	80.30	88.02	97.07	96.44	91.66	66.64	79.97	84.50	66.37	83.24

Gambar 2.6. Hasil performa model pada IndoNLU benchmark (Wilie et al., 2020)

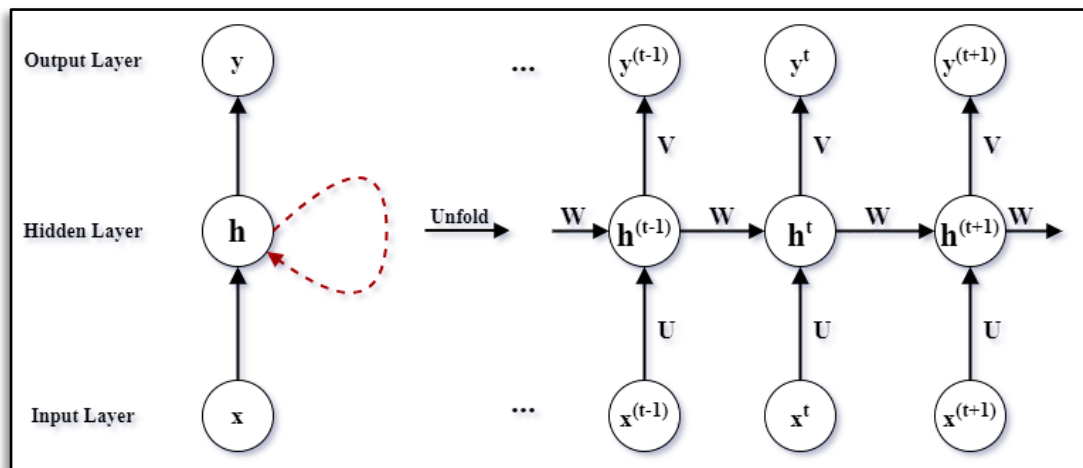
Gambar 2.6 menunjukkan tiga model yang menghasilkan performa terbaik untuk tugas klasifikasi adalah IndoBERT_{LARGE}, XLM-R_{LARGE}, dan IndoBERT_{BASE} sedangkan tiga model dengan performa terbaik untuk tugas *sequence labeling* adalah XLM-R_{LARGE}, IndoBERT_{LARGE}, dan XLM-R_{BASE}. Hasil pengujian pada beberapa model juga menunjukkan bahwa model yang lebih besar menghasilkan performa yang

lebih unggul dibandingkan model yang lebih kecil. Pengujian ini menunjukkan keefektivitas model terlatih (*pretrained model*) jika dibandingkan dengan model yang dilatih dari awal (*scratch model*). Hasil pengujian juga menunjukkan keunggulan model *contextualized embedding* dibandingkan pendekatan *word embedding* yang dibuktikan melalui model *contextualized* yang menungguli model berbasis *word embedding* yang ditunjukkan pada Gambar 2.6.

2.11 Long Short-Term Memory (LSTM)

Long Short-Term Memory atau biasa dikenal dengan LSTM merupakan jenis khusus dari arsitektur *Recurrent Neural Network* (RNN) yang mampu mempelajari ketergantungan jangka panjang (*long-term dependency*) pada data *sequential*. Model LSTM diusulkan oleh Hochreiter dan Schmidhuber (1997) untuk mengatasi masalah gradien *exploding* dan *vanishing* pada saat melatih model RNN tradisional.

RNN merupakan *multilayer perceptrons* yang membentuk suatu koneksi jaringan yang berulang atau siklus dalam jaringan. Koneksi berulang akan membentuk suatu *memory* yang memungkinkan informasi dari *hidden state* di *timestep* sebelumnya bertahan di dalam jaringan dan diteruskan ke *timestep* berikutnya untuk memengaruhi pemrosesan *input* selanjutnya (Elman, 1990). RNN pada dasarnya dapat memetakan seluruh riwayat *input* pada unit di *timestep* sebelumnya ke setiap *output* pada *timestep* berikutnya, berbeda dengan *multilayer perceptrons* yang hanya dapat memetakan dari *input* ke vektor *output*. Pada saat membuka gulungan atau rantai RNN, didapati RNN sebagai arsitektur yang tersusun dari serangkaian salinan dari jaringan yang sama yang saling berhubungan untuk setiap langkah waktu dalam sekuens. Cara yang efektif untuk menggambarkan bagaimana aliran informasi (*forward and backward pass*) mengalir dari waktu ke waktu dan bagaimana cara mempertahankan *memory* input masa lalu melalui *hidden state* adalah dengan membuka jaringan RNN di sepanjang urutan input. Tampilan RNN yang terbuka (*unfolding*) ditunjukkan pada Gambar 2.7, parameter dasar yang diterapkan di dalam RNN adalah sama pada setiap langkah urutan.



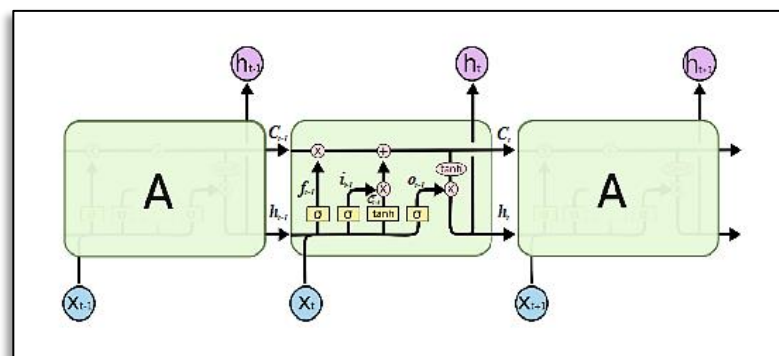
Gambar 2.7. Unfolded RNN berdasarkan timesteps (Graves, 2012)

Grafik RNN terbuka (*unfolding*) yang ditunjukkan pada Gambar 2.7 berada di sebelah kanan, memiliki ukuran yang ditentukan oleh panjang sekuens. Setiap *node* pada grafik RNN terbuka merepresentasikan lapisan unit jaringan pada satu *timestep*. Koneksi *input-to-hidden* diparameterisasi oleh matriks bobot U , koneksi *hidden-to-hidden* diparameterisasi oleh matriks bobot W , dan koneksi *hidden-to-output* diparameterisasi oleh matriks bobot V . Setiap *timestep* menggunakan matriks bobot yang sama (Graves, 2012).

Penelitian dalam beberapa tahun terakhir menunjukkan RNN bekerja dengan baik dalam berbagai tugas *machine learning*, terutama ketika panjang *input* dan *output* jaringannya bervariasi seperti, *image captioning*, pengenalan atau sintesis ucapan, pemodelan bahasa, analisis video, *musical information retrieval*, *machine translation*, dan lain-lain (A. Zhang et al., 2023). Namun dalam praktiknya, menurut pengamatan Bengio dan Hochreiter (Bengio et al., 1994) bahwa melatih RNN untuk menyimpan informasi jangka panjang tampaknya sulit dilakukan dan sangat memakan waktu. RNN adalah jaringan yang memiliki *memory* jangka pendek yang dilatih menggunakan algoritma pembelajaran *backpropagation through time* (BPTT) di mana pada saat *timelag* antara *input* dan *output* nya panjang membuat proses pembelajarannya membutuhkan waktu yang sangat lama dan bahkan pembelajarannya bahkan tidak bekerja dengan baik sama sekali, sehingga memungkinkan munculnya konflik antara bobot *input* dan *output* yang membuat pembelajaran menjadi sulit (Hochreiter & Schmidhuber, 1997). Hal ini terjadi karena gradien atau eror pada proses *backpropagation* cenderung (1) meledak (*explode*) atau (2) menghilang (*vanish*).

Ketika gradien meningkat atau meledak (1) mengakibatkan bobot berosilasi atau tidak stabil sehingga proses *training* nya juga tidak stabil, sedangkan ketika gradien meluruh (2) menjadi terlalu kecil yaitu mendekati 0 membuat jaringan berhenti belajar karena tidak ada yang dapat dipelajari. Oleh karena itu, untuk mengatasi masalah *error backflow* diperkenalkan pendekatan baru yang dikenal dengan LSTM (Hochreiter & Schmidhuber, 1997).

Arsitektur LSTM memiliki struktur yang sama seperti RNN tetapi pada *node hidden layer* berulang yang dimiliki oleh RNN digantikan dengan subnet yang dikenal dengan blok memori (*memory block*). Setiap blok memori memiliki satu atau lebih *cell memory* yang dilengkapi dengan tiga unit *multiplicative gates*, yaitu *forget*, *input*, dan *output gate* (Graves, 2012). LSTM yang dilengkapi dengan sejumlah *multiplicative gates* memungkinkan *cell memory* untuk menyimpan dan mengakses informasi dalam jangka waktu yang lama sehingga dapat menghindari masalah *vanishing gradient*. Pada dasarnya, *unit gate* memiliki kemampuan untuk membuka dan menutup akses aliran informasi dalam *cell memory*. *Unit gate* dikenal juga sebagai *sigmoidal gate* karena memiliki struktur yang terdiri dari lapisan jaringan saraf dengan fungsi aktivasi sigmoid dan *pointwise operation* (Yan et al., 2018). Lapisan sigmoid mengeluarkan nilai antara 0 dan 1. Nilai 0 berarti *gate* tertutup yang berarti tidak membiarkan aliran informasi apapun masuk, sedangkan nilai 1 berarti *gate* terbuka yang membiarkan semua aliran informasi lewat atau masuk. Jadi, *unit gate* dapat memelihara dan mengontrol informasi dalam *cell memory* dari terjadinya konflik bobot *input* dan *output*. Struktur LSTM ditunjukkan pada Gambar 2.8.



Gambar 2.8. Struktur LSTM

Berdasarkan diagram LSTM standar yang diilustrasikan pada Gambar 2.8, *gate* dipengaruhi oleh *input* pada langkah waktu saat ini (x_t) dan *hidden state* dari langkah

waktu sebelumnya (h_{t-1}). Selain itu, setiap gerbang memiliki sumber internal dari sel blok memorinya yang berupa *cell state* (c_{t-1}). Pada diagram LSTM yang ditunjukkan pada Gambar 2.8 setiap baris membawa vektor dari *output* suatu *node* ke *input* pada *node* lainnya. Komponen yang berbentuk lingkaran merah muda merepresentasikan *pointwise operation*, sedangkan komponen yang berbentuk persegi kuning adalah lapisan *neural network*. Penjabaran bagaimana aliran informasi di dalam blok sel arsitektur LSTM mengalir langkah demi langkah dijelaskan sebagai berikut:

1. *Forget Gate*

Forget gate adalah salah satu lapisan *sigmoid* pada LSTM yang memiliki fungsi untuk memutuskan apakah informasi pada *cell state* sebelumnya dapat digunakan untuk *cell state* saat ini atau harus dihapus. Lapisan ini memperhitungkan *input* saat ini dan *hidden state* di masa lalu untuk menghasilkan nilai *forget gate* (f_t). Nilai yang dihasilkan adalah antara 0 dan 1, angka 1 menunjukkan “simpan atau pertahankan seluruhnya” dan angka 0 menunjukkan “hapus seluruhnya”.

2. *Input Gate*

Input gate diperkenalkan untuk melindungi *cell state* dari gangguan oleh *input* atau konten baru yang tidak relevan. *Input gate* melibatkan dua lapisan *neural networks* dalam pemrosesannya yang dilengkapi dengan dua aktivasi yang berbeda yaitu lapisan *sigmoid* dan lapisan *tanh*.

- Pada lapisan *sigmoid*, *input gate* memutuskan apakah *input* atau konten baru yang ditambahkan itu penting atau tidak. *Input gate* memperhitungkan *input* saat ini dan *hidden state* di masa lalu untuk menentukan apakah masukan tersebut memiliki dampak pada *cell state* dan dengan demikian layak dipertahankan atau tidak sehingga perlu diperbarui. *Input gate* juga mengatur berapa banyak konten memori baru yang ditambahkan ke *cell state*.
- Lapisan *tanh* menghasilkan vektor kandidat memori baru $\tilde{c}_t$ yang dapat ditambahkan ke *cell state*.

Pada tahap selanjutnya, untuk memperbarui *cell state* yang lama ($\tilde{c}_{t-1}$), keluaran dari lapisan *input gate* dan *tanh* akan digabungkan untuk menghasilkan *cell state* baru ($\tilde{c}$).

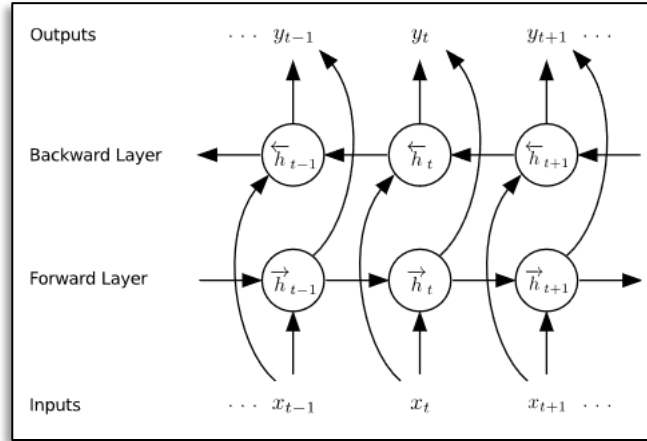
3. *Output Gate*

Output gate diperkenalkan dengan tujuan untuk memisahkan memori akhir atau *cell state* baru (c_t) dari *hidden state* (h_t). Banyaknya informasi yang ada pada c_t

belum tentu perlu disimpan ke dalam h_t dan dengan demikian *output gate* dapat melindungi h_t dari gangguan konten yang tidak relevan yang disimpan oleh c_t .

Model RNN dan LSTM standar memproses data sekuens secara berurutan berdasarkan urutan waktu, sehingga model cenderung mengabaikan konteks masa depan dan dengan demikian model standar RNN dan LSTM bersifat *unidirectional* (searah). Sifat model seperti ini sesuai dengan sebagian besar tugas pemodelan bahasa (*language modeling*) yang ketika melatih modelnya mengondisikan konteks kiri ke kanan, di mana pemodelan bahasa memprediksi token berikutnya dengan mempertimbangkan semua token sebelumnya di dalam sekuens. Namun, tidak sedikit juga tugas dalam pemodelan bahasa yang mencapai hasil yang baik bahkan mengungguli model standar ketika model dilatih dengan tidak hanya memanfaatkan konteks kiri ke kanan namun juga mengondisikan konteks kanan ke kiri, yaitu membuat prediksi token melalui penggabungan representasi token yang muncul di masa lalu dan masa depan. Teknik yang menggabungkan konteks masa lalu dan masa depan untuk memprediksi suatu sekuens dikenal sebagai arsitektur *Bidirectional Recurrent Neural Network* (BRNN) (Schuster & Paliwal, 1997). Ide dasar BRNN adalah mengubah RNN *unidirectional* (searah) menjadi RNN *bidirectional* (dua arah) dengan menghadirkan pada setiap sekuens pelatihan maju dan mundur (*forward* dan *backward training*) ke dalam dua jaringan yang terpisah di mana keluaran dari kedua jaringan terhubung ke lapisan yang sama.

Agar jaringan *bidirectional* mampu mempelajari ketergantungan jangka panjang pada setiap urutan *input*, Graves dan Schmidhuber (Graves & Schmidhuber, 2005) memperkenalkan arsitektur Bi-LSTM yang menggabungkan konsep dasar BRNN dengan arsitektur LSTM. Bi-LSTM menggunakan dua lapisan LSTM terpisah untuk mempelajari informasi dari arah maju dan mundur sebuah kalimat serta dapat menggabungkan informasi yang keluar dari dua arah tersebut, yaitu lapisan *forward* LSTM dan *backward* LSTM.



Gambar 2.9. Bidirectional LSTM (Graves et al., 2013)

Seperti yang ditunjukkan pada Gambar 2.9, Bi-LSTM memproses data sekuens dalam dua arah menggunakan dua *hidden layers* yang terpisah, satu lapisan memproses sekuens dari kiri ke kanan dan lapisan lainnya memproses sekuens dari kanan ke kiri, yang kemudian keduanya dimasukkan ke dalam lapisan *output* yang sama. Pada setiap langkah waktu t , token di lapisan *hidden forward* ($\vec{h}$) dihitung berdasarkan *hidden state* sebelumnya ($\vec{h}_{t-1}$) dan *input* pada *timestep* saat ini (x_t), token di lapisan *hidden backward* ($\tilde{h}$) dihitung berdasarkan *hidden state* setelahnya ($\tilde{h}_{t+1}$) dan *input* pada *timestep* saat ini (x_t). Representasi akhir token diperoleh dengan menggabungkan konteks kiri ($\vec{h}_{t-1}$) pada *timestep* $t = T$ ke 1 dan konteks kanan ($\tilde{h}_{t+1}$) pada *timestep* $t = 1$ ke T , sehingga didapatkan *output* atau *hidden state* akhir dari Bi-LSTM yang diformulasikan sebagai persamaan (2.5) (Niu et al., 2017).

$$h_t = \vec{h}_t \oplus \tilde{h}_t \quad (2.5)$$

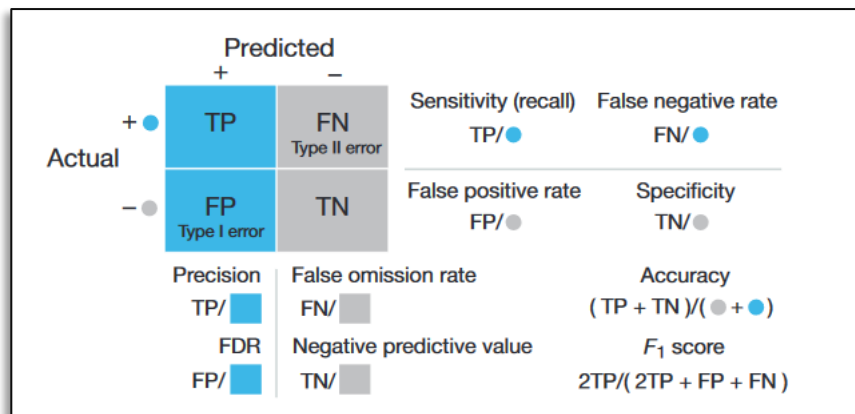
di mana $\oplus$ menunjukkan fungsi penggabungan. Parameter pada lapisan dua arah yang terpisah bersifat independent, tetapi menggunakan *word embedding* atau *input representation* yang sama dalam suatu sekuens (*casuality*).

2.12 Pengukuran Kinerja Model

Pengukuran kinerja model atau *performance metrics* merupakan tahap evaluasi dalam membangun model. Tahap evaluasi sangat penting untuk penggunaan dan pembangunan model *supervised learning* terutama dalam tugas klasifikasi dikarenakan dengan adanya tahap ini kinerja algoritma atau arsitektur yang digunakan untuk membangun suatu model klasifikasi (*classifier*) dapat dipahami. Pengukuran

kinerja suatu model memiliki tujuan untuk (i) membandingkan kinerja dari dua model yang berbeda atau (ii) menganalisis perilaku dari satu model yang sama dengan *tuning* parameter yang berbeda pada domain tertentu (Grandini et al., 2020).

Banyak pengukuran yang tersedia untuk mengevaluasi model klasifikasi (Kowsari et al., 2019). Kinerja model klasifikasi umumnya dievaluasi menggunakan pengukuran numerik, seperti akurasi, skor F_β , *Matthews Correlation Coefficient* (MCC), atau direpresentasikan secara visual atau grafis, seperti kurva *receiver operating characteristic* (ROC), dan kurva *area under ROC* (AUC) (Kowsari et al., 2019; Lever et al., 2016). Pengukuran kinerja model untuk tugas klasifikasi yang biasa digunakan adalah metrik yang digunakan dalam tugas *information extraction* (IE) (Sokolova & Lapalme, 2009). Metrik ini mencakup akurasi, *recall*, *precision*, F_1 score, *micro average*, dan *macro average*. Metrik klasifikasi ini dihitung berdasarkan *true positives* (TP), *false positives* (FP), *false negatives* (FN), dan *true negatives* (TN) yang dirangkum dalam apa yang disebut sebagai *confusion matrix* yang ditunjukkan pada Gambar 2.10.



Gambar 2.10. Confusion Matrix (Lever et al., 2016)

Pengukuran kinerja model dilakukan pada kumpulan data uji, yaitu data yang tidak diikutsertakan ketika melatih model klasifikasi. *Confusion matrix* pada Gambar 2.10 menunjukkan jumlah data yang diprediksi benar dan salah berdasarkan data uji yang diketahui labelnya. TP dan TN memberikan informasi bahwa model klasifikasi memprediksi data secara benar, di mana TP menunjukkan suatu kumpulan data positif yang diprediksi secara benar oleh model klasifikasi dan TN merupakan kumpulan data negatif yang diprediksi secara benar oleh model. Sebaliknya, FP dan FN merupakan istilah ketika model klasifikasi melakukan kesalahan, seperti memberi label yang

salah, FP menunjukkan kumpulan data negatif yang salah diberi label positif oleh model klasifikasi dan FN merupakan kumpulan data positif yang salah diberi label negatif oleh model klasifikasi (Han et al., 2023).

Akurasi adalah ukuran keseluruhan mengenai seberapa tepat model klasifikasi memprediksi seluruh kumpulan data, akurasi dihitung melalui persamaan (2.6).

$$akurasi = \frac{TP+TN}{TP+TN+FP+FN} \quad (2.6)$$

Precision adalah jumlah data atau *instances* positif yang diprediksi secara benar oleh model klasifikasi diantara semua *instances* yang diprediksi positif oleh model, *precision* dihitung melalui persamaan (2.7).

$$precision = \frac{TP}{TP+FP} \quad (2.7)$$

Recall adalah jumlah *instances* positif yang diprediksi secara benar oleh model klasifikasi terhadap semua *instances* yang berlabel positif dalam kumpulan data, *recall* disebut juga sebagai *sensitivity*, persamaan yang digunakan untuk menghitung *recall* ditunjukkan pada persamaan (2.8).

$$recall = \frac{TP}{TP+FN} \quad (2.8)$$

F_1 score atau *F – measure* merupakan salah satu metrik agregat untuk mengevaluasi model klasifikasi. *F – measure* adalah *harmonic mean* dari nilai *precision* dan *recall*. Bentuk *F – measure* yang umum didefinisikan sebagai F_β dalam persamaa (2.9).

$$F_\beta = \frac{(1+\beta^2)(precision \times recall)}{\beta^2 \times precision + recall} \quad (2.9)$$

di mana parameter β digunakan untuk mengatur keseimbangan hasil dari *precision* dan *recall*. Jika $\beta > 1$ maka F menjadi lebih berorientasi terhadap *recall* dan jika $\beta < 1$ maka F lebih berorientasi kepada *precision*. Parameter β yang paling umum digunakan adalah $\beta = 1$, yang memberikan bobot yang sama kepada *precision* dan *recall* sehingga memperoleh *harmonic mean* yang seimbang antara keduanya. Persamaan F_1 score diturunkan dari persamaan F_β , sehingga didapatkan persamaan (2.10) untuk menghitung F_1 score:

$$F_1 score = \frac{2 \times precision \times recall}{precision + recall} \quad (2.10)$$

BAB III

METODE PENELITIAN

3.1 Tempat Dan Waktu Penelitian

3.1.1 Tempat Penelitian

Penelitian Tugas Akhir ini dilakukan di Laboratorium Basis Data dan Pemrograman Gedung Fakultas MIPA Blok A Lantai 3 Jurusan Informatika Universitas Syiah Kuala.

3.2 Alat Dan Bahan

3.2.1 Alat

Alat yang digunakan untuk mendukung pengerjaan Tugas Akhir terdiri dari perangkat keras dan perangkat lunak yang dideskripsikan sebagai berikut:

a. Perangkat Keras

Perangkat keras yang digunakan untuk melakukan penelitian Tugas Akhir ini adalah VivoBook_ASUSLAPTOP X409FJ_A409FJ dengan spesifikasi RAM 8 GB Intel(R) Core(TM) i5-8265U CPU @ 1.60GHz 1.80 GHz.

b. Perangkat Lunak

Perangkat lunak yang digunakan untuk mendukung penelitian Tugas Akhir ini adalah Jupyter Notebook, Google Colaboratory, Google Chrome, Visual Studio Code, Docker.

3.2.2 Bahan

Terdapat dua kumpulan data yang digunakan dalam penelitian Tugas akhir ini yaitu, kumpulan data berlabel tunggal dan kumpulan data tidak berlabel atau *unsupervised dataset*.

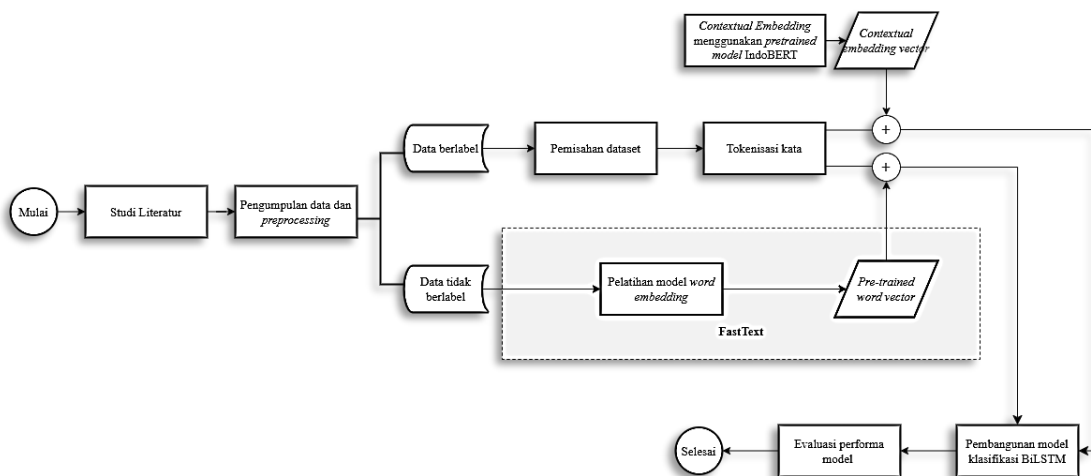
a. Kumpulan data berlabel tunggal adalah kumpulan dari dokumen pertanyaan pengguna konsultasi kesehatan di forum *online* yang diperoleh dari dua situs kesehatan yaitu, Alodokter dan SehatQ. Kumpulan data ini terdiri dari 21.000 pertanyaan yang terbagi ke dalam tujuh kategori atau subtopik spesialisasi kedokteran. Kumpulan data berlabel digunakan untuk membangun model klasifikasi subtopik pertanyaan.

b. Kumpulan data tidak berlabel adalah kumpulan pertanyaan ditambah dengan kumpulan artikel kesehatan yang diambil dari beberapa situs kesehatan, yaitu

HelloSehat dan SehatQ. Jumlah data tidak berlabel yang digunakan adalah sebanyak 50.000 dokumen. Kumpulan data ini digunakan untuk memperluas ruang representasi vektor kata dalam domain kesehatan dan dengan demikian dapat meningkatkan kinerja model klasifikasi saat vektor tersebut digunakan sebagai vektor *input*.

3.3 Metode Penelitian

Metode penelitian yang digunakan dalam penelitian Tugas akhir ini ditunjukkan melalui diagram alir pada Gambar 3.1.



Gambar 3.1. Diagram alir tahapan penelitian

3.3.1 Studi Literatur

Studi literatur merupakan tahap menganalisa pengetahuan atau teori terkini terkait topik yang diangkat pada penelitian Tugas Akhir ini, yaitu *deep learning*, *language modeling*, dan klasifikasi teks. Tahap ini dilakukan dengan cara mengevaluasi dan membandingkan penelitian yang relevan mengenai topik tersebut sehingga dapat diketahui keterbatasan, kesenjangan, atau permasalahan apa yang sedang terjadi dalam penelitian saat ini. Melalui studi literatur ini ditentukan bagaimana tujuan penelitian yang akan diusulkan sehingga dapat berkontribusi dan sebagai pemecah masalah pada bidang topik penelitian tersebut.

3.3.2 Pembangunan Kumpulan Data

Penelitian ini menggunakan dua kumpulan data sehingga tahap pembangunan kedua kumpulan data tersebut akan dijelaskan ke dalam tahap yang berbeda yang akan dijelaskan sebagai berikut:

a. Kumpulan data berlabel

Data berlabel yang digunakan dalam penelitian ini diperoleh dari forum konsultasi kesehatan *online* pada situs web kesehatan. Data yang diambil adalah data teks dari kumpulan pertanyaan pengguna di forum tersebut. Data dikumpulkan menggunakan teknik *web scraping* terhadap dua situs web kesehatan, yaitu Alodokter¹ dan SehatQ². *Web scraping* dilakukan menggunakan dua teknik, yaitu *framework* Scrapy³ dan ChromeDriver yang disediakan oleh *library* Selenium. Kumpulan data yang sudah melalui proses *scraping* disimpan dalam bentuk *comma-separated values* (.csv).

Kumpulan data berlabel yang diperoleh melalui *web scraping* terdiri dari 21.000 pertanyaan. Setiap situs web telah menetapkan label atau kategori pada kumpulan pertanyaan berdasarkan spesialisasi kedokteran. Setiap situs web memiliki kategori pertanyaan yang beragam tetapi pada penelitian ini ditetapkan hanya menggunakan tujuh kategori spesialisasi kedokteran sebagai label tugas klasifikasi berdasarkan kategori yang ada di kedua situs web tersebut. Pada penelitian ini, setiap dokumen pertanyaan pengguna hanya dapat dimasukkan ke dalam satu label sehingga kumpulan data yang digunakan disebut sebagai kumpulan data berlabel tunggal. Perincian mengenai distribusi pertanyaan dan kategori yang digunakan dalam penelitian ini ditunjukkan melalui Tabel 3.1.

Tabel 3.1. Distribusi pertanyaan dan kategori kumpulan data

No	Kategori	Jumlah Pertanyaan
1	Kandungan	3000
2	Kulit dan Kelamin	3000
3	Pencernaan	3000
4	Pernapasan	3000
5	Gigi dan Mulut	3000
6	Psikologi	3000
7	Lainnya	3000
Total		21000

¹ <https://www.alodokter.com/>

² <https://www.sehatq.com/>

³ <https://scrapy.org/>

Contoh kumpulan data berlabel yang mencakup teks pertanyaan sesuai dengan kategorinya berdasarkan spesialisasi kedokteran ditunjukkan melalui Tabel 3.2.

Tabel 3.2. Contoh kumpulan data berlabel

Pertanyaan	Kategori
Dok saya mau tanya. Saya lagi hamil 27 minggu. Saya merasakan bayi itu seringnya di bawah pusar tapi menyamping. Apakah bayi saya sungsang dok? Apakah bisa membedakan gerakan bayi sungsang dalam kandungan dok? terima kasih	Kandungan
Permisi alodokter, jadi gini dok saya punya keluhan benjolan kecil2 berderet di lekukan lengan tangan kanan. Penampakan kulitnya seperti kulit melepuh gitu yg isinya cairan warna putih. Jika di pegang agak sakit rasanya. Boleh tolong penjelasannya. Dok apa itu dan gimana penanganannya? Terima kasih	Kulit dan Kelamin
Sakit perut, mual, muntah-muntah, dan BAB cair. Sudah minum obat parasetamol, tapi ga membaik,, apakah obat yang cocok untuk gejala di atas ya dok?	Pencernaan
Jadi saya 2 bulan yang lalu mengalami kondisi kritis dan sampai dibuat lubang trakeostomi untuk bantu nafas, setelah itu lubang trakeostomi ditutup tetapi saat ini nafas terasa berat dibagian bekas trakeostomi tersebut apakah ini merupakan efek setelah trakeostomi?	Pernapasan
Dok saya mau tanya, saya kan pasang crown gigi sudah 5 tahun, efek apa yang akan terjadi kalo sudah lama? Dan apa solusinya jika saya mengalami pendarahan pada crown gigi?	Gigi dan Mulut
Selamat siang dok.. saya mau tanya.. star syndrome itu penyakit apa ya dok? terus bagaimana cara mencegahnya atau bagaimana cara mengatasinya agar saya tidak sampai terkena penyakit itu dok.. terima kasih	Psikologi
Dok saya mau tanya mata kanan saya itu visus 6/9 lalu mata kiri saya 6/7,5. Saya ingin Tes Angakatan dok apakah bisa sembuh klo bisa dengan cara apa dok	Lainnya

b. Kumpulan data tidak berlabel

Kumpulan data tidak berlabel pada penelitian ini akan digunakan sebagai data pelatihan oleh model *word embedding* FastText untuk mempelajari representasi vektor kata terlatih dalam domain kesehatan. Data tidak berlabel ini diperoleh melalui teknik *web scraping* menggunakan Scrapy yang diambil melalui situs web kesehatan SehatQ dan HelloSehat⁴ serta kumpulan pertanyaan dan jawaban konsultasi kesehatan yang berada dalam situs web SehatQ dan Alodokter. Data tidak berlabel ini berupa

⁴ <https://helohehat.com/>

kumpulan konten artikel kesehatan dan pertanyaan serta jawaban di forum konsultasi pada situs web kesehatan. Total data tidak berlabel yang digunakan untuk menghasilkan vektor representasi kata menggunakan *word embedding* adalah 50.000 dokumen.

3.3.3 *Preprocessing Data*

Preprocessing merupakan salah satu tahapan dalam tugas klasifikasi teks untuk mempersiapkan data teks sebelum dimasukkan ke dalam model baik model *embedding* maupun model klasifikasi dengan mengubah bentuk tidak terstruktur dari teks menjadi representasi yang terstruktur. Pada umumnya, tahap *preprocessing* melibatkan beberapa tugas, seperti *case folding*, tokenisasi, penghapusan *stopwords*, dan *stemming*. Namun, tugas penghapusan *stopwords* dan *stemming* tidak diterapkan dalam penelitian ini karena kata-kata dalam bahasa Indonesia dibangun dengan morfologi yang kaya, seperti afiksasi, penggabungan, dan reduplikasi sehingga dengan tidak diterapkannya kedua tahapan tersebut dapat menyimpan informasi kontekstual kata dalam suatu urutan kata (Wilie et al., 2020). Selain itu, penelitian ini juga tidak menerapkan normalisasi pada kata-kata informal, seperti kata yang disingkat, salah eja, dan kata gaul ketika menghasilkan vektor kata, tujuannya adalah agar model *embedding* dapat belajar dari kata-kata tersebut. Pada saat proses pelatihan, model *embedding* diharapkan dapat menangkap kesamaan kata-kata tersebut dengan bentuk kata aslinya dengan menghitung kemungkinan atau probabilitas kemunculan kata-kata tersebut pada konteks tertentu dalam suatu urutan kata sehingga dapat membantu meningkatkan kinerja model klasifikasi teks ketika digunakan sebagai vektor *input*. Tahap *preprocessing* yang diterapkan pada penelitian ini dijelaskan sebagai berikut:

a. *Case folding*

Case folding adalah tahap mengubah semua huruf besar (*uppercase*) pada kumpulan data menjadi huruf kecil (*lowercase*). Tahap ini dilakukan untuk menetapkan bahwa huruf kecil atau besar tidak memiliki perbedaan. Contoh penerapan *case folding* pada salah satu data dari kumpulan pertanyaan yang digunakan dalam penelitian ini ditunjukkan pada Tabel 3.3.

Tabel 3.3. Contoh penerapan case folding

Pertanyaan	Hasil Case Folding
Sakit perut, mual, muntah-muntah, dan BAB cair. Sudah minum obat parasetamol, tapi ga membaik,, apakah obat yang cocok untuk gejala di atas ya dok?	sakit perut, mual, muntah-muntah, dan bab cair. sudah minum obat parasetamol, tapi ga membaik,, apakah obat yang cocok untuk gejala di atas ya dok?

b. Tokenisasi

Tokenisasi adalah proses pemecahan teks menjadi token baik itu dalam bentuk kata, subkata, frasa, tanda baca, maupun bentuk bermakna lainnya. Sebelum dilakukan tahap tokenisasi, akan baik jika dimulai dengan melakukan *data cleaning*. Pada tahap *data cleaning*, akan dilakukan penghapusan *noise* yang dapat memengaruhi suatu teks, seperti kata dengan karakter yang berulang, spasi berlebih, tanda baca, dan karakter-karakter khusus lainnya. Namun, perlu diperhatikan bahwa perlakuan terhadap suatu kata dan apa yang dianggap sebagai *noise* itu berbeda-beda tergantung pada tugas apa yang dilakukan oleh model. Misalnya, penelitian ini akan dilakukan tugas klasifikasi teks untuk mengelompokkan suatu urutan teks ke dalam kategori atau subtopik tertentu sehingga karakter khusus seperti tanda baca tidak terlalu diperhatikan dan tidak memiliki peran atau pengaruh yang berarti terhadap hasil dari model klasifikasi. Sebaliknya, tanda baca mungkin memiliki peran dan dianggap berpengaruh terhadap tugas-tugas tertentu, seperti tugas analisa sentiment, *text generation*, dll. Penerapan tokenisasi salah satu data dari kumpulan pertanyaan yang digunakan dalam penelitian ini ditunjukkan pada Tabel 3.4.

Tabel 3.4. Contoh penerapan tokenisasi

Pertanyaan	Hasil Tokenisasi
sakit perut, mual, muntah-muntah, dan bab cair. sudah minum obat parasetamol, tapi ga membaik,, apakah obat yang cocok untuk gejala di atas ya dok?	'sakit', 'perut', 'mual', 'muntahmuntah', 'dan', 'bab', 'cair', 'sudah', 'minum', 'obat', 'parasetamol', 'tapi', 'ga', 'membaik', 'apakah', 'obat', 'yang', 'cocok', 'untuk', 'gejala', 'di', 'atas', 'ya', 'dok'

3.3.4 Pembangunan Model *Embedding*

Pembangunan model *embedding* pada penelitian ini dibagi menjadi dua bagian dengan tahapan pembangunan model yang berbeda-beda, yaitu *word embedding* dan *contextual embedding*.

a. *Word embedding*

Tahap pelatihan model *word embedding* fastText terdiri dari dua tahap, yaitu persiapan korpus dan pelatihan model *word embedding*. Pelatihan model *word embedding* akan menggunakan Gensim. Gensim adalah *open-source library* Python yang digunakan untuk merepresentasikan dokumen sebagai vektor semantik menggunakan algoritma *unsupervised learning*. Gensim dirancang untuk memproses teks mentah dan tidak terstruktur. Tahap-tahap pembangunan *word embedding* dapat dijelaskan sebagai berikut:

1) Persiapan korpus

Persiapan korpus dilakukan untuk menyesuaikan bentuk data *input* yang dapat diterima oleh *library* Gensim. *Input* yang diterima oleh Gensim berupa satu data direpresentasikan sebagai satu baris, di mana satu baris merupakan *list* token dari teks yang sudah ditokenisasi. Setiap satu artikel atau pertanyaan pada kumpulan data tidak berlabel akan dimasukkan ke dalam satu baris dan dilakukan tokenisasi sehingga menghasilkan kumpulan data yang dapat diterima oleh Gensim dan dimasukkan sebagai data pelatihan model *word embedding*.

2) Pelatihan model *word embedding*

Pada penelitian ini, model fastText akan dilatih menggunakan kumpulan data tidak berlabel yang berjumlah 50.000 data dalam domain kesehatan untuk menghasilkan vektor representasi kata yang dapat menangkap semantik dan konsep terkait domain kesehatan sehingga dapat membantu meningkatkan kinerja model klasifikasi ketika hasil representasi dari model fastText digunakan sebagai vektor *input* pada BiLSTM. Model fastText pada penelitian ini tidak bangun dari awal melainkan menggunakan model yang sudah tersedia di *library* Gensim.

Kualitas vektor *word embedding* yang baik bergantung pada parameter yang digunakan pada saat pelatihan, yaitu ukuran dimensi *word embedding*, ukuran jendela konteks, dan distribusi *negative sampling*. Pengaturan parameter yang digunakan untuk melatih model *word embedding* fastText pada penelitian ini ditunjukkan pada Tabel 3.5.

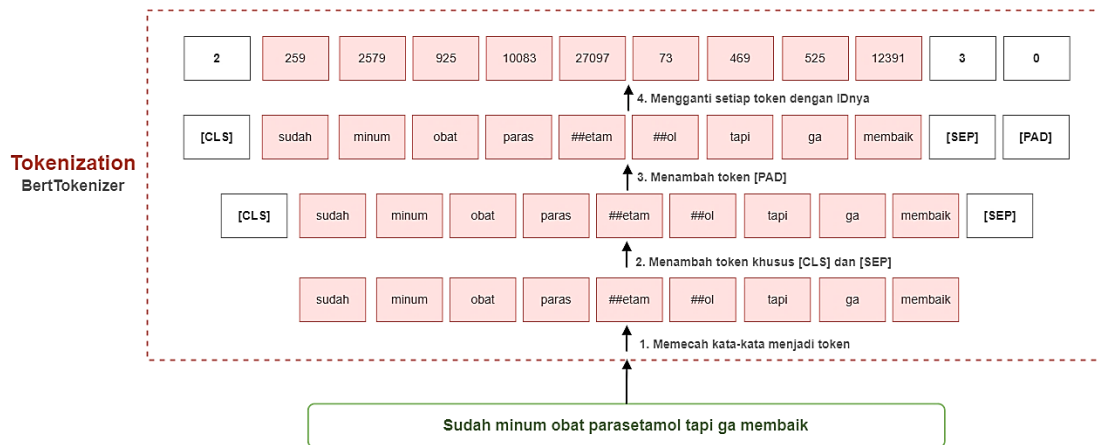
Tabel 3.5. Parameter *word embedding* fastText

Parameter	Nilai
Dimensi <i>word embedding</i>	200
Ukuran jendela (<i>windows size</i>)	5
Distribusi <i>negative sampling</i>	10

Ukuran n-gram yang digunakan sama seperti yang dilakukan oleh Bojanowski et al. (2017), yaitu mengekstraksi karakter n-gram untuk semua ukuran n , di mana $3 \leq n \leq 6$.

b) *Contextual Embedding*

Contextual embedding yang digunakan pada penelitian ini adalah model *pretrained* IndoBERT. Model terlatih IndoBERT yang digunakan adalah IndoBERT_{BASE} dengan konfigurasi model (L=12, H=768, A=12, Parameters=110M), di mana L, H, dan A masing-masing merujuk kepada jumlah lapisan *encoder*, *hidden size*, dan jumlah *self-attention heads*. Pada model ini juga dilakukan tahap persiapan korpus untuk menyesuaikan kumpulan data yang dapat diterima sebagai representasi *input* oleh model IndoBERT sesuai dengan proses yang ditunjukkan melalui Gambar 2.4. Proses persiapan korpus menjadi representasi *input* IndoBERT dilakukan oleh IndoBERT Tokenizer. IndoBERT Tokenizer menggunakan algoritma Sentence Piece dengan *byte-pair encoding* (BPE) *tokenizer* yang dilatih dengan korpus Indo4B berukuran 30.522 kosakata. Pada penelitian ini, model terlatih IndoBERT hanya digunakan untuk mengekstraksi vektor atau fitur *input* berupa *contextual embedding* yang akan dimasukkan ke dalam model BiLSTM untuk mengklasifikasi pertanyaan konsultasi kesehatan *online*. Sebaliknya, pada penelitian ini tidak menerapkan proses *fine-tuning* model IndoBERT ke dalam *downstream tasks*. Proses mempersiapkan kalimat pertanyaan menjadi representasi *input* yang dapat dimasukkan ke dalam model IndoBERT disebut sebagai proses tokenisasi yang ditunjukkan melalui Gambar 3.2.



Gambar 3.2. Proses tokenisasi menggunakan IndoBERT Tokenizer

Proses tokenisasi sekuens *input* yang dilakukan oleh *tokenizer* dijelaskan melalui langkah-langkah berikut:

- 1) Setiap rangkaian pertanyaan akan ditokenisasi menjadi token yang berupa kata dengan huruf kecil, subkata awal dari kata yang lebih besar, atau subkata terakhir dari kata yang lebih besar. *Tokenizer* memeriksa apakah suatu kata dalam sekuens ada pada kosakata atau tidak. Jika *tokenizer* tidak menemukan kata tersebut maka *tokenizer* akan memecah kata menjadi subkata yang peluang kemunculannya lebih besar pada kosakata atau sebagai upaya terakhir *tokenizer* akan memecah kata menjadi karakter-karakter individual. Pada Gambar 3.2 menunjukkan bagaimana kata “parasetamol” ditokenisasi menjadi subkata yang lebih kecil. Simbol ## yang mendahului sebagian subkata menunjukkan bahwa subkata tersebut adalah bagian atau subkata terakhir dari kata yang lebih besar yang didahului oleh subkata lain.
- 2) Selanjutnya, dua token khusus akan ditambahkan ke dalam rangkaian pertanyaan yang sudah ditokenisasi, yaitu [CLS] dan [SEP]. Token [CLS] digunakan sebagai token pertama dari setiap sekuens yang ditokenisasi, sedangkan token [SEP] digunakan di akhir sekuens sebagai tanda akhir dari satu sekuens.
- 3) Setelah itu, token [PAD] ditambahkan pada *input* sekuens untuk menunjukkan token kosong atau *padding*. Umumnya, kalimat *input* yang dimasukkan ke dalam model IndoBERT memiliki panjang yang tetap. Panjang maksimum yang dapat diterima oleh model disesuaikan berdasarkan pada kumpulan data. Kalimat dengan panjang yang lebih pendek dari panjang maksimum akan diberi *padding*,

sedangkan kalimat dengan panjang yang melebihi panjang maksimum akan dipotong.

- 4) Langkah terakhir yang dilakukan oleh *tokenizer* adalah mengganti setiap token dengan IDnya sesuai dengan indeksnya dalam kosakata atau berdasarkan tabel *embedding* yang diperoleh dari tahap *pretraining* model.

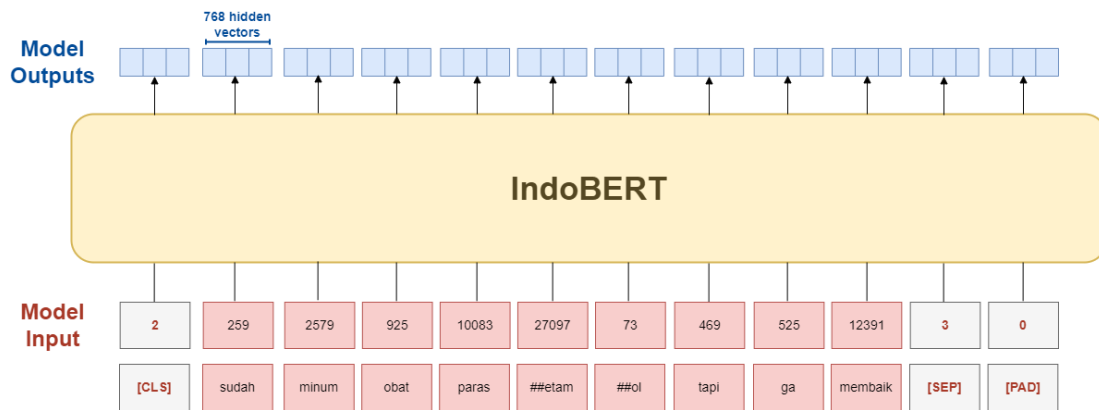
Pada dasarnya, representasi *input* model IndoBERT merupakan jumlah dari tahap *token embedding*, *segment embedding*, dan *positional embedding* seperti yang ditunjukkan pada Gambar 2.5. *Token embedding* adalah istilah untuk proses tokenisasi sekuens *input* yang dilakukan oleh *tokenizer* seperti yang ditunjukkan oleh Gambar 3.2. Sebaliknya, *segment embedding* digunakan untuk membedakan apakah token tertentu termasuk ke dalam kalimat atau token *padding* [PAD]. Jika token tersebut adalah kalimat atau token asli maka akan diberi nilai 1, sedangkan untuk token [PAD] akan diberi nilai 0. *Position embedding* adalah tahap yang digunakan untuk menunjukkan posisi urutan token dalam suatu kalimat. Contoh representasi *input* model IndoBERT ditunjukkan melalui Gambar 3.3.

Input	[CLS]	sudah	minum	obat	paras	##etam	##ol	tapi	ga	membalik	[SEP]	[PAD]
Token	E _[CLS]	E _{sudah}	E _{minum}	E _{obat}	E _{paras}	E _{##etam}	E _{##ol}	E _{tapi}	E _{ga}	E _{membalik}	E _[SEP]	E _[PAD]
Embeddings	+	+	+	+	+	+	+	+	+	+	+	+
Segment	E ₁	E ₁	E ₁	E ₁	E ₁	E ₁	E ₁	E ₁	E ₁	E ₁	E ₁	E ₀
Embeddings	+	+	+	+	+	+	+	+	+	+	+	+
Position	E ₀	E ₁	E ₂	E ₃	E ₄	E ₄	E ₅	E ₆	E ₇	E ₈	E ₉	E ₁₀
Embeddings												

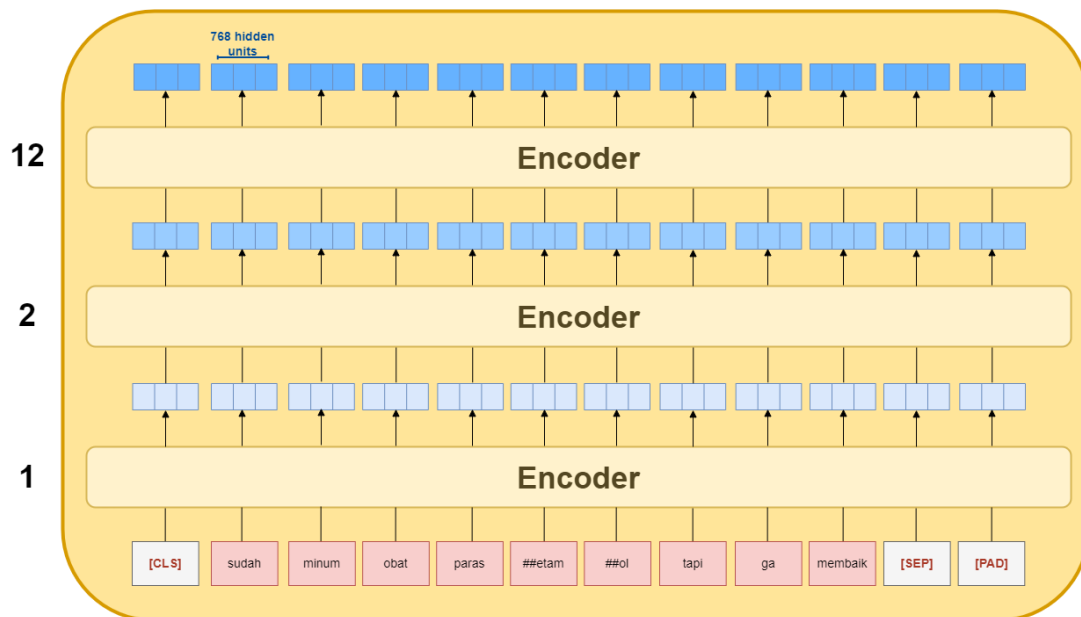
Gambar 3.3 Contoh representasi input IndoBERT

Akhirnya, kalimat atau sekuens *input* sudah menjadi bentuk yang tepat untuk diteruskan ke dalam model IndoBERT seperti yang ditunjukkan oleh Gambar 3.4. Sekuens *input* akan dimasukkan ke dalam model IndoBERT melalui tumpukan lapisan *encoder*. Pada penelitian ini, model yang digunakan adalah IndoBERT_{BASE} sehingga jumlah lapisan *encoder* yang akan dimasuki oleh vektor *input* adalah sebanyak 12 kali. Setiap blok *encoder* akan menerapkan mekanisme *self-attention* pada sekuens *input* dan menghasilkan *output* masing-masing berupa vektor untuk tiap token dengan panjang 768. *Output* dari satu lapisan *encoder* akan diteruskan menjadi *input* pada *encoder* selanjutnya. Jadi, setiap token pada sekuens *input* memiliki 12 vektor *embedding* yang dihasilkan melalui 12 lapisan *encoder*. Banyak penelitian umumnya

menggabungkan atau menjumlahkan empat lapisan terakhir *encoder* untuk mendapatkan satu vektor untuk masing-masing token. Namun, banyak metode lainnya yang dapat digunakan untuk mendapatkan vektor *embedding* dengan menggunakan vektor *output* dari berbagai lapisan *encoder*. Pemrosesan sekuens *input* dalam tumpukan lapisan *encoder* ditunjukkan oleh Gambar 3.5.



Gambar 3.4 Input dan output dalam model IndoBERT



Gambar 3.5 Sekuens Input dalam lapisan encoder model IndoBERT

3.3.5 Pembangunan Model Klasifikasi BiLSTM

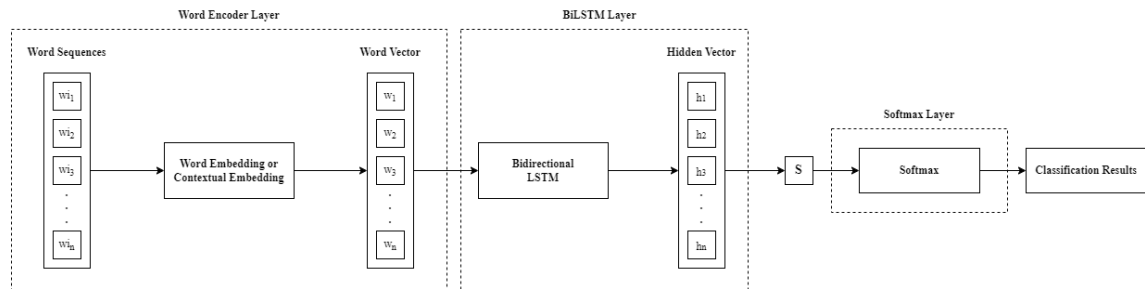
Pada pembangunan model klasifikasi, kumpulan data berlabel yang telah dilakukan tahap *preprocessing* akan dibagi ke dalam dua subdata, yaitu data *training* dan data *testing*. Kumpulan data yang terdiri dari 21.000 dokumen, 80% dari kumpulan data tersebut akan digunakan sebagai data *training* dan 20% dari kumpulan data tersebut akan digunakan sebagai data *testing*. Kumpulan data *training* digunakan untuk

melatih model klasifikasi teks dan menemukan parameter yang optimal untuk model sedangkan kumpulan data *testing* digunakan untuk menguji keberhasilan model klasifikasi yang dibangun setelah dilatih menggunakan data *training* dengan parameter yang optimal. Model klasifikasi teks yang digunakan pada penelitian ini terdiri dari dua model yang ditunjukkan pada Tabel 3.6.

Tabel 3.6. Model klasifikasi teks yang diusulkan

<i>Embedding Model</i>	<i>Pretrained Vector</i>	<i>Proposed Model</i>
<i>Word Embedding</i>	FastText	BiLSTM + FastText
<i>Contextual embedding</i>	IndoBERT	BiLSTM + IndoBERT

Selanjutnya tahapan untuk pembangunan model klasifikasi teks menggunakan BiLSTM dan ilustrasi jaringan ditunjukkan melalui Gambar 3.6.



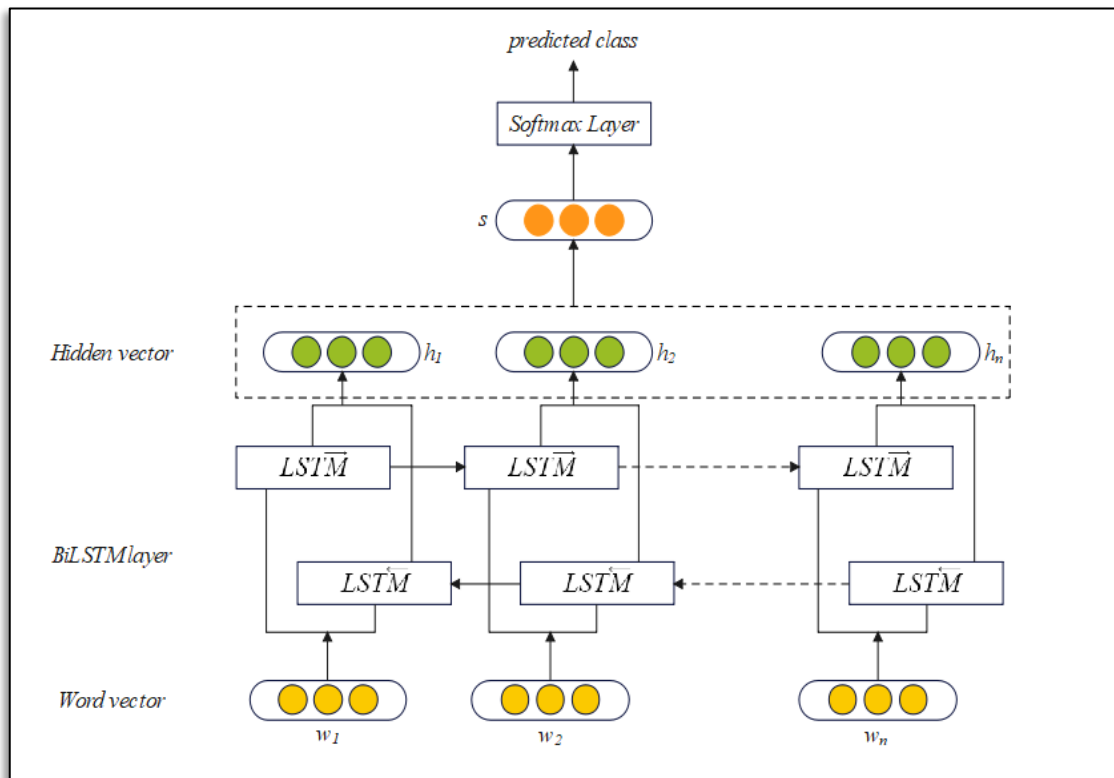
Gambar 3.6 Diagram Klasifikasi Teks pada BiLSTM (Xie et al., 2019)

Penjabaran terkait komponen-komponen yang terdapat pada Gambar 3.6 dijelaskan melalui langkah-langkah berikut:

- 1) Pada lapisan *word encoder*, setiap urutan kata pada kumpulan data dokumen pertanyaan akan diubah ke dalam vektor kata yang sesuai melalui dua teknik *embedding*, yaitu *word embedding* dan *contextual embedding* sehingga vektor kata akan dijadikan sebagai representasi *input* untuk lapisan BiLSTM.
- 2) Pada lapisan BiLSTM, vektor kata akan dipelajari menggunakan dua model LSTM yang terpisah dan berlawanan arah. Selanjutnya, model akan mempelajari *hidden vector* untuk setiap kata.
- 3) S pada diagram menunjukkan representasi akhir dari suatu kalimat. Representasi ini diperoleh melalui menggabungkan *hidden state* pertama dari *backward* LSTM dengan *hidden state* terakhir dari *forward* LSTM: $s_{emb} = [\vec{h}_n, \tilde{h}_1]$ sesuai dengan persamaan (2.5).

- 4) Selanjutnya representasi akhir kalimat atau *sentence embedding* (s_{emb}) akan dimasukkan ke dalam lapisan *softmax* untuk mengklasifikasi kalimat pertanyaan ke dalam kategori atau subtopik spesialisasi kedokteran yang berbeda. Lapisan *softmax* akan mengeluarkan nilai probabilitas yang merupakan nilai prediksi oleh model.

Tahapan-tahapan yang dilakukan pada saat pembangunan model klasifikasi teks menggunakan arsitektur BiLSTM diilustrasikan melalui Gambar 3.7.



Gambar 3.7 Arsitektur BiLSTM untuk tugas klasifikasi teks (Xie et al., 2019)

Terdapat beberapa *hyperparameter* yang digunakan dalam penelitian ini ketika melatih model klasifikasi BiLSTM, yaitu *optimization*, *learning rate*, *batch size*, *epoch*, dan *hidden units* atau *node LSTM*. Penetapan nilai untuk setiap *hyperparameter* ditunjukkan melalui Tabel 3.7.

Tabel 3.7. Pengaturan hyperparameter BiLSTM

<i>Hyperparameter</i>	Nilai
<i>Optimizer</i>	Adam
<i>Learning rate</i>	$1 \times 10^{-3}, 1 \times 10^{-4}, 1 \times 10^{-5}$
<i>Batch size</i>	16 dan 32

<i>Hyperparameter</i>	Nilai
<i>Epochs</i>	15, 30
<i>Hidden units LSTM</i>	128

Pada penelitian ini akan mempertimbangkan teknik *regularization* untuk diterapkan ketika melatih model klasifikasi, seperti *dropout*, *early stopping*, dan *L2 regularization* untuk membuat model lebih kuat terhadap *overfitting* serta menangani masalah kinerja model apabila tidak mengalami peningkatan.

3.3.6 Evaluasi Model

Pada tahap ini, akan dilakukan pengujian terhadap performa model BiLSTM menggunakan IndoBERT dan fastText sebagai vektor *input*. Performa model akan diuji menggunakan akurasi, *recall*, *precision*, dan *F1 score* sebagai metrik performa menggunakan persamaan (2.6) hingga (2.10) berdasarkan pada tabel *confusion matrix* ditunjukkan pada Gambar 2.10. Setelah didapatkan performa untuk setiap model klasifikasi yang digunakan, tahap selanjutnya adalah melakukan perbandingan dan analisa terhadap performa setiap model klasifikasi yang digunakan.

BAB IV

HASIL DAN PEMBAHASAN

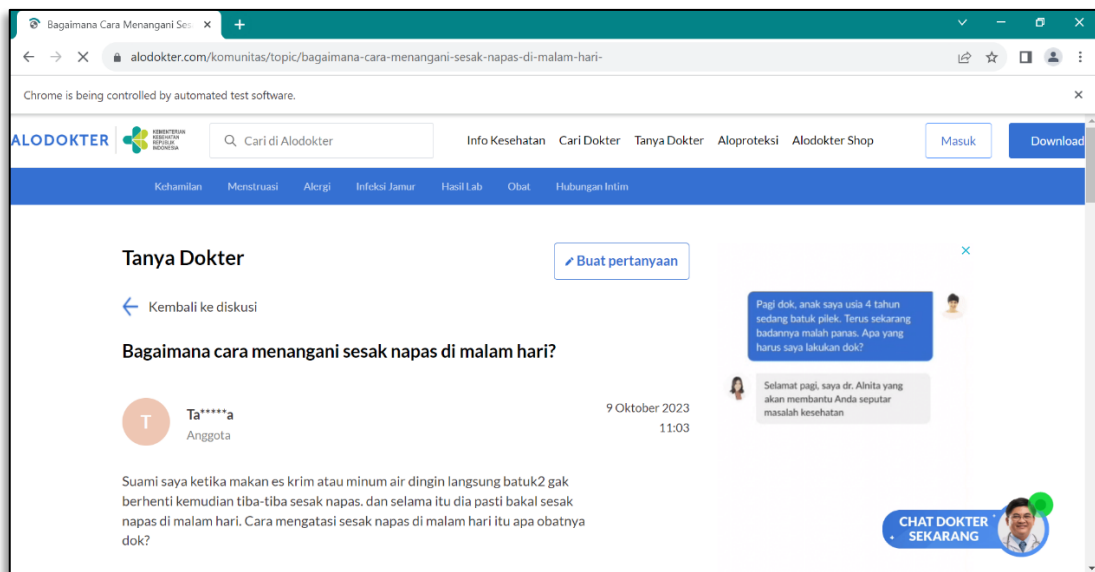
4.1 Pembangunan Kumpulan Data

Terdapat dua korpus atau kumpulan data yang digunakan dalam penelitian ini, yaitu data berlabel dan data tidak berlabel. Proses pembangunan kedua kumpulan data tersebut akan dijelaskan sebagai berikut:

4.1.1 Kumpulan Data Berlabel

Kumpulan data berlabel yang digunakan pada penelitian ini adalah kumpulan pertanyaan pengguna layanan konsultasi kesehatan *online* yang dipublikasikan secara terbuka dalam situs web kesehatan SehatQ dan Alodokter. Pengumpulan data dilakukan menggunakan dua teknik, yaitu *web crawling* dan *web scraping*. Teknik *web crawling* digunakan untuk menelusuri situs konsultasi kesehatan SehatQ dan Alodokter sedangkan teknik *web scraping* digunakan untuk mengekstrak pertanyaan pengguna dari halaman web tersebut melalui proses *parsing*. *Web scraping* pada penelitian ini diimplementasikan menggunakan bahasa pemrograman Python dengan dua teknik *scraping*, yaitu *framework* Scrapy dan ChromeDriver Selenium.

Data yang diambil dari situs web merupakan halaman HTML. Ekstraksi data pertanyaan dari halaman HTML menggunakan *selector* yang disediakan oleh Scrapy dan *web driver* Selenium. *Selector* akan memilih bagian tertentu pada halaman HTML yang berisi konten yang ingin diekstraksi dari dokumen HTML situs web seperti yang ditunjukkan melalui Gambar 4.1. Jumlah data pertanyaan yang dikumpulkan melalui proses *scraping* adalah sebanyak 21.000 data teks pertanyaan. Total token yang digunakan adalah sebanyak 626.383 token dengan 22.990 kata unik. Data yang dikumpulkan melalui *scraping* disimpan dalam bentuk file CSV seperti yang ditunjukkan melalui Gambar 4.2. Tampilan situs web ketika proses *scraping* dilakukan menggunakan ChromeDriver Selenium ditunjukkan melalui Gambar 4.3.

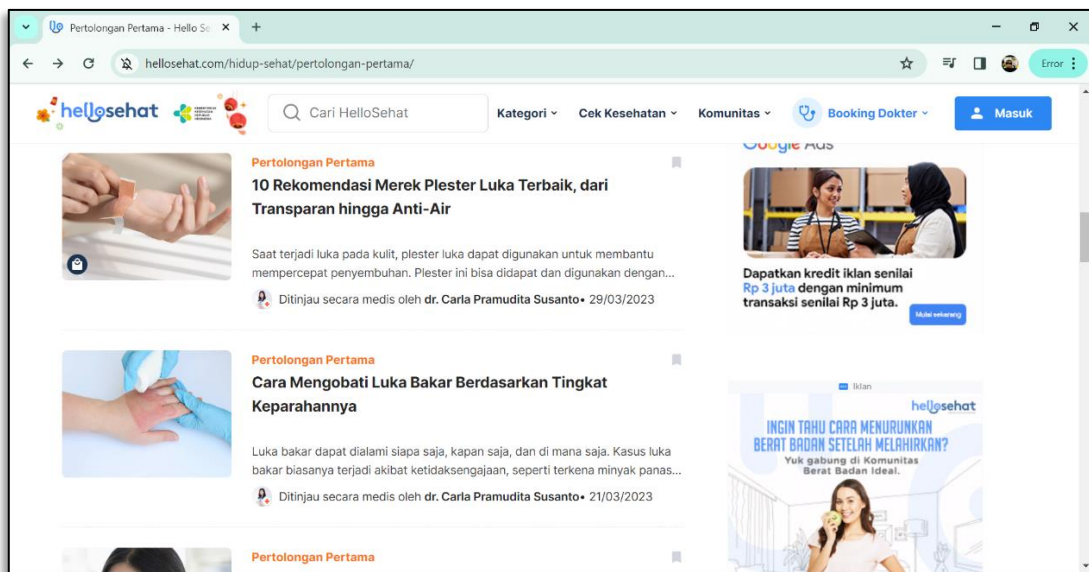


Gambar 4.3. Proses *scraping* menggunakan ChromeDriver Selenium

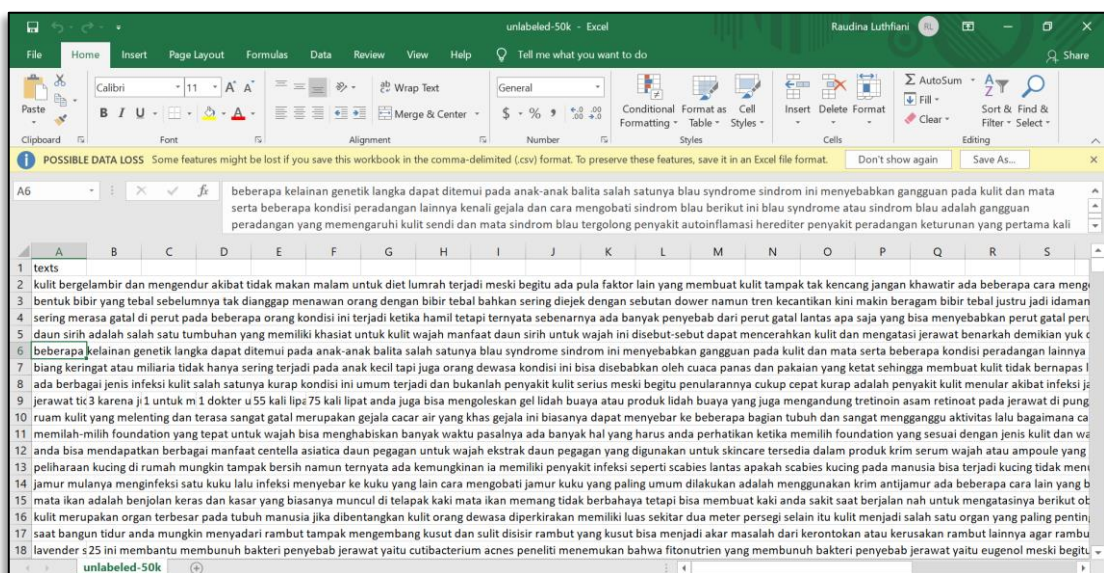
4.1.2 Kumpulan Data Tidak Berlabel

Kumpulan data tidak berlabel adalah korpus yang digunakan untuk membangun model *word embedding* fasttext. Model *embedding* ini akan digunakan untuk mengekstraksi fitur yang digunakan sebagai vektor input pada model klasifikasi BiLSTM. Kumpulan data tidak berlabel yang digunakan dalam penelitian ini adalah data dalam domain kesehatan, yaitu kumpulan konten artikel kesehatan yang diambil melalui situs web kesehatan SehatQ dan HelloSehat serta kumpulan pertanyaan dan jawaban konsultasi kesehatan yang berada dalam situs web SehatQ dan Alodokter yang ditunjukkan melalui Gambar 4.4.

Teknik yang digunakan untuk mengumpulkan data tidak berlabel adalah teknik yang sama seperti yang dilakukan ketika mengumpulkan data berlabel, yaitu menggunakan teknik *crawling* dan *scraping* yang diimplementasikan menggunakan *framework* Scrapy. Jumlah data yang dikumpulkan melalui proses *scraping* adalah 50.000 data, sedangkan total token yang digunakan adalah sebanyak 16.057.318 token. Kumpulan data tidak berlabel yang berhasil dikumpulkan melalui proses *scraping* disimpan dalam file CSV seperti yang ditunjukkan melalui Gambar 4.5.



Gambar 4.4. Situs web untuk pengumpulan data tidak berlabel

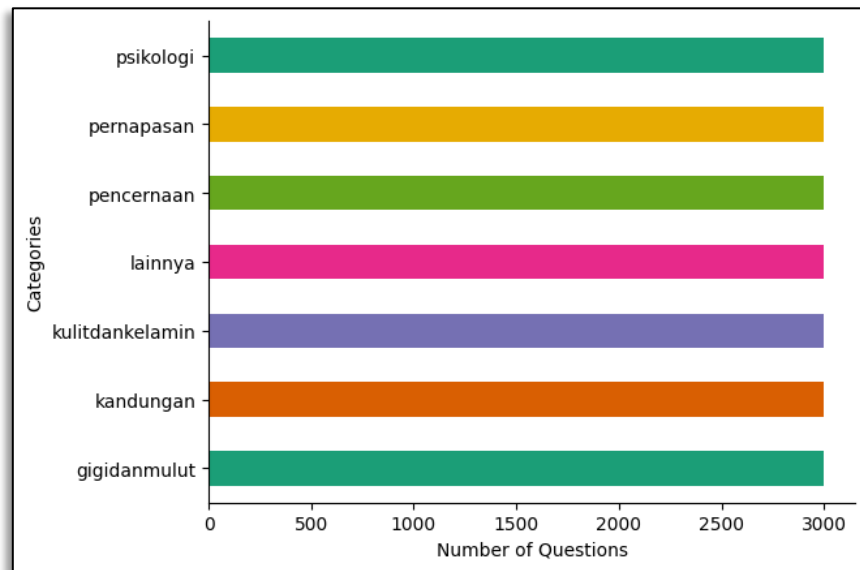


Gambar 4.5. Kumpulan data tidak berlabel disimpan dalam bentuk file CSV

4.2 Pelabelan Kumpulan Data

Kumpulan data tidak berlabel dengan jumlah 21.000 data yang sudah berhasil dikumpulkan kemudian akan dikategorikan ke dalam tujuh kategori berdasarkan spesialisasi kedokteran yang sudah ditentukan sebelumnya. Pada penelitian ini, kumpulan pertanyaan yang dikumpulkan melalui situs web SehatQ dan HelloSehat sudah dikategorisasi pada situs web tersebut ke dalam beberapa topik atau kategori berdasarkan spesialisasi kedokteran. Kategori yang digunakan dalam penelitian ini terdiri dari tujuh kategori, yaitu kandungan, kulit dan kelamin, pencernaan,

pernapasan, gigi dan mulut, psikologi, dan lainnya. Penyebaran dataset pada penelitian ini ditunjukkan melalui Gambar 4.6.



Gambar 4.6. Visualisasi persebaran data pertanyaan dan kategori kumpulan data

4.3 Text Preprocessing Kumpulan Data

Text preprocessing adalah tahap meneruskan data ke dalam suatu metode yang bertujuan untuk menghilangkan *noise* sehingga menghasilkan data yang lebih bersih sebelum data tersebut diteruskan ke dalam teknik ekstraksi fitur melalui proses *word embedding*. *Noise* adalah elemen dalam teks yang berisi informasi yang tidak berarti di dalamnya dan tidak relevan dengan tugas tertentu yang dilakukan oleh model. Pada penelitian ini, data yang masuk ke dalam proses *text preprocessing* akan melalui langkah-langkah, seperti penghapusan tanda baca (*punctuation removal*), penghapusan URL, penghapusan kata non-alfanumerik, dan penghapusan spasi di depan dan di belakang, serta spasi yang berlebihan.

```

1. def text_preprocessing(text):
2.     remover = r"^[^a-zA-Z' ]+"
3.     url_pattern = re.compile(r'https?://\S+|www\.\S+')
4.     text = url_pattern.sub(' ', text)
5.     text = re.sub(remover, ' ', text)
6.     text = re.sub('\s+', ' ', text)
7.     text = re.sub("\.{2,}", ".", text)
8.     text = text.lower()
9.     tokens = nltk.word_tokenize(text)
10.    text = " ".join([token for token in tokens if token not in
                        string.punctuation])
11.    text = text.translate(str.maketrans("", "", string.punctuation))
12.
13.    return text.strip()

```

Gambar 4.7. Text Preprocessing kumpulan data

Fungsi *text preprocessing* yang ditunjukkan melalui Gambar 4.7, menerima satu parameter berupa teks. Pada baris ke-4 sampai ke-11, teks akan dimasukkan ke dalam langkah-langkah *preprocessing* secara berurutan, yaitu penghapusan URL, penghapusan kata alfanumerik, penghapusan spasi berlebih, pengubahan kata menjadi huruf kecil atau *lower case*, dan penghapusan tanda baca. Fungsi ini akan mengembalikan teks yang sudah bersih dari *noise* sehingga layak untuk dimasukkan ke dalam model baik itu model *embedding* untuk menghasilkan representasi numerik pada teks maupun model klasifikasi teks.

4.4 Penerapan Model Word Embedding FastText

Pada penelitian ini, tujuan digunakannya pendekatan *word embedding fasttext* adalah untuk menghasilkan representasi kata berupa suatu vektor yang dapat menangkap informasi bahasa terkait domain kesehatan. Implementasi model *fasttext* yang digunakan pada penelitian ini didasarkan pada *toolkit fasttext* yang disediakan oleh Facebook. Selanjutnya, vektor kata ini akan digunakan sebagai vektor atau fitur *input* dalam model klasifikasi teks.

Pembangunan vektor kata menggunakan *fasttext* memerlukan suatu korpus atau kumpulan data yang besar. Informasi yang akan ditangkap atau digambarkan oleh vektor model sangat bergantung pada korpus yang digunakan selama pelatihan atau pembelajaran model ketika merepresentasikan suatu kata. Pada penelitian ini, model diharapkan untuk dapat menangkap informasi terkait kesehatan, sehingga korpus yang digunakan dalam penelitian ini berupa kumpulan teks yang berada dalam domain kesehatan, seperti kumpulan artikel kesehatan dan pertanyaan-jawaban pengguna pada konsultasi kesehatan.

Korpus yang digunakan dalam penelitian ini didasarkan pada data tidak berlabel yang sudah disiapkan sebelumnya melalui proses yang dilakukan dalam subbab 4.1. Sebelum korpus tersebut dimasukkan ke dalam model untuk dilatih, korpus sebaiknya sudah melewati tahap *text preprocessing* untuk menghilangkan *noise* atau informasi yang tidak relevan. Implementasi proses pelatihan model *word embedding fastText* pada korpus sehingga menghasilkan vektor atau fitur yang dapat merepresentasikan suatu kata seperti yang ditunjukkan melalui Gambar 4.8. Fungsi *train_unsupervised* digunakan untuk mempelajari vektor *embedding* kata yang terdapat dalam korpus

selama proses pelatihan. Korpus yang diteruskan ke dalam fungsi `train_unsupervised` merupakan file dalam bentuk TXT.

```
1. %%time
2. import fasttext
3.
4. FILE_PATH = "C:/Users/ASUS/OneDrive/Documents/tugas-akhir-source-
code/dataset/unlabeled-dataset/unlabeled-60k.txt"
5.
6. print("Mulai training")
7. model = fasttext.train_unsupervised(FILE_PATH, model='skipgram', dim=200,
ws=5, neg=10, epoch=30)
8. print("Selesai training")
```

Gambar 4.8. Proses pelatihan *word embedding* fastText

Model *word embedding* terlatih yang dihasilkan melalui proses pelatihan dapat digunakan untuk mengambil informasi terkait kata yang terdapat di dalam korpus, seperti mendapatkan vektor kata. Sebagai contoh, untuk mendapatkan vektor dari kata “parasetamol” maka akan ditampilkan vektor *real-valued* dengan dimensi atau panjang vektor 200 sesuai dengan pengaturan parameter saat melatih model *embedding*. Implementasi mendapatkan vektor kata melalui model *word embedding* terlatih fasttext ditunjukkan melalui Gambar 4.9. Selain itu, model fastText juga dapat menampilkan kedekatan suatu kata dengan kata lain yang muncul di dalam korpus, seperti yang ditunjukkan melalui gambar 4.10 yang menampilkan 10 kata terdekat dari kata “parasetamol”.

```

fasttext_model.get_word_vector('parasetamol')

array([-0.02591312,  0.04402022,  0.23040293,  0.19143572, -0.14204535,
        0.10381508,  0.19324087, -0.14182429, -0.09953444,  0.2364456 ,
        0.23449168, -0.4142868 ,  0.00910973,  0.05723324, -0.15371643,
        0.33624116, -0.00898808, -0.46855378,  0.16170043, -0.09047651,
        -0.22615585,  0.09509496, -0.0505137 ,  0.20260969, -0.2565318 ,
        -0.42618653, -0.14477035, -0.27083978, -0.3478033 , -0.29525107,
        0.8398949 ,  0.2184238 , -0.06320299, -0.15592103,  0.17513728,
        -0.18601088,  0.1484221 ,  0.08415821, -0.416398 ,  0.04565441,
        -0.36724535, -0.10285877, -0.07675793, -0.18934806,  0.16008824,
        0.37647033,  0.4462101 ,  0.22070591, -0.23528343,  0.08946362,
        0.03564674, -0.47222173, -0.16784659,  0.3200607 ,  0.26861092,
        -0.3170123 ,  0.10472757,  0.04974405, -0.00234783, -0.1184036 ,
        0.42456588, -0.34070292,  0.24897219,  0.33564234,  0.04689807,
        0.0348223 ,  0.37851894, -0.43569204, -0.32502648,  0.50712913,
        0.40295312,  0.18249296,  0.12281648,  0.13313381, -0.38105187,
        0.3444915 , -0.02046444, -0.08915098, -0.07249269, -0.23266348,
        -0.22036366, -0.04525817, -0.33613065, -0.58231723,  0.05313607,
        -0.12791601,  0.29850328,  0.08426698, -0.21253113, -0.3327593 ,
        0.01469735,  0.13919762, -0.09802137, -0.27617258,  0.01630471,
        0.38404506, -0.24809371, -0.4341317 ,  0.05034837, -0.09468108,
        0.04444869,  0.14860442,  0.07321306,  0.18200143,  0.00769613,
        0.07232647,  0.5936156 ,  0.00197378, -0.01755214, -0.09077319,
        0.18298066, -0.19370688,  0.02864609,  0.47319975, -0.14318103,
        -0.17852436,  0.41860425,  0.16652566, -0.12336308, -0.06587534,
        -0.22335467,  0.37780714,  0.15146166, -0.275664 ,  0.34259823,
        0.2630018 ,  0.4106194 , -0.04723669, -0.11302777, -0.07509843,
        -0.01672206,  0.22751468, -0.21316697,  0.48670653, -0.0234143 ,
        -0.04982146, -0.11418917,  0.05905622, -0.45400336,  0.20961353,
        0.18795465,  0.21338828,  0.37811086,  0.04994857,  0.3622282 ,
        0.02269537,  0.19182156, -0.41819876,  0.11242174, -0.16040042,
        0.0727198 ,  0.5877897 , -0.09777977, -0.26294336, -0.4670372 ,
        -0.47173873, -0.2503531 , -0.15307438,  0.2584126 , -0.43124154,
        -0.04057184, -0.80613613,  0.19225779, -0.33435723, -0.51752174,
        -0.0930266 ,  0.579495 ,  0.18542844, -0.11874283,  0.12879983,
        0.17492805,  0.27076414, -0.25080797,  0.05327351,  0.01147917,
        -0.1384916 ,  0.08635544, -0.00153027, -0.05663752,  0.1566903 ,
        0.15231964, -0.53180885,  0.17993575, -0.03356681, -0.1562182 ,
        -0.17807953,  0.10446029, -0.2147551 , -0.06784406,  0.14420427,
        0.33775344,  0.44496042, -0.32143924,  0.0238739 ,  0.20625377,
        -0.06262226,  0.22592865, -0.19804814,  0.06514356,  0.08354999],
      dtype=float32)

```

Gambar 4.9. Implementasi menampilkan vektor suatu kata

```

fasttext_model.get_nearest_neighbors('parasetamol')

[(0.8755280375480652, 'paracetamol'),
 (0.7517370581626892, 'peredai'),
 (0.7405918836593628, 'ibuprofen'),
 (0.6831239461898804, 'acetaminophen'),
 (0.6490201950073242, 'antinyeri'),
 (0.6468615531921387, 'acetaminofen'),
 (0.643150269985199, '3x1'),
 (0.6321232318878174, '500mg'),
 (0.62559574842453, 'asetaminofen'),
 (0.6131779551506042, 'chlorpeniramine')]

```

Gambar 4.10. Implementasi menampilkan kedekatan suatu kata dengan kata lain

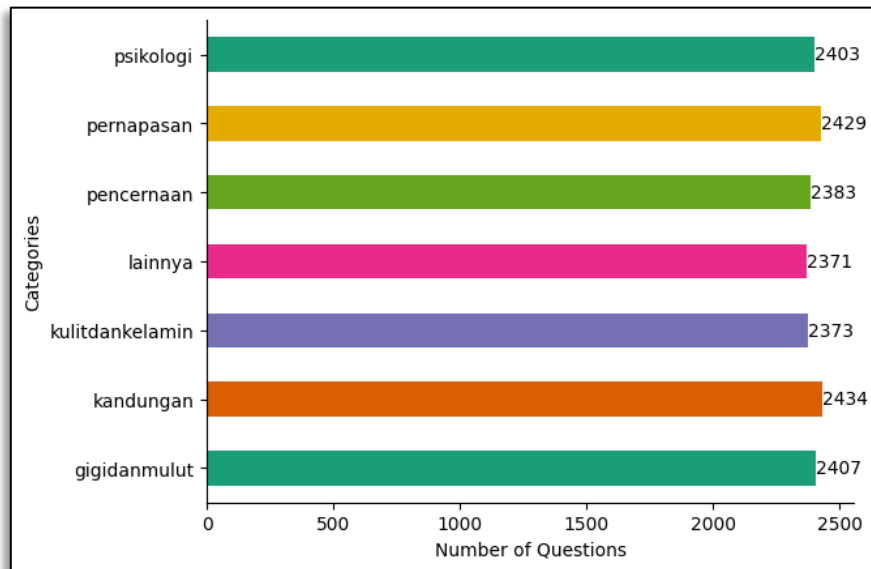
4.5 Pemisahan Kumpulan Data

Pemisahan kumpulan data diterapkan pada data berlabel yang digunakan sebagai kumpulan data untuk tugas klasifikasi pertanyaan pengguna konsultasi kesehatan menggunakan arsitektur BiLSTM. Kumpulan data dipisah menggunakan fungsi *train_test_split* dari kelas *model_selection* pada *library* Scikit-Learn, seperti yang ditunjukkan melalui Gambar 4.11. Pada penelitian ini, kumpulan data berlabel yang terdiri dari 21.000 kumpulan pertanyaan dipisah ke dalam dua bagian, yaitu 16.800 untuk data *training* (80% dari kumpulan data) dan 4.200 untuk data *testing* (20% dari

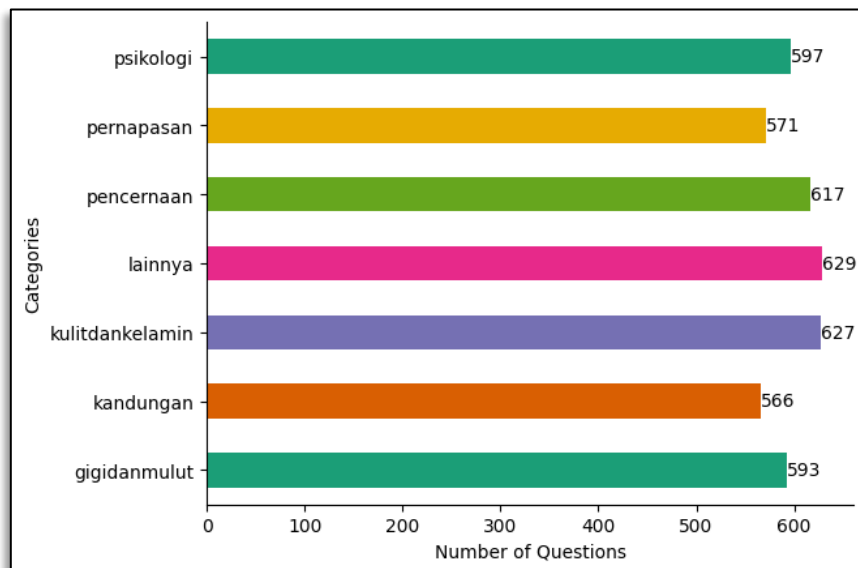
kumpulan data). Statistik kumpulan data *training* untuk setiap kategori ditunjukkan melalui Gambar 4.12, sedangkan statistik untuk kumpulan data *testing* di setiap kategori ditunjukkan pada Gambar 4.13.

```
1. #Split the dataset into training and testing sets
2. train_df, test_df = train_test_split(dataset, test_size=0.2, random_state=42)
```

Gambar 4.11. Proses pemisahan kumpulan data



Gambar 4.12. Statistik kumpulan data *training* untuk setiap kategori



Gambar 4.13. Statistik kumpulan data *testing* untuk setiap kategori

4.6 Pembangunan Model Klasifikasi BiLSTM+FastText

Berdasarkan diagram yang ditunjukkan pada Gambar 3.2, pembangunan model klasifikasi BiLSTM terdiri dari dua komponen utama, yaitu lapisan *word encoder* atau lapisan *embedding* dan lapisan BiLSTM.

4.6.1 Lapisan *Word Encoder*

a. Tokenisasi

Sebelum data *training* dimasukkan ke dalam model klasifikasi untuk dipelajari selama proses pelatihan, data *training* perlu diubah ke dalam format yang tepat sehingga dapat diterima oleh model klasifikasi. Secara umum, model *neural networks* tidak dapat memproses data teks mentah sehingga perlu diubah menjadi angka yang dapat dimengerti oleh model. Pada penelitian ini, lapisan *word encoder* mengubah urutan kata pada data teks pertanyaan menjadi vektor penyematan kata atau *word embedding* menggunakan pendekatan fastText. Vektor *word embedding* dihasilkan melalui proses pelatihan pendekatan fastText dijelaskan pada subbab 4.4.

Setiap kata masukan pada urutan teks pertanyaan akan dipetakan ke dalam bilangan bulat yang sesuai dan unik atau disebut sebagai indeks. Indeks ini akan digunakan sebagai *input* dalam lapisan *word encoder* lalu indeks tersebut diubah menjadi vektor *word embedding* fastText oleh lapisan *word encoder*. Proses ini diimplementasi menggunakan kelas Keras Tokenizer dari *library* TensorFlow. Cuplikan kode program tokenisasi berdasarkan data *training* ditampilkan melalui Gambar 4.14.

```
1. from tensorflow.keras.preprocessing.text import Tokenizer
2. from tensorflow.keras.preprocessing.sequence import pad_sequences
3. from os import truncate
4.
5. def fit_tokenizer(train_sentences, num_words, oov_token):
6.     tokenizer = Tokenizer(num_words=num_words, oov_token=oov_token)
7.
8.     tokenizer.fit_on_texts(train_sentences)
9.
10.    return tokenizer
11.
12. def seq_and_pad(sentences, tokenizer, padding, maxlen):
13.     # convert training sentences to list of indices
14.     sequences = tokenizer.texts_to_sequences(sentences)
15.
16.     # pad the sequences using the proper padding and maxlen
17.     padded_sequences = pad_sequences(sequences, maxlen=maxlen,
18.                                     padding=padding, truncating='post')
19.
20.    return padded_sequences
21.
22. #fit the tokenizer based on train set
```

```

23. tokenizer = fit_tokenizer(df['questions'], NUM_WORDS, OOV_TOKEN)
24.
25. # convert train set and test set to indices
26. X_train = seq_and_pad(train_df['questions'], tokenizer, PADDING, MAX_LEN)
27. X_test = seq_and_pad(test_df['questions'], tokenizer, PADDING, MAX_LEN)

```

Gambar 4.14. Kode program proses tokenisasi berdasarkan data *training*

Proses tokenisasi berdasarkan cuplikan kode program pada Gambar 4.14 bertujuan untuk memisahkan teks *input* menjadi kata atau subkata yang disebut sebagai token dan memetakan setiap token tersebut ke bilangan bulat atau ID. Proses ini akan menghasilkan *vocabulary* atau kosakata berupa kata-kata unik yang didasarkan pada data *training*. Proses tokenisasi ini diimplementasikan melalui fungsi *fit_tokenizer()* pada kode program. Fungsi *fit_tokenizer()* yang ditunjukkan pada baris ke-5 sampai baris ke-10 dan baris ke-23 merupakan proses untuk menginisialisasi objek *tokenizer* yang bertugas untuk menetapkan indeks berupa bilangan bulat ke tiap kata dalam kosakata berdasarkan frekuensi kemunculan kata dalam data *training*. *Tokenizer* juga dapat memberi informasi terkait berapa banyak kata dan jumlah total kata dalam teks. Sementara itu, fungsi *seq_and_pad()* yang ditunjukkan pada baris ke-12 sampai baris ke-20 adalah fungsi untuk mengubah teks atau kalimat *input* menjadi daftar indeks melalui fungsi *texts_to_sequences()* serta menyamakan panjang sekuens teks *input* sehingga *input* memiliki panjang yang sama sesuai dengan panjang maksimum yang ditetapkan melalui fungsi *pad_sequences()*. Selanjutnya, kode program pada baris ke-26 dan ke-25 merupakan penerapan dari fungsi *seq_and_pad()* terhadap data *training* dan *testing* untuk mengubah teks dalam kumpulan data tersebut menjadi kumpulan indeks sehingga menghasilkan data dengan format yang tepat untuk diterima dan diproses selanjutnya oleh model klasifikasi BiLSTM.

b. Word Embedding

Fitur yang dimasukkan ke dalam model *neural networks* BiLSTM adalah vektor *word embedding* yang sebelumnya sudah dipelajari selama proses pelatihan menggunakan pendekatan fastText yang dijelaskan dalam subbab 4.4. Sebelum fitur dimasukkan ke dalam model, diperlukan sebuah variabel yang menyimpan vektor *embedding* untuk setiap kata yang merepresentasikan kata tersebut, yaitu *embedding_matrix*. Cuplikan kode program untuk menyimpan vektor *embedding* ditunjukkan melalui Gambar 4.15.

```

1. %%time
2. import fasttext

```

```

3. import numpy as np
4.
5. MODEL_PATH = '/content/drive/MyDrive/Colab Notebooks/tugas-akhir-source-
code/model_embedding/fasttext/medical-fasttext-30epochs.bin'
6.
7. fasttext_model = fasttext.load_model(MODEL_PATH)
8.
9. embedding_matrix = []
10.
11. for word in tokenizer.word_index.keys():
12.     embedding_matrix.append(fasttext_model[word])
13.
14. embedding_matrix = np.array(embedding_matrix)
15. weights_embedding = [embedding_matrix]

```

Gambar 4.15. Kode Program proses menyimpan vektor *word embedding*

Variabel *embedding_matrix* yang ditunjukkan pada Gambar 4.15 di baris ke-14 berisi daftar vektor *word embedding* yang didasarkan pada indeks ID kata yang sebelumnya ditentukan melalui proses tokenisasi sehingga apabila indeks kata dipanggil maka indeks tersebut akan mengembalikan vektor *word embedding* dengan ukuran 200 dimensi. Variabel *embedding_matrix* akan dimasukkan ke dalam model BiLSTM sebagai fitur model atau bobot.

c. *Label Encoding*

Label atau kategori pada kumpulan data pada penelitian ini merupakan atribut kategorikal dalam bentuk teks sehingga perlu diubah menjadi label numerik melalui proses *label encoding* sebelum dimasukkan ke dalam model klasifikasi BiLSTM. Tabel 4.1 menunjukkan perbandingan label kategorikal dalam bentuk teks dan label yang sudah dikonversi ke dalam bentuk numerik.

Tabel 4.1. Perbandingan label dalam bentuk teks dan numerik

Label (Teks)	Label (Numerik)
kandungan	0
kulitdankelamin	1
gigidanmulut	2
pencernaan	3
pernapasan	4
psikologi	5
lainnya	6

4.6.2 Lapisan BiLSTM

Keluaran dari lapisan *embedding* yang berupa vektor *word embedding* fastText akan diteruskan ke dalam lapisan BiLSTM sebagai *input*. Cuplikan kode program yang

ditunjukkan pada Gambar 4.16 menjelaskan bagaimana membangun dan melatih model klasifikasi BiLSTM menggunakan vektor *word embedding* fastText sebagai vektor *input* nya. Lapisan pertama model klasifikasi ditunjukkan pada baris ke-9 merupakan lapisan *embedding* yang mengubah indeks ID kata menjadi vektor *embedding* yang prosesnya dijelaskan dalam subbab 4.6.1. Selanjutnya, model diikuti oleh satu lapisan BiLSTM yang ditunjukkan pada baris ke-13 yang terdiri dari lapisan LSTM dua arah yang berlawanan yang masing-masing terdiri dari 150 *hidden unit*. Lapisan terakhir atau lapisan *output* yang ditunjukkan pada baris ke-17 dalam cuplikan kode program merupakan lapisan *dense* atau *fully connected* dengan tujuh unit yang disesuaikan dengan kelas atau kategori. Lapisan ini menggunakan aktivasi *softmax* yang menghitung probabilitas untuk setiap kategori sehingga menghasilkan prediksi *output* model klasifikasi dengan melihat kategori dengan nilai probabilitas tertinggi.

```

1. %%time
2. # Build and train the BiLSTM model
3. maxim_words = len(embedding_matrix)
4. embedding_dim = 200
5. hidden_dim = 150
6. num_classes = len(df['label'].unique())
7.
8. model = Sequential()
9. model.add(Embedding(maxim_words, embedding_dim, input_length=MAX_LEN,
10.                    weights=weights_embedding,
11.                    trainable=False))
12. model.add(Dropout(0.2))
13. model.add(Bidirectional(LSTM(hidden_dim, activation='tanh',
14.                               kernel_regularizer=l2(0.0001),
15.                               kernel_constraint=MaxNorm(4))))
16. model.add(Dropout(0.3))
17. model.add(Dense(num_classes, activation='softmax'))
18. optimizer = Adam(learning_rate=0.001)
19. model.compile(loss='sparse_categorical_crossentropy', optimizer=optimizer,
20.               metrics=['accuracy'])
21. model.summary()
22. history = model.fit(X_train, y_train, epochs=EPOCHS, batch_size=16,
23.                    validation_split=0.1)

```

Gambar 4.16. Kode program untuk membangun dan melatih model klasifikasi BiLSTM+FastText

Terdapat beberapa pendekatan *regularization* yang digunakan ketika membangun model klasifikasi pada penelitian yang bertujuan untuk mengatasi masalah *overfitting*, salah satunya adalah *dropout*. Teknik *dropout* diterapkan pada lapisan *embedding* dengan nilai 0.2 yang ekuivalen dengan 20% dan diterapkan juga pada lapisan BiLSTM dengan nilai 0.3 atau 30%. Selanjutnya, model dikompilasi menggunakan fungsi kerugian *sparse_categorical_crossentropy* dan *optimizer* yang

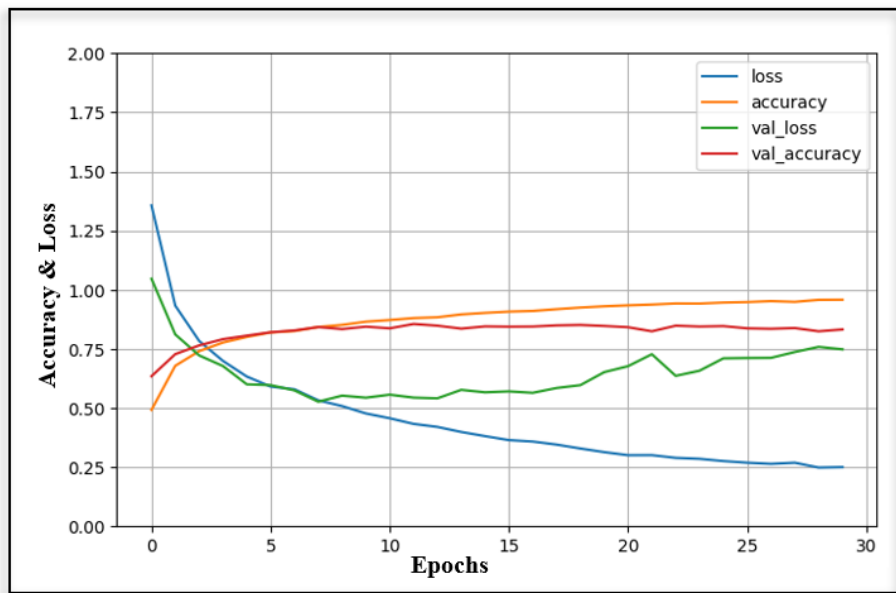
digunakan untuk melatih model adalah Adam dengan tiga skenario nilai parameter *learning rate*, yaitu 0.001, 0.0001, dan 0.00001 serta metrik yang dihitung selama pelatihan dan evaluasi model adalah akurasi. Terakhir, model dilatih dalam 30 *epoch* dengan ukuran mini-batch 16 dan 32. Gambar 4.17 menunjukkan rangkuman model yang telah dibangun dengan menampilkan semua lapisan model, termasuk nama masing-masing lapisan, bentuk *output* nya, dan jumlah parameternya. Pada akhir ringkasan model ditampilkan jumlah total parameter yang digunakan untuk melatih model, yaitu 5.021.307 parameter dengan 4.598.000 parameter *non-trainable* yang berasal dari lapisan *embedding* yang bobot *input* pada lapisan tersebut tidak terpengaruh oleh *backpropagation* atau tidak dilatih selama proses pelatihan.

Model: "sequential_1"		
Layer (type)	Output Shape	Param #
embedding_1 (Embedding)	(None, 200, 200)	4598000
dropout_2 (Dropout)	(None, 200, 200)	0
bidirectional_1 (Bidirectional)	(None, 300)	421200
dropout_3 (Dropout)	(None, 300)	0
dense_1 (Dense)	(None, 7)	2107
=====		
Total params: 5021307 (19.15 MB)		
Trainable params: 423307 (1.61 MB)		
Non-trainable params: 4598000 (17.54 MB)		
=====		
CPU times: user 990 ms, sys: 75.9 ms, total: 1.07 s		
Wall time: 2.68 s		

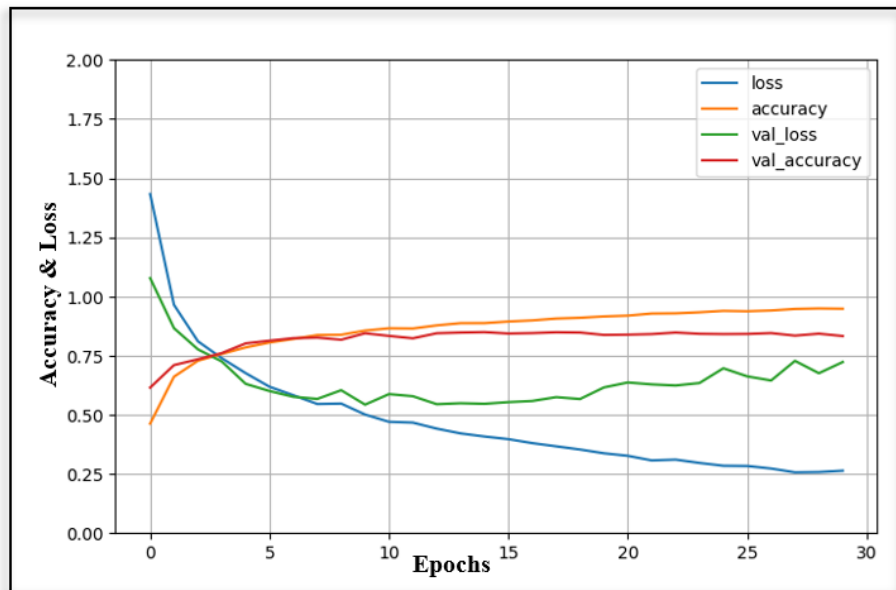
Gambar 4.17. Rangkuman model klasifikasi BiLSTM+FastText

4.7 Pelatihan Model BiLSTM+FastText

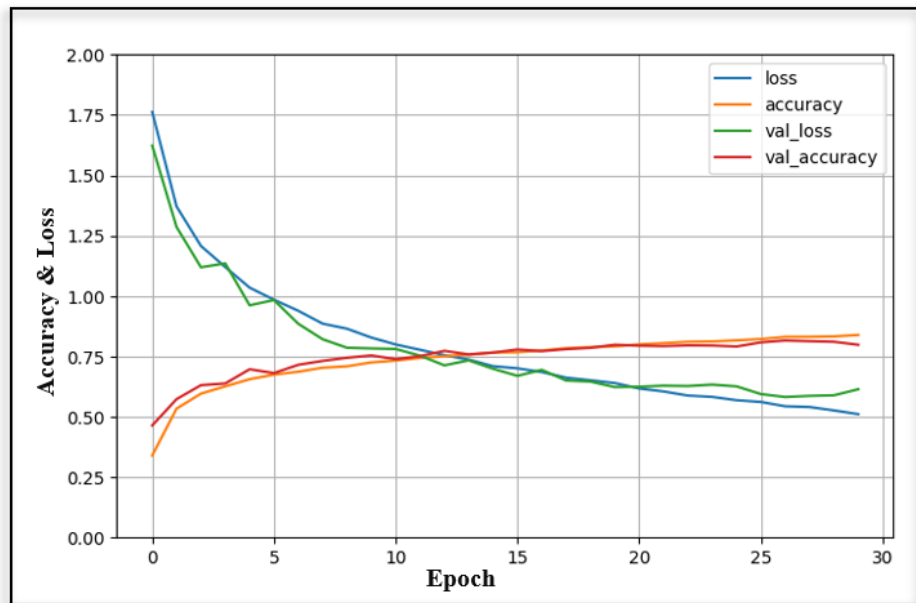
Data yang digunakan untuk pembelajaran dan pelatihan model adalah data *training* yang terdiri dari 16.800 pertanyaan dan 10% dari data *training* tersebut digunakan sebagai data *validation*. Model BiLSTM+FastText dilatih dalam enam set percobaan. Terdapat tiga skenario yang digunakan untuk penetapan nilai *learning rate*, yaitu 0.001, 0.0001, dan 0.00001 dan dua skenario untuk nilai *batch size*, yaitu 16 dan 32. Grafik perbandingan nilai akurasi dan *loss* pada data *training* dan *validation* yang didapatkan selama proses pelatihan model untuk keenam percobaan ditunjukkan melalui Gambar 4.18, Gambar 4.19, Gambar 4.20, Gambar 4.21, Gambar 4.22, dan Gambar 4.23.



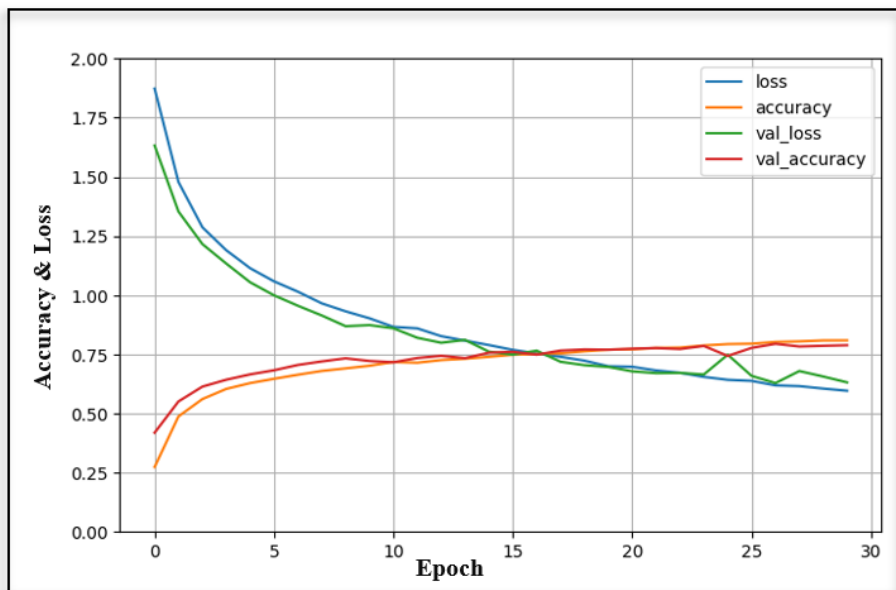
Gambar 4.18. Grafik akurasi dan *loss* model BiLSTM+FastText dengan $lr=0.001$ dan *batch size* 16



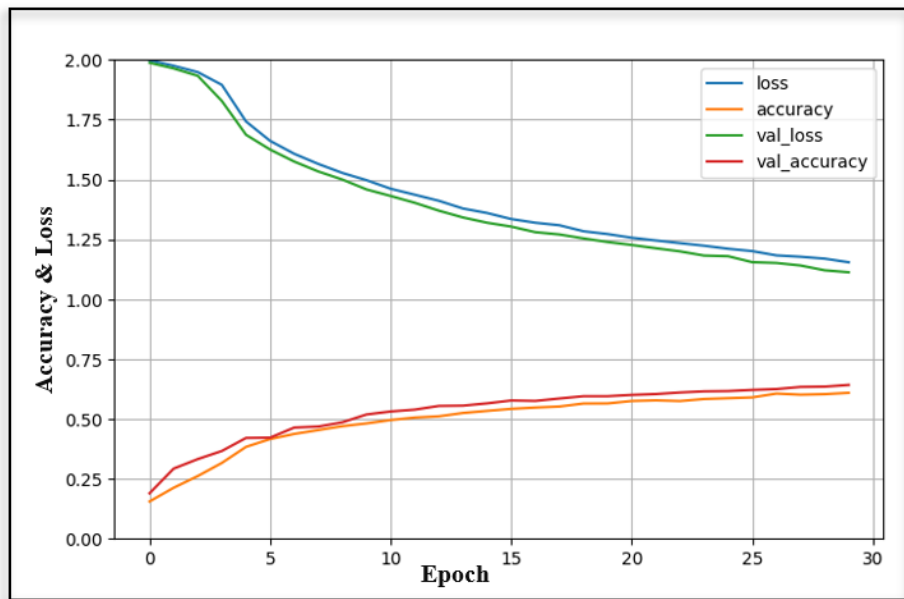
Gambar 4.19. Grafik akurasi dan *loss* model BiLSTM+FastText dengan $lr=0.001$ dan *batch size* 32



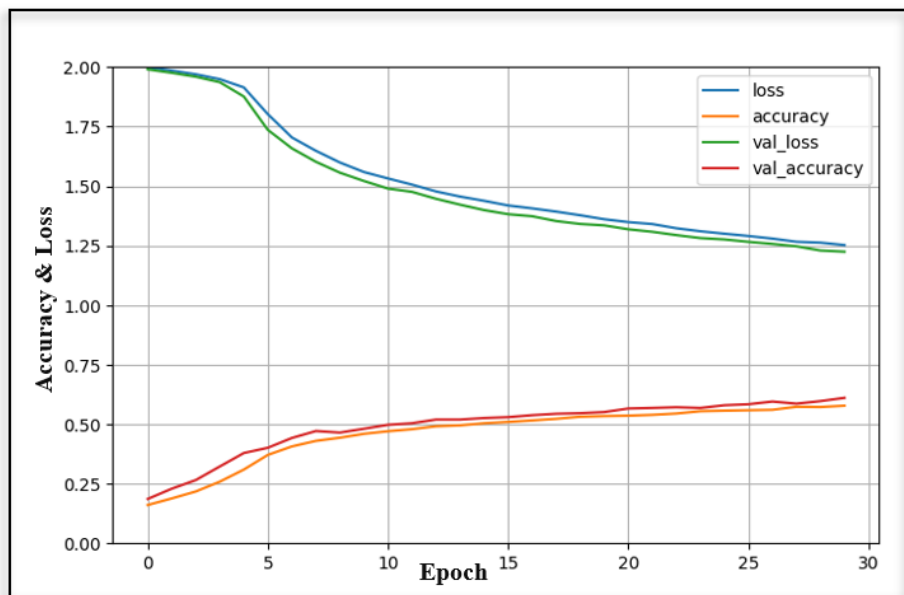
Gambar 4.20. Grafik akurasi dan *loss* model BiLSTM+FastText dengan $lr=0.0001$ dan *batch size* 16



Gambar 4.21. Grafik akurasi dan *loss* model BiLSTM+FastText dengan $lr=0.0001$ dan *batch size* 32



Gambar 4.22. Grafik akurasi dan *loss* model BiLSTM+FastText dengan $lr=0.00001$ dan *batch size* 16



Gambar 4.23. Grafik akurasi dan *loss* model BiLSTM+FastText dengan $lr=0.00001$ dan *batch size* 32

Berdasarkan grafik hasil pengamatan terhadap keenam percobaan, kinerja model diamati melalui dua kurva pembelajaran, yaitu kurva akurasi atau kurva kinerja dan kurva *loss* atau kurva optimasi pada kedua data *training* dan *validation* selama pelatihan model. Pengamatan terhadap kurva akurasi dan *loss* memberikan pemahaman tentang bagaimana keakuratan model memprediksi selama pelatihan dan

kemampuan model dalam melakukan generalisasi pada data baru. Secara umum, kurva akurasi grafik pada Gambar 4.18, Gambar 4.19, Gambar 4.20, dan Gambar 4.21 menunjukkan akurasi model saat dilatih menggunakan data *training* lebih baik daripada akurasi saat model dievaluasi menggunakan data *validation*. Sebaliknya, grafik pada percobaan kelima dan keenam yang ditunjukkan pada Gambar 4.22 dan Gambar 4.23 menunjukkan akurasi *validation* lebih baik daripada akurasi pada data *training*.

Berdasarkan hasil penelitian, terdapat tiga bentuk dan dinamika kurva pembelajaran model BiLSTM+fastText ketika dilatih menggunakan tiga *learning rate* yang berbeda. Grafik kurva pembelajaran model dengan $lr=0.001$ ditunjukkan pada Gambar 4.18 dan Gambar 4.19 dengan *batch size* 16 dan 32 berturut-turut. Pada kedua grafik tersebut, kurva akurasi pada data *training* terus meningkat secara signifikan sampai akhir *epoch*, sementara akurasi pada data *validation* tidak berubah atau bahkan menurun setelah mengalami peningkatan akurasi pada awalnya di beberapa *epoch*. Terdapat perbedaan akurasi antara kurva data *training* dan *validation* dengan akurasi data *training* secara konsisten terus melampaui akurasi data *validation* sepanjang *epoch*. Hal ini menandakan model dengan $lr=0.001$ baik menggunakan *batch size* 16 maupun 32 mengalami *overfitting* terhadap data *training*. Indikasi terjadinya *overfitting* juga ditunjukkan melalui kesenjangan atau jarak antara kurva *loss* pada data *training* dan *loss* pada data *validation* terlalu besar. Nilai *loss* data *training* mengecil secara cepat selama beberapa *epoch*, sedangkan nilai *loss* data *validation* menurun di beberapa awal *epoch* dan setelah titik tertentu nilai *loss* mulai meningkat. Kurva pembelajaran dengan bentuk *overfitting* menunjukkan model terlalu cocok dengan data *training* yang akhirnya menangkap *noise* dan pola yang tidak relevan sehingga model tidak bekerja dengan baik saat menggeneralisasi data yang tidak terlihat sebelumnya. Terdapat beberapa kemungkinan yang menyebabkan terjadinya *overfitting* pada model BiLSTM+FastText penelitian ini, seperti kumpulan data *training* yang kecil dibandingkan dengan kompleksitas model yang digunakan sehingga model tidak dapat menangkap pola dasar dari data, teknik *text preprocessing* yang digunakan kurang tepat sehingga masih terdapat *noise*, fitur yang tidak relevan, atau *outlier* di dalam data *training*, atau *hyperparameter* yang digunakan pada saat

melatih model tidak optimal, seperti *learning rate*, *batch size*, atau teknik *regularization* yang terlalu kecil atau terlalu besar.

Selanjutnya, grafik kurva pembelajaran model dengan $lr=0.0001$ dan *batch size* 16 dan 32 ditunjukkan melalui Gambar 4.20 dan Gambar 4.21. Kurva akurasi data *training* pada kedua grafik tersebut menunjukkan kurva yang secara konsisten meningkat dengan perlahan di setiap *epoch*, sedangkan kurva akurasi data *validation* meningkat di awal *epoch* dan setelah beberapa *epoch* akurasi model mencapai titik *plateau* yaitu kinerja model yang tidak atau sedikit bergerak. Kesenjangan atau perbedaan antara kedua kurva akurasi data *training* dan *validation* sangat kecil bahkan selama beberapa *epoch* kedua kurva tersebut mencapai konvergen atau menyatu pada nilai akurasi yang sama. Kesenjangan yang kecil antara kurva *training* dan *validation* menunjukkan bahwa model melakukan generalisasi dengan baik terhadap data yang tidak terlihat. Hal ini juga ditunjukkan melalui kurva *loss* pada data *training* dan *validation* yang keduanya berkurang seiring dengan bertambahnya *epoch* yang pada akhirnya kurva *loss* menurun secara lambat sampai hampir mencapai titik stabilitas serta kesenjangan antara nilai *loss* data *training* dan *validation* kecil. Pada kedua grafik menunjukkan kurva *loss* pada data *validation* yang fluktuatif di beberapa *epoch* yang menunjukkan adanya ketidakstabilan terhadap data *validation*. Walaupun demikian, kedua grafik menunjukkan kurva pembelajaran dengan nilai *loss* pada data *training* lebih rendah daripada nilai *loss* pada data *validation* dan ini adalah metrik yang diharapkan dalam model *neural network*, yaitu memperkecil nilai *loss* data *training* dengan kesenjangan yang minimal antara data *training* dan *validation*. Selain itu, hasil kedua percobaan ini menunjukkan nilai *learning rate* yang digunakan sesuai dengan kapasitas model yang digunakan pada penelitian ini dan lebih efektif untuk mendorong model melakukan generalisasi dibandingkan kedua percobaan sebelumnya.

Selanjutnya, grafik model dengan $lr=0.00001$ dan *batch size* 16 dan 32 ditunjukkan pada Gambar 4.22 dan Gambar 4.23. Berdasarkan kurva *loss* yang ditampilkan pada kedua grafik, model diindikasikan mengalami *underfitting*. Selama proses pelatihan, kurva *loss* pada data *training* dan *validation* dimulai dengan nilai *loss* yang relatif tinggi yang kemudian hanya sedikit menurun sepanjang *epoch* sehingga kedua nilai *loss* tetap berada pada nilai yang tinggi. Walaupun kesenjangan antara kurva *loss* data *training* dan *validation* kecil, nilai *loss* data *validation* lebih rendah

dibandingkan nilai *loss* pada data *training*. Dalam hal ini menunjukkan model tidak memiliki kapasitas yang cocok dengan kompleksitas data *training* sehingga model gagal dalam menangkap pola yang mendasar pada data *training*. Oleh karena itu, model tidak dapat memperoleh nilai *loss* yang rendah pada saat melatih data *training* yang akhirnya model mengalami *underfitting*. Pada penelitian ini, *underfitting* juga dapat terjadi disebabkan oleh proses pelatihan yang diberhentikan sebelum waktunya. Dengan kata lain, jika *epoch* diperpanjang, model berkemungkinan untuk melakukan pembelajaran lebih lanjut sehingga berpeluang untuk meningkatkan kinerja model.

Tabel 4.2 menampilkan perbandingan kinerja model pada *epoch* terbaik berdasarkan nilai *loss* yang minimal untuk setiap percobaan model BiLSTM+FastText selama pelatihan model. Selanjutnya, keenam model dievaluasi menggunakan kumpulan data *testing*.

Tabel 4.2. Kinerja model pelatihan BiLSTM+FastText untuk setiap percobaan

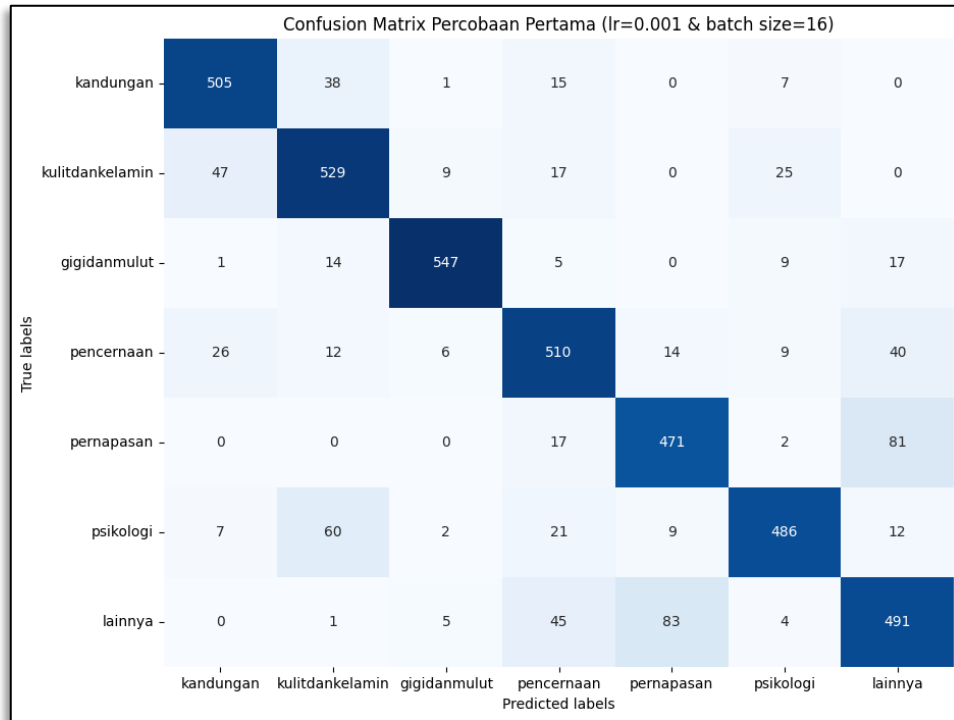
Model	Optimizer	Hidden Unit LSTM	Batch Size	Learning Rate	Training Accuracy	Training Loss	Validation Accuracy	Validation Loss	Epochs
Percobaan 1	Adam	150	16	0.001	0.8427	0.5327	0.8423	0.5269	8
Percobaan 2	Adam	150	32	0.001	0.8559	0.5011	0.8446	0.5430	10
Percobaan 3	Adam	150	16	0.0001	0.8310	0.5438	0.8161	0.5821	27
Percobaan 4	Adam	150	32	0.0001	0.8024	0.6150	0.7946	0.6287	27
Percobaan 5	Adam	150	16	0.00001	0.6093	1.1544	0.6423	1.1128	30
Percobaan 6	Adam	150	32	0.00001	0.5778	1.2524	0.6107	1.2245	30

4.8 Pengujian Model BiLSTM+FastText

Setelah model klasifikasi BiLSTM+FastText dilatih menggunakan enam set percobaan dengan skenario yang berbeda, model tersebut perlu diuji terhadap data baru untuk mengukur kinerja model dan menentukan skenario terbaik yang menghasilkan model berkinerja tinggi di antara enam set percobaan model. Pengujian model dilakukan menggunakan data *testing* yang sudah ditentukan sebelumnya pada subbab 4.5. Data *testing* yang digunakan terdiri dari 4.200 kumpulan pertanyaan dengan rincian kategori, yaitu 566 data kandungan, 627 data kulit dan kelamin, 593 data gigi dan mulut, 617 data pencernaan, 571 data pernapasan, 597 data psikologi, dan 629 data lainnya.

Model klasifikasi yang telah diuji perlu dilakukan evaluasi untuk mengetahui kinerja model terhadap data *testing*. Evaluasi dilakukan menggunakan empat metrik kinerja, yaitu *precision*, *recall*, *F1 score*, dan akurasi. Keempat metrik tersebut

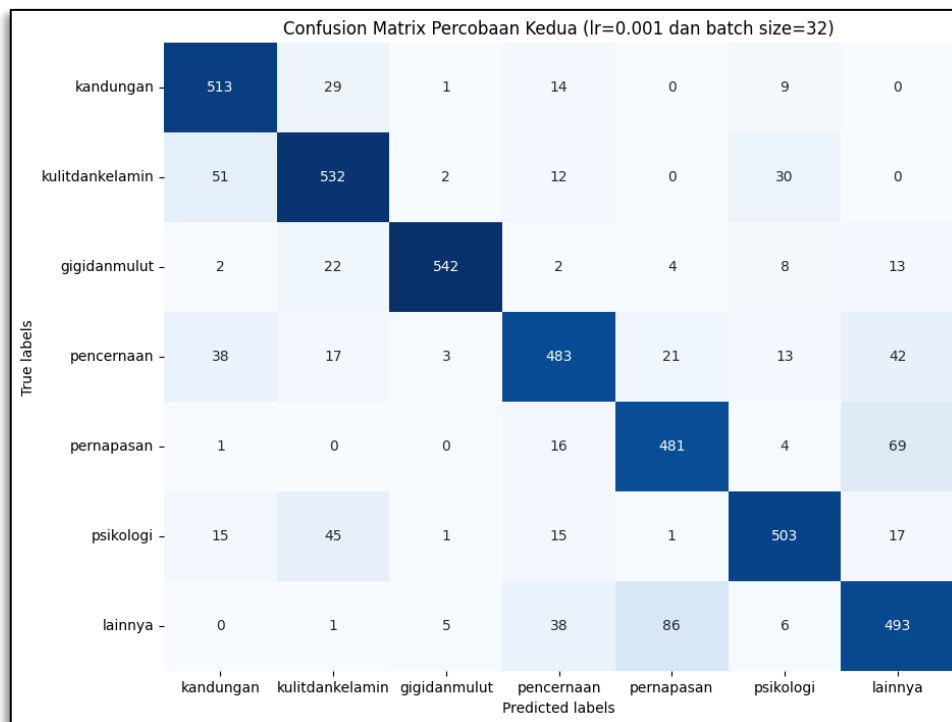
ditentukan berdasarkan diagram *confusion matrix* untuk setiap percobaan. Diagram *confusion matrix* dan metrik kinerja model BiLSTM+FastText dengan lr=0.001 serta batch size dengan ukuran 16 dan 32 beturut-turut ditunjukkan pada Gambar 4.24, Gambar 4.25, Gambar 4.26, dan Gambar 4.27.



Gambar 4.24. *Confusion matrix* model BiLSTM+FastText percobaan pertama

	precision	recall	f1-score	support
kandungan	0.8618	0.8922	0.8767	566
kulitdankelamin	0.8089	0.8437	0.8259	627
gigidanmulut	0.9596	0.9224	0.9407	593
pencernaan	0.8095	0.8266	0.8180	617
pernapasan	0.8163	0.8249	0.8206	571
psikologi	0.8967	0.8141	0.8534	597
lainnya	0.7660	0.7806	0.7732	629
accuracy			0.8426	4200
macro avg	0.8455	0.8435	0.8441	4200
weighted avg	0.8445	0.8426	0.8431	4200

Gambar 4.25. Metrik kinerja model BiLSTM+FastText percobaan pertama



Gambar 4.26. Confusion matrix model BiLSTM+FastText percobaan kedua

	precision	recall	f1-score	support
kandungan	0.8274	0.9064	0.8651	566
kulitdankelamin	0.8235	0.8485	0.8358	627
gigidanmulut	0.9783	0.9140	0.9451	593
pencernaan	0.8328	0.7828	0.8070	617
pernapasan	0.8111	0.8424	0.8265	571
psikologi	0.8778	0.8425	0.8598	597
lainnya	0.7776	0.7838	0.7807	629
accuracy			0.8445	4200
macro avg	0.8469	0.8458	0.8457	4200
weighted avg	0.8464	0.8445	0.8448	4200

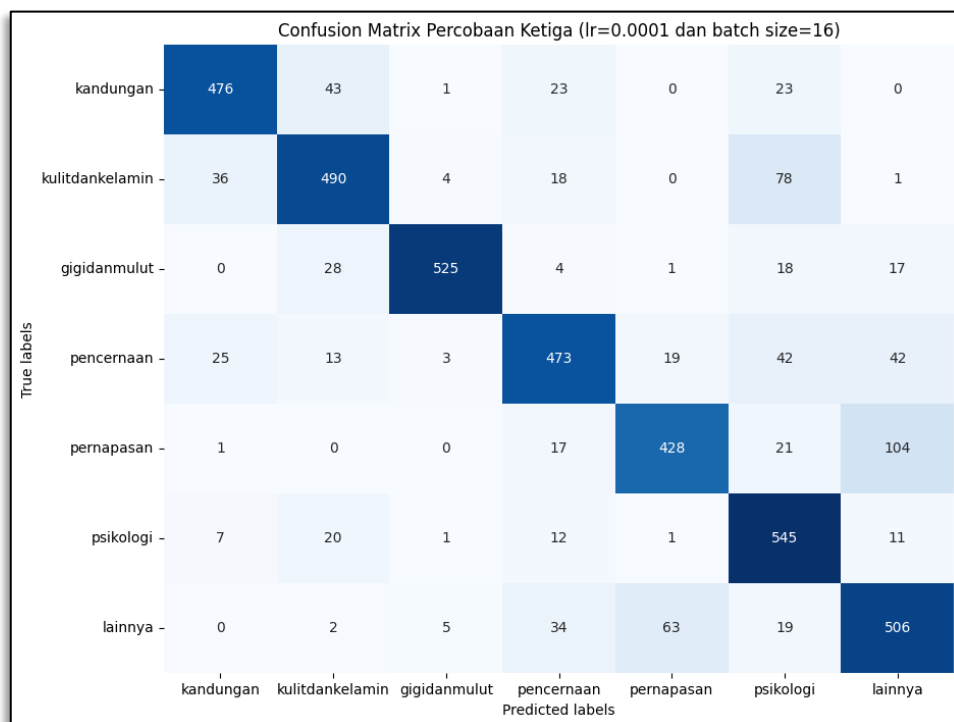
Gambar 4.27. Metrik kinerja model BiLSTM+FastText percobaan kedua

Gambar 4.25 dan Gambar 4.27 menunjukkan metrik kinerja yang dihasilkan oleh model BiLSTM+FastText dengan pengaturan lr=0.001 dengan *batch size* 16 dan 32. Model dengan *batch size* 32 menunjukkan hasil kinerja yang lebih baik dibandingkan ketika model menggunakan *batch size* 16 meskipun tidak menunjukkan perbedaan yang signifikan. Nilai akurasi yang diperoleh model menggunakan *batch size* 16 dan 32 masing-masing sebesar 0.8426 dan 0.8445, sedangkan nilai *F1 score* yang didapatkan oleh model dengan *batch size* 16 dan 32 adalah 0.8441 dan 0.8457. Jika

metrik kinerja model ditinjau berdasarkan kategorinya, kategori gigi dan mulut memiliki nilai *precision*, *recall*, dan *F1 score* tertinggi dibandingkan tujuh kategori lainnya pada kedua model percobaan. Nilai *precision*, *recall*, dan *F1 score* untuk model dengan *batch size* 16 masing-masing sebesar 0.9596, 0.9224, dan 0.9407 yang ditunjukkan pada Gambar 4.25, sedangkan nilai *precision*, *recall*, dan *F1 score* untuk model dengan *batch size* 32 adalah 0.9783, 0.9140, dan 0.9451 seperti yang ditunjukkan pada Gambar 4.27. Sementara itu, kategori lainnya adalah kategori yang mendapatkan nilai *precision* dan *F1 score* terendah pada kedua model percobaan yang masing-masing nilainya sebesar 0.7660 dan 0.7732 untuk model dengan *batch size* 16 dan 0.7776 dan 0.7807 untuk model dengan *batch size* 32, sedangkan nilai *recall* yang paling rendah adalah kategori lainnya dengan nilai 0.7806 dan kategori pencernaan dengan nilai 0.7828 untuk kedua model dengan *batch size* 16 dan 32.

Berdasarkan diagram *confusion matrix* yang ditunjukkan pada Gambar 4.24 dan Gambar 4.26, model dengan *batch size* 16 berhasil mengklasifikasikan 3.539 pertanyaan dari 4.200 data *testing* dengan benar, sedangkan model dengan *batch size* 32 berhasil mengklasifikasikan secara benar 3.547 pertanyaan dari kumpulan 4.200 data *testing*. Berdasarkan *confusion matrix* pada Gambar 4.24, kategori yang paling diprediksi secara akurat oleh model dengan *batch size* 16 adalah kategori gigi dan mulut yang menghasilkan lebih sedikit *False Positive* (FP), yaitu dari 570 pertanyaan yang diprediksi ke dalam kategori gigi dan mulut hanya 23 pertanyaan yang salah diprediksi oleh model, sedangkan sisanya diprediksi dengan benar. Sebaliknya, kategori yang paling banyak salah diprediksi oleh model sehingga menghasilkan nilai FP yang tinggi adalah kategori lainnya, sebanyak 150 pertanyaan yang salah diprediksi oleh model dari total 641 pertanyaan yang diprediksi ke dalam kategori lainnya. Sementara itu, kategori yang paling banyak dikenali atau diidentifikasi oleh model adalah kategori gigi dan mulut yang ditunjukkan melalui 547 pertanyaan berhasil diklasifikasi dengan benar oleh model dari 593 data *testing* yang berada dalam kategori gigi dan mulut. Sementara itu, kategori yang paling sulit diklasifikasi dengan benar oleh model adalah kategori lainnya yang ditunjukkan melalui model yang hanya berhasil mengidentifikasi 491 pertanyaan dari 629 data *testing* yang berada dalam kategori lainnya.

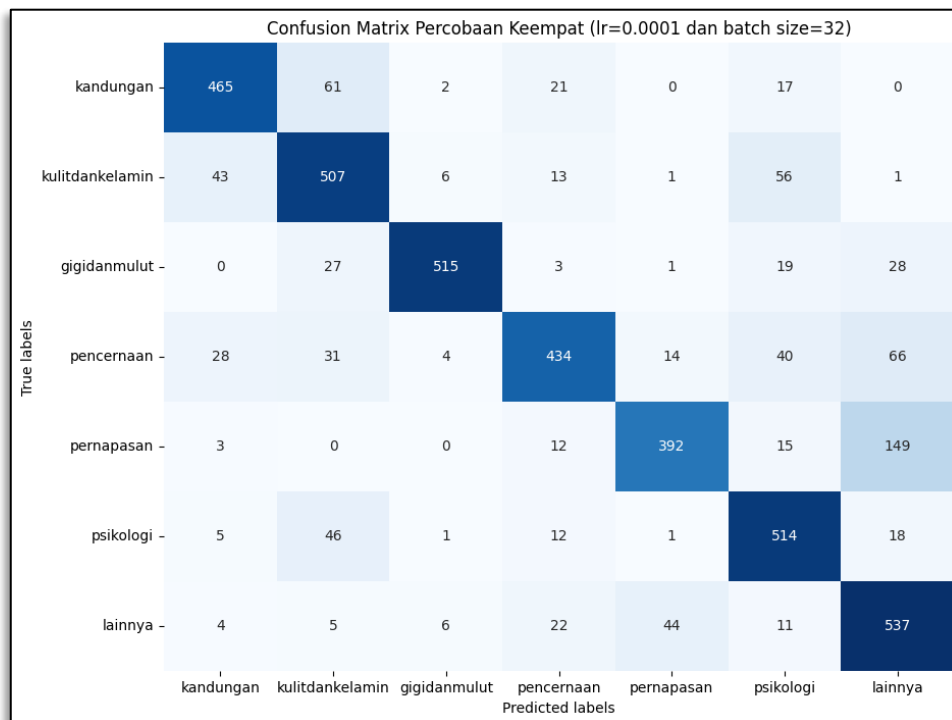
Di samping itu, berdasarkan diagram *confusion matrix* yang ditunjukkan pada Gambar 4.26, kategori yang paling banyak diprediksi secara akurat oleh model dengan *batch size* 32 adalah kategori gigi dan mulut, yaitu dari 554 pertanyaan yang diprediksi oleh model ke dalam kategori gigi dan mulut hanya 12 pertanyaan di antaranya yang salah diprediksi oleh model, sedangkan sisanya diprediksi dengan benar. Sebaliknya, kategori yang paling banyak salah diprediksi oleh model adalah kategori lainnya, yaitu sebanyak 141 pertanyaan salah diprediksi oleh model dari total 634 pertanyaan yang diprediksi ke dalam kategori lainnya. Sementara itu, kategori yang paling banyak dikenali atau diidentifikasi oleh model adalah kategori gigi dan mulut yang ditunjukkan melalui 542 pertanyaan berhasil diklasifikasi dengan benar oleh model dari 593 data dalam kategori gigi dan mulut. Sebaliknya, kategori yang paling sulit diklasifikasi dengan benar oleh model dengan *batch size* 32 adalah kategori pencernaan yang ditunjukkan melalui model yang hanya berhasil mengidentifikasi 481 pertanyaan dari 617 data *testing* yang berada dalam kategori pencernaan. Selanjutnya, hasil evaluasi kinerja model dengan $lr=0.0001$ dengan *batch size* 16 dan 32 ditampilkan melalui diagram *confusion matrix* dan tabel metrik kinerja yang ditunjukkan melalui Gambar 4.28, Gambar 4.29, Gambar 4.30, dan Gambar 4.31.



Gambar 4.28. *Confusion matrix* model BiLSTM+FastText percobaan ketiga

	precision	recall	f1-score	support
kandungan	0.8734	0.8410	0.8569	566
kulitdankelamin	0.8221	0.7815	0.8013	627
gigidanmulut	0.9740	0.8853	0.9276	593
pencernaan	0.8141	0.7666	0.7896	617
pernapasan	0.8359	0.7496	0.7904	571
psikologi	0.7306	0.9129	0.8116	597
lainnya	0.7430	0.8045	0.7725	629
accuracy			0.8198	4200
macro avg	0.8276	0.8202	0.8214	4200
weighted avg	0.8263	0.8198	0.8206	4200

Gambar 4.29. Metrik kinerja model BiLSTM+FastText percobaan ketiga



Gambar 4.30. Confusion matrix model BiLSTM+FastText percobaan keempat

	precision	recall	f1-score	support
kandungan	0.8485	0.8216	0.8348	566
kulitdankelamin	0.7489	0.8086	0.7776	627
gigidanmulut	0.9644	0.8685	0.9139	593
pencernaan	0.8395	0.7034	0.7654	617
pernapasan	0.8653	0.6865	0.7656	571
psikologi	0.7649	0.8610	0.8101	597
lainnya	0.6721	0.8537	0.7521	629
accuracy			0.8010	4200
macro avg	0.8148	0.8005	0.8028	4200
weighted avg	0.8127	0.8010	0.8019	4200

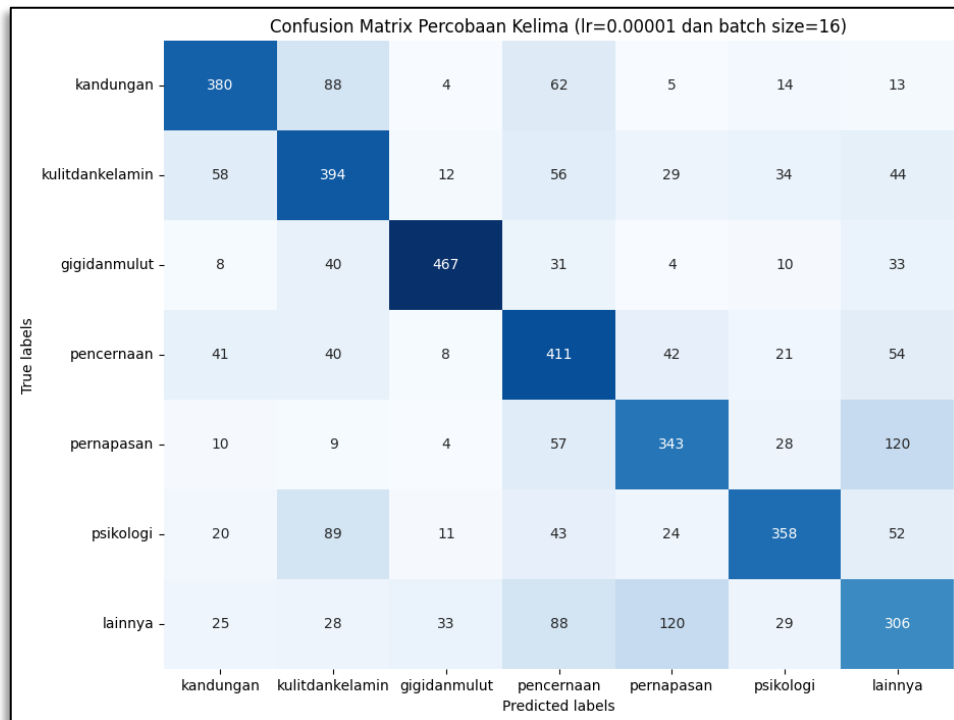
Gambar 4.31. Metrik kinerja model BiLSTM+FastText percobaan keempat

Gambar 4.29 dan Gambar 4.31 menunjukkan metrik kinerja yang dihasilkan oleh model BiLSTM+FastText dengan pengaturan $lr=0.0001$ dengan *batch size* 16 dan 32. Secara umum, model dengan *batch size* 16 menunjukkan hasil kinerja yang lebih baik dibandingkan ketika model menggunakan *batch size* 32. Hal ini berbeda dengan kinerja yang didapatkan oleh model percobaan 1 dan 2 sebelumnya. Nilai akurasi yang diperoleh model menggunakan *batch size* 16 dan 32 masing-masing sebesar 0.8198 dan 0.8010, sedangkan nilai *F1 score* yang didapatkan oleh model dengan *batch size* 16 dan 32 adalah 0.8214 dan 0.8028. Jika metrik kinerja model ditinjau berdasarkan kategorinya, nilai *precision* dan *F1 score* tertinggi didapatkan oleh kategori gigi dan mulut untuk model dengan *batch size* 16. Nilai *precision* dan *F1 score* masing-masing sebesar 0.9740 dan 0.9276, sedangkan nilai *recall* tertinggi dimiliki oleh kategori psikologi dengan nilai sebesar 0.9129. Sementara itu, nilai *precision*, *recall*, dan *F1 score* tertinggi untuk model dengan *batch size* 32 adalah kategori gigi dan mulut dengan nilai masing-masing 0.9644, 0.8686, dan 0.9139. Berdasarkan metrik kinerja kedua model percobaan yang ditunjukkan pada Gambar 4.29 dan Gambar 4.31, kategori dengan nilai *F1 score* terendah untuk kedua model adalah kategori lainnya dengan nilai masing-masing sebesar 0.7725 dan 0.7521 untuk *batch size* 16 dan 32. Selanjutnya, nilai *precision* terendah dimiliki oleh kategori psikologi dan lainnya yang masing-masing nilainya sebesar 0.7306 dan 0.6721 untuk kedua model dengan *batch size* 16 dan 32, sedangkan nilai *recall* terendah untuk kedua model diperoleh kategori pernapasan dan kategori kulit dan kelamin dengan nilai sebesar 0.7496 dan 0.8086 untuk *batch size* 16 dan 32.

Berdasarkan diagram *confusion matrix* yang ditunjukkan pada Gambar 4.28 dan Gambar 4.30, model dengan *batch size* 16 berhasil mengklasifikasikan 3.443 pertanyaan dari 4.200 data *testing* dengan benar, sedangkan model dengan *batch size* 32 berhasil mengklasifikasikan secara benar 3.364 pertanyaan dari kumpulan 4.200 data *testing*. Berdasarkan *confusion matrix* pada Gambar 4.28, kategori yang paling banyak diprediksi secara akurat oleh model dengan *batch size* 16 adalah kategori gigi dan mulut yang menghasilkan lebih sedikit *False Positive* (FP), yaitu dari 539 pertanyaan yang diprediksi ke dalam kategori gigi dan mulut hanya 14 pertanyaan yang salah diprediksi oleh model, sedangkan sisanya diprediksi dengan benar. Sebaliknya, kategori yang paling banyak salah diprediksi oleh model sehingga menghasilkan nilai FP yang tinggi adalah kategori psikologi, sebanyak 175 pertanyaan salah diprediksi oleh model dari total 681 pertanyaan yang diprediksi ke dalam kategori psikologi. Namun, kategori psikologi adalah kategori yang paling baik dikenal atau diidentifikasi oleh model yang ditunjukkan melalui 545 pertanyaan berhasil diklasifikasi dengan benar oleh model dari 597 data *testing* yang berada dalam kategori psikologi. Sementara itu, kategori yang paling sulit diklasifikasi dengan benar oleh model adalah kategori pernapasan yang ditunjukkan melalui model yang hanya berhasil mengidentifikasi 428 pertanyaan dari 571 data *testing* yang berada dalam kategori pernapasan.

Di samping itu, berdasarkan diagram *confusion matrix* yang ditunjukkan pada Gambar 4.30, kategori yang paling banyak diprediksi secara akurat oleh model dengan *batch size* 32 adalah kategori gigi dan mulut, yaitu dari 534 pertanyaan yang diprediksi oleh model ke dalam kategori gigi dan mulut hanya 19 pertanyaan yang salah diprediksi oleh model, sedangkan sisanya diprediksi dengan benar oleh model. Sebaliknya, kategori yang paling banyak salah diprediksi oleh model adalah kategori lainnya, yaitu sebanyak 262 pertanyaan salah diprediksi oleh model dari total 799 pertanyaan yang diprediksi ke dalam kategori lainnya. Sementara itu, kategori yang paling banyak diidentifikasi oleh model adalah kategori gigi dan mulut yang ditunjukkan melalui 515 pertanyaan berhasil diklasifikasi dengan benar oleh model dari 593 data dalam kategori gigi dan mulut. Sebaliknya, kategori yang paling sulit diklasifikasi dengan benar oleh model dengan *batch size* 32 adalah kategori pernapasan yang ditunjukkan melalui model yang hanya berhasil mengidentifikasi 392

pertanyaan dari 571 data *testing* yang berada dalam kategori pernapasan. Selanjutnya, hasil evaluasi kinerja model dengan $lr=0.00001$ dengan *batch size* 16 dan 32 ditampilkan melalui diagram *confusion matrix* dan tabel metrik kinerja yang ditunjukkan melalui Gambar 4.32, Gambar 4.33, Gambar 4.34, dan Gambar 4.35.



Gambar 4.32. *Confusion matrix* model BiLSTM+FastText percobaan kelima

	precision	recall	f1-score	support
kandungan	0.7011	0.6714	0.6859	566
kulitdankelamin	0.5727	0.6284	0.5992	627
gigidanmulut	0.8664	0.7875	0.8251	593
pencernaan	0.5495	0.6661	0.6022	617
pernapasan	0.6049	0.6007	0.6028	571
psikologi	0.7247	0.5997	0.6563	597
lainnya	0.4920	0.4865	0.4892	629
accuracy			0.6331	4200
macro avg	0.6445	0.6343	0.6372	4200
weighted avg	0.6420	0.6331	0.6354	4200

Gambar 4.33. Metrik kinerja model BiLSTM+FastText percobaan kelima

Confusion Matrix Percobaan Keenam ($lr=0.00001$ dan $batch\ size=32$)

True labels	kandungan	374	101	3	45	8	22	13
	kulitdankelamin	57	359	11	41	57	50	52
	gigidanmulut	6	32	467	20	28	12	28
	pencernaan	35	46	11	377	90	17	41
	pernapasan	10	16	6	65	312	18	144
	psikologi	24	85	7	33	33	352	63
	lainnya	27	19	26	75	158	29	295
		Predicted labels						
		kandungan	kulitdankelamin	gigidanmulut	pencernaan	pernapasan	psikologi	lainnya

Gambar 4.34. *Confusion matrix* model BiLSTM+FastText percobaan keenam

	precision	recall	f1-score	support
kandungan	0.7017	0.6608	0.6806	566
kulitdankelamin	0.5456	0.5726	0.5588	627
gigidanmulut	0.8795	0.7875	0.8310	593
pencernaan	0.5747	0.6110	0.5923	617
pernapasan	0.4548	0.5464	0.4964	571
psikologi	0.7040	0.5896	0.6418	597
lainnya	0.4638	0.4690	0.4664	629
accuracy			0.6038	4200
macro avg	0.6177	0.6053	0.6096	4200
weighted avg	0.6160	0.6038	0.6080	4200

Gambar 4.35. Metrik kinerja model BiLSTM+FastText percobaan keenam

Gambar 4.33 dan Gambar 4.35 menunjukkan metrik kinerja yang dihasilkan oleh model BiLSTM+FastText dengan pengaturan *hyperparameter* $lr=0.00001$ dengan *batch size* 16 dan 32. Secara umum, sama seperti percobaan 3 dan 4, model dengan *batch size* 16 menunjukkan hasil kinerja yang lebih baik dibandingkan ketika model menggunakan *batch size* 32. Nilai akurasi yang diperoleh model menggunakan *batch size* 16 dan 32 masing-masing sebesar 0.6331 dan 0.6038, sedangkan nilai *F1 score* yang didapatkan oleh model dengan *batch size* 16 dan 32 adalah 0.6331 dan 0.6053. Jika metrik kinerja model ditinjau berdasarkan kategorinya, nilai *precision*, *recall*, dan

F1 score tertinggi didapatkan oleh kategori gigi dan mulut untuk model dengan *batch size* 16. Nilai *precision*, *recall*, dan *F1 score* masing-masing sebesar 0.8664, 0.7875, dan 0.8251. Demikian juga, nilai *precision*, *recall*, dan *F1 score* tertinggi untuk model dengan *batch size* 32 dimiliki oleh kategori gigi dan mulut dengan nilai masing-masing 0.8795, 0.7875, dan 0.8310. Sementara itu, kategori dengan nilai *F1 score* terendah untuk kedua model adalah kategori lainnya dengan nilai masing-masing sebesar 0.4892 dan 0.4664 untuk *batch size* 16 dan 32. Selanjutnya, nilai *precision* terendah dimiliki oleh kategori lainnya dan pernapasan untuk kedua model dengan *batch size* 16 dan 32 yang masing-masing nilainya sebesar 0.4920 dan 0.4548, sedangkan nilai *recall* terendah untuk kedua model diperoleh kategori lainnya dengan nilai sebesar 0.4865 dan 0.4690 untuk *batch size* 16 dan 32.

Berdasarkan diagram *confusion matrix* yang ditunjukkan pada Gambar 4.32 dan Gambar 4.34, model dengan *batch size* 16 berhasil mengklasifikasikan 2.659 pertanyaan dari 4.200 data *testing* dengan benar, sedangkan model dengan *batch size* 32 berhasil mengklasifikasikan secara benar 2.536 pertanyaan dari kumpulan 4.200 data *testing*. Berdasarkan *confusion matrix* pada Gambar 4.32, kategori yang paling diprediksi secara akurat oleh model dengan *batch size* 16 adalah kategori gigi dan mulut yang menghasilkan lebih sedikit *False Positive* (FP), yaitu dari 539 pertanyaan yang diprediksi ke dalam kategori gigi dan mulut, 72 pertanyaan salah diprediksi oleh model, sedangkan sisanya diprediksi dengan benar. Sebaliknya, kategori yang paling banyak salah diprediksi oleh model sehingga menghasilkan nilai FP yang tinggi adalah kategori lainnya, sebanyak 316 pertanyaan salah diprediksi oleh model dari total 622 pertanyaan yang diprediksi ke dalam kategori lainnya. Sementara itu, kategori yang paling banyak dikenali atau diidentifikasi oleh model adalah kategori gigi dan mulut yang ditunjukkan melalui 467 pertanyaan berhasil diklasifikasi dengan benar oleh model dari 593 data *testing* yang berada dalam kategori gigi dan mulut. Sementara itu, kategori yang paling sulit diklasifikasi dengan benar oleh model adalah kategori lainnya yang ditunjukkan melalui model yang hanya berhasil mengidentifikasi 306 pertanyaan dari 629 data *testing* yang berada dalam kategori lainnya.

Di samping itu, berdasarkan diagram *confusion matrix* yang ditunjukkan pada Gambar 4.30, kategori yang paling banyak diprediksi secara akurat oleh model dengan *batch size* 32 adalah kategori gigi dan mulut, yaitu dari 531 pertanyaan yang diprediksi

oleh model ke dalam kategori gigi dan mulut, 64 pertanyaan di antaranya salah diprediksi oleh model, sedangkan sisanya diprediksi dengan benar. Sebaliknya, kategori yang paling banyak salah diprediksi oleh model adalah kategori pernapasan, yaitu sebanyak 374 pertanyaan salah diprediksi oleh model dari total 686 pertanyaan yang diprediksi ke dalam kategori pernapasan. Sementara itu, kategori yang dikenali paling banyak oleh model adalah kategori gigi dan mulut yang ditunjukkan melalui 467 pertanyaan berhasil diklasifikasi dengan benar oleh model dari 593 data dalam kategori gigi dan mulut. Sebaliknya, kategori yang paling sulit diklasifikasi dengan benar oleh model dengan *batch size* 32 adalah kategori lainnya yang ditunjukkan melalui model yang hanya berhasil mengidentifikasi 295 pertanyaan dari 629 data *testing* yang berada dalam kategori lainnya.

Tabel 4.3 menampilkan perbandingan keseluruhan metrik kinerja hasil percobaan model BiLSTM menggunakan *word embedding* FastText sebagai vektor *input* dengan mempertimbangkan tiga skenario pengaturan *hyperparameter learning rate* (0.001, 0.0001, dan 0.00001) dan dua skenario nilai *batch size* (16 dan 32). Pada semua percobaan, model dioptimalkan oleh *optimizer* Adam dan jumlah *hidden unit* pada LSTM yang digunakan adalah 150 (300 *hidden units* BiLSTM).

Tabel 4.3. Perbandingan metrik kinerja hasil percobaan model BiLSTM+FastText

Model	Optimizer	Hidden Unit LSTM	Batch Size	Learning Rate	Accuracy	F1 Score	Precision	Recall	Epochs
Percobaan 1	Adam	150	16	0.001	0.8426	0.8441	0.8455	0.8435	8
Percobaan 2	Adam	150	32	0.001	0.8445	0.8457	0.8469	0.8458	10
Percobaan 3	Adam	150	16	0.0001	0.8198	0.8214	0.8276	0.8202	27
Percobaan 4	Adam	150	32	0.0001	0.8010	0.8028	0.8148	0.8005	27
Percobaan 5	Adam	150	16	0.00001	0.6335	0.6372	0.6445	0.6343	30
Percobaan 6	Adam	150	32	0.00001	0.6053	0.6096	0.6177	0.6053	30

Berdasarkan hasil kinerja model BiLSTM+FastText pada keenam percobaan, percobaan 2 mencapai kinerja tertinggi di semua metrik yang masing-masing bernilai 0.8445, 0.8457, 0.8469, dan 0.8458 untuk akurasi, *F1 score*, *precision*, dan *recall*. Hal ini menunjukkan penggunaan *learning rate* 0.001 dan *batch size* 32 adalah konfigurasi model yang paling efektif untuk tugas klasifikasi pertanyaan konsultasi kesehatan. Sebaliknya, model percobaan 5 dan 6 menghasilkan kinerja terendah di seluruh metrik yang menunjukkan penggunaan *learning rate* (0.00001) yang sangat rendah tidak efektif untuk tugas klasifikasi pada penelitian ini. Berdasarkan Tabel 4.3

yang menampilkan perbandingan metrik kinerja untuk keenam percobaan menunjukkan bahwa penetapan nilai *learning rate* atau *step size* pada *optimizer* Adam memiliki pengaruh terhadap kinerja yang dihasilkan oleh model. Pada Tabel 4.3 memperlihatkan bagaimana penggunaan *learning rate* yang lebih tinggi, yaitu 0.001 menghasilkan kinerja yang lebih baik dibandingkan dengan *learning rate* yang lebih rendah, yaitu 0.0001 dan 0.00001. Begitu pula dengan nilai *batch size*, hasil penelitian menunjukkan hubungan antara nilai *learning rate* dan *batch size* yang saling mempengaruhi terhadap kinerja model. Pada saat model menggunakan *learning rate* yang lebih tinggi (0.001), *batch size* yang lebih besar (32) menghasilkan kinerja yang sedikit lebih optimal dibandingkan dengan *batch size* 16 meskipun perbedaan kinerja di antara keduanya tidak begitu signifikan, di mana keduanya sama-sama menghasilkan kinerja yang tinggi. Namun, saat model menggunakan *learning rate* yang lebih rendah (0.0001 dan 0.00001), *batch size* yang lebih kecil (16) menunjukkan kinerja yang lebih baik dibandingkan dengan *batch size* 32.

Setelah mengamati diagram *confusion matrix* dan membandingkannya dengan hasil metrik kinerja yang didapatkan dari keenam percobaan, kategori gigi dan mulut adalah kategori yang memiliki nilai *F1 score* tertinggi yang didapatkan oleh model di seluruh percobaan. Dalam hal ini, model berkinerja baik pada metrik *precision* dan *recall* yang ditunjukkan melalui model yang dapat mengidentifikasi dengan benar sebagian besar pertanyaan yang terdapat dalam kategori gigi dan mulut serta model menghasilkan kesalahan prediksi yang sedikit. Selain itu, metrik kinerja yang tinggi juga menunjukkan bahwa vektor *embedding* yang dihasilkan oleh model *embedding fastText* efektif dalam merepresentasikan teks pertanyaan dalam kategori gigi dan mulut sehingga model dapat membedakan antara pertanyaan yang termasuk dalam kategori gigi dan mulut dan yang tidak karena model dapat menangkap fitur atau varian dari pertanyaan yang berhubungan dengan kategori tersebut pada saat pelatihan model klasifikasi. Sebaliknya, kategori yang memiliki nilai *F1 score* terendah untuk seluruh model percobaan adalah kategori lainnya yang berarti model tidak menghasilkan kinerja yang baik pada *precision* dan *recall*. *F1 score* yang rendah ditandai dengan nilai *precision*, *recall*, atau keduanya rendah. Artinya, model gagal dalam mengidentifikasi sebagian besar pertanyaan yang terdapat dalam kategori

lainnya serta banyak pertanyaan yang salah diprediksi oleh model ke dalam kategori lainnya.

4.9 Pembangunan Model Klasifikasi BiLSTM+IndoBERT

Sama seperti yang dilakukan ketika membangun model BiLSTM+FastText, pembangunan model klasifikasi BiLSTM+IndoBERT juga terdiri dari dua komponen, yaitu lapisan *embedding* atau lapisan *word encoder* dan lapisan *neural networks* BiLSTM yang keduanya akan dijelaskan pada subbab berikut.

4.9.1 Persiapan Lapisan Word Encoder

Penelitian ini menggunakan model IndoBERT⁵ yang sudah dilatih sebelumnya untuk menghasilkan vektor penyematan kata yang kontekstual dari teks pertanyaan pada kumpulan data *training* kemudian vektor tersebut akan digunakan sebagai fitur *input* untuk model klasifikasi BiLSTM. Sebelum data *training* dimasukkan ke dalam model IndoBERT, teks pada data *training* perlu diubah dan disesuaikan ke dalam bentuk yang tepat sehingga menghasilkan representasi *input* yang dapat diterima oleh model IndoBERT. Dalam hal ini, *library* Transformer menyediakan fungsi *encoding* dari kelas AutoTokenizer berdasarkan pada model IndoBERT untuk menangani proses penyesuaian teks pertanyaan pada data *training* menjadi representasi *input* yang sesuai untuk model IndoBERT. Cuplikan kode program terkait *encoding* teks pada data *training* sehingga menjadi bentuk yang tepat untuk diteruskan ke dalam model IndoBERT ditunjukkan pada Gambar 4.36.

```
1. from transformers import AutoTokenizer, TFAutoModel, AutoModel
2. config = AutoConfig.from_pretrained("indobenchmark/indobert-base-p1")
3. tokenizer = AutoTokenizer.from_pretrained("indobenchmark/indobert-base-p1")
4. indobert_model = TFAutoModel.from_pretrained("indobenchmark/indobert-base-p1",
5.                                             config=config)
6.
7. max_seq_length = 128
8.
9. train_features = tokenizer.batch_encode_plus(
10.     train_df.questions.tolist(),
11.     add_special_tokens=True,
12.     max_length=max_seq_length,
13.     pad_to_max_length=True,
14.     truncation=True,
15.     return_attention_mask=True,
16.     return_tensors='tf'
17. )
18.
19. X_train = [np.array(train_features['input_ids']),
20.            np.array(train_features['token_type_ids']),
21.            np.array(train_features['attention_mask'])]
```

⁵ <https://github.com/indobenchmark/indobenchmark-toolkit>

Gambar 4.36. Kode Program *encoding* data *training* untuk representasi *input* model IndoBERT

Baris ke-2 sampai baris ke-5 menunjukkan bagaimana memuat model IndoBERT dan *tokenizer* IndoBERT menggunakan kelas `TFAutoModel` dan `AutoTokenizer` yang disediakan oleh *library* Transformer. Ukuran model IndoBERT terlatih yang digunakan adalah IndoBERT_{BASE} *phase* 1 atau `indobert-base-p1` yang disediakan oleh *platform* Hugging Face transformer⁶. Model IndoBERT_{BASE} *phase* 1 menerima kalimat atau sekuens dengan panjang maksimum 128 token. Penetapan panjang maksimum sekuens ditunjukkan pada baris ke-7. Proses *encoding* untuk mengubah teks pertanyaan pada data *training* ke dalam bentuk yang sesuai untuk representasi *input* model IndoBERT dilakukan menggunakan fungsi `tokenizer.batch_encode_plus` yang ditunjukkan pada baris ke-9 sampai baris ke-16. Terdapat beberapa langkah yang dilakukan oleh fungsi `tokenizer.batch_encode_plus`, yaitu pada baris ke-10 memecah sekuens pertanyaan menjadi token, baris ke-11 menambahkan token khusus [CLS] dan [SEP] serta memetakan setiap token ke ID yang sesuai, baris ke-12 sampai baris ke-14 menambahkan atau memotong token sehingga menghasilkan panjang sekuens yang sama, baris ke-15 menambahkan *attention mask* untuk membedakan token yang asli dengan token yang ditambah, yaitu [PAD]. *Output* dari proses *encoding* ini ditunjukkan melalui Gambar 4.37.

```
{'input_ids': <tf.Tensor: shape=(16800, 128), dtype=int32, numpy=
array([[ 2, 1197, 209, ..., 0, 0, 0],
       [ 2, 264, 30358, ..., 0, 0, 0],
       [ 2, 2368, 1217, ..., 0, 0, 0],
       ...,
       [ 2, 264, 30358, ..., 0, 0, 0],
       [ 2, 264, 30358, ..., 0, 0, 0],
       [ 2, 1197, 209, ..., 0, 0, 0]], dtype=int32)>, 'token_type_ids': <tf.Tensor: shape=(16800, 128), dtype=int32, numpy=
array([[0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       ...,
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0]], dtype=int32)>, 'attention_mask': <tf.Tensor: shape=(16800, 128), dtype=int32, numpy=
array([[1, 1, 1, ..., 0, 0, 0],
       [1, 1, 1, ..., 0, 0, 0],
       [1, 1, 1, ..., 0, 0, 0],
       ...,
       [1, 1, 1, ..., 0, 0, 0],
       [1, 1, 1, ..., 0, 0, 0],
       [1, 1, 1, ..., 0, 0, 0]], dtype=int32)>>}
```

Gambar 4.37. *Output* proses *encoding* data *training* untuk representasi *input* model IndoBERT

Berdasarkan *output* dari proses *encoding* data *training* yang ditunjukkan pada Gambar 4.37, terdapat tiga fitur yang akan digunakan sebagai representasi *input* model

⁶ <https://huggingface.co/indobenchmark/indobert-base-p1>

IndoBERT, yaitu *input_ids*, *token_type_ids*, dan *attention_mask*. Setiap nilai dari masing-masing fitur ini akan dimasukkan ke dalam sebuah *array list* yang ditunjukkan pada baris ke-19 di Gambar 4.36 yang selanjutnya digunakan sebagai *input* untuk diteruskan ke dalam model IndoBERT dengan tujuan untuk mengekstrak penyematan kata yang kontekstual dari setiap token pada teks pertanyaan.

4.9.2 Lapisan *Encoder* dan BiLSTM

Proses ekstraksi vektor *word embedding* yang kontekstual dari setiap token *input* yang diteruskan ke dalam model IndoBERT dan menggunakan vektor penyematan tersebut sebagai fitur *input* untuk model klasifikasi BiLSTM dijelaskan pada cuplikan kode program yang ditunjukkan pada Gambar 4.38. Terdapat dua lapisan pada kode program pembangunan model BiLSTM+IndoBERT, yaitu lapisan pertama adalah lapisan *encoder* atau lapisan *embedding* yang prosesnya ditunjukkan pada baris ke-5 sampai baris ke-12. Pada baris ke-12 menunjukkan proses ekstraksi vektor *embedding* kontekstual dari setiap token *input* untuk teks pertanyaan pada data *training*. Pada penelitian ini, vektor *embedding* yang digunakan adalah vektor yang dihasilkan oleh lapisan terakhir atau lapisan ke-12 dari lapisan *encoder* model IndoBERT. Setiap vektor *embedding* untuk tiap-tiap token *input* berukuran 768 dimensi. Selanjutnya, vektor *embedding* ini dimasukkan ke dalam lapisan kedua, yaitu lapisan model BiLSTM yang ditunjukkan pada baris ke-16. Konfigurasi model pada BiLSTM sama seperti yang digunakan ketika membangun model BiLSTM+FastText, yaitu lapisan BiLSTM terdiri dari dua lapisan LSTM dua arah yang masing-masing terdiri dari 150 *hidden units*. Setelah itu, *output* dari lapisan BiLSTM dimasukkan ke lapisan *softmax* yang menghitung probabilitas setiap kategori untuk menghasilkan prediksi model klasifikasi yang ditunjukkan pada baris ke-22.

```

1. # Load and freeze IndoBERT model
2. indobert_model = TFAutoModel.from_pretrained("indobenchmark/indobert-base-p1")
3. indobert_model.trainable = False
4.
5. # Define input for IndoBERT layers
6. input_ids = Input(shape=(max_seq_length,), dtype=tf.int32, name="input_ids")
7. input_type = Input(shape=(max_seq_length,), dtype=tf.int32, name="token_type_ids")
8. input_mask = Input(shape=(max_seq_length,), dtype=tf.int32, name="attention_mask")
9. inputs = [input_ids, input_mask, input_type]
10.
11. # Get the last hidden state layer (layer 12th) from Transformer Encoder Layers
12. contextual_embedding = indobert_model(inputs, output_hidden_states=True)[2][-1]
13. # Apply dropout on the contextual embedding layer
14. dropout_embedding = Dropout(0.2, name='dropout_1')(contextual_embedding)
15. # Pass the contextual embedding through BiLSTM Layer
16. lstm_output = Bidirectional(LSTM(150, activation='tanh',
17.                                kernel_regularizer=l2(0.0001),

```

```

18.                                     kernel_constraint=MaxNorm(4)))(dropout_embedd)
19. # Apply dropout on the BiLSTM Layer
20. dropout_lstm = Dropout(0.3, name='dropout_2')(lstm_output)
21. # Calculate the output using softmax activation function
22. outputs = Dense(7, activation='softmax')(dropout_lstm)
23.
24. # Create and compile the model
25. model = Model(inputs=inputs, outputs=outputs)
26. model.compile(loss='sparse_categorical_crossentropy',
27.               optimizer=Adam(learning_rate=0.001),
28.               metrics=['accuracy'])
29. #train the model
30. history = model.fit(X_input, y_train, epochs=15, batch_size=16,
31.                     validation_split=0.1, callbacks=callbacks_list)

```

Gambar 4.38. Kode program untuk membangun model klasifikasi BiLSTM+IndoBERT

Pada pembangunan model BiLSTM+IndoBERT juga menggunakan pendekatan *regularization* yang sama seperti model BiLSTM+FastText, yaitu teknik *dropout* yang diterapkan pada lapisan *embedding* dengan nilai 0.2 dan pada lapisan BiLSTM dengan nilai 0.3 yang masing-masing ditunjukkan pada baris ke-14 dan baris ke-20. Selanjutnya, model dikompilasi menggunakan fungsi kerugian *sparse_categorical_crossentropy* dan *optimizer* yang digunakan untuk melatih model adalah Adam dengan tiga skenario nilai parameter *learning rate*, yaitu 0.001, 0.0001, dan 0.00001 serta metrik yang dihitung selama pelatihan dan evaluasi model adalah akurasi. Konfigurasi model BiLSTM+IndoBERT dapat dirangkum pada Gambar 4.39 yang menampilkan semua lapisan model beserta nama tiap-tiap lapisan, bentuk *output* nya, dan jumlah parameternya. Pada akhir ringkasan model ditampilkan jumlah total parameter yang digunakan untuk melatih model, yaitu 125.546.251 parameter dengan 124.441.344 parameter *non-trainable* yang berasal dari lapisan *embedding* yang bobot *input* pada lapisan tersebut tidak terpengaruh oleh *backpropagation* atau tidak dilatih selama proses pelatihan.

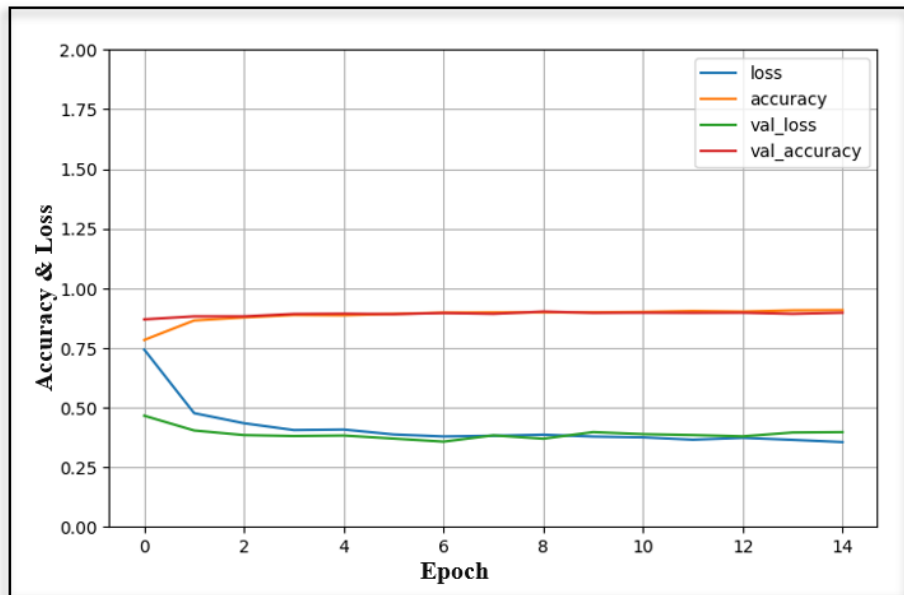
Layer (type)	Output Shape	Param #	Connected to
input_ids (InputLayer)	[(None, 128)]	0	[]
attention_mask (InputLayer)	[(None, 128)]	0	[]
token_type_ids (InputLayer)	[(None, 128)]	0	[]
tf_bert_model_1 (TFBertModel)	TFBaseModelOutputWithPoolingAndCrossAttentions(last_hidden_state=(None, 128, 768), pooler_output=(None, 768), past_key_values=None, hidden_states=((None, 128, 768), (None, 128, 768), (None, 128, 768), (None, 128, 768), (None, 128, 768), (None, 128, 768), (None, 128, 768), (None, 128, 768), (None, 128, 768), (None, 128, 768), (None, 128, 768)), attentions=None, cross_attentions=None)	124441344	['input_ids[0][0]', 'attention_mask[0][0]', 'token_type_ids[0][0]']
dropout_1 (Dropout)	(None, 128, 768)	0	['tf_bert_model_1[0][12]']
bidirectional (Bidirectional)	(None, 300)	1102800	['dropout_1[0][0]']
dropout_2 (Dropout)	(None, 300)	0	['bidirectional[0][0]']
dense (Dense)	(None, 7)	2107	['dropout_2[0][0]']
Total params: 125546251 (478.92 MB) Trainable params: 1104907 (4.21 MB) Non-trainable params: 124441344 (474.71 MB)			

Gambar 4.39. Rangkuman model klasifikasi BiLSTM+IndoBERT

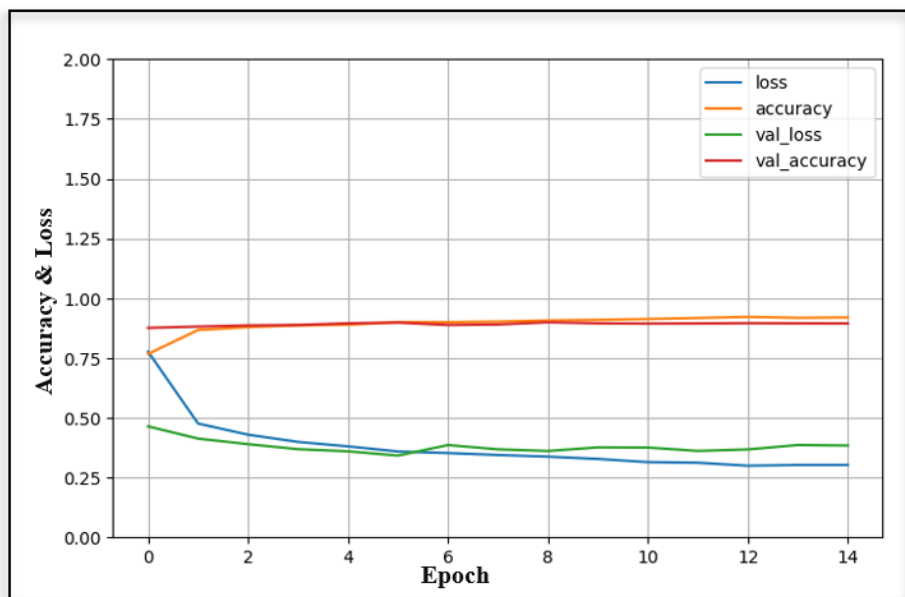
4.10 Pelatihan Model BiLSTM+IndoBERT

Sama seperti yang dilakukan ketika melatih model BiLSTM+FastText, model BiLSTM+IndoBERT juga dilatih dalam enam set percobaan dan data yang digunakan untuk pelatihan model adalah data *training* yang terdiri dari 16.800 pertanyaan dan 10% dari data *training* tersebut digunakan sebagai data *validation*. Terdapat tiga skenario yang digunakan untuk penetapan nilai *learning rate*, yaitu 0.001, 0.0001, dan 0.00001 dan dua skenario untuk nilai *batch size*, yaitu 16 dan 32. Berbeda dengan model BiLSTM+FastText yang dilatih selama 30 *epochs*, model BiLSTM+IndoBERT dilatih dalam 15 *epochs*. Grafik perbandingan nilai akurasi dan *loss* pada data *training* dan *validation* yang didapatkan selama proses pelatihan model untuk keenam

percobaan ditunjukkan melalui Gambar 4.40, Gambar 4.41, Gambar 4.42, Gambar 4.43, Gambar 4.44, dan Gambar 4.45.



Gambar 4.40. Grafik akurasi dan *loss* model BiLSTM+FastText dengan $lr=0.001$ dan *batch size* 16

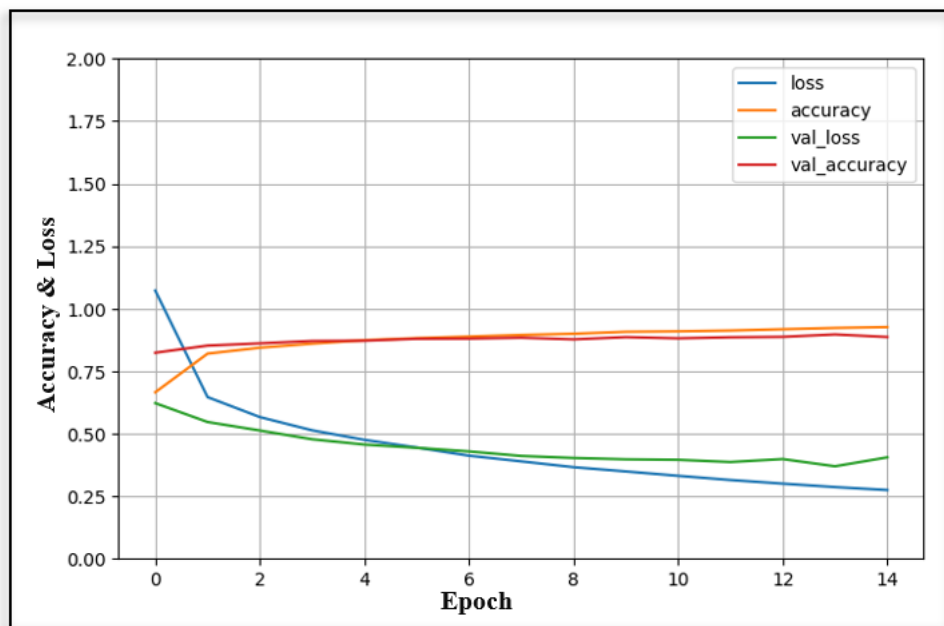


Gambar 4.41. Grafik akurasi dan *loss* model BiLSTM+FastText dengan $lr=0.001$ dan *batch size* 32

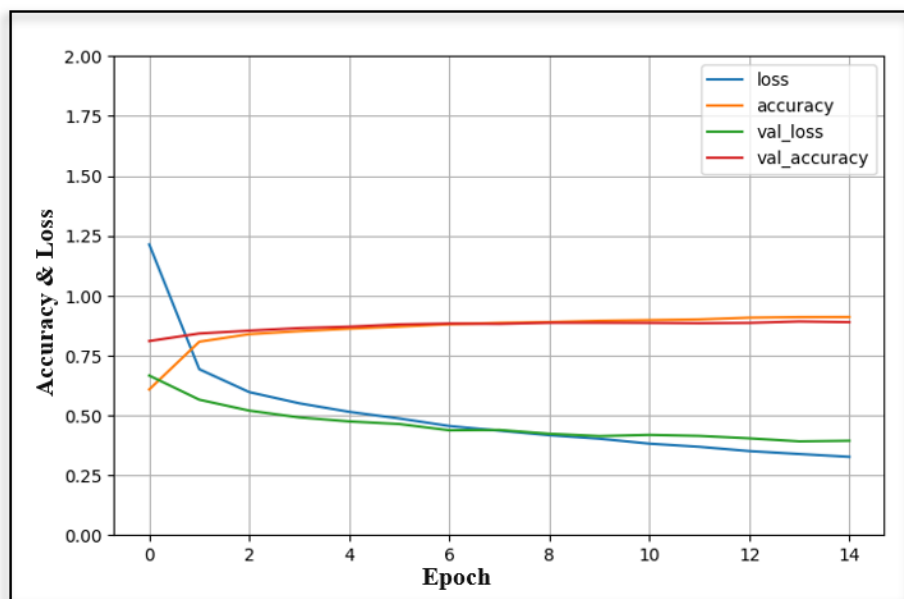
Grafik kurva pembelajaran model menggunakan $lr=0.001$ dengan *batch size* 16 dan 32 yang masing-masing ditunjukkan pada Gambar 4.40 dan Gambar 4.41 memperlihatkan kurva akurasi data *training* pada kedua grafik tersebut meningkat

secara cepat pada awal *epoch* dan setelah melewati empat *epoch* kurva mencapai titik stabil dan meningkat secara konstan sampai akhir *epoch*. Namun, kurva akurasi data *validation* meningkat secara perlahan pada awal *epoch* tetapi secara cepat mencapai kondisi stabil yang kemudian selama beberapa *epoch* nilai akurasinya tetap konstan setelah kenaikan awal dan tidak ada kenaikan yang signifikan. Secara umum, nilai akurasi data *training* sedikit lebih tinggi dibandingkan dengan akurasi data *validation* walaupun tidak terdapat perbedaan yang signifikan antara kurva akurasi *training* dan *validation* untuk model dengan *batch size* 16 bahkan selama beberapa *epoch* kedua kurva akurasi tersebut mencapai konvergensi, sedangkan untuk model dengan *batch size* 32 terdapat sedikit perbedaan akurasi antara kurva akurasi *training* dan *validation*. Dalam hal ini, kurva akurasi pada model dengan $lr=0.001$ dan *batch size* 16 dan 32 menunjukkan bahwa model mempelajari data *training* dengan sangat baik bahkan hampir sempurna dan mencapai keseimbangan yang baik dengan data *validation*.

Sementara itu, kurva *loss* data *training* pada kedua model *batch size* 16 dan 32 menurun atau berkurang dengan cepat pada beberapa awal *epoch* pertama dan kemudian menurun secara perlahan sampai kepada stabilitas pada sisa *epoch* yang menunjukkan bahwa model secara efektif meminimalkan kesalahan pada data *training*, sedangkan kurva *loss* data *validation* untuk kedua model menurun di awal *epoch* dan setelah beberapa *epoch* nilai *loss* nya tidak berubah atau menurun dengan konstan dan kemudian pada *epoch* tertentu nilai *loss* mulai meningkat dan berfluktuasi lalu pada akhirnya kurva *loss* mencapai stabilitas menjelang akhir *epoch*. Artinya, pada tahap awal pelatihan model menggeneralisasi dengan baik data yang tidak terlihat tetapi seiring dengan penambahan *epoch* model mulai *overfitting* dengan data *training*. Akan tetapi, kesenjangan antara kurva *loss* data *training* dan *validation* minimal yang menunjukkan model telah digeneralisasi ke data *validation* tanpa *overfitting* yang signifikan. Selanjutnya, grafik kurva akurasi dan *loss* untuk model yang menggunakan $lr=0.0001$ dengan *batch size* 16 dan 32 masing-masing ditunjukkan pada Gambar 4.42 dan Gambar 4.43.



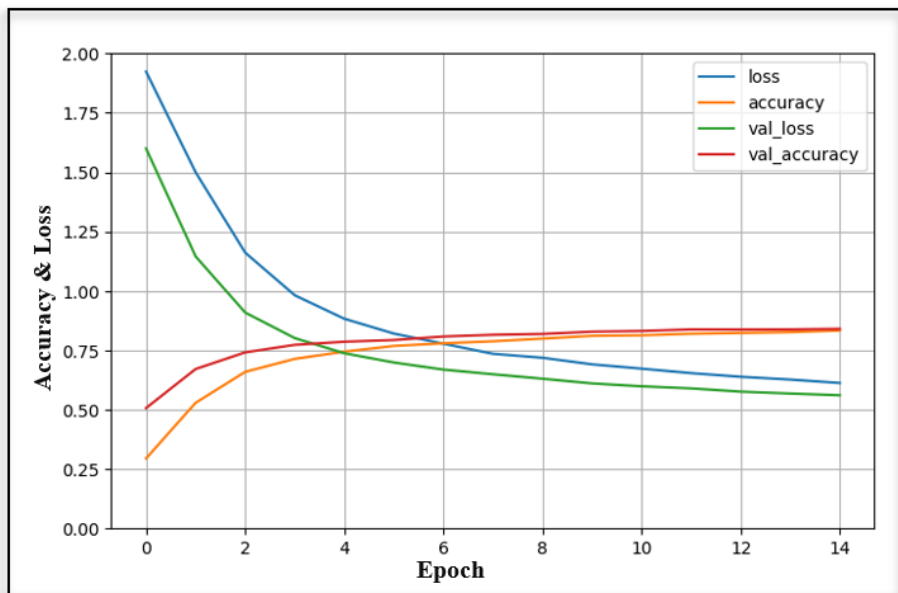
Gambar 4.42. Grafik akurasi dan *loss* model BiLSTM+FastText dengan $lr=0.0001$ dan *batch size* 16



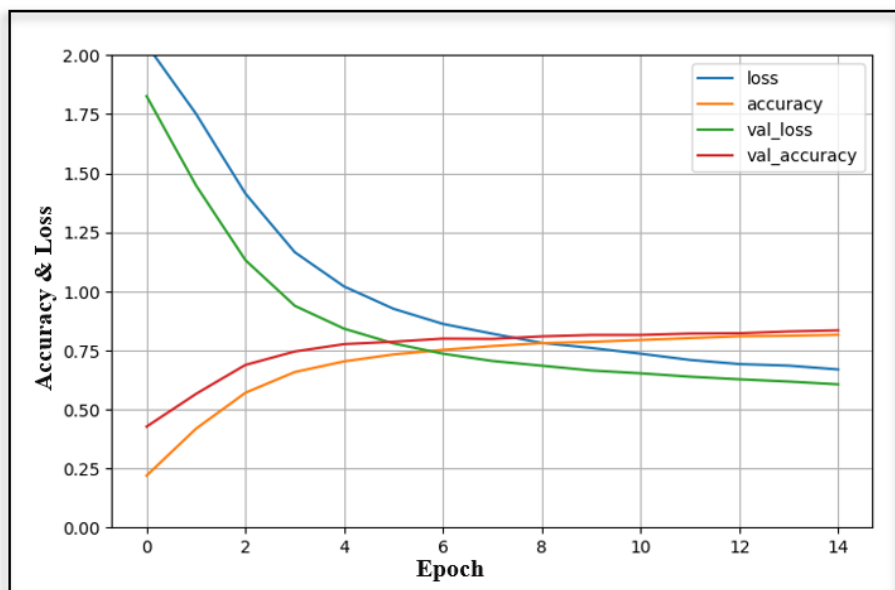
Gambar 4.43. Grafik akurasi dan *loss* model BiLSTM+FastText dengan $lr=0.0001$ dan *batch size* 32

Berdasarkan grafik kurva pembelajaran yang ditunjukkan melalui Gambar 4.42 dan 4.43, model dengan *batch size* 16 dan 32 menunjukkan kurva akurasi data *training* yang meningkat secara cepat di awal *epoch* dan kemudian pada beberapa *epoch* selanjutnya kurva akurasi terus meningkat sampai akhir *epoch*, sedangkan kurva

akurasi pada data *validation* menunjukkan kurva yang meningkat secara perlahan yang tidak terlalu signifikan pada awal *epoch* yang kemudian kurva mencapai titik stabil tetapi untuk model dengan *batch size* 16 akurasi, data *validation* mulai menurun pada akhir *epoch* sehingga terdapat sedikit kesenjangan antara kurva akurasi data *training* dan data *validation*. Sementara itu, kurva *loss* pada data *training* untuk kedua model menurun dengan cepat pada awal *epoch*, penurunan ini terjadi karena model belajar secara cepat untuk menyesuaikan dengan data *training* dan kemudian kurva *loss* pada data *training* menurun secara terus menerus sampai akhir *epoch*. Berbeda dengan kurva *loss* data *training*, kurva *loss* data *validation* pada kedua model menurun secara perlahan dan pada akhirnya selama beberapa *epoch* kurva mencapai titik stabil. Namun setelah kurva *loss* data *validation* pada model dengan *batch size* 16 mencapai stabilitas, kurva *loss* mengalami fluktuasi yang ditunjukkan melalui kurva yang mulai meningkat lalu menurun dan akhirnya pada menjelang akhir *epoch* kurva *loss* mulai meningkat lagi sehingga terdapat kesenjangan yang cukup signifikan antara kurva *loss* data *training* dan *validation* yang dapat dilihat melalui Gambar 4.42. Berdasarkan Analisa terhadap kurva akurasi dan *loss*, pada awalnya kedua kurva pembelajaran tersebut menunjukkan bahwa model mempelajari data *training* dengan baik serta menggeneralisasi data *validation* secara wajar tetapi pada akhirnya untuk model dengan $lr=0.0001$ dan *batch size* 16 mengalami *overfitting* terhadap data *training*. Di samping itu, grafik kurva akurasi dan *loss* untuk model yang menggunakan $lr=0.0001$ dengan *batch size* 16 dan 32 masing-masing ditunjukkan pada Gambar 4.44 dan Gambar 4.45.



Gambar 4.44. Grafik akurasi dan *loss* model BiLSTM+FastText dengan $lr=0.00001$ dan *batch size* 16



Gambar 4.45. Grafik akurasi dan *loss* model BiLSTM+FastText dengan $lr=0.00001$ dan *batch size* 32

Berdasarkan grafik kurva pembelajaran model dengan $lr=0.00001$ yang ditunjukkan melalui Gambar 4.44 dan 4.45 untuk *batch size* 16 dan 32, kurva akurasi data *training* dan *validation* meningkat seiring dengan berjalannya *epoch* hingga mencapai titik stabil dan kemudian kedua kurva pada akhirnya mencapai konvergensi. Hal ini menunjukkan bahwa model belajar secara efektif dari data *training* dan melakukan generalisasi pada data *validation* dengan baik sehingga grafik pada Gambar 4.44 dan Gambar 4.45 menunjukkan model yang performa baik dan stabil.

Sebenarnya, akurasi pada data *training* yang diharapkan lebih tinggi daripada akurasi data *validation* tetapi berlawanan dengan hasil yang didapatkan pada penelitian ini ini. Namun, hal tersebut tidak menjadi masalah karena akurasi yang tinggi pada data *validation* menunjukkan bahwa *regularization* yang digunakan pada saat konfigurasi model efektif untuk mengatas *overfitting*.

Model dengan $lr=0.00001$ berkinerja dengan baik dan stabil juga ditunjukkan melalui kurva *loss*, kedua kurva *loss* data *training* dan *validation* sama-sama menurun sampai titik stabil saat menuju akhir *epoch* dengan kesenjangan yang semakin mengecil dengan seiring berjalannya *epoch*. Hal ini menunjukkan model belajar secara efektif dan menunjukkan generalisasi yang baik. Kinerja model dengan $lr=0.00001$ dan *batch size* 16 dan 32 dapat ditingkatkan dan memiliki kemungkinan untuk mencapai kinerja yang lebih maksimal apabila terdapat penambahan pada masa pelatihan atau *epoch*. Perbandingan kinerja untuk keenam percobaan model BiLSTM+IndoBERT pada *epoch* terbaik berdasarkan pada nilai *loss* yang minimal selama pelatihan model ditampilkan pada Tabel 4.4. Selanjutnya, keenam model pada Tabel 4.4 dievaluasi menggunakan *data testing*.

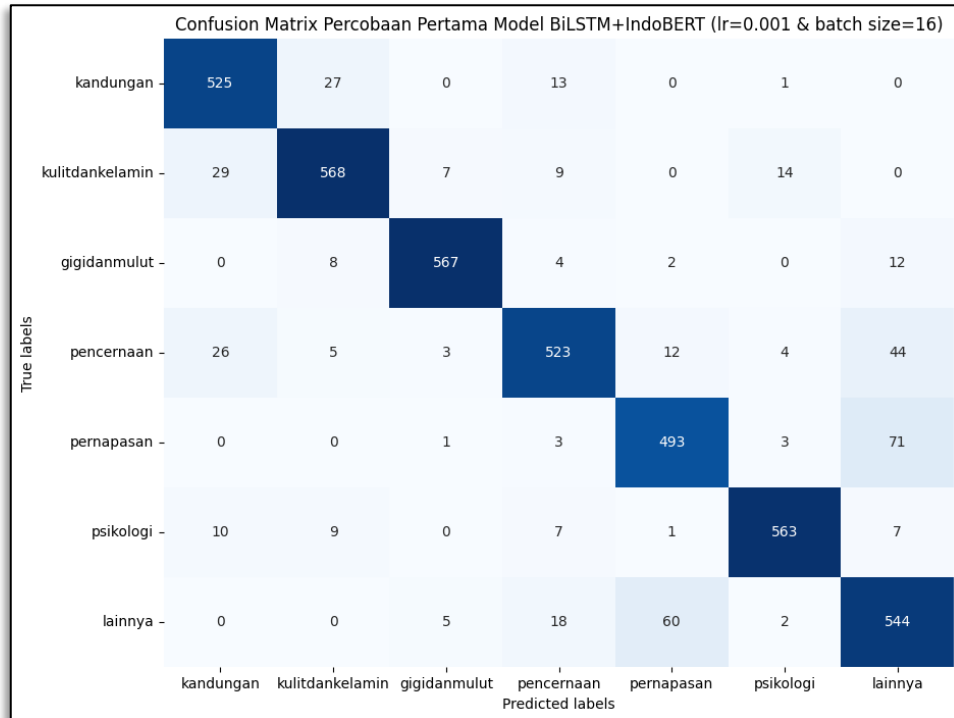
Tabel 4.4. Kinerja pelatihan model BiLSTM+IndoBERT untuk setiap percobaan

Model	Optimizer	Hidden Unit LSTM	Batch Size	Learning Rate	Training Accuracy	Training Loss	Validation Accuracy	Validation Loss	Epochs
Percobaan 1	Adam	150	16	0.001	0.8978	0.3788	0.8964	0.3572	7
Percobaan 2	Adam	150	32	0.001	0.9005	0.3591	0.8982	0.3421	6
Percobaan 3	Adam	150	16	0.0001	0.9231	0.2866	0.8970	0.3699	14
Percobaan 4	Adam	150	32	0.0001	0.9105	0.3387	0.8923	0.3914	14
Percobaan 5	Adam	150	16	0.00001	0.8334	0.6126	0.8405	0.5610	15
Percobaan 6	Adam	150	32	0.00001	0.8158	0.6691	0.8345	0.6059	15

4.11 Pengujian Model BiLSTM+IndoBERT

Setelah model klasifikasi BiLSTM+IndoBERT dilatih menggunakan enam set percobaan dengan skenario yang berbeda, model tersebut perlu diuji terhadap data *testing* untuk mengukur kinerja model dan menentukan skenario terbaik yang menghasilkan model berkinerja tinggi di antara enam set percobaan model. Distribusi data pengujian yang digunakan untuk pengujian model sudah ditentukan sebelumnya pada subbab 4.5 dan sudah dirincikan untuk tiap kategorinya pada subbab 4.8 ketika menguji model BiLSTM+FastText.

Model yang telah diuji selanjutnya dievaluasi menggunakan empat metrik kinerja, yaitu *precision*, *recall*, *F1 score*, dan akurasi. Keempat metrik tersebut ditentukan berdasarkan diagram *confusion matrix* untuk setiap percobaan. Diagram *confusion matrix* dan metrik kinerja model BiLSTM+IndoBERT dengan $lr=0.001$ serta batch size dengan ukuran 16 dan 32 beturut-turut ditunjukkan pada Gambar 4.46, Gambar 4.47, Gambar 4.48, dan Gambar 4.49.



Gambar 4.46. *Confusion matrix* model BiLSTM+IndoBERT percobaan pertama

	precision	recall	f1-score	support
kandungan	0.8898	0.9276	0.9083	566
kulitdankelamin	0.9206	0.9059	0.9132	627
gigidanmulut	0.9726	0.9562	0.9643	593
pencernaan	0.9064	0.8476	0.8760	617
pernapasan	0.8680	0.8634	0.8657	571
psikologi	0.9591	0.9430	0.9510	597
lainnya	0.8024	0.8649	0.8324	629
accuracy			0.9007	4200
macro avg	0.9027	0.9012	0.9016	4200
weighted avg	0.9023	0.9007	0.9011	4200

Gambar 4.47. Metrik kinerja model BiLSTM+IndoBERT percobaan pertama

Confusion Matrix Percobaan Ketiga Model BiLSTM+IndoBERT (lr=0.0001 & batch size=16)

kandungan	519	32	0	14	0	1	0
kulitdankelamin	34	557	6	9	1	19	1
gigidanmulut	0	7	568	3	3	4	8
pencernaan	25	8	8	519	9	9	39
pernapasan	0	0	5	4	504	4	54
psikologi	3	9	1	8	1	571	4
lainnya	0	0	10	24	65	7	523
	kandungan	kulitdankelamin	gigidanmulut	pencernaan	pernapasan	psikologi	lainnya

True labels

Predicted labels

Gambar 4.48. Confusion matrix model BiLSTM+IndoBERT percobaan kedua

	precision	recall	f1-score	support
kandungan	0.8974	0.9276	0.9123	566
kulitdankelamin	0.9011	0.9155	0.9082	627
gigidanmulut	0.9744	0.9629	0.9686	593
pencernaan	0.9022	0.8671	0.8843	617
pernapasan	0.8555	0.9229	0.8880	571
psikologi	0.9509	0.9414	0.9461	597
lainnya	0.8682	0.8172	0.8419	629
accuracy			0.9067	4200
macro avg	0.9071	0.9078	0.9071	4200
weighted avg	0.9071	0.9067	0.9065	4200

Gambar 4.49. Metrik kinerja model BiLSTM+IndoBERT percobaan kedua

Gambar 4.47 dan Gambar 4.49 menunjukkan metrik kinerja yang dihasilkan oleh model BiLSTM+IndoBERT dengan pengaturan lr=0.001 dengan *batch size* 16 dan 32. Sama seperti kinerja yang didapatkan oleh model BiLSTM+FastText, model dengan *batch size* 32 menunjukkan hasil kinerja yang lebih baik dibandingkan ketika model dilatih menggunakan *batch size* 16 meskipun tidak menunjukkan perbedaan yang begitu besar. Nilai akurasi yang diperoleh model menggunakan *batch size* 16 dan 32 masing-masing sebesar 0.9007 dan 0.9067, sedangkan nilai *F1 score* yang didapatkan

oleh model dengan *batch size* 16 dan 32 adalah 0.9016 dan 0.9071. Jika metrik kinerja model ditinjau berdasarkan kategorinya, kategori gigi dan mulut memiliki nilai *precision*, *recall*, dan *F1 score* tertinggi dibandingkan tujuh kategori lainnya pada kedua model percobaan. Nilai *precision*, *recall*, dan *F1 score* untuk model dengan *batch size* 16 masing-masing sebesar 0.9726, 0.9562, dan 0.96643 yang ditunjukkan pada Gambar 4.47, sedangkan nilai *precision*, *recall*, dan *F1 score* untuk model dengan *batch size* 32 adalah 0.9744, 0.9629, dan 0.9686 seperti yang ditunjukkan pada Gambar 4.49. Sementara itu, kategori yang memiliki nilai *precision* dan *F1 score* terendah untuk model dengan *batch size* 16 adalah kategori lainnya yang masing-masing nilainya sebesar 0.8024 dan 0.8324, sedangkan nilai *recall* terendah dimiliki oleh kategori pencernaan dengan nilai 0.8476. Sama seperti model dengan *batch size* 16, kategori lainnya memiliki nilai *recall* dan *F1 score* terendah untuk model dengan *batch size* 32 dengan masing-masing nilainya 0.8172 dan 0.8419, sedangkan nilai *precision* terendah dimiliki oleh kategori pernapasan dengan nilai 0.8555.

Berdasarkan diagram *confusion matrix* yang ditunjukkan pada Gambar 4.46 dan Gambar 4.48, model dengan *batch size* 16 berhasil mengklasifikasikan 3.783 pertanyaan dari 4.200 data *testing* dengan benar, sedangkan model dengan *batch size* 32 berhasil mengklasifikasikan secara benar 3.808 pertanyaan dari kumpulan 4.200 data *testing*. Berdasarkan *confusion matrix* pada Gambar 4.46, kategori yang paling diprediksi secara akurat oleh model dengan *batch size* 16 adalah kategori gigi dan mulut yang menghasilkan lebih sedikit *False Positive* (FP), yaitu dari 583 pertanyaan yang diprediksi ke dalam kategori gigi dan mulut hanya 16 pertanyaan yang salah diprediksi oleh model, sedangkan sisanya diprediksi dengan benar. Sebaliknya, kategori yang paling banyak salah diprediksi oleh model sehingga menghasilkan nilai FP yang tinggi adalah kategori lainnya, sebanyak 134 pertanyaan yang salah diprediksi oleh model dari total 678 pertanyaan yang diprediksi ke dalam kategori lainnya. Sementara itu, kategori yang paling banyak dikenali atau diidentifikasi oleh model adalah kategori gigi dan mulut yang ditunjukkan melalui 568 pertanyaan berhasil diklasifikasi dengan benar oleh model dari 593 data *testing* yang berada dalam kategori gigi dan mulut. Sementara itu, kategori yang paling sulit diklasifikasi dengan benar oleh model adalah kategori pencernaan yang ditunjukkan melalui model yang hanya

berhasil mengidentifikasi 523 pertanyaan dari 617 data *testing* yang berada dalam kategori pencernaan.

Di samping itu, berdasarkan diagram *confusion matrix* yang ditunjukkan pada Gambar 4.26, kategori yang paling banyak diprediksi secara akurat oleh model dengan *batch size* 32 adalah kategori gigi dan mulut, yaitu dari 586 pertanyaan yang diprediksi oleh model ke dalam kategori gigi dan mulut hanya 15 pertanyaan di antaranya yang salah diprediksi oleh model, sedangkan sisanya diprediksi dengan benar. Sebaliknya, kategori yang paling banyak salah diprediksi oleh model adalah kategori pernapasan, yaitu sebanyak 89 pertanyaan salah diprediksi oleh model dari total 616 pertanyaan yang diprediksi ke dalam kategori pernapasan. Sementara itu, kategori yang paling banyak dikenali atau diidentifikasi oleh model adalah kategori gigi dan mulut yang ditunjukkan melalui 574 pertanyaan berhasil diklasifikasi dengan benar oleh model dari 593 data dalam kategori gigi dan mulut. Sebaliknya, kategori yang paling sulit diklasifikasi dengan benar oleh model dengan *batch size* 32 adalah kategori lainnya yang ditunjukkan melalui model yang hanya berhasil mengidentifikasi 514 pertanyaan dari 629 data *testing* yang berada dalam kategori pencernaan. Selanjutnya, hasil evaluasi kinerja model dengan $lr=0.0001$ dengan *batch size* 16 dan 32 ditampilkan melalui diagram *confusion matrix* dan tabel metrik kinerja yang ditunjukkan melalui Gambar 4.50, Gambar 4.51, Gambar 4.52, dan Gambar 4.53.

Confusion Matrix Percobaan Ketiga Model BiLSTM+IndoBERT (lr=0.0001 & batch size=16)

kandungan	519	32	0	14	0	1	0
kulitdankelamin	34	557	6	9	1	19	1
gigidanmulut	0	7	568	3	3	4	8
pencernaan	25	8	8	519	9	9	39
pernapasan	0	0	5	4	504	4	54
psikologi	3	9	1	8	1	571	4
lainnya	0	0	10	24	65	7	523
	kandungan	kulitdankelamin	gigidanmulut	pencernaan	pernapasan	psikologi	lainnya

True labels

Predicted labels

Gambar 4.50. Confusion matrix model BiLSTM+IndoBERT percobaan ketiga

	precision	recall	f1-score	support
kandungan	0.8933	0.9170	0.9050	566
kulitdankelamin	0.9086	0.8884	0.8984	627
gigidanmulut	0.9498	0.9578	0.9538	593
pencernaan	0.8933	0.8412	0.8664	617
pernapasan	0.8645	0.8827	0.8735	571
psikologi	0.9285	0.9564	0.9422	597
lainnya	0.8315	0.8315	0.8315	629
accuracy			0.8955	4200
macro avg	0.8956	0.8964	0.8958	4200
weighted avg	0.8954	0.8955	0.8952	4200

Gambar 4.51. Metrik kinerja model BiLSTM+IndoBERT percobaan ketiga

Confusion Matrix Percobaan Keempat Model BiLSTM+IndoBERT (lr=0.0001 & batch size=32)

kandungan	525	25	0	15	0	1	0
kulitdankelamin	39	558	6	8	0	16	0
gigidanmulut	0	10	569	2	4	0	8
pencernaan	25	12	6	524	8	9	33
pernapasan	1	0	4	5	495	2	64
psikologi	3	10	1	9	0	570	4
lainnya	1	1	7	28	66	6	520
	kandungan	kulitdankelamin	gigidanmulut	pencernaan	pernapasan	psikologi	lainnya

True labels

Predicted labels

Gambar 4.52. Confusion matrix model BiLSTM+IndoBERT percobaan keempat

	precision	recall	f1-score	support
kandungan	0.8838	0.9276	0.9052	566
kulitdankelamin	0.9058	0.8900	0.8978	627
gigidanmulut	0.9595	0.9595	0.9595	593
pencernaan	0.8866	0.8493	0.8675	617
pernapasan	0.8639	0.8669	0.8654	571
psikologi	0.9437	0.9548	0.9492	597
lainnya	0.8267	0.8267	0.8267	629
accuracy			0.8955	4200
macro avg	0.8957	0.8964	0.8959	4200
weighted avg	0.8955	0.8955	0.8953	4200

Gambar 4.53. Metrik kinerja model BiLSTM+IndoBERT percobaan keempat

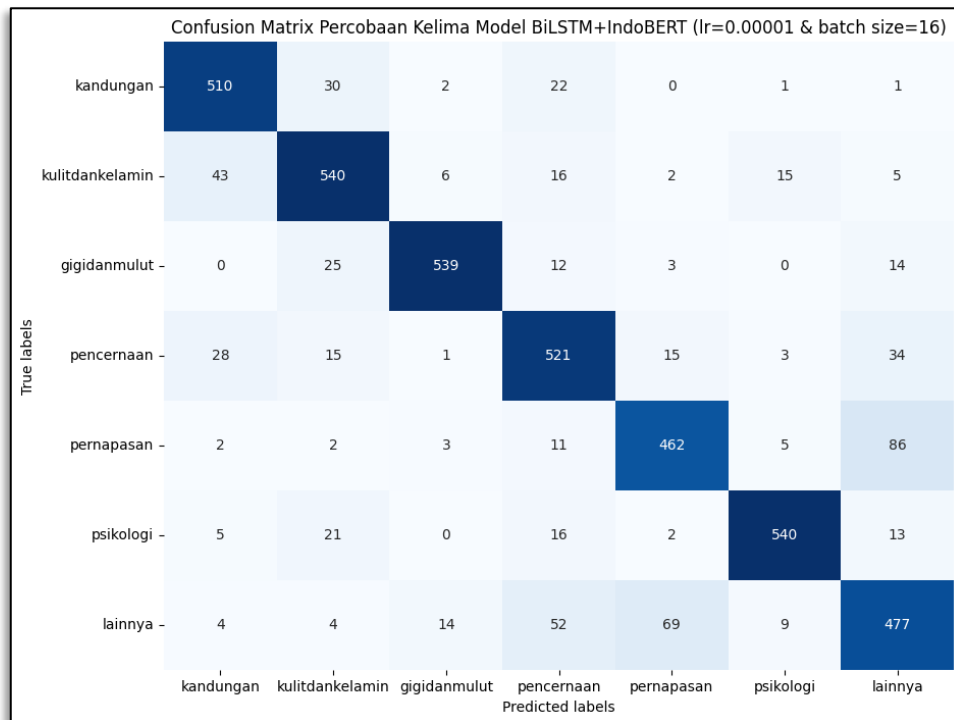
Gambar 4.51 dan Gambar 4.53 menunjukkan metrik kinerja yang dihasilkan oleh model BiLSTM+IndoBERT dengan pengaturan lr=0.0001 dengan *batch size* 16 dan 32. Model dengan *batch size* 16 menunjukkan hasil kinerja yang sama dengan model menggunakan *batch size* 32. Nilai akurasi yang diperoleh model menggunakan *batch size* 16 dan 32 adalah 0.8955, sedangkan nilai *F1 score* yang didapatkan oleh model dengan *batch size* 16 dan 32 adalah 0.8958 dan 0.8959. Jika metrik kinerja model ditinjau berdasarkan kategorinya, kategori gigi dan mulut memiliki nilai *precision*,

recall, dan *F1 score* tertinggi dibandingkan tujuh kategori lainnya pada untuk kedua model percobaan. Nilai *precision*, *recall*, dan *F1 score* untuk model dengan *batch size* 16 masing-masing sebesar 0.9498, 0.9578, dan 0.9538 yang ditunjukkan pada Gambar 4.51, sedangkan nilai *precision*, *recall*, dan *F1 score* untuk model dengan *batch size* 32 adalah 0.9595, 0.9595, dan 0.9595 seperti yang ditunjukkan pada Gambar 4.53. Berdasarkan metrik kinerja kedua model percobaan, kategori dengan nilai *precision*, *recall*, dan *F1 score* terendah untuk kedua model adalah kategori lainnya dengan nilai masing-masing sebesar 0.8315, 0.8315, dan 0.8315 untuk *batch size* 16 dan 0.8267, 0.8267, dan 0.8627 untuk *batch size* 32.

Berdasarkan diagram *confusion matrix* yang ditunjukkan pada Gambar 4.50 dan Gambar 4.52, kedua model dengan *batch size* 16 dan 32 berhasil mengklasifikasikan 3.761 pertanyaan dari 4.200 data *testing* dengan benar. Berdasarkan *confusion matrix* pada Gambar 4.50, kategori yang paling banyak diprediksi secara akurat oleh model dengan *batch size* 16 adalah kategori gigi dan mulut yang menghasilkan lebih sedikit *False Positive* (FP), yaitu dari 598 pertanyaan yang diprediksi ke dalam kategori gigi dan mulut hanya 30 pertanyaan yang salah diprediksi oleh model, sedangkan sisanya diprediksi dengan benar. Sebaliknya, kategori yang paling banyak salah diprediksi oleh model sehingga menghasilkan nilai FP yang tinggi adalah kategori lainnya, sebanyak 106 pertanyaan salah diprediksi oleh model dari total 629 pertanyaan yang diprediksi ke dalam kategori lainnya. Kategori yang paling baik dikenal atau diidentifikasi oleh model adalah kategori gigi dan mulut yang ditunjukkan melalui 568 pertanyaan berhasil diklasifikasi dengan benar oleh model dari 593 data *testing* yang berada dalam kategori gigi dan mulut. Sementara itu, kategori yang paling sulit diklasifikasi dengan benar oleh model adalah kategori lainnya yang ditunjukkan melalui model yang hanya berhasil mengidentifikasi 523 pertanyaan dari 629 data *testing* yang berada dalam kategori pernapasan.

Di samping itu, berdasarkan diagram *confusion matrix* yang ditunjukkan pada Gambar 4.52, kategori yang paling banyak diprediksi secara akurat oleh model dengan *batch size* 32 adalah kategori gigi dan mulut, yaitu dari 569 pertanyaan yang diprediksi oleh model ke dalam kategori gigi dan mulut hanya 24 pertanyaan yang salah diprediksi oleh model, sedangkan sisanya diprediksi dengan benar oleh model. Sebaliknya, kategori yang paling banyak salah diprediksi oleh model adalah kategori

lainnya, yaitu sebanyak 109 pertanyaan salah diprediksi oleh model dari total 629 pertanyaan yang diprediksi ke dalam kategori lainnya. Sementara itu, kategori yang paling banyak diidentifikasi oleh model adalah kategori gigi dan mulut yang ditunjukkan melalui 569 pertanyaan berhasil diklasifikasi dengan benar oleh model dari 593 data dalam kategori gigi dan mulut. Sebaliknya, kategori yang paling sulit diklasifikasi dengan benar oleh model dengan *batch size* 32 adalah kategori lainnya yang ditunjukkan melalui model yang hanya berhasil mengidentifikasi 520 pertanyaan dari 629 data *testing* yang berada dalam kategori pernapasan. Selanjutnya, hasil evaluasi kinerja model dengan $lr=0.00001$ dengan *batch size* 16 dan 32 ditampilkan melalui diagram *confusion matrix* dan tabel metrik kinerja yang ditunjukkan melalui Gambar 4.54, Gambar 4.55, Gambar 4.56, dan Gambar 4.57.



Gambar 4.54. *Confusion matrix* model BiLSTM+IndoBERT percobaan kelima

	precision	recall	f1-score	support
kandungan	0.8615	0.9011	0.8808	566
kulitdankelamin	0.8477	0.8612	0.8544	627
gigidanmulut	0.9540	0.9089	0.9309	593
pencernaan	0.8015	0.8444	0.8224	617
pernapasan	0.8354	0.8091	0.8221	571
psikologi	0.9424	0.9045	0.9231	597
lainnya	0.7571	0.7583	0.7577	629
accuracy			0.8545	4200
macro avg	0.8571	0.8554	0.8559	4200
weighted avg	0.8560	0.8545	0.8550	4200

Gambar 4.55. Metrik kinerja model BiLSTM+IndoBERT percobaan kelima

Confusion Matrix Percobaan Keenam Model BiLSTM+IndoBERT (lr=0.00001 & batch size=32)

kandungan -	512	25	2	22	0	3	2
kulitdankelamin -	41	531	10	15	2	25	3
gigidanmulut -	0	17	542	13	5	1	15
pencernaan -	26	12	5	508	20	8	38
pernapasan -	2	2	4	10	461	5	87
psikologi -	6	16	2	16	5	543	9
lainnya -	5	9	22	44	95	9	445
	kandungan	kulitdankelamin	gigidanmulut	pencernaan	pernapasan	psikologi	lainnya

True labels

Predicted labels

Gambar 4.56. Confusion matrix model BiLSTM+IndoBERT percobaan keenam

	precision	recall	f1-score	support
kandungan	0.8649	0.9046	0.8843	566
kulitdankelamin	0.8676	0.8469	0.8571	627
gigidanmulut	0.9233	0.9140	0.9186	593
pencernaan	0.8089	0.8233	0.8161	617
pernapasan	0.7840	0.8074	0.7955	571
psikologi	0.9141	0.9095	0.9118	597
lainnya	0.7429	0.7075	0.7248	629
accuracy			0.8433	4200
macro avg	0.8437	0.8447	0.8440	4200
weighted avg	0.8431	0.8433	0.8430	4200

Gambar 4.57. Metrik kinerja model BiLSTM+IndoBERT percobaan keenam

Gambar 4.55 dan Gambar 4.57 menunjukkan metrik kinerja yang dihasilkan oleh model BiLSTM+IndoBERT dengan pengaturan *hyperparameter* $lr=0.00001$ dengan *batch size* 16 dan 32. Secara umum, model dengan *batch size* 16 menunjukkan hasil kinerja yang lebih baik dibandingkan ketika model menggunakan *batch size* 32. Nilai akurasi yang diperoleh model menggunakan *batch size* 16 dan 32 masing-masing sebesar 0.8545 dan 0.8433, sedangkan nilai *F1 score* yang didapatkan oleh model dengan *batch size* 16 dan 32 adalah 0.8559 dan 0.8440. Jika metrik kinerja model ditinjau berdasarkan kategorinya, nilai *precision*, *recall*, dan *F1 score* tertinggi didapatkan oleh kategori gigi dan mulut untuk kedua model. Nilai *precision*, *recall*, dan *F1 score* masing-masing sebesar 0.9540, 0.9089, dan 0.9309 untuk model dengan *batch size* 16 dan 0.9233, 0.9140, dan 0.9186 untuk model dengan *batch size* 32. Sementara itu, nilai *precision*, *recall*, dan *F1 score* terendah dimiliki oleh kategori lainnya untuk kedua model. Nilai *precision*, *recall*, dan *F1 score* masing-masing sebesar 0.7571, 0.7583, dan 0.7577 untuk model dengan *batch size* 16 yang ditunjukkan melalui Gambar 4.55 dan 0.7429, 0.7075, dan 0.7248 untuk model dengan *batch size* 32 yang ditunjukkan pada Gambar 4.57.

Berdasarkan diagram *confusion matrix* yang ditunjukkan pada Gambar 4.54 dan Gambar 4.56, model dengan *batch size* 16 berhasil mengklasifikasikan 3.589 pertanyaan dari 4.200 data *testing* dengan benar, sedangkan model dengan *batch size* 32 berhasil mengklasifikasikan secara benar 3.562 pertanyaan dari kumpulan 4.200 data *testing*. Berdasarkan *confusion matrix* pada Gambar 4.54, kategori yang paling diprediksi secara akurat oleh model dengan *batch size* 16 adalah kategori gigi dan

mulut yang menghasilkan lebih sedikit *False Positive* (FP), yaitu dari 565 pertanyaan yang diprediksi ke dalam kategori gigi dan mulut hanya 26 pertanyaan yang salah diprediksi oleh model, sedangkan sisanya diprediksi dengan benar. Sebaliknya, kategori yang paling banyak salah diprediksi oleh model sehingga menghasilkan nilai FP yang tinggi adalah kategori lainnya, sebanyak 153 pertanyaan salah diprediksi oleh model dari total 630 pertanyaan yang diprediksi ke dalam kategori lainnya. Sementara itu, kategori yang paling banyak dikenali atau diidentifikasi oleh model adalah kategori gigi dan mulut yang ditunjukkan melalui 539 pertanyaan berhasil diklasifikasi dengan benar oleh model dari 593 data *testing* yang berada dalam kategori gigi dan mulut. Sementara itu, kategori yang paling sulit diklasifikasi dengan benar oleh model adalah kategori lainnya yang ditunjukkan melalui model yang hanya berhasil mengidentifikasi 477 pertanyaan dari 629 data *testing* yang berada dalam kategori lainnya.

Di samping itu, berdasarkan diagram *confusion matrix* yang ditunjukkan pada Gambar 4.56, kategori yang paling banyak diprediksi secara akurat oleh model dengan *batch size* 32 adalah kategori gigi dan mulut, yaitu dari 587 pertanyaan yang diprediksi oleh model ke dalam kategori gigi dan mulut hanya 45 pertanyaan di antaranya salah diprediksi oleh model, sedangkan sisanya diprediksi dengan benar. Sebaliknya, kategori yang paling banyak salah diprediksi oleh model adalah kategori lainnya, yaitu sebanyak 154 pertanyaan salah diprediksi oleh model dari total 599 pertanyaan yang diprediksi ke dalam kategori lainnya. Sementara itu, kategori yang dikenali paling banyak oleh model adalah kategori gigi dan mulut yang ditunjukkan melalui 542 pertanyaan berhasil diklasifikasi dengan benar oleh model dari 593 data dalam kategori gigi dan mulut. Sebaliknya, kategori yang paling sulit diklasifikasi dengan benar oleh model dengan *batch size* 32 adalah kategori lainnya yang ditunjukkan melalui model yang hanya berhasil mengidentifikasi 445 pertanyaan dari 629 data *testing* yang berada dalam kategori lainnya.

Tabel 4.4 menampilkan perbandingan keseluruhan metrik kinerja hasil percobaan model BiLSTM menggunakan *contextual embedding* IndoBERT sebagai vektor *input* dengan mempertimbangkan tiga skenario pengaturan *hyperparameter learning rate* (0.001, 0.0001, dan 0.00001) dan dua skenario nilai *batch size* (16 dan

32). Pada semua percobaan, model dioptimalkan oleh *optimizer* Adam dan jumlah *hidden unit* pada LSTM yang digunakan adalah 150 (300 *hidden units* BiLSTM).

Tabel 4.4. Perbandingan metrik kinerja hasil percobaan model BiLSTM+IndoBERT

Model	Optimizer	Hidden Unit LSTM	Batch Size	Learning Rate	Accuracy	F1 Score	Precision	Recall	Epochs
Percobaan 1	Adam	150	16	0.001	0.9007	0.9016	0.9027	0.9012	7
Percobaan 2	Adam	150	32	0.001	0.9067	0.9071	0.9071	0.9078	6
Percobaan 3	Adam	150	16	0.0001	0.8955	0.8958	0.8956	0.8964	14
Percobaan 4	Adam	150	32	0.0001	0.8955	0.8959	0.8957	0.8964	14
Percobaan 5	Adam	150	16	0.00001	0.8545	0.8559	0.8571	0.8554	15
Percobaan 6	Adam	150	32	0.00001	0.8433	0.8440	0.8437	0.8447	15

Berdasarkan hasil kinerja model BiLSTM+IndoBERT pada keenam percobaan, percobaan 2 mencapai kinerja tertinggi di semua metrik yang masing-masing bernilai 0.9067, 0.9071, 0.9071, dan 0.9078 untuk akurasi, *F1 score*, *precision*, dan *recall*. Hal ini menunjukkan penggunaan *learning rate* 0.001 dan *batch size* 32 adalah konfigurasi model yang paling efektif untuk tugas klasifikasi pertanyaan konsultasi kesehatan baik menggunakan *word embedding* fastText maupun IndoBERT. Sebaliknya, model percobaan 5 dan 6 menghasilkan kinerja terendah di seluruh metrik dibandingkan dengan percobaan lainnya. Akan tetapi, kinerja yang didapatkan oleh model percobaan 5 dan 6 lebih baik dibandingkan dengan keenam percobaan model BiLSTM+FastText. Berdasarkan Tabel 4.4 yang menampilkan perbandingan metrik kinerja untuk keenam percobaan menunjukkan bahwa penetapan nilai *learning rate* atau *step size* pada *optimizer* Adam memiliki pengaruh terhadap kinerja yang dihasilkan oleh model yang ditunjukkan melalui penggunaan *learning rate* yang lebih tinggi, yaitu 0.001 menghasilkan kinerja yang lebih baik dibandingkan dengan *learning rate* yang lebih rendah, yaitu 0.0001 dan 0.00001. Begitu pula dengan nilai *batch size*, hasil penelitian menunjukkan hubungan antara nilai *learning rate* dan *batch size* yang saling mempengaruhi terhadap kinerja model. Pada saat model menggunakan *learning rate* yang lebih tinggi (0.001), *batch size* yang lebih besar (32) menghasilkan kinerja yang sedikit lebih optimal dibandingkan dengan *batch size* 16. Namun, saat model menggunakan *learning rate* yang lebih rendah, yaitu 0.00001, *batch size* yang lebih kecil (16) menunjukkan kinerja yang lebih baik dibandingkan dengan *batch size* 32.

Setelah mengamati diagram *confusion matrix* dan membandingkannya dengan hasil metrik kinerja yang didapatkan dari keenam percobaan, sama seperti metrik

kinerja yang dihasilkan oleh model BiLSTM+FastText yang menunjukkan bahwa kategori gigi dan mulut adalah kategori yang memiliki nilai *F1 score* tertinggi yang didapatkan oleh model di seluruh percobaan. Model menghasilkan kinerja yang tinggi baik pada metrik *precision* maupun *recall* yang ditunjukkan melalui model yang dapat mengidentifikasi dengan benar sebagian besar pertanyaan yang terdapat dalam kategori gigi dan mulut serta model menghasilkan kesalahan prediksi yang sedikit. Sebaliknya, kategori yang memiliki nilai *F1 score* terendah untuk seluruh model percobaan adalah kategori lainnya yang berarti model tidak menghasilkan kinerja yang baik pada *precision* dan *recall*. Artinya, model gagal dalam mengidentifikasi beberapa pertanyaan yang terdapat dalam kategori lainnya. Walaupun demikian, secara umum model klasifikasi BiLSTM+IndoBERT cukup baik dalam menangkap konteks kumpulan pertanyaan untuk tiap-tiap kategori.

4.12 Perbandingan Kinerja Model Klasifikasi

Sebelumnya, penelitian terkait klasifikasi teks pertanyaan pengguna konsultasi kesehatan sudah pernah dilakukan oleh Hakim et al. (2017) yang mengembangkan skema klasifikasi pertanyaan berdasarkan spesialisasi kedokteran untuk pertanyaan pengguna non-profesional pada konsultasi kesehatan *online*. Penelitian tersebut berbasis pendekatan *machine learning* yang menggunakan algoritma SVM dengan menerapkan dua ekstraksi fitur, yaitu 100 kata teratas di setiap kategori dan *controlled vocabulary*. Model SVM dari penelitian Hakim et al. (2017) digunakan sebagai *baseline* untuk mengevaluasi kinerja model yang diusulkan dalam penelitian ini, yaitu model BiLSTM+FastText dan BiLSTM+IndoBERT. Tabel 4.5 menyajikan perbandingan metrik kinerja model yang diusulkan dengan model SVM berbasis *machine learning* sebagai *baseline*. Tabel 4.5 menunjukkan kedua model klasifikasi BiLSTM yang diusulkan pada penelitian ini memiliki kinerja yang lebih tinggi dibandingkan *baseline* yang menggunakan model SVM sebagai *classifier*. Hal ini menunjukkan bahwa model BiLSTM berbasis *deep learning* memberikan hasil yang lebih baik dibandingkan hasil metrik kinerja model SVM yang berbasis *machine learning* karena model klasifikasi yang diusulkan pada penelitian ini menggunakan vektor kata terlatih yang berdasarkan pendekatan fastText dan IndoBERT untuk meningkatkan pembelajaran representasi kata pada teks pertanyaan. Jadi, penggunaan

IndoBERT dan *word embedding* fastText sebagai vektor *input* untuk model BiLSTM dapat digunakan sebagai alternatif untuk klasifikasi teks dalam domain kesehatan.

Tabel 4.5. Perbandingan metrik kinerja *baseline* dengan model BiLSTM menggunakan fastText dan IndoBERT

Baseline: Precision (73.0%), Recall (72.7%), F1 Score (72.1%)				
Model	Precision (%)	Recall (%)	F1 Score (%)	Accuracy (%)
BiLSTM+FastText	84.69 (+11.69)	84.58 (+11.85)	84.57 (+12.47)	84.45
BiLSTM+IndoBERT	90.71 (+17.71)	90.78 (+18.08)	90.71 (+18.61)	90.67

Tabel 4.5 juga menampilkan perbandingan metrik kinerja yang dihasilkan oleh model BiLSTM menggunakan pendekatan IndoBERT dan *word embedding* fastText. Model BiLSTM yang dilatih menggunakan IndoBERT (BiLSTM+IndoBERT) sebagai vektor *input* menunjukkan kinerja yang lebih tinggi dibandingkan kinerja yang dihasilkan oleh model BiLSTM menggunakan *word embedding* fastText (BiLSTM+FastText). Model BiLSTM+IndoBERT memperoleh nilai akurasi dan *F1 score* masing-masing sebesar, 90.67% dan 90.71%. Hal ini menunjukkan bahwa model klasifikasi berbasis IndoBERT lebih baik mengklasifikasikan teks pertanyaan kesehatan ke dalam kategori yang sesuai secara tepat. Pada dasarnya, ini sejalan dengan IndoBERT sebagai model *contextual embedding* yang merepresentasikan suatu kata dengan vektor penyematan kata yang kontekstual. Artinya, model IndoBERT menetapkan vektor yang berbeda dari kata yang sama bergantung pada konteks penggunaan kata tersebut yang memungkinkan model untuk menangkap makna kata yang berbeda dalam konteks yang berbeda sehingga dapat menghasilkan representasi *input* yang baik untuk model klasifikasi. Pemahaman konteks kata yang dimiliki oleh IndoBERT ini dapat meningkatkan kemampuan pembelajaran model klasifikasi BiLSTM untuk mengklasifikasi secara akurat teks pertanyaan dalam domain kesehatan yang benar-benar memerlukan pemahaman konteks yang kuat. Ditambah lagi, IndoBERT telah dilatih sebelumnya pada kumpulan data teks bahasa Indonesia berskala besar yang membuat IndoBERT memiliki pemahaman mendalam terkait sintaktik dan semantik bahasa Indonesia serta pengetahuan terkait domain kesehatan dari data *training* saat pelatihan model klasifikasi sehingga membantu model klasifikasi untuk membedakan kategori atau subtopik pertanyaan dalam domain kesehatan.

Sementara itu, pendekatan *word embedding* fastText adalah penyematan kata statis yang menetapkan vektor penyematan tunggal untuk satu kata walaupun kata tersebut berada dalam konteks kalimat yang berbeda. Ini merupakan keterbatasan dari fastText sebagai model *static embedding* yang membuat model klasifikasi mengalami kesulitan dalam memahami dan mengklasifikasi teks pertanyaan yang terdiri dari kata-kata yang ambigu dan memiliki banyak arti berdasarkan konteks. Walaupun demikian, model BiLSTM+FastText sebagian besar mampu mengklasifikasikan teks pertanyaan kesehatan dengan baik yang ditunjukkan melalui perolehan akurasi dan *F1 score* yang cukup baik dengan masing-masing bernilai 84.45% dan 84.57%. Hal ini menunjukkan bahwa *word embedding* fastText yang dilatih menggunakan kumpulan data yang besar dalam domain kesehatan secara umum juga mampu menangkap pemahaman dan hubungan kata dalam konteks kesehatan walaupun tidak dapat beradaptasi secara dinamis terhadap konteks spesifik seperti model IndoBERT. Selanjutnya, Tabel 4.6 menampilkan perbandingan kinerja model BiLSTM+IndoBERT dan BiLSTM+FastText berdasarkan kategori spesialisasi kedokteran.

Tabel 4.6. Perbandingan kinerja model berdasarkan kategori spesialisasi kedokteran

Category	BiLSTM+IndoBERT			BiLSTM+FastText		
	Precision	Recall	F1 Score	Precision	Recall	F1 Score
Kandungan	0.8974	0.9276	0.9123	0.8274	0.9064	0.8651
Kulit dan Kelamin	0.9011	0.9155	0.9082	0.8235	0.8485	0.8358
Gigi dan Mulut	0.9744	0.9629	0.9686	0.9783	0.9140	0.9451
Pencernaan	0.9022	0.8671	0.8843	0.8328	0.7828	0.8070
Pernapasan	0.8555	0.9229	0.8880	0.8111	0.8424	0.8265
Psikologi	0.9509	0.9414	0.9461	0.8778	0.8425	0.8598
Lainnya	0.8682	0.8172	0.8419	0.7776	0.7838	0.7807
Macro Average	0.9071	0.9078	0.9071	0.8469	0.8458	0.8445

Berdasarkan Tabel 4.6 menunjukkan bahwa kategori Gigi dan Mulut memiliki kinerja tertinggi untuk kedua model klasifikasi dengan nilai *precision*, *recall*, dan *F1 score* masing-masing sebesar 97.44%, 96.29%, dan 96.86% untuk model BiLSTM+IndoBERT, sedangkan nilai *precision*, *recall*, dan *F1 score* untuk model BiLSTM+FastText adalah masing-masing sebesar 82.35%, 84.85%, dan 83.58%. Selain itu, kategori Kandungan dan Psikologi juga mendapatkan metrik kinerja yang cukup baik dibandingkan yang lainnya. Hal ini menunjukkan bahwa kumpulan teks pertanyaan pada data *training* untuk kategori Gigi dan Mulut, Kandungan, dan Psikologi merupakan data yang cukup representatif. Artinya, data *training* untuk

ketiga kategori tersebut terdiri dari data yang bervariasi dengan keberagaman topik ataupun jenis pertanyaan yang mencerminkan tiap-tiap kategori tersebut yang dapat membantu model mempelajari fitur dan pola yang relevan dalam kategori tersebut dan meningkatkan kemampuan model untuk melakukan generalisasi sehingga menghasilkan model yang memprediksi dan mengklasifikasi pertanyaan secara akurat untuk kategori tersebut. Sementara itu, kategori Lainnya adalah kategori yang memiliki nilai *precision*, *recall*, dan *F1 score* terendah untuk kedua model. Nilai *precision*, *recall*, dan *F1 score* untuk model BiLSTM+IndoBERT adalah masing-masing sebesar 0.8682, 0.8172, dan 0.8419, sedangkan nilai *precision*, *recall*, dan *F1 score* untuk model BiLSTM+FastText adalah sebesar 0.7776, 0.7838, dan 0.7807. Setelah dilakukan pengamatan lebih lanjut terhadap kategori lainnya, kumpulan pertanyaan yang terdapat di dalam kategori Lainnya merupakan kumpulan pertanyaan dari empat kategori spesialisasi kedokteran, yaitu penyakit Infeksi, Mata, Saraf, Telinga Hidung Tenggorokan (THT), dan Gizi. Terdapat beberapa pertanyaan dari kategori Lainnya yang memiliki hubungan atau keterkaitan dengan spesialisasi medis lainnya, terutama dengan kategori Pencernaan dan Pernapasan sehingga terdapat ketidakkonsistenan pada label sebenarnya untuk pertanyaan dalam kategori Lainnya karena pertanyaan yang bersifat ambigu. Hal ini yang menyebabkan terdapat beberapa pertanyaan yang kategori sebenarnya adalah kategori Lainnya tetapi model memprediksi pertanyaan tersebut ke dalam kategori Pencernaan atau Pernapasan. Begitu juga sebaliknya, pertanyaan dengan label Pencernaan atau Pernapasan diprediksi oleh model ke dalam kategori Lainnya. Contohnya, pada teks pertanyaan berikut ini yang diambil dari kumpulan data kategori Lainnya dan salah diprediksi oleh model ke dalam kategori Pernapasan:

“dok kenapa pada tenggorokan saya kerasa ada bau dan ditenggorokan terasa panas apakah ada obat resep penyembuhan nya dok”.

Kumpulan pertanyaan yang bersifat ambigu ini yang membuat model kesulitan dalam melakukan klasifikasi pertanyaan dalam kategori Lainnya sehingga menghasilkan kinerja model klasifikasi yang lebih rendah untuk kategori ini.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil dan perbandingan metrik kinerja model untuk tugas klasifikasi teks pertanyaan pengguna konsultasi kesehatan *online* berbasis *deep learning* menggunakan *word embedding* FastText dan *contextual embedding* IndoBERT sebagai vektor *input* pada arsitektur BiLSTM, kesimpulan yang didapatkan dari penelitian Tugas Akhir ini adalah sebagai berikut:

1. Klasifikasi teks pertanyaan pengguna konsultasi kesehatan *online* menggunakan pendekatan *contextual embedding* IndoBERT sebagai fitur atau vektor *input* pada model klasifikasi BiLSTM menghasilkan kinerja yang lebih tinggi dibandingkan ketika model menggunakan *word embedding* FastText sebagai vektor *input*. Hal ini didasari oleh akurasi pengujian model BiLSTM+IndoBERT yang mencapai akurasi 0.9067 atau 90.67% dengan metrik *precision*, *recall*, dan *F1 score* yang masing-masing mencapai nilai sebesar 0.9071, 0.9078, dan 0.9071.
2. Berdasarkan percobaan model dengan *learning rate* 0.001, 0.0001, dan 0.00001 serta *batch size* 16 dan 32, didapatkan bahwa *learning rate* dengan nilai 0.001 dan *batch size* 32 merupakan *hyperparameter* terbaik yang menghasilkan model klasifikasi yang berkinerja tinggi ketika dikonfigurasi kepada kedua model klasifikasi baik model BiLSTM+IndoBERT maupun model BiLSTM+FastText untuk tugas klasifikasi teks pertanyaan pengguna konsultasi kesehatan *online*.

5.2 Saran

Saran untuk pengembangan dan peningkatan lebih lanjut terhadap hasil yang didapatkan dari penelitian ini adalah sebagai berikut:

1. Mengumpulkan lebih banyak data pada kumpulan data *training* untuk meningkatkan keragaman data *training* sehingga dapat membantu model untuk mempelajari pola yang kompleks pada saat pelatihan model dan

melakukan generalisasi yang baik terhadap data baru yang akhirnya menghasilkan model klasifikasi yang berkinerja dengan lebih baik.

2. Meningkatkan kualitas data *training* dan *testing* dengan memastikan data bersih, data diberi label yang benar dan akurat, dan data yang merepresentasikan domain tugas klasifikasi. Pemeriksaan keakuratan pada label suatu data dapat dilakukan dengan anotasi secara manual dengan anotator yang profesional atau terlatih untuk domain tertentu.
3. Meningkatkan kualitas *text preprocessing* yang disesuaikan dengan tugas klasifikasi tertentu, seperti menghilangkan *stopwords* pada data *training* karena *stopwords* memiliki relevansi yang rendah terhadap kategori pada domain kesehatan.
4. Mengeksplorasi konfigurasi kapasitas model, seperti menambahkan atau mengurangi lapisan *neural network* dan jumlah *neuron* atau *hidden unit* sehingga model dapat menangkap kompleksitas distribusi data.
5. Melakukan teknik *hyperparameter tuning*, seperti *grid search* atau *random search* untuk menetapkan *hyperparameter* yang optimal dan *regularization* yang sesuai dan efektif untuk mencegah *overfitting* dan *underfitting* sehingga menghasilkan model yang berkinerja tinggi dan *good fit*.
6. Meningkatkan kualitas *word embedding* fastText dengan meningkatkan jumlah data teks pada saat melatih vektor *word embedding*, yaitu dengan cara mengumpulkan data teks yang beragam topik dalam domain kesehatan sehingga dapat menangkap hubungan semantik antar kata dalam nuansa medis dengan baik yang akhirnya menghasilkan vektor *input* yang efektif untuk model klasifikasi.
7. Melakukan eksplorasi pada 12 lapisan *encoder* IndoBERT untuk menentukan lapisan yang menghasilkan vektor *contextual embedding* terbaik untuk tugas klasifikasi dalam domain kesehatan.

DAFTAR KEPUSTAKAAN

- Aggarwal, C. C. (2022). *Machine Learning for Text*. Springer International Publishing. <https://doi.org/10.1007/978-3-030-96623-2>
- Alpaydin, E. (2020). *Introduction to machine learning* (Fourth edition). The MIT Press.
- Bengio, Y., Ducharme, R., Vincent, P., & Janvin, C. (2003). A Neural Probabilistic Language Model. *J. Mach. Learn. Res.*, 3(null), 1137–1155.
- Bengio, Y., Simard, P., & Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5(2), 157–166. <https://doi.org/10.1109/72.279181>
- Bird, S., Klein, E., & Loper, E. (2009). *Natural Language Processing with Python* (1st ed.). O'Reilly Media, Inc.
- Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017). Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics*, 5, 135–146. https://doi.org/10.1162/tacl_a_00051
- Chollet, F. (2017). Deep Learning with Python. In *Deep Learning with Python*. Manning Publications Co. <https://doi.org/10.1007/978-1-4842-5364-9>
- Chollet, F. (2018). *Deep learning with Python*. Manning Publications Co.
- Dale, R. (2010). Classical approaches to natural language processing. In *Handbook of Natural Language Processing, Second Edition*.
- Deng, L., & Liu, Y. (2018). *Deep Learning in Natural Language Processing* (1st ed.). Springer Publishing Company, Incorporated.
- Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding* (arXiv:1810.04805). arXiv. <http://arxiv.org/abs/1810.04805>
- Elman, J. L. (1990). Finding structure in time. *Cognitive Science*, 14(2), 179–211. [https://doi.org/10.1016/0364-0213\(90\)90002-E](https://doi.org/10.1016/0364-0213(90)90002-E)
- Geron, A. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems* (2nd ed.). O'Reilly Media, Inc.
- Goldberg, Y., & Hirst, G. (2017). *Neural Network Methods in Natural Language Processing*. Morgan & Claypool Publishers.
- Goodfellow, I. J., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.

- Goyal, P., Pandey, S., & Jain, K. (2018). *Deep Learning for Natural Language Processing*. Apress. <https://doi.org/10.1007/978-1-4842-3685-7>
- Grandini, M., Bagli, E., & Visani, G. (2020). *Metrics for Multi-Class Classification: An Overview* (arXiv:2008.05756). arXiv. <http://arxiv.org/abs/2008.05756>
- Graves, A. (2012). *Supervised Sequence Labelling with Recurrent Neural Networks* (Vol. 385). Springer Berlin Heidelberg. <https://doi.org/10.1007/978-3-642-24797-2>
- Graves, A., Mohamed, A., & Hinton, G. (2013). *Speech Recognition with Deep Recurrent Neural Networks* (arXiv:1303.5778). arXiv. <http://arxiv.org/abs/1303.5778>
- Graves, A., & Schmidhuber, J. (2005). Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks*, 18(5–6), 602–610. <https://doi.org/10.1016/j.neunet.2005.06.042>
- Gurney, K. (1997). *An Introduction to Neural Networks*. Taylor & Francis, Inc.
- Hakim, A. N., Mahendra, R., Adriani, M., & Ekakristi, A. S. (2017). Corpus development for Indonesian consumer-health question answering system. *2017 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, 222–227. <https://doi.org/10.1109/ICACSIS.2017.8355037>
- Han, J., Pei, J., & Tong, H. (2023). *Data mining: Concepts and techniques* (Fourth edition). Morgan Kaufmann is an imprint of Elsevier.
- Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Comput.*, 9(8), 1735–1780. <https://doi.org/10.1162/neco.1997.9.8.1735>
- Joulin, A., Grave, E., Bojanowski, P., & Mikolov, T. (2016). *Bag of Tricks for Efficient Text Classification* (arXiv:1607.01759). arXiv. <http://arxiv.org/abs/1607.01759>
- Khattak, F. K., Jeblee, S., Pou-Prom, C., Abdalla, M., Meaney, C., & Rudzicz, F. (2019). A survey of word embeddings for clinical text. *Journal of Biomedical Informatics*, 100, 100057. <https://doi.org/10.1016/j.yjbinox.2019.100057>
- Khurana, D., Koli, A., Khatter, K., & Singh, S. (2023). Natural language processing: State of the art, current trends and challenges. *Multimedia Tools and Applications*, 82(3), 3713–3744. <https://doi.org/10.1007/s11042-022-13428-4>
- Kowsari, K., Meimandi, K. J., Heidarysafa, M., Mendu, S., Barnes, L. E., & Brown, D. E. (2019). Text Classification Algorithms: A Survey. *Information*, 10(4), 150. <https://doi.org/10.3390/info10040150>
- Lane, H., Howard, C., & Hapke, H. (2019). *Natural Language Processing in Action*. Manning Publications Co.

- Lever, J., Krzywinski, M., & Altman, N. (2016). Classification evaluation. *Nature Methods*, 13(8), 603–604. <https://doi.org/10.1038/nmeth.3945>
- Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). *Efficient Estimation of Word Representations in Vector Space* (arXiv:1301.3781). arXiv. <http://arxiv.org/abs/1301.3781>
- Minaee, S., Kalchbrenner, N., Cambria, E., Nikzad, N., Chenaghlu, M., & Gao, J. (2021). *Deep Learning Based Text Classification: A Comprehensive Review* (arXiv:2004.03705). arXiv. <http://arxiv.org/abs/2004.03705>
- Niu, X., Hou, Y., & Wang, P. (2017). Bi-Directional LSTM with Quantum Attention Mechanism for Sentence Modeling. In D. Liu, S. Xie, Y. Li, D. Zhao, & E.-S. M. El-Alfy (Eds.), *Neural Information Processing* (Vol. 10635, pp. 178–188). Springer International Publishing. https://doi.org/10.1007/978-3-319-70096-0_19
- Osinga, D. (2018). *Deep Learning Cookbook*. O'Reilly Media, Inc.
- Pennington, J., Socher, R., & Manning, C. (2014). Glove: Global Vectors for Word Representation. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1532–1543. <https://doi.org/10.3115/v1/D14-1162>
- Priddy, K. L., & Keller, P. E. (2005). *Artificial neural networks: An introduction*. SPIE Press.
- Samuel, A. L. (1962). Artificial Intelligence: A Frontier of Automation. *The Annals of the American Academy of Political and Social Science*, 340, 10–20.
- Schuster, M., & Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11), 2673–2681. <https://doi.org/10.1109/78.650093>
- Sokolova, M., & Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, 45(4), 427–437. <https://doi.org/10.1016/j.ipm.2009.03.002>
- Uysal, A. K., & Gunal, S. (2014). The impact of preprocessing on text classification. *Information Processing & Management*, 50(1), 104–112. <https://doi.org/10.1016/j.ipm.2013.08.006>
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2023). *Attention Is All You Need* (arXiv:1706.03762). arXiv. <http://arxiv.org/abs/1706.03762>
- Wang, P., Qian, Y., Soong, F. K., He, L., & Zhao, H. (2016). Learning Distributed Word Representations For Bidirectional LSTM Recurrent Neural Network. *Proceedings of the 2016 Conference of the North American Chapter of the*

Association for Computational Linguistics: Human Language Technologies, 527–533. <https://doi.org/10.18653/v1/N16-1064>

- Wilie, B., Vincentio, K., Winata, G. I., Cahyawijaya, S., Li, X., Lim, Z. Y., Soleman, S., Mahendra, R., Fung, P., Bahar, S., & Purwarianti, A. (2020). *IndoNLU: Benchmark and Resources for Evaluating Indonesian Natural Language Understanding* (arXiv:2009.05387). arXiv. <http://arxiv.org/abs/2009.05387>
- Xie, J., Chen, B., Gu, X., Liang, F., & Xu, X. (2019). Self-Attention-Based BiLSTM Model for Short Text Fine-Grained Sentiment Classification. *IEEE Access*, 7, 180558–180570. <https://doi.org/10.1109/ACCESS.2019.2957510>
- Yan, Y., Wang, Y., Gao, W.-C., Zhang, B.-W., Yang, C., & Yin, X.-C. (2018). LSTM²: Multi-Label Ranking for Document Classification. *Neural Processing Letters*, 47(1), 117–138. <https://doi.org/10.1007/s11063-017-9636-0>
- Zhang, A., Lipton, Z. C., Li, M., & Smola, A. J. (2023). *Dive into Deep Learning*. Cambridge University Press.
- Zhang, Y., Chen, M., & Liu, L. (2015). A review on text mining. *2015 6th IEEE International Conference on Software Engineering and Service Science (ICSESS)*, 681–685. <https://doi.org/10.1109/ICSESS.2015.7339149>
- Zhang, Y., Chen, Q., Yang, Z., Lin, H., & Lu, Z. (2019). BioWordVec, improving biomedical word embeddings with subword information and MeSH. *Scientific Data*, 6(1), 52. <https://doi.org/10.1038/s41597-019-0055-0>

LAMPIRAN

Lampiran 1. *Word Cloud* kumpulan data pada kategori Kandungan



Lampiran 2. Word Cloud kumpulan data pada kategori Kulit dan Kelamin



Lampiran 5. *Word Cloud* kumpulan data pada kategori Pernapasan



Lampiran 6. Word Cloud kumpulan data pada kategori Psikologi



Lampiran 7. Word Cloud kumpulan data pada kategori Lainnya



Lampiran 8. Hasil prediksi pertanyaan konsultasi kesehatan menggunakan model BiLSTM+FastText

Index	Questions	Actual label	Predicted label
0	apr sore dok saya mau tanya apakah ada wanita yang hamil minggu mengalami haid	kandungan	kandungan
1	malam dok saya mau tanya mengenai ibu saya yang memiliki banyak keluhan serta mudah sekali lemas ibu saya umurnya tahun tapi sudah tidak menstruasi lagi terakhir katanya menstruasi hanya dua tetes saja dan badan ibu saya selalu lemas karena beliau juga mengidap penyakit lambung maag lalu telinga yg terinfeksi juga akibat kemasukan air dan di korek ibu saya sudah berobat di klinik mana saja tapi tetap tidak sembuh katanya selalu mual dan lemas bagaimana ya dok cara mengatasinya agar tidak selalu mual dan lemas apa ada cara lain selain mengkonsumsi obat maag karena obat yang ibu saya konsumsi sebelumnya sudah terlalu banyak dan bermacam terimakasih	pencernaan	pencernaan
2	feb halo dok permisi mau tanya barusan gigi saya yang bagian depan patah ketika saya makan apa bisa tumbuh kembali ya dok saya usia tahun	gigidanmulut	gigidanmulut
3	dok saya mau tanya dok nafas saya mengi tapi tidak merasakan sesak nafas dok atau pun nyeri dada dan kalo sehabis tidur saya sering hidung tersumbat dok dan kalau setelah minum air putih tuh air liur ada darahnya dikit dok tapi itu tidak sering tapi cuma beberapa saat aja dok cuman sedikit itu knp ya dok	pernapasan	pernapasan
4	okt saya sering sakit kepala berat ingin tidur terus sering cemas dan jantung sering berdetak kencang	psikologi	psikologi
5	kemarin saya tidak sengaja makan daging ayam yang kurang matang apa yang harus saya lakukan dok saya takut masih ada bakteri atau virus lainnya	pencernaan	lainnya
6	mar permisi dok pada tanggal maret saya dan pacar coba untuk memasukkan penisnya ke vagina saya tanpa alat kontrasepsi dan yang masuk hanya ujungnya serta tdk merobek hymen ini berlangsung tidak ada mnt saya dan dia ykn tak ad cairan pra eja dan sperma yang msk ke dlm tapi bln ini saya terlambat haid janua dan feb saya haid kisaran tgl apakh ada kemungkinan hamil dia juga mengeluarkan cairannya di luar	kandungan	kandungan
7	sep perut saya sakit tepat dibawah pusar itu kira kenapa ya	pencernaan	pencernaan
8	dok anak saya berusia hari sejak sekitar minggu setelah dia lahir dia sudah mengalami kesulitan bernafas yg makin parah hingga sekarang dia sering terganggu tidurnya karena kesulitan bernafas kadang dia terbangun sesaat dan menggerakkan kaki dan tangannya karna sepertinya tidak bisa nafas kemudian setelah dia bisa	pernapasan	lainnya

Index	Questions	Actual label	Predicted label
	nafas dia tidur lagi kata ibu saya kepalanya harus d bikin tinggi saat tidur jdi bayi saya d suruh pakai bantal besar jd posisi tidurnya setengah duduk gt sebenarnya itu boleh nggak dok dan aman gak utk bayi saya		
9	agt permisi dok perkenalkan saya wanita bernama nana usia tahun saya introvert yang memiliki gangguan depresi dan cemas dahulu saya di beri obat risperidone mg saya mendapati diri saya keracunan obat tersebut overdosis efek samping serius mulai dari perasaan ingin pingsan otot kaku kejang tremor sakit kepala kesulitan bernafas tenggorokan bengkak bibir biru keunguan wajah pucat wajah bengkak kesulitan menelan pandangan kabur tidak bisa buka mata hampir jam keringat dingin terus menerus lemas sempoyongan dan kepala sakit luar biasa saya sudah menghentikan obat tersebut awalnya saya memang memiliki halusinasi bau an sehari saja ketika saya sedang depresi berat beberapa tahun lalu dan hanya berlangsung beberapa jam setelah itu saya tidak pernah mengalami halusinasi lagi sampai sekarang saya pun sudah membicarakan kepada psikeater saya bahwa saya tidak memiliki halusinasi lagi tapi saya tetap di beri obat antipsikotik oleh psikeaternya dan saya masih trauma meminum obat psikeater golongan antipsikotik sedangkan saya merasa saya hanya depresi biasa dan cemas biasa saya tidak pernah mengalami halusinasi waham atau gejala sakit mental serius lainnya kemudian saya diberi resep racikan obat seperti stelazine mg amytripilin mg diazepam mg serta coblazam tablet yang jadi pertanyaan saya apakah aman saya mengonsumsi obat an tersebut sedangkan saya masih takut serta trauma meminum lagi obat golongan antipsikotik apakah obat racikan tersebut akan membuat saya meninggal melihat dosis yang di berikan menurut saya terlalu banyak sedangkan ketika keracunan risperidone saya hanya mengonsumsi dosis paling rendah terimakasih banyak dok jawabannya saya begitu stress karena takut meminum obat golongan antipsikotik lagi saya sudah berulang kali mengatakan pada psikeater bahwa saya tdk mengalami halusinasi tetapi saya tetap diberi obat tsb mohon bantuannya	psikologi	psikologi
10	okt dok obat yg cocok untuk anak jika gusi bengkak apa ya	gigidanmulut	gigidanmulut
11	feb hallo dok saya tanya nih kenapa ya setiap saya mendapat luka dan berdarah atau bahkan melihat luka berdarah badan saya langsung lemas dan kedua lutut saya langsung gemetar sampai membuat saya sedikit kesusahan untuk bergerak atau mengambil barang terima kasih dok	psikologi	kulitdankelamin
12	mengapa ya dok tiap kali anak saya aktifitas di rumah tiap kali mau mandi entah pagi siang malam kadang keluar sedikit kotoran di celana dalamnya kira kira gejala apa ya dok anak saya perempuan usia tahun	pencernaan	lainnya

Index	Questions	Actual label	Predicted label
13	nov saya mudah nangis untuk hal kecil satu hal buruk bisa membuat saya hancur mood saya jga sering berubah kadang saya bersemangat tiba merasa sedih saya biasa bangun dengan merasa sangat kosong dlu saya sangat suka menulis skrg saya tidak bersemangat lgi untuk menulis sbnarnya saya knp	psikologi	psikologi
14	agt saya punya bisul dan selalu timbul hanya di pantat bagian kiri tidak pernah tumbuh di area yang lain penyebabnya apa ya dok karena kalau saya baca baca telur tidak menjadi salah satu penyebabnya apa benar atau mungkin dikarenakan mengenakan celana dalam yang terlalu ketat selain itu saya dulu juga pernah bisulan setelah bersepeda apakah faktor gesekan juga mempengaruhinya terima kasih	kulitdankelamin	kulitdankelamin
15	halo dok nama sy dawiyah ibu dari anak yg satu masih dakam kandungan mau tanya anak saya yg pertama umur tahun suka gigitin baju nya sampai basah itu kenapa ya dok penyebab nya bahayakah n bagaimana cara mengatasi nya terimakasih	gigidanmulut	lainnya
16	okt met subuh dokter saya ada keluhan nihsaya sperti diare bolak balik ke kamar mandi tp yg kluar bab dikit dikit trus terasa sesak di ulu hati itu kanepa ya	pencernaan	pencernaan
17	nov ass dok wajah berjerawat terus saya pakai bedak dari dokter jerawat udah mulai mengering tapi pada saat saya menggunakan cream itu wajah saya panas gatal berih dan setelah itu wajah saya memerah kulit pun mengelupas lalu wajah saya bengkak gitu dok itu kenapa ya dok harus saya hentikan creamnya apa harus berhenti	kulitdankelamin	kulitdankelamin
18	jul dok saya telat haid dan keputihan deres kadang encer dan kadang berwarna putih dan kuning sampai sekarang blom haid apakah terjadi kehamilan atau gimana dok	kandungan	kandungan
19	agt tadi siang kontrol ke dokter selang drain udah di ambil sama dokter n habis di ambil selang ya ko ga di jahit apa di lem	kulitdankelamin	kulitdankelamin
20	okt selamat pagi dok saya mau tanya hari selasa kemarin saya baru cabut gigi geraham dan dihari sabtu minggu ini rencananya saya mau naik gunung apakah boleh dok	gigidanmulut	gigidanmulut
21	apr pagi dok maaf jika saya tidak berhubungan badan tetapi saya mengeluarkan sperma diperut perempuan tsb kemudian sperma tsb mengalir kebawah dok menuju vagina nya apakah bisa menyebabkan kehamilan dok terimakasih	kandungan	kandungan
22	saya mau bertanya dok seminggu yang lalu saya scalling gigi untuk membersihkan karang beberapa hari setelahnya saya makan duren kemudian gigi geraham saya sakit nyeri usia tambalan sekitar tahun sebelumnya	gigidanmulut	gigidanmulut

Index	Questions	Actual label	Predicted label
	tidak ada keluhan apakah sebenarnya tambalan gigi saya memang bermasalah tetapi baru ketahuan setelah karang giginya dibersihkan		
23	sep dok saya mau nanya m p saya kan kecil tapi panjang saat ereksi hampir cm itu pun sya ukur dengan penggaris biasa apakah bisa kalo bisa dengan panjang saat ereksi penis lebih dari cm tapi dengan ukuranya kecil apakah normal dok	kulitdankelamin	kulitdankelamin
24	okt saya mau tanya dok apakah boleh ibu hamil muda makan buah anggur merah	kandungan	kandungan
25	mar assalamualaikum dok mau nanya setiap saya mau bab knapa ga keluar tapi malah kaya ada darah dikit yang keluar di anus saya itu penyakit apa ya dok	pencernaan	pencernaan
26	jan dokter saya sudah bulan menggunakan extension kuku dan mau saya lepas bagaimana ya cara melepas extension kuku agar kuku asli tidak rusak	kulitdankelamin	kulitdankelamin
27	sep selamat malam dok saya mau bertanya awalnya di kaki terdapat luka kecil dan beberapa hari kemudian mengeluarkan cairan seperti nanah sudah sempet periksa ke dokter dan alhamdulillah mulai membaik tapi tidak selang beberapa hari luka nya muncul lagi yg awalnya hanya di kaki sekarang muncul juga di tangan bisa dikatakan semakin banyak dan sama mengeluarkan cairan seperti nanah juga mohon dibantu dok untuk penjelasan dan cara menangani nya terimakasih	kulitdankelamin	kulitdankelamin
28	jun malam dok saya mau tanya saya habis melakukan hubungan seksual untuk pertama kalinya ketika malam saya melakukannya dan mengeluarkan sedikit darah dan disertai sakit lalu keesokannya saya melakukan lagi tetapi ada hentakan sedikit kencang terbentur sehingga menyebabkan sakit yg luar biasa bingga ke perut bawah saya seperti sakit sedang menstruasi setelah itu saya mengeluarkan darah yg banyak disertai beberapa kali ada gumpalan darah dan trs menerus mengeluarkan darah ini kenapa ya dok	kulitdankelamin	kandungan
29	dok saya hamil weeks dan sering pusing kaya gemeter gitu badan hangat tetapi kringetnya terasa dingin trs kalo pusingnya udah ilang sakit gigi nya kumat kenapa ya	gigidanmulut	gigidanmulut
30	pagi dok maaf mau bertanya dodok dada terasa sakit seperti ditekan engap untuk bernafas ada sakit menelan terkadang ada sakit perut sebelah kanan gejala apa ya dok	pernapasan	pencernaan
31	sep dok mengapa banyak orang suka menyimpan masalahnya sendiri apa dampak dari orang orang yang suka memendam masalahnya tanpa membaginya pada orang lain lalu bagaimana cara meyakinkan mereka agar mereka mau terbuka akan masalah yang sedang dihadapinya	psikologi	psikologi

Index	Questions	Actual label	Predicted label
32	dok knpa yah telingaku yg sebelah kanan kayak ada bunyi kresek kresek gitu sprti ada hewan tpi bukan semacam ada yang mau keluar tpi sdah hari belum keluar	lainnya	lainnya
33	mau tanya dok saya kan di diagnosa tb paru aktif cuma saya tanpa gejala dok mau tanya gimana caranya mengeluarkan dahak untuk tes dahak ya dok soalnya tidak batuk dok sudah di pakai uap juga tetep gabisa dok apakah ada solusinya dok	pernapasan	pernapasan
34	dok gigi dewasa saya kan baru tumbuh sedikit namun jika saya sentuh pakai lidah ada sedikit bagian yg terasa kasar dan seperti bukan gigi terasa seperti menyentuh sikat	gigidanmulut	gigidanmulut
35	permisi dokter dok izin bertanya istri saya saat ini alhamdulillah sedang hamil dok ini kehamilan pertamanya jadi kami sama belum ada pengalaman dok yg dirasakan sata ini terutama saya sebagai suami adalah emosi istri saya yg sangat tidak stabil dok dia bisa dengan mudah merasa senang lalu berubah marah hingga kadang sampe mennangis dok saya terkadang sampe kewalahan dokter apakah hal seperti ini normal terjadi pada wanita hamil	psikologi	kandungan
36	malam dok ada yg aneh dg payudara kiri sy sdhh hampir th mengeluarkan cairan bening agak putih dg volumenya tdk sampai tetes cairan tsb keluar jk dipencet sj dn ketika sdh dipencet dn keluar kemudian dipencet lg itu sdh tdk keluar harus dijeda beberapa menit baru bs keluar lg jk dipencet tdk ada benjolan tdk sakit puting normal bentuk payudaranya normal tdk ada penebalan kulit dll fnab hsilnya ada sedikit sel radang tp msh normal tp sy ttp takut kira knp y dok n	lainnya	lainnya
37	pagi dok mau tanya beberapa hari ini badan sering lemes keluar keringat dingin kadang panas pusing perut kembung mual sesak pandangan agak kabur kira kira gejala apa ya dok makasih	pencernaan	pencernaan
38	jan dok mau tanya apakah kekurangan darah menjadi tanda kehamilan atau itu tanda telat menstruasi	kandungan	kandungan
39	okt dok saya mau tanya kenapa di dalam perut saya terasa panas kadang kya ada yg bergerak tp saya tiap bulan halangan apakah itu penyakit	pencernaan	pencernaan
40	des dok saya positif hbsag apakah saya bisa lahiran di luar daerah dengan menggunakan bpjs berdomisili jakarta dan apakah saya mendapatkan vaksin hbsag	kandungan	kulitdankelamin
41	nov halo dok dok seminggu lalu udah haid terus kenapa keluar darah lagi y dok tapi sedikit	kulitdankelamin	kandungan
42	des testis turun satu kenapa	kulitdankelamin	kulitdankelamin

Index	Questions	Actual label	Predicted label
43	nov gimana yah caranya mencurangi komedo dan ngecilin pori pori soalnya udah pake banyak produk tetap aja gak berkurang	kulitdankelamin	kulitdankelamin
44	mar assalamu'alaikum dok kalau setiap hari ngocok sendiri bisa menyebabkan ke hamilan gak dok atau perut sakit di bagian kanan bawah	kulitdankelamin	kandungan
45	okt hallo dok saya haid tgl oktober dan selesai tgl sore tadi saya keluar darah kirain saya mau haid soalnya siklus saya pendek namun bercak dan warnanya merah tp bbrpa menit menjadi gelap tidak sprt biasanya saya haid apakah itu darah implansi dok saya ga pernah hubungan badan dan waktu keluar darah pun tidak di sertai kram	kandungan	kandungan
46	dok obat panadol apa bisa buat sakit gigi yg bngkak	gigidanmulut	gigidanmulut
47	sep bismillah saya mau bertanya dok saya uda telat haid uda minggu tiga hari yg lalu saya mengalami keputihan saya coba tespek hasilnya negatif apakah itu bisa dikatakan normal dok	kandungan	kandungan
48	dok didalem kulit mata saya ada benjolan trs mata ngeluarin cairan berbahaya gk sih dok kan biasanya klw bintitan di area luar mata	lainnya	lainnya
49	sore dok saya mau tanya saya kan sudah hari sakit gigi ya terus selama sakit gigi ini saya bingung dok makan apalagi paling makan bubur aja saya bosan makan bubur terus dok saya minta rekomendasi makanan lembut untuk orang sakit gigi dok makasih	gigidanmulut	gigidanmulut
50	dok saya mau tanya saya perokok dan saya bekerja sebagai tukang las yg saya tanyakan belakangan ini saya merasa kurang enak untuk bernafas dok ya bisa dibilang seperti sesak untuk bernafas dan selalu bernafas besar yang saya tanyakan apakah saya perlu untuk terapi nebul dok dan apa obat nya trima kasih dok	pernapasan	pernapasan
51	apr dokter gigi depan saya satunya ompong sekarang saya umur tahun apakah gigi saya masih bisa tumbuh lagi	gigidanmulut	gigidanmulut
52	selamat siang dok mohon penjelasannya dok saya ketika mejamin mata terasa puyeng dan di ujung kaki terasa dingin dok	lainnya	lainnya
53	jan hallo dok kenapa bagian ulu hati nyeri tapi berjeda sebentar ilang kerasa lagi bab juga cair pusing bagian kepala mual tapi ga muntah	pencernaan	pencernaan
54	apr dok gigi sya yang sementara perawatan saluran akar geraham bawah kiri itu sakit lalu tambalan sementara yg ketiga kalinya dibongkar setelah dibongkar giginya masih kadang sakit kalau bertemu dengan gigi bagian	gigidanmulut	gigidanmulut

Index	Questions	Actual label	Predicted label
	atasnya lalu kata dokternya itu nga papa apakah benar dok gigi tidak apa apa karena kata dokter mau dicabut tpi kadang masih sakit jika bertemu gigi atasnya dok		
55	sep dok saya udah hampir bulanan terkena ruam pada kaki awalnya muncul bintik merah kecil di betis tidak terasa gatal tapi lama kelamaan menjadi banyak hingga menjalar ke paha dan telapak kaki dan rasanya gatal ukurannya pun mulai besar dan bengkak seperti gigitan nyamuk sebenarnya saya sakit apa yah dok	kulitdankelamin	kulitdankelamin
56	dok jadi dulu saya pas smp termasuk yang tinggi nya menjulang mungkin sekitar cm lebih tinggi dari rata teman saya tapi pas saya sma saya tergolong pendek dari teman seumuran pertanyaan saya bisakah saya mengejar tinggi teman yang lain atau saya akan berhenti tumbuh duluan dibanding yang lain	lainnya	lainnya
57	dokter apa obat tradisional untuk bayi demam	lainnya	lainnya
58	des dok saya mau nanya dokter berhubungan seks di dalam air dengan menggunakan kondom apakah bisa terjadi kehamilan dok tapi saat saya tiup kondom nya dan saya isi air dia tidak bocor dan tidak kempes dok saya ngelakuin hubungan seks bulan kali dok dengan menggunakan kondom apakah mempengaruhi siklus mens dok soalnya biasa nya istri saya haid tanggal sampe skrng belum haid dok kenapa ya	kulitdankelamin	kandungan
59	sore dok sudah hari ini mata kanan saya terkadang perih dan sensitif pada cahaya siang hari dan kadang berair dikarenakan saya begadang karena tuntutan pekerjaan saat ini mata saya pakaikan insto kira kira ada apa ya dok dgn mata saya dan cara mengatasinya karna begadang ini tuntutan pekerjaan saya dok saya dapat shift malam thx	lainnya	lainnya
60	saya lelaki tahun beberapa hari yg lalu saya di vonis menderita pneumonia dan tbc saya sempat opname selama hari di rumah sakit tapi lucunya saat keluar dari rumah sakit saya tidak terlalu tau tentang informasi penyakit saya selama di rumah sakit dokter paru yg menangani penyakit saya cuma bisa di temui kali dan ketika bertemu dia tidak menjelaskan apapun hanya tanya keluhan habis itu langsung jalan lg mungkin dia terlalu sibuk saat keluar dari rumah sakit saya mendapatkan buah obat megabal mecobalamin mg x ranitidin hcl mg dan terakhir cefixime trihydrate mg x sudah sekitar seminggu saya mengalami perawatan di rumah jujur tidak banyak perkembangan yg terjadi karna minim nya informasi saya coba cari di internet tapi informasi di internet masih bersifat acak yg saya tau ketika udara sangat dingin maka saya akan batuk dan dada saya terasa gatal untuk batuk selain itu saya sering kelaparan tengah malam saya ingin tau apa pantangan dalam pengobatan penyakit saya apa hal yg bisa mendukung dalam proses penyembuhan saya dan berapa	pernapasan	pernapasan

Index	Questions	Actual label	Predicted label
	lama biasanya orang di rawat untuk menghilangkan flek yg ada di paru apakah saya harus menemui dokter spesialis paru lg untuk konsultasi		
61	nov salah minum pil kb sebelum haid minum warna putih sampe habiskonsumsi pil kb namun tidak haid kenapa	kandungan	kandungan
62	des halo dok saya ingin bertanya apakah onani setiap seminggu sekali itu hal yg wajar tidak	kulitdankelamin	kulitdankelamin
63	dok belakangan ini ada kabar bakteri masuk ke indonesia lagi yg viral yaitu mycoplasma pneumoniae yg bisa menyebabkan infeksi dan saya kan punya anak kecil nih dok kabarnya kebanyakan usia anak yg rentan kena usia anak yg rentan terkena itu usia berapa dokter	pernapasan	lainnya
64	apr halo dok ayah saya berusia selalu mengalami rasa tertekan di perut bagian tengah ulu hati dan sudah minum beberapa obat untuk pencernaan tapi tidak berpengaruh kira kira apakah penyebabnya tidak ada muntah dan mual dan tidak ada demam obat apakah yang sebaiknya di berikan yang tidak terlalu keras untuk lansia terimakasih	pencernaan	pencernaan
65	feb permisi dok bu dok sebelumnya saya ingin menanyakan dengan serius dan tolong kasih jawaban yang bisa dipahami intinya saya melihat di google ciri wanita tdk perawan dilihat dari fisiknya dan itu benar ada diciri pasangan saya tapi dia tidak mau jujur atau mengaku masih perawan tolong kasih ciri jawaban yang memang benar kalau dia itu tdk perawan lagi terima kasih	kulitdankelamin	kulitdankelamin
66	des dok kuyup bengkak akibat minyak kemiri gimana cara mengatasi x	kulitdankelamin	kulitdankelamin
67	mei dokter saya ingin bertanya saya remaja thn saya sering merasakan tekanan yg membuat sy ingin melakukan hal yg gila tp itu hanya sesaat tp setiap kali saya terdiam sendiri di kamar sy tdk mau melakukan hal apapun dan bahkan jika saya merasa tersinggung sy tdk ingin melihat org itu walaupun itu org tua sy bahkan saat ditanya sy menangis seolah tidak sadar itu sy saat di tampar pun sy tdk merasakan sakit dok sy ingin sekali sembuh bagaimana penjelasannya terimakasih	psikologi	psikologi
68	apr sakit di bagian kanan bawah payudara itu karna apa	pencernaan	gigitanmulut
69	halo dok saya beberapa minggu ini rasa sesak saat cuaca panas dok dan kaki bengkak karena panas dok kira kira itu penyakit apa ya dok terimakasih dok	pernapasan	pernapasan
70	okt dok belakangan ini sy orang nya jijik an hal tsb membuat sy tbtb ingin membuang ludah tp sy tdk bisa membuang ludah secara leluasa apalagi sedang dlm pandemi bagaimana solusinya dok sy jg mau bertanya td	psikologi	psikologi

Index	Questions	Actual label	Predicted label
	saya terkena debu ke mata dan air mata saya keluar dan kena masker saya terus saja jijik an sy kambuh lg karna mulut saya terkena masker saya apakah jijik an saya berlebihan		
71	hi dok aku mau tanya udah seminggu ini tinjaku berwarna hitam terus aku juga merasakan sakit dikepala dibagin belakang dan perutku terasa kembung padahal aku tidak makan yg berlebihan kira kira aku sakit apa ya dok td nya aku pikir karena aku memakan mie jajangmyon yg berwarna hitam tetapi aku makan mie itu sudah lebih dr minggu yg lalu dr sekarang apakah memang itu berasal dr pewarna mie yg aku makan atau ada penyakit lainn dok mohon bantuannya dok	pencernaan	lainnya
72	haii dok saya amah mau tanya kl tenggorokan sakit itu gejalanya aph dok awalnya kk saya plg kerja batuk trs ayah saya jga batuk udh ada bln blm sembuh jga dok itu knp yh awalnya ayah saya narikin uang kematian trs keujanan dan meriang keesokkannya suaranya serak dan skrg batuk smpe skrg blm sembuh dok dan adik saya jga batuk panas dok kmrn smpe c dan saya skrg ikutan batuk jga kira itu knp dok saya takutny terkena covid amit si	pernapasan	pernapasan
73	apr assalamualaikum dok saya telat mens sudah hari dok tadi pagi tespek dan hasilnya satu tebal dan satu samar saya mau tespek lgi tp nnti nunggu beberpa hari lagi krn mungkin menurut saya kadar hcg masih rendah pertanyaan saya andaikan saya hamil bolehkah selama saya menunggu hari untuk tespek lg dan memutuskan kedokter bolehkah saya berhubungan dengan suami bahayakah buat bayinya karena saya pernah dengar dari dokternya sodara saya kalau hamil muda tdk boleh berhubungan mohon penjelasannya dok	kandungan	kandungan
74	maap dok saya mau tanya ini teh anak saya panas sma pilek batuk sma menycret udh hari tus di periksa ke bidan udah turn panas nya sekarang teh anak saya bab nya ada darah nya sedikit sedikit tp sering udah hari sama ga buang air kecil ga pipis sama sekali bahaya ga dok	lainnya	pencernaan
75	mar dok saya mau tanya bagaiman cara memutih kan gigi padahal sya sering gosok gigi	gigidanmulut	gigidanmulut
76	dokter apa faktor terjadinya pernapasan	pernapasan	lainnya
77	halo dok mau tanya anak usia bulansaya pupnya jadi agak encer tapi masih ada teksturnya tidak terlalu encer seperti air pupnya seperti itu hari kemarin awal mulanya saya beri jagung rebus untuk finger food nya dan pada saat pup masih ada butiran jagungnya karena anak saya blm tumbuh gigi sampai hari ini pupnya masih encer dan waktu bab nya normal kali sehariapakah itu termasuk diare atau hanya bab cair aja yg normal ya dok	pencernaan	gigidanmulut

Index	Questions	Actual label	Predicted label
78	sep salam dok suami sy megang v saya lalu melakukan gesek gesek jari dia di v saya tapi suami saya abis megang punya dia yg udah keluar cairan bukan sperma tp apakah dr tangan itu bs menyebabkan kehamilan dok terimakasih	kandungan	kandungan
79	agt sakit gigi boleh dicabut	gigidanmulut	gigidanmulut
80	jun dok bulan april tanggal saya haid dan berhenti di tanggal tetapi sampai sekarang saya tidak haid sebelum nya tanggal saya melakukan seks apakah itu bertanda hamil dok apakah ada kemungkinan saya mengalami kehamilan	kandungan	kandungan
81	sep selamat siang dok saya mau bertanyakemaren pengunci behel saya copot bagian pojok kiri lalu kawat nya bengkok apa itu tidak apa apa karena jadwal kontrol saya masih minggu lagi	gigidanmulut	kulitdankelamin
82	dok umur saya th tiba di mata kaki saya muncul benjolan apakah itu dok dan apa obat nya	lainnya	lainnya
83	halo dok izin bertanya sudah seminggu ini gigi saya sakit dikarenakan berlubang jika saya lihat di google nama gigi tersebut gigi geraham kecil kedua sakit yang saya rasakan hilang timbul namun sekalinya timbul sakitnya sampai ke telinga dan itu membuat saya tidak nyaman sakit pada gigi masih bisa saya tahan rasanya sedangkan untuk telinga saya tidak kuat menahannya lebih sakit telinga daripada gigi saya yang berlubang tetapi semakin sakit gigi saya semakin sakit pula telinga saya apakah itu yang bermasalah gigi atau telinga saya ya dok apakah saya harus ke dokter gigi atau ke tht dan apakah ada obat untuk pereda sakit pada gigi dan telinga saya terimakasih	gigidanmulut	gigidanmulut
84	jan halo dok saya laki laki usia tahun saya memiliki bulu kaki yang lebat bahkan seperti orang dewasa tidak hanya di kaki tetapi di tangan dan di badan juga kadang kalau saya pakai celana panjang suka ketarik sakit tapi memang bulu kaki ayah saya memang lebat kira kira karena apa ya dok lebih baik diapakan saya agak tidak percaya diri dan selalu saya tutupi pakai celana panjang mohon bantuannya dok	kulitdankelamin	psikologi
85	garis mata tidak sejajar krna bekas bintitan cara nyembuhinnya gimana dok	lainnya	lainnya
86	dok tadi sore saya makan nasi jagung terus selang beberapa jam perut terasa tidak enak seperti orang lapar tapi kenyang terus ketika malam mual muntah dan diare bagaimana dok terimakasih	pencernaan	pencernaan
87	nov cara membesar kan mr p secara alami	kulitdankelamin	kulitdankelamin
88	mau nanya bagian pantat sebelah kanan terasa nyeri dibuat lari langsung terasa pincang dibuat jongkok juga sakit itu kenapa ya	lainnya	lainnya

Index	Questions	Actual label	Predicted label
89	mar hi dok saya mau nanya apakah masih bisa hamil ketika sudah mengalami menstruasi kali dalam bulan bulan lalu saya melakukan patting tepatnya tanggal february lalu tanggal saya menstruasi lalu bulan ini saya mens tanggal setelah tanggal itu saya tidak melakukan hubungan seksual lagi dok sampai haid berikutnya apakah masih bisa hamil dok	kandungan	kandungan
90	dok bagaimana kondisi ketika paru mengalami gangguan apakah jika saturasi oksigen kita baik oksigen dan nadi bisa meniup sesuatu dengan kuat dan normal saat melakukan nafas dalam bisa menjadi rujukan bahwa kondisi paru kita masih sehat jika mengalami apa kita harus mencari pertolongan medis	pernapasan	pernapasan
91	nov dok saya mau tanya mengapa saya merasakan benci iri dan rindu bersamaan dok apakah itu normal	psikologi	kulitdankelamin
92	dok saya kebetulan memiliki riwayat asma yang kronis apakah ini membuat saya lebih mudah tertular virus corona	pernapasan	pernapasan
93	halo dok maaf mengganggu saya memakai behel sudah masuk minggu ketiga pada awal pemakaian terasa sakit dan hilang setelah seminggu lalu kemarin breket saya lepas jadi saya kontrol dan mengganti karet juga tetapi semalam tiba tiba saja gigi saya bagian atas ujung terasa sakit ngilu dan tidak bisa menggigit padahal sebelumnya tidak sakit seperti ini	gigidanmulut	gigidanmulut
94	dok kenapa pada tenggorokan saya kerasa ada bau seperti bau bangke tikus mati dan ditenggorokan terasa panas apakah ada obat resep penyembuhan nya dok	lainnya	pernapasan
95	agt halo dokter saya mau bertanya gigi bagian atas saya setiap digunakan untuk beraktivitas seperti lompat lari bahkan berjalan selalu terasa cenat cunut kira kira itu kenapa apakah perlu dicabut gigi saya terimakasih dok	gigidanmulut	gigidanmulut
96	apakah ada jeda antara ibuprofen dan concor	lainnya	lainnya
97	dokter saya ingin bertanya ada benjolan di atas areola tidak sakit tidak nyeri warna payudara normal tidak ada penebalan kira kira kenapa ya kalau memang kanker jika masih awal apa kita bisa menghilangkan sendiri tanpa operasi makanan apa saja yang memicu kanker makanan apa yang dibenci sel kanker terimakasih semoga dokter sehat selalu	lainnya	lainnya
98	jan sekitar minggu yg lalu stres berat dan sekrang telat haid apakah ada hubungan nya dok	kandungan	kandungan
99	apr kenapa ya dok saya kram perut dan sedikit bengkak juga sakit punggung lemas sering buang air kecil dan dada sesak	pencernaan	pencernaan

Lampiran 9. Hasil prediksi pertanyaan konsultasi kesehatan menggunakan model BiLSTM+IndoBERT

Index	Questions	Actual label	Predicted label
0	apr sore dok saya mau tanya apakah ada wanita yang hamil minggu mengalami haid	kandungan	kandungan
1	malam dok saya mau tanya mengenai ibu saya yang memiliki banyak keluhan serta mudah sekali lemas ibu saya umurnya tahun tapi sudah tidak menstruasi lagi terakhir katanya menstruasi hanya dua tetes saja dan badan ibu saya selalu lemas karena beliau juga mengidap penyakit lambung maag lalu telinga yg terinfeksi juga akibat kemasukan air dan di korek ibu saya sudah berobat di klinik mana saja tapi tetap tidak sembuh katanya selalu mual dan lemas bagaimana ya dok cara mengatasinya agar tidak selalu mual dan lemas apa ada cara lain selain mengkonsumsi obat maag karena obat yang ibu saya konsumsi sebelumnya sudah terlalu banyak dan bermacam terimakasih	pencernaan	pencernaan
2	feb halo dok permisi mau tanya barusan gigi saya yang bagian depan patah ketika saya makan apa bisa tumbuh kembali ya dok saya usia tahun	gigidanmulut	gigidanmulut
3	dok saya mau tanya dok nafas saya mengi tapi tidak merasakan sesak nafas dok atau pun nyeri dada dan kalo sehabis tidur saya sering hidung tersumbat dok dan kalau setelah minum air putih tuh air liur ada darahnya dikit dok tapi itu tidak sering tapi cuma beberapa saat aja dok cuman sedikit itu knp ya dok	pernapasan	pernapasan
4	okt saya sering sakit kepala berat ingin tidur terus sering cemas dan jantung sering berdetak kencang	psikologi	psikologi
5	kemarin saya tidak sengaja makan daging ayam yang kurang matang apa yang harus saya lakukan dok saya takut masih ada bakteri atau virus lainnya	pencernaan	lainnya
6	mar permisi dok pada tanggal maret saya dan pacar coba untuk memasukkan penisnya ke vagina saya tanpa alat kontrasepsi dan yang masuk hanya ujungnya serta tdk merobek hymen ini berlangsung tidak ada mnt saya dan dia ykn tak ad cairan pra eja dan sperma yang msk ke dlm tapi bln ini saya terlambat haid janua dan feb saya haid kisaran tgl apakh ada kemungkinan hamil dia juga mengeluarkan cairannya di luar	kandungan	kandungan
7	sep perut saya sakit tepat dibawah pusar itu kira kenapa ya	pencernaan	pencernaan
8	dok anak saya berusia hari sejak sekitar minggu setelah dia lahir dia sudah mengalami kesulitan bernafas yg makin parah hingga sekarang dia sering terganggu tidurnya karena kesulitan bernafas kadang dia terbangun sesaat dan menggerakkan kaki dan tangannya karna sepertinya tidak bisa nafas kemudian setelah dia bisa nafas dia tidur lagi kata ibu saya kepalanya harus d bikin tinggi saat tidur jdi bayi saya d suruh pakai bantal besar jd posisi tidurnya setengah duduk gt sebenarnya itu boleh nggak dok dan aman gak utk bayi saya	pernapasan	pernapasan

Index	Questions	Actual label	Predicted label
9	agt permisi dok perkenalkan saya wanita bernama nana usia tahun saya introvert yang memiliki gangguan depresi dan cemas dahulu saya di beri obat risperidone mg saya mendapati diri saya keracunan obat tersebut overdosis efek samping serius mulai dari perasaan ingin pingsan otot kaku kejang tremor sakit kepala kesulitan bernafas tenggorokan bengkak bibir biru keunguan wajah pucat wajah bengkak kesulitan menelan pandangan kabur tidak bisa buka mata hampir jam keringat dingin terus menerus lemas sempoyongan dan kepala sakit luar biasa saya sudah menghentikan obat tersebut awalnya saya memang memiliki halusinasi bau an sehari saja ketika saya sedang depresi berat beberapa tahun lalu dan hanya berlangsung beberapa jam setelah itu saya tidak pernah mengalami halusinasi lagi sampai sekarang saya pun sudah membicarakan kepada psikeater saya bahwa saya tidak memiliki halusinasi lagi tapi saya tetap di beri obat antipsikotik oleh psikeaternya dan saya masih trauma meminum obat psikeater golongan antipsikotik sedangkan saya merasa saya hanya depresi biasa dan cemas biasa saya tidak pernah mengalami halusinasi waham atau gejala sakit mental serius lainnya kemudian saya diberi resep racikan obat seperti stelazine mg amytripilin mg diazepam mg serta coblazam tablet yang jadi pertanyaan saya apakah aman saya mengonsumsi obat an tersebut sedangkan saya masih takut serta trauma meminum lagi obat golongan antipsikotik apakah obat racikan tersebut akan membuat saya meninggal melihat dosis yang di berikan menurut saya terlalu banyak sedangkan ketika keracunan risperidone saya hanya mengonsumsi dosis paling rendah terimakasih banyak dok jawabannya saya begitu stress karena takut meminum obat golongan antipsikotik lagi saya sudah berulang kali mengatakan pada psikeater bahwa saya tdk mengalami halusinasi tetapi saya tetap diberi obat tsb mohon bantuannya	psikologi	psikologi
10	okt dok obat yg cocok untuk anak jika gusi bengkak apa ya	gigidanmulut	gigidanmulut
11	feb hallo dok saya tanya nih kenapa ya setiap saya mendapat luka dan berdarah atau bahkan melihat luka berdarah badan saya langsung lemas dan kedua lutut saya langsung gemetar sampai membuat saya sedikit kesusahan untuk bergerak atau mengambil barang terima kasih dok	psikologi	kulitdankelamin
12	mengapa ya dok tiap kali anak saya aktifitas di rumah tiap kali mau mandi entah pagi siang malam kadang keluar sedikit kotoran di celana dalamnya kira kira gejala apa ya dok anak saya perempuan usia tahun	pencernaan	lainnya
13	nov saya mudah nangis untuk hal kecil satu hal buruk bisa membuat saya hancur mood saya jga sering berubah kadang saya bersemangat tiba merasa sedih saya biasa bangun dengan merasa sangat kosong dlu saya sangat suka menulis skrg saya tidak bersemangat lgi untuk menulis sbnarnya saya knp	psikologi	psikologi
14	agt saya punya bisul dan selalu timbul hanya di pantat bagian kiri tidak pernah tumbuh di area yang lain penyebabnya apa ya dok karena kalau saya baca baca telur tidak menjadi salah satu penyebabnya apa benar	kulitdankelamin	kulitdankelamin

Index	Questions	Actual label	Predicted label
	atau mungkin dikarenakan mengenakan celana dalam yang terlalu ketat selain itu saya dulu juga pernah bisulan setelah bersepeda apakah faktor gesekan juga mempengaruhinya terima kasih		
15	halo dok nama sy dawiyah ibu dari anak yg satu masih dakam kandungan mau tanya anak saya yg pertama umur tahun suka gigitin baju nya sampai basah itu kenapa ya dok penyebab nya bahayakah n bagaimana cara mengatasi nya terimakasih	gigidanmulut	lainnya
16	okt met subuh dokter saya ada keluhan nihsaya sperti diare bolak balik ke kamar mandi tp yg keluar bab dikit dikit trus terasa sesak di ulu hati itu kanepa ya	pencernaan	pencernaan
17	nov ass dok wajah berjerawat terus saya pakai bedak dari dokter jerawat udah mulai mengering tapi pada saat saya menggunakan cream itu wajah saya panas gatal berih dan setelah itu wajah saya memerah kulit pun mengelupas lalu wajah saya bengkak gitu dok itu kenapa ya dok harus saya hentikan creamnya apa harus berhenti	kulitdankelamin	kulitdankelamin
18	jul dok saya telat haid dan keputihan deres kadang encer dan kadang berwarna putih dan kuning sampai sekarang blom haid apakah terjadi kehamilan atau gimana dok	kandungan	kandungan
19	agt tadi siang kontrol ke dokter selang drain udah di ambil sama dokter n habis di ambil selang ya ko ga di jahit apa di lem	kulitdankelamin	kulitdankelamin
20	okt selamat pagi dok saya mau tanya hari Selasa kemarin saya baru cabut gigi geraham dan dihari Sabtu minggu ini rencananya saya mau naik gunung apakah boleh dok	gigidanmulut	gigidanmulut
21	apr pagi dok maaf jika saya tidak berhubungan badan tetapi saya mengeluarkan sperma diperut perempuan tsb kemudian sperma tsb mengalir kebawah dok menuju vagina nya apakah bisa menyebabkan kehamilan dok terimakasih	kandungan	kandungan
22	saya mau bertanya dok seminggu yang lalu saya scalling gigi untuk membersihkan karang beberapa hari setelahnya saya makan duren kemudian gigi geraham saya sakit nyeri usia tambalan sekitar tahun sebelumnya tidak ada keluhan apakah sebenarnya tambalan gigi saya memang bermasalah tetapi baru ketahuan setelah karang giginya dibersihkan	gigidanmulut	gigidanmulut
23	sep dok saya mau nanya m p saya kan kecil tapi panjang saat ereksi hampir cm itu pun sya ukur dengan penggaris biasa apakah bisa kalo bisa dengan panjang saat ereksi penis lebih dari cm tapi dengan ukuranya kecil apakah normal dok	kulitdankelamin	kulitdankelamin

Index	Questions	Actual label	Predicted label
24	okt saya mau tanya dok apakah boleh ibu hamil muda makan buah anggur merah	kandungan	kandungan
25	mar assalamualaikum dok mau nanya setiap saya mau bab knapa ga keluar tapi malah kaya ada darah dikit yang keluar di anus saya itu penyakit apa ya dok	pencernaan	pencernaan
26	jan dokter saya sudah bulan menggunakan extension kuku dan mau saya lepas bagaimana ya cara melepas extension kuku agar kuku asli tidak rusak	kulitdankelamin	kulitdankelamin
27	sep selamat malam dok saya mau bertanya awalnya di kaki terdapat luka kecil dan beberapa hari kemudian mengeluarkan cairan seperti nanah sudah sempet periksa ke dokter dan alhamdulillah mulai membaik tapi tidak selang beberapa hari luka nya muncul lagi yg awalnya hanya di kaki sekarang muncul juga di tangan bisa dikatakan semakin banyak dan sama mengeluarkan cairan seperti nanah juga mohon dibantu dok untuk penjelasan dan cara menangani nya terimakasih	kulitdankelamin	kulitdankelamin
28	jun malam dok saya mau tanya saya habis melakukan hubungan seksual untuk pertama kalinya ketika malam saya melakukannya dan mengeluarkan sedikit darah dan disertai sakit lalu keesokannya saya melakukan lagi tetapi ada hentakan sedikit kencang terbentur sehingga menyebabkan sakit yg luar biasa binggga ke perut bawah saya seperti sakit sedang menstruasi setelah itu saya mengeluarkan darah yg banyak disertai beberapa kali ada gumpalan darah dan trs menerus mengeluarkan darah ini kenapa ya dok	kulitdankelamin	kandungan
29	dok saya hamil weeks dan sering pusing kaya gemeter gitu badan hangat tetapi kringetnya terasa dingin trs kalo pusingnya udah ilang sakit gigi nya kumat kenapa ya	gigidanmulut	lainnya
30	pagi dok maaf mau bertanya dodok dada terasa sakit seperti ditekan engap untuk bernafas ada sakit menelan terkadang ada sakit perut sebelah kanan gejala apa ya dok	pernapasan	pernapasan
31	sep dok mengapa banyak orang suka menyimpan masalahnya sendiri apa dampak dari orang orang yang suka memendam masalahnya tanpa membaginya pada orang lain lalu bagaimana cara meyakinkan mereka agar mereka mau terbuka akan masalah yang sedang dihadapinya	psikologi	psikologi
32	dok knpa yah telingaku yg sebelah kanan kayak ada bunyi kresek kresek gitu sprti ada hewan tpi bukan semacam ada yang mau keluar tpi sdah hari belum keluar	lainnya	lainnya
33	mau tanya dok saya kan di diagnosa tb paru aktif cuma saya tanpa gejala dok mau tanya gimana caranya mengeluarkan dahak untuk tes dahak ya dok soalnya tidak batuk dok sudah di pakai uap juga tetep gabisa dok apakah ada solusinya dok	pernapasan	pernapasan

Index	Questions	Actual label	Predicted label
34	dok gigi dewasa saya kan baru tumbuh sedikit namun jika saya sentuh pakai lidah ada sedikit bagian yg terasa kasar dan seperti bukan gigi terasa seperti menyentuh sikat	gigidanmulut	gigidanmulut
35	permisi dokter dok izin bertanya istri saya saat ini alhamdulillah sedang hamil dok ini kehamilan pertamanya jadi kami sama belum ada pengalaman dok yg dirasakan sata ini terutama saya sebagai suami adalah emosi istri saya yg sangat tidak stabil dok dia bisa dengan mudah merasa senang lalu berubah marah hingga kadang sampe mennangis dok saya terkadang sampe kewalahan dokter apakah hal seperti ini normal terjadi pada wanita hamil	psikologi	psikologi
36	malam dok ada yg aneh dg payudara kiri sy sdhh hampir th mengeluarkan cairan bening agak putih dg volumenya tdk sampai tetes cairan tsb keluar jk dipencet sj dn ketika sdh dipencet dn keluar kemudian dipencet lg itu sdh tdk keluar harus dijeda beberapa menit baru bs keluar lg jk dipencet tdk ada benjolan tdk sakit puting normal bentuk payudaranya normal tdk ada penebalan kulit dll fnab hsilnya ada sedikit sel radang tp msh normal tp sy ttp takut kira knp y dok n	lainnya	lainnya
37	pagi dok mau tanya beberapa hari ini badan sering lemes keluar keringat dingin kadang panas pusing perut kembung mual sesak pandangan agak kabur kira kira gejala apa ya dok makasih	pencernaan	lainnya
38	jan dok mau tanya apakah kekurangan darah menjadi tanda kehamilan atau itu tanda telat menstruasi	kandungan	kandungan
39	okt dok saya mau tanya kenapa di dalam perut saya terasa panas kadang kya ada yg bergerak tp saya tiap bulan halangan apakah itu penyakit	pencernaan	pencernaan
40	des dok saya positif hbsag apakah saya bisa lahiran di luar daerah dengan menggunakan bpjs berdomisili jakarta dan apakah saya mendapatkan vaksin hbsag	kandungan	kandungan
41	nov halo dok dok seminggu lalu udah haid terus kenapa keluar darah lagi y dok tapi sedikit	kulitdankelamin	kandungan
42	des testis turun satu kenapa	kulitdankelamin	kulitdankelamin
43	nov gimana yah caranya mencurangi komedo dan ngecilin pori pori soalnya udah pake banyak produk tetap aja gak berkurang	kulitdankelamin	kulitdankelamin
44	mar assalamu'alaikum dok kalau setiap hari ngocok sendiri bisa menyebabkan ke hamilan gak dok atau perut sakit di bagian kanan bawah	kulitdankelamin	pencernaan
45	okt halo dok saya haid tgl oktober dan selesai tgl sore tadi saya keluar darah kirain saya mau haid soalnya siklus saya pendek namun bercak dan warnanya merah tp bbrpa menit menjadi gelap tidak sprt biasanya saya	kandungan	kandungan

Index	Questions	Actual label	Predicted label
	haid apakah itu darah implansi dok saya ga pernah hubungan badan dan waktu keluar darah pun tidak di sertai kram		
46	dok obat panadol apa bisa buat sakit gigi yg bngkak	gigidanmulut	gigidanmulut
47	sep bismillah saya mau bertanya dok saya uda telat haid uda minggu tiga hari yg lalu saya mengalami keputihan saya coba tespek hasilnya negatif apakah itu bisa dikatakan normal dok	kandungan	kandungan
48	dok didalem kulit mata saya ada benjolan trs mata ngeluarin cairan berbahaya gk sih dok kan biasanya klw bintitan di area luar mata	lainnya	lainnya
49	sore dok saya mau tanya saya kan sudah hari sakit gigi ya terus selama sakit gigi ini saya bingung dok makan apalagi paling makan bubur aja saya bosan makan bubur terus dok saya minta rekomendasi makanan lembut untuk orang sakit gigi dok makasih	gigidanmulut	gigidanmulut
50	dok saya mau tanya saya perokok dan saya bekerja sebagai tukang las yg saya tanyakan belakangan ini saya merasa kurang enak untuk bernafas dok ya bisa dibilang seperti sesak untuk bernafas dan selalu bernafas besar yang saya tanyakan apakah saya perlu untuk terapi nebul dok dan apa obat nya trima kasih dok	pernapasan	pernapasan
51	apr dokter gigi depan saya satunya ompong sekarang saya umur tahun apakah gigi saya masih bisa tumbuh lagi	gigidanmulut	gigidanmulut
52	selamat siang dok mohon penjelasannya dok saya ketika mejamin mata terasa puyeng dan di ujung kaki terasa dingin dok	lainnya	lainnya
53	jan halo dok kenapa bagian ulu hati nyeri tapi berjeda sebentar ilang kerasa lagi bab juga cair pusing bagian kepala mual tapi ga muntah	pencernaan	pencernaan
54	apr dok gigi sya yang sementara perawatan saluran akar geraham bawah kiri itu sakit lalu tambalan sementara yg ketiga kalinya dibongkar setelah dibongkar giginya masih kadang sakit kalau bertemu dengan gigi bagian atasnya lalu kata dokternya itu nga papa apakah benar dok gigi tidak apa apa karena kata dokter mau dicabut tpi kadang masih sakit jika bertemu gigi atasnya dok	gigidanmulut	gigidanmulut
55	sep dok saya udah hampir bulanan terkena ruam pada kaki awalnya muncul bintik merah kecil di betis tidak terasa gatal tapi lama kelamaan menjadi banyak hingga menjalar ke paha dan telapak kaki dan rasanya gatal ukurannya pun mulai besar dan bengkak seperti gigitan nyamuk sebenarnya saya sakit apa yah dok	kulitdankelamin	kulitdankelamin

Index	Questions	Actual label	Predicted label
56	dok jadi dulu saya pas smp termasuk yang tinggi nya menjulang mungkin sekitar cm lebih tinggi dari rata teman saya tapi pas saya sma saya tergolong pendek dari teman seumuran pertanyaan saya bisakah saya mengejar tinggi teman yang lain atau saya akan berhenti tumbuh duluan dibanding yang lain	lainnya	lainnya
57	dokter apa obat tradisional untuk bayi demam	lainnya	lainnya
58	des dok saya mau nanya dokter berhubungan seks di dalam air dengan menggunakan kondom apakah bisa terjadi kehamilan dok tapi saat saya tiup kondom nya dan saya isi air dia tidak bocor dan tidak kempes dok saya ngelakuin hubungan seks bulan kali dok dengan menggunakan kondom apakah mempengaruhi siklus mens dok soalnya biasa nya istri saya haid tanggal sampe skrng belum haid dok kenapa ya	kulitdankelamin	kandungan
59	sore dok sudah hari ini mata kanan saya terkadang perih dan sensitif pada cahaya siang hari dan kadang berair dikarenakan saya begadang karena tuntutan pekerjaan saat ini mata saya pakaikan insto kira kira ada apa ya dok dgn mata saya dan cara mengatasinya karna begadang ini tuntutan pekerjaan saya dok saya dapat shift malam thx	lainnya	lainnya
60	saya lelaki tahun beberapa hari yg lalu saya di vonis menderita pneumonia dan tbc saya sempat opname selama hari di rumah sakit tapi lucunya saat keluar dari rumah sakit saya tidak terlalu tau tentang informasi penyakit saya selama di rumah sakit dokter paru yg menangani penyakit saya cuma bisa di temui kali dan ketika bertemu dia tidak menjelaskan apapun hanya tanya keluhan habis itu langsung jalan lg mungkin dia terlalu sibuk saat keluar dari rumah sakit saya mendapatkan buah obat megabal mecobalamin mg x ranitidin hcl mg dan terakhir cefixime trihydrate mg x sudah sekitar seminggu saya mengalami perawatan di rumah jujur tidak banyak perkembangan yg terjadi karna minim nya informasi saya coba cari di internet tapi informasi di internet masih bersifat acak yg saya tau ketika udara sangat dingin maka saya akan batuk dan dada saya terasa gatal untuk batuk selain itu saya sering kelaparan tengah malam saya ingin tau apa pantangan dalam pengobatan penyakit saya apa hal yg bisa mendukung dalam proses penyembuhan saya dan berapa lama biasanya orang di rawat untuk menghilangkan flek yg ada di paru apakah saya harus menemui dokter spesialis paru lg untuk konsultasi	pernapasan	pernapasan
61	nov salah minum pil kb sebelum haid minum warna putih sampe habiskonsumsi pil kb namun tidak haid kenapa	kandungan	kandungan
62	des halo dok saya ingin bertanya apakah onani setiap seminggu sekali itu hal yg wajar tidak	kulitdankelamin	kulitdankelamin

Index	Questions	Actual label	Predicted label
63	dok belakangan ini ada kabar bakteri masuk ke indonesia lagi yg viral yaitu mycoplasma pneumoniae yg bisa menyebabkan infeksi dan saya kan punya anak kecil nih dok kabarnya kebanyakan usia anak yg rentan kena usia anak yg rentan terkena itu usia berapa dokter	pernapasan	pernapasan
64	apr halo dok ayah saya berusia selalu mengalami rasa tertekan di perut bagian tengah ulu hati dan sudah minum beberapa obat untuk pencernaan tapi tidak berpengaruh kira kira apakah penyebabnya tidak ada muntah dan mual dan tidak ada demam obat apakah yang sebaiknya di berikan yang tidak terlalu keras untuk lansia terimakasih	pencernaan	pencernaan
65	feb permisi dok bu dok sebelumnya saya ingin menanyakan dengan serius dan tolong kasih jawaban yang bisa dipahami intinya saya melihat di google ciri wanita tdk perawan dilihat dari fisiknya dan itu benar ada diciri pasangan saya tapi dia tidak mau jujur atau mengaku masih perawan tolong kasih ciri jawaban yang memang benar kalau dia itu tdk perawan lagi terima kasih	kulitdankelamin	psikologi
66	des dok kuyup bengkak akibat minyak kemiri gimana cara mengatasi x	kulitdankelamin	kulitdankelamin
67	mei dokter saya ingin bertanya saya remaja thn saya sering merasakan tekanan yg membuat sy ingin melakukan hal yg gila tp itu hanya sesaat tp setiap kali saya terdiam sendiri di kamar sy tdk mau melakukan hal apapun dan bahkan jika saya merasa tersinggung sy tdk ingin melihat org itu walaupun itu org tua sy bahkan saat ditanya sy menangis seolah tidak sadar itu sy saat di tampar pun sy tdk merasakan sakit dok sy ingin sekali sembuh bagaimana penjelasnya terimakasih	psikologi	psikologi
68	apr sakit di bagian kanan bawah payudara itu karna apa	pencernaan	kulitdankelamin
69	halo dok saya beberapa minggu ini rasa sesak saat cuaca panas dok dan kaki bengkak karena panas dok kira kira itu penyakit apa ya dok terimakasih dok	pernapasan	lainnya
70	okt dok belakangan ini sy orang nya jijik an hal tsb membuat sy tbtb ingin membuang ludah tp sy tdk bisa membuang ludah secara leluasa apalagi sedang dlm pandemi bagaimana solusinya dok sy jg mau bertanya td saya terkena debu ke mata dan air mata saya keluar dan kena masker saya terus saja jijik an sy kambuh lg karna mulut saya terkena masker saya apakah jijik an saya berlebihan	psikologi	psikologi
71	hi dok aku mau tanya udah seminggu ini tinjaku berwarna hitam terus aku juga merasakan sakit dikepala dibagian belakang dan perutku terasa kembung padahal aku tidak makan yg berlebihan kira kira aku sakit apa ya dok td nya aku pikir karena aku memakan mie jajangmyon yg berwarna hitam tetapi aku makan mie itu sudah lebih	pencernaan	pencernaan

Index	Questions	Actual label	Predicted label
	dr minggu yg lalu dr sekarang apakah memang itu berasal dr pewarna mie yg aku makan atau ada penyakit lainn dok mohon bantuannya dok		
72	haii dok saya amah mau tanya kl tenggorokan sakit itu gejalanya aph dok awalnya kk saya plg kerja batuk trs ayah saya jga batuk udh ada bln blm sembuh jga dok itu knp yh awalnya ayah saya narikin uang kematian trs keujanan dan meriang keesokkannya suaranya serak dan skrg batuk smpe skrg blm sembuh dok dan adik saya jga batuk panas dok kmrn smpe c dan saya skrg ikutan batuk jga kira itu knp dok saya takutny terkena covid amit si	pernapasan	pernapasan
73	apr assalamualaikum dok saya telat mens sudah hari dok tadi pagi tespek dan hasilnya satu tebal dan satu samar saya mau tespek lgi tp nnti nunggu beberpa hari lagi krn mungkin menurut saya kadar hcg masih rendah pertanyaan saya andaikan saya hamil bolehkah selama saya menunggu hari untuk tespek lg dan memutuskan kedokter bolehkah saya berhubungan dengan suami bahayakah buat bayinya karena saya pernah dengar dari dokternya sodara saya kalau hamil muda tdk boleh berhubngan mohon penjelasannya dok	kandungan	kandungan
74	maap dok saya mau tanya ini teh anak saya panas sma pilek batuk sma menycret udh hari tus di periksa ke bidan udah turn panas nya sekarang teh anak saya bab nya ada darah nya sedikit sedikit tp sering udah hari sama ga buang air kecil ga pipis sama sekali bahaya ga dok	lainnya	lainnya
75	mar dok saya mau tanya bagaimana cara memutih kan gigi padahal sya sering gosok gigi	gigidanmulut	gigidanmulut
76	dokter apa faktor terjadinya pernapasan	pernapasan	pernapasan
77	halo dok mau tanya anak usia bulansaya pupnya jadi agak encer tapi masih ada teksturnya tidak terlalu encer seperti air pupnya seperti itu hari kemarin awal mulanya saya beri jagung rebus untuk finger food nya dan pada saat pup masih ada butiran jagungnya karena anak saya blm tumbuh gigi sampai hari ini pupnya masih encer dan waktu bab nya normal kali sehariapakah itu termasuk diare atau hanya bab cair aja yg normal ya dok	pencernaan	pencernaan
78	sep salam dok suami sy megang v saya lalu melakukan gesek gesek jari dia di v saya tapi suami saya abis megang punya dia yg udah keluar cairan bukan sperma tp apakah dr tangan itu bs menyebabkan kehamilan dok terimakasih	kandungan	kandungan
79	agt sakit gigi boleh dicabut	gigidanmulut	gigidanmulut

Index	Questions	Actual label	Predicted label
80	jun dok bulan april tanggal saya haid dan berhenti di tanggal tetapi sampai sekarang saya tidak haid sebelum nya tanggal saya melakukan seks apakah itu bertanda hamil dok apakah ada kemungkinan saya mengalami kehamilan	kandungan	kandungan
81	sep selamat siang dok saya mau bertanyakemaren pengunci behel saya copot bagian pojok kiri lalu kawat nya bengkok apa itu tidak apa apa karena jadwal kontrol saya masih minggu lagi	gigidanmulut	gigidanmulut
82	dok umur saya th tiba di mata kaki saya muncul benjolan apakah itu dok dan apa obat nya	lainnya	lainnya
83	halo dok izin bertanya sudah seminggu ini gigi saya sakit dikarenakan berlubang jika saya lihat di google nama gigi tersebut gigi geraham kecil kedua sakit yang saya rasakan hilang timbul namun sekalinya timbul sakitnya sampai ke telinga dan itu membuat saya tidak nyaman sakit pada gigi masih bisa saya tahan rasanya sedangkan untuk telinga saya tidak kuat menahannya lebih sakit telinga daripada gigi saya yang berlubang tetapi semakin sakit gigi saya semakin sakit pula telinga saya apakah itu yang bermasalah gigi atau telinga saya ya dok apakah saya harus ke dokter gigi atau ke tht dan apakah ada obat untuk pereda sakit pada gigi dan telinga saya terimakasih	gigidanmulut	gigidanmulut
84	jan halo dok saya laki laki usia tahun saya memiliki bulu kaki yang lebat bahkan seperti orang dewasa tidak hanya di kaki tetapi di tangan dan di badan juga kadang kalau saya pakai celana panjang suka ketarik sakit tapi memang bulu kaki ayah saya memang lebat kira kira karena apa ya dok lebih baik diapakan saya agak tidak percaya diri dan selalu saya tutupi pakai celana panjang mohon bantuannya dok	kulitdankelamin	kulitdankelamin
85	garis mata tidak sejajar krna bekas bintitan cara nyembuhinnya gimana dok	lainnya	lainnya
86	dok tadi sore saya makan nasi jagung terus selang beberapa jam perut terasa tidak enak seperti orang lapar tapi kenyang terus ketika malam mual muntah dan diare bagaimana dok terimakasih	pencernaan	pencernaan
87	nov cara membesar kan mr p secara alami	kulitdankelamin	kulitdankelamin
88	mau nanya bagian pantat sebelah kanan terasa nyeri dibuat lari langsung terasa pincang dibuat jongkok juga sakit itu kenapa ya	lainnya	lainnya
89	mar hi dok saya mau nanya apakah masih bisa hamil ketika sudah mengalami menstruasi kali dalam bulan bulan lalu saya melakukan patting tepatnya tanggal february lalu tanggal saya menstruasi lalu bulan ini saya mens tanggal setelah tanggal itu saya tidak melakukan hubungan seksual lagi dok sampai haid berikutnya apakah masih bisa hamil dok	kandungan	kandungan

Index	Questions	Actual label	Predicted label
90	dok bagaimana kondisi ketika paru mengalami gangguan apakah jika saturasi oksigen kita baik oksigen dan nadi bisa meniup sesuatu dengan kuat dan normal saat melakukan nafas dalam bisa menjadi rujukan bahwa kondisi paru kita masih sehat jika mengalami apa kita harus mencari pertolongan medis	pernapasan	pernapasan
91	nov dok saya mau tanya mengapa saya merasakan benci iri dan rindu bersamaan dok apakah itu normal	psikologi	psikologi
92	dok saya kebetulan memiliki riwayat asma yang kronis apakah ini membuat saya lebih mudah tertular virus corona	pernapasan	pernapasan
93	halo dok maaf mengganggu saya memakai behel sudah masuk minggu ketiga pada awal pemakaian terasa sakit dan hilang setelah seminggu lalu kemarin breket saya lepas jadi saya kontrol dan mengganti karet juga tetapi semalam tiba tiba saja gigi saya bagian atas ujung terasa sakit ngilu dan tidak bisa menggigit padahal sebelumnya tidak sakit seperti ini	gigidanmulut	gigidanmulut
94	dok kenapa pada tenggorokan saya kerasa ada bau seperti bau bangke tikus mati dan ditenggorokan terasa panas apakah ada obat resep penyembuhan nya dok	lainnya	pernapasan
95	agt halo dokter saya mau bertanya gigi bagian atas saya setiap digunakan untuk beraktivitas seperti lompat lari bahkan berjalan selalu terasa cenat cenut kira kira itu kenapa apakah perlu dicabut gigi saya terimakasih dok	gigidanmulut	gigidanmulut
96	apakah ada jeda antara ibuprofen dan concor	lainnya	pernapasan
97	dokter saya ingin bertanya ada benjolan di atas areola tidak sakit tidak nyeri warna payudara normal tidak ada penebalan kira kira kenapa ya kalau memang kanker jika masih awal apa kita bisa menghilangkan sendiri tanpa operasi makanan apa saja yang memicu kanker makanan apa yang dibenci sel kanker terimakasih semoga dokter sehat selalu	lainnya	lainnya
98	jan sekitar minggu yg lalu stres berat dan sekrang telat haid apakah ada hubungan nya dok	kandungan	kandungan
99	apr kenapa ya dok saya kram perut dan sedikit bengkak juga sakit punggung lemas sering buang air kecil dan dada sesak	pencernaan	pencernaan