

**ANALISIS SENTIMEN BERBASIS ASPEK PADA ULASAN
APLIKASI NOVEL ONLINE DI MEDIA SOSIAL
MENGUNAKAN LATENT DIRICHLET ALLOCATION DAN
BIDIRECTIONAL ENCODER REPRESENTATION FROM
TRANSFORMERS**

SKRIPSI



Syauqatun Nabilah

11170940000052

**PROGRAM STUDI MATEMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UIN SYARIF HIDAYATULLAH JAKARTA
2022 M / 1443 H**

**ANALISIS SENTIMEN BERBASIS ASPEK PADA ULASAN
APLIKASI NOVEL ONLINE DI MEDIA SOSIAL
MENGUNAKAN LATENT DIRICHLET ALLOCATION DAN
BIDIRECTIONAL *ENCODER* REPRESENTATION FROM
TRANSFORMERS**

Skripsi

Diajukan kepada

Universitas Islam Negeri SYARIF HIDAYATULLAH JAKARTA

FAKULTAS SAINS DAN TEKNOLOGI

Untuk memenuhi Salah Satu Persyaratan dalam

Memperoleh Gelar Sarjana Matematika (S.Mat)

Oleh:

Syauqatun Nabilah

11170940000052

PROGRAM STUDI MATEMATIKA

FAKULTAS SAINS DAN TEKNOLOGI

UIN SYARIF HIDAYATULLAH JAKARTA

2022 M / 1443 H

PERSEMBAHAN DAN MOTTO

Segala puji bagi Allah dan shalawat serta salam atas Nabi Muhammad SAW.
sehingga penulis dapat menyelesaikan skripsi ini.

Skripsi ini dipersembahkan untuk kedua orang tua, keluarga dan teman-teman
penulis yang telah memberikan do'a dan dukungan berupa moral dan moril.

MOTTO

“Perlakukan orang lain sebagaimana kamu ingin diperlakukan oleh orang lain”

“Allah akan memberikan yang terbaik untuk setiap hambanya, maka jangan lupa
bersyukur di setiap harinya”

“I will, I can, I must”

ABSTRAK

Syauqatun Nabilah, Analisis Sentimen Berbasis Aspek Pada Ulasan Aplikasi Novel Online di Media Sosial Menggunakan Latent Dirichlet Allocation dan Bidirectional Encoder Representations from Transformers, dibawah bimbingan **Muhaza Liebenlito, M.Si** dan **Muhammad Manaqib, M.Sc.**

Penelitian ini membahas tentang *Aspect-Based Sentiment Analysis* yang bertujuan untuk mengevaluasi atau mengetahui kelebihan dan kekurangan berdasarkan *review* pengguna aplikasi novel online yang digunakan bahan evaluasi agar perusahaan dapat meningkatkan kualitas aplikasi. Pada penelitian ini, dilakukan klasifikasi sentimen berdasarkan aspek menggunakan metode *Bidirectional Encoder Representations from Transformers*. Penentuan jumlah aspek dilakukan dengan pemodelan topik menggunakan *Latent Dirichlet Allocation*, menghasilkan 4 aspek dominan yaitu manfaat, pelayanan, tampilan, dan sistem. Karena data yang digunakan pada penelitian ini *imbalanced* maka dilakukanlah teknik SMOTE. Untuk klasifikasi sentiment pada aspek Manfaat mendapatkan akurasi sebesar 76%, untuk teknik SMOTE diperoleh nilai akurasi (89%) dan AUC cukup baik (97%). Klasifikasi sentiment pada aspek Tampilan mendapatkan akurasi sebesar 63%, untuk teknik SMOTE diperoleh nilai akurasi (74%) dan AUC cukup baik (88%). Klasifikasi sentiment pada aspek sistem mendapatkan akurasi sebesar 64%, untuk teknik SMOTE diperoleh nilai akurasi (73%) dan AUC cukup baik (86%). Terakhir, untuk klasifikasi sentiment pada aspek sistem mendapatkan akurasi sebesar 72%, untuk teknik SMOTE diperoleh nilai akurasi (86%) dan AUC cukup baik (95%). Bentuk Trend dari tiap aspek dominan per bulan menunjukkan bentuk trend naik dan menurun di setiap bulannya pada masing masing aspek dominan dan aspek yang harus diperbaiki oleh perusahaan aplikasi novel online adalah aspek pelayanan dan manfaat. Manfaat dari penelitian ini adalah untuk mengetahui aspek dominan apa saja yang dibicarakan oleh pengguna aplikasi novel online di media sosial dan menjadi bahan evaluasi perusahaan aplikasi novel online untuk menentukan kebijakan optimal untuk meningkatkan kualitas aplikasi.

Kata Kunci: *Aspect-Based Sentiment Analysis, Bidirectional Encoder Representations from Transformers, SMOTE, Trend.*

ABSTRACT

Syauqatun Nabilah, Aspect Based Sentiment Analysis of Online Novel Applications Review On Social Media Using Latent Dirichlet Allocation and Bidirectional *Encoder* Representations from Transformers, under the guindace of **Muhaza Liebenlito, M.Si** and **Muhammad Manaqib, M.Sc.**

The research discusses Aspect-Based Sentiment Analysis which aims to evaluate or find out the advantages and disadvantages based on user reviews of online novel applications that are used as evaluation materials so that companies can improve application quality. In this study, two classifications were carried out, namely sentiment and aspect using the Bidirectional Encoder Representations from Transformers method. Determination of the number of aspects is done by modeling the topic using Latent Dirichlet Allocation, resulting in 4 dominant aspects, namely benefits, services, appearance, and system. Because the data used in this study were imbalanced, the SMOTE technique was used. For the classification of sentiment in the Benefit aspect, the accuracy is 76%, for the SMOTE technique, the accuracy value is (89%) and the AUC is quite good (97%). Sentiment classification in the aspect of display gets an accuracy of 63%, for the SMOTE technique the accuracy value is (74%) and the AUC is quite good (88%). Sentiment classification on the system aspect gets an accuracy of 64%, for the SMOTE technique the accuracy value is (73%) and the AUC is quite good (86%). Finally, for the classification of sentiment on the system aspect, the accuracy is 72%, for the SMOTE technique, the accuracy is (86%) and the AUC is quite good (95%). The trend form of each dominant aspect per month shows an increasing and decreasing trend in each month in each dominant aspect and the aspect that must be improved by the online novel application company is the service and benefit aspect. The benefit of this research is to find out what dominant aspects are discussed by online novel application users on social media and to be used as evaluation material for online novel application companies to determine optimal policies to improve application quality.

Keywords: Aspect-Based Sentiment Analysis, Bidirectional *Encoder* Representation of Transformers, SMOTE, Trend.

KATA PENGANTAR

Assalamu'alaikum Wr. Wb.

Alhamdulillah, Puji serta syukur penulis panjatkan kehadirat Allah SWT yang telah memberikan taufik, hidayah dan inayah-Nya, sehingga peneliti dapat melaksanakan dan menyelesaikan penelitian ini. Shalawat serta salam peneliti curahkan kepada junjungan Nabi Muhammad SAW dan para sahabat serta kerabatnya.

Penulisan penelitian ini berguna sebagai syarat untuk memperoleh gelar sarjana Matematika, pada program studi S-1 Matematika Fakultas Sains dan Teknologi Universitas Islam Negeri Syarif Hidayatullah Jakarta. Dalam penulisan ini terdapat sedikit banyak kesulitan dan hambatan, tetapi terdapat pula pihak-pihak yang telah membantu peneliti dalam bentuk do'a, bimbingan, serta dukungan. Maka dari itu, peneliti ingin menyampaikan rasa terima kasih kepada:

1. Ir. Nashrul Hakiem, S.Si., M.T., Ph.D selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Syarif Hidayatullah Jakarta.
2. Ibu Dr. Suma'innah, M.Si, selaku Ketua Program Studi Matematika Fakultas Sains dan Teknologi Universitas Islam Negeri Syarif Hidayatullah Jakarta.
3. Bapak Muhaza Liebenlito, M.Si selaku dosen pembimbing I dan Bapak Muhammad Manaqib, M.Sc selaku dosen pembimbing II atas segala ilmu dan saran kepada peneliti selama proses penyusunan skripsi ini.
4. Bapak Dr. Taufik Edy Sutanto, M.Sc.Tech selaku dosen penguji I dan Bapak Ary Santoso, M.Si selaku dosen penguji II atas segala saran dan kritiknya kepada penulis, serta kesediaan waktunya untuk menghadiri seminar hasil dan siding skripsi.
5. Kedua orang tua, kakak serta kerabat keluarga lainnya yang tiada hentinya memberikan dukungan, semangat serta do'a sehingga peneliti dapat menyelesaikan skripsi ini.

6. Teman-teman terdekat matematika di Arzalia, Adeline, Intan yang selalu memberikan semangat dan berdiskusi dan memberi saran selama proses pembuatan skripsi.
7. Teman-teman Matematika 2017 yang tidak bisa disebutkan satu – persatu.
8. Seluruh pihak yang secara langsung ataupun tidak langsung telah membantu, mendoakan dan mendukung peneliti dalam menyelesaikan skripsi ini.
9. *The last but not least, I wanna thank me. Thank your for being you*, Nabilah.

Peneliti menyadari bahwa masih banyak terdapat kesalahan dalam penyusunan skripsi ini. Oleh karena itu, peneliti berharap adanya penyampaian kritik dan saran demi penyempurnaan skripsi ini. Semoga skripsi ini dapat bermanfaat untuk banyak pihak.

Wassalamu'alaikum Wr. Wb.

DAFTAR ISI

PERSEMBAHAN DAN MOTTO	ii
ABSTRAK	iii
ABSTRACT	iv
KATA PENGANTAR.....	vi
DAFTAR ISI	viii
DAFTAR TABLE	xi
DAFTAR GAMBAR.....	xiii
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	4
1.3. Batasan Masalah.....	5
1.4. Tujuan Masalah	5
1.5. Manfaat Masalah	5
BAB II LANDASAN TEORI.....	7
2.1. Aspect-Based Sentiment Analysis (Analisis Sentimen Berbasis Aspek). 7	
2.2. <i>Neural Network</i>	7
2.3. Fungsi Aktivasi.....	9
2.4. Lapisan Fully Connected	10
2.6. <i>Adaptive Moment Estimation Algorithm (ADAM)</i>	11
2.7. Teknik Pengambilan Data	12
2.8. <i>Preprocessing</i>	12
2.9. <i>Vector Space Model</i>	13

2.10.	<i>Hyperparameters</i>	14
2.11.	<i>WordPiece Tokenization</i>	15
2.12.	<i>Principal Component Analysis</i>	16
BAB III METODE PENELITIAN		17
3.1.	Sumber Data	17
3.2.	Alur Penelitian.....	18
3.3.	Latent Dirichlet Allocation (LDA)	21
3.4.	BERT (<i>Bidirectional Encoder Representations from Transformers</i>)	26
3.4.1.	Penjelasan BERT	27
3.4.2.	Representasi <i>Input</i> BERT.....	33
3.5.	Evaluasi Model.....	41
3.6.	Kurva ROC dan AUC.....	42
BAB IV PEMBAHASAN		45
4.1.	Kondisi Data.....	45
4.2.	Hasil Preprocessing	48
4.3.	BoW.....	49
4.4.	Penentuan Jumlah Aspek.....	50
4.5.	Hasil Klasifikasi Sentimen Berbasis Aspek	53
4.5.1.	Hasil Klasifikasi Sentimen Pada Aspek Manfaat.....	54
4.5.2.	Hasil Klasifikasi Sentimen Pada Aspek Tampilan.....	60
4.5.3.	Hasil Klasifikasi Sentimen Pada Aspek Sistem	67
4.5.4.	Hasil Klasifikasi Sentimen Pada Aspek Pelayanan	74
4.6.	Visualisasi dan Interpretasi Data	81
4.7.	Analisis Trend per Bulan.....	85
BAB V PENUTUP		91

5.1. Kesimpulan.....	91
5.2. Saran.....	92
Daftar Pustaka	94
Lampiran.....	98

DAFTAR TABLE

Tabel 2. 1 Pembuatan Dictionary dengan Fungsi doc2bow	14
Tabel 2. 2 Proses doc2bow	14
Tabel 3. 1 Tabel Awal Hasil Scrapping.....	17
Tabel 3. 2 Contoh LDA untuk Dokumen, Token dan Kata Unik	23
Tabel 3. 3 Hasil Sebaran Kata Unik di Setiap Topik	23
Tabel 3. 4 Topik Setiap Kata Unik dalam Masing-Masing Kalimat.....	24
Tabel 3. 5 Jumlah Kata Unik di Setiap Topik pada Masing-Masing Dokumen...	24
Tabel 3. 6 Peluang Topik di setiap Dokumen	25
Tabel 3. 7 Peluang Kata di setiap Topik.....	25
Tabel 3. 8 Confusion matrix	42
Tabel 3. 9 Rumus Akurasi, Precision, dan Recall	42
Tabel 3. 10 Kategori Klasifikasian Model Berdasarkan Nilai AUC	44
Tabel 4. 1 Contoh Beberapa Ulasan yang dihapus.....	45
Tabel 4. 2 Jumlah Ulasan Tiap Rating	46
Tabel 4. 3 Beberapa Ulasan Awal dan Hasil Processing.....	48
Tabel 4. 4 Bentuk Bag of Words (BoW).....	49
Tabel 4. 5 Penentuan jumlah Aspek	50
Tabel 4. 6 Nilai Principal Components Untuk Setiap Cluster Topik	52
Tabel 4. 7 Sebaran Kata Tiap Cluster.....	53
Tabel 4. 8 Hasil Confusion Matrix Sentimen pada Aspek Manfaat dengan learning rate 10^{-5}	55
Tabel 4. 9 Evaluasi Confusion Matrix Sentimen pada Aspek Manfaat dengan learning rate $2 \cdot 10^{-5}$	57
Tabel 4. 10 Evaluasi Confusion Matrix Sentimen pada Aspek Manfaat dengan learning rate $3 \cdot 10^{-5}$	58
Tabel 4. 11 Perbandingan Nilai Akurasi dari Setiap Learning Rate pada Aspek Manfaat	58
Tabel 4. 12 Evaluasi Confusion Matrix Sentimen pada Aspek Manfaat dengan Teknik SMOTE.....	59

Tabel 4. 13 Evaluasi Confusion Matrix Sentimen pada Aspek Tampilan dengan learning rate 10^{-5}	62
Tabel 4. 14 Evaluasi Confusion Matrix Sentimen pada Aspek Tampilan dengan learning rate $2 \cdot 10^{-5}$	63
Tabel 4. 15 Evaluasi Confusion Matrix Sentimen pada Aspek Tampilan dengan learning rate $2 \cdot 10^{-5}$	65
Tabel 4. 16 Perbandingan Nilai Akurasi dari Setiap Learning Rate pada Aspek Tampilan	65
Tabel 4. 17 Evaluasi Confusion Matrix Sentimen pada Aspek Tampilan dengan Teknik SMOTE.....	66
Tabel 4. 18 Evaluasi Confusion Matrix Sentimen pada Aspek Sistem dengan learning rate 10^{-5}	69
Tabel 4. 19 Evaluasi Confusion Matrix Sentimen pada Aspek Sistem dengan learning rate $2 \cdot 10^{-5}$	70
Tabel 4. 20 Evaluasi Confusion Matrix Sentimen pada Aspek Sistem dengan learning rate $3 \cdot 10^{-5}$	71
Tabel 4. 21 Perbandingan nilai akurasi dari setiap learning rate.....	72
Tabel 4. 22 Evaluasi Confusion Matrix Sentimen pada Aspek Sistem dengan Teknik SMOTE.....	73
Tabel 4. 23 Evaluasi Confusion Matrix Sentimen pada Aspek Pelayanan dengan learning rate 10^{-5}	75
Tabel 4. 24 Evaluasi Confusion Matrix Sentimen pada Aspek Pelayanan dengan learning rate $2 \cdot 10^{-5}$	77
Tabel 4. 25 Evaluasi Confusion Matrix Sentimen pada Aspek Pelayanan dengan learning rate $3 \cdot 10^{-5}$	78
Tabel 4. 26 Perbandingan Nilai Akurasi dari Setiap Learning Rate pada Aspek Pelayanan	78
Tabel 4. 27 Evaluasi Confusion Matrix Sentimen pada Aspek Pelayanan dengan Teknik SMOTE.....	79

DAFTAR GAMBAR

Gambar 2. 1 Arsitektur Feed-forward Network [11].....	8
Gambar 2. 2 Arsitektur Recurrent Neural Network [11].....	9
Gambar 2. 3 Lapisan Fully Connected [17]	10
Gambar 2. 4 Contoh Kalimat yang Menggunakan Tokenisasi Wordpiece	15
Gambar 3. 1 Alur Penelitian	19
Gambar 3. 2 Representasi Model LDA [26].....	21
Gambar 3. 3 Proses Self Attention Layer [32]	29
Gambar 3. 4 Proses Encoder [32].....	29
Gambar 3. 5 Proses Masked Language Modelling [29]	31
Gambar 3. 6 Proseses Training BERT [29].....	31
Gambar 3. 7 Representasi Input BERT [29].....	32
Gambar 3. 8 Ilustrasi Fine-tuning pada Single Sentence [29]	33
Gambar 3. 9 Ilustrasi proses Pre-training dan Fine-tuning [29]	33
Gambar 3. 10 Tahap Tokonisasi Menggunakan WordPiece	35
Gambar 3. 11 Tahap menjadi Token Embedding.....	35
Gambar 3. 12 Tahapan Pemberian Token Padding	36
Gambar 3. 13 Indeks pada Vocabulary Model	36
Gambar 3. 14 Tahap Substitusi ID dari Setiap Token.....	37
Gambar 3. 15 Tahap Sentence Embedding	37
Gambar 3. 16 Tahap Positional Embedding.....	38
Gambar 3. 17 Proses Tokenisasi BERT	38
Gambar 3. 18 Representasi Input dan Output pada Tokenisasi BERT.	38
Gambar 3. 19 Input dan Output pada Model BERT.....	39
Gambar 3. 20 Ilustrasi layer untuk Analisis Klasifikasi Sentimen.....	40
Gambar 3. 21 Ilustrasi Proses Klasifikasi BERT	41
Gambar 4. 1 Grafik Jumlah Ulasan Tiap Bulan	46
Gambar 4. 2 Jumlah Sentimen Tiap Aspek	47
Gambar 4. 3 Jumlah Label Sentimen.....	47
Gambar 4. 4 Jumlah Label Aspek	47

Gambar 4. 5 Visualisasi Visualisasi pyLDAvis 20 topik	51
Gambar 4. 6 Grafik Nilai Coherence untuk iterasi 1-5 Aspek	52
Gambar 4. 7 Visualisasi PyLDAvis num topic 4	52
Gambar 4. 8 Jumlah Ulasan Per Kelas pada Aspek Manfaat	54
Gambar 4. 9 Hasil Confusion Matrix Sentimen pada Aspek Manfaat dengan learning rate 10^{-5}	55
Gambar 4. 10 Hasil Confusion Matrix Sentimen pada Aspek Manfaat dengan learning rate $2 \cdot 10^{-5}$	56
Gambar 4. 11 Hasil Confusion Matrix Sentimen pada Aspek Manfaat dengan learning rate $3 \cdot 10^{-5}$	57
Gambar 4. 12 kurva ROC dan AUC Aspek Manfaat menggunakan Teknik SMOTE	60
Gambar 4. 13 jumlah ulasan per kelas pada aspek tampilan	60
Gambar 4. 14 Hasil Confusion matrix Sentimen pada Aspek Tampilan dengan learning rate 10^{-5}	61
Gambar 4. 15 Hasil Confusion Matrix Sentimen pada Aspek Tampilan dengan learning rate $2 \cdot 10^{-5}$	63
Gambar 4. 16 Hasil Confusion Matrix Sentimen pada Aspek Tampilan dengan learning rate $2 \cdot 10^{-5}$	64
Gambar 4. 17 kurva ROC dan AUC Aspek Tampilan menggunakan Teknik SMOTE	67
Gambar 4. 18 jumlah ulasan per kelas pada aspek sistem.....	67
Gambar 4. 19 Hasil Confusion matrix Sentimen pada Aspek Sistem dengan learning rate 10^{-5}	68
Gambar 4. 20 Hasil Confusion Matrix Sentimen pada Aspek Sistem dengan learning rate $2 \cdot 10^{-5}$	69
Gambar 4. 21 Hasil Confusion Matrix Sentimen pada Aspek Sistem dengan learning rate $3 \cdot 10^{-5}$	71
Gambar 4. 22 kurva ROC dan AUC Aspek Sistem menggunakan Teknik SMOTE	73
Gambar 4. 23 Jumlah Ulasan per Kelas pada Aspek Pelayanan	74

Gambar 4. 24 Hasil Confusion Matrix Sentimen pada Aspek Pelayanan dengan learning rate 10^{-5}	75
Gambar 4. 25 Hasil Confusion Matrix Sentimen pada Aspek Pelayanan dengan learning rate $2 \cdot 10^{-5}$	76
Gambar 4. 26 Hasil Confusion Matrix Sentimen pada Aspek Pelayanan dengan learning rate $3 \cdot 10^{-5}$	77
Gambar 4. 27 Kurva ROC dan AUC Aspek Pelayanan Menggunakan Teknik SMOTE	80
Gambar 4. 28 Wordcloud Aspek Manfaat dengan Sentimen (a) Negatif (b) Netral (c) Positif.....	81
Gambar 4. 29 Wordlink Aspek Manfaat dengan Sentimen (a) Negatif (b) Netral (c) Positif.....	81
Gambar 4. 30 Wordcloud Aspek Pelayanan dengan sentimen (a) Negatif (b) Netral (c) Positif.....	82
Gambar 4. 31 Wordlink Aspek Pelayanan dengan sentimen (a) Negatif (b) Netral (c) Positif.....	82
Gambar 4. 32 Wordcloud Aspek Tampilan dengan sentimen (a) Negatif (b) Netral (c) Positif.....	83
Gambar 4. 33 Wordlink Aspek Tampilan dengan sentimen (a) Negatif (b) Netral (c) Positif.....	84
Gambar 4. 34 Wordcloud Aspek Sistem dengan sentimen (a) Negatif (b) Netral (c) Positif.....	84
Gambar 4. 35 Wordlink Aspek Sistem dengan sentimen (a) Negatif (b) Netral (c) Positif	85
Gambar 4. 36 Trend Kelas Aspek per Bulan.....	86
Gambar 4. 37 Trend Kelas Sentimen per Bulan.....	87
Gambar 4. 38 Visualisasi Jumlah Data Kelas Sentimen Tiap Aspek (a) Positif (b) Netral dan (c) Negatif.....	88
Gambar 4. 39 Visualisasi Selisih Jumlah Data Sentimen Positif dan Negatif	89

BAB I

PENDAHULUAN

1.1. Latar Belakang

Dalam Al-qur'an telah diperintahkan bahwa:

اقْرَأْ بِاسْمِ رَبِّكَ الَّذِي خَلَقَ ۝ ۱ خَلَقَ الْإِنْسَانَ مِنْ عَلَقٍ ۝ ۲ اقْرَأْ وَرَبُّكَ الْأَكْرَمُ ۝ ۳
الَّذِي عَلَّمَ بِالْقَلَمِ ۝ ۴ عَلَّمَ الْإِنْسَانَ مَا لَمْ يَعْلَمْ ۝ ۵

“Bacalah dengan (menyebut) nama Tuhanmu yang Menciptakan (1) Dia telah menciptakan manusia dari segumpal darah (2) Bacalah, dan Tuhanmulah yang Maha pemurah (3) yang mengajar (manusia) dengan perantaraan kalam (4) Dia mengajar kepada manusia apa yang tidak diketahuinya (5)”. (QS. Al-‘Alaq[96] : 1-5). Berdasarkan ayat tersebut, manusia telah diperintahkan untuk membaca, tentu dimaknai bukan hanya sebatas membaca lembaran-lembaran yang terdapat di buku, melainkan juga “buku” dunia.

Pada saat ini teknologi sudah berkembang dengan pesat yang memudahkan masyarakat dalam berbagai hal, misalnya transaksi, gaya hidup dan hobi yang bisa dipermudah dengan kemajuan teknologi. Segala kebutuhan masyarakat bisa didapat dalam satu genggam, termasuk masyarakat yang mempunyai hobi menulis atau membaca. Dengan adanya novel online yang dapat diakses melalui website atau aplikasi, maka dapat mempermudah masyarakat tanpa harus keluar rumah untuk membeli novel di toko buku. Selain efisien waktu dan tenaga, lebih murah, anti rusak, ramah lingkungan dan tidak perlu takut kehabisan stok karena novel online akan selalu *ready stock* tanpa perlu cek toko buku mana yang masih menyediakan buku, itulah membuat masyarakat senang membaca buku atau novel online [1]. Namun, kelemahan dari aplikasi online adalah performa aplikasi yang dimiliki aplikasi terkadang membuat pengguna merasa kecewa karena tidak sesuai dengan apa yang diharapkan pengguna.

Pada saat ini ulasan pada media sosial banyak digunakan oleh masyarakat untuk mencurahkan atau mereview aplikasi yang sedang digunakan oleh

pengguna. Dengan bertambahnya pengguna baru yang mengunduh aplikasi novel berkaitan juga dengan bertambahnya ulasan pengguna yang ada pada media sosial. Ulasan tersebut sangat berguna karena dapat membangun kepercayaan pengguna lainnya. Pada penelitian ini, peneliti menggunakan data ulasan pada aplikasi novel online berbahasa Indonesia sebagai data penelitian yang sebagai sasarannya adalah kaum anak muda yang menggunakan aplikasi tersebut, yang di mana biasanya mempunyai hobi membaca atau menulis novel yang dapat mengembangkan bakat kemampuan dan dapat mengangkat penulis menjadi penulis buku best seller. Banyak perusahaan yang bersaing untuk membuat aplikasi novel online yang terbagus menurut masing-masing perusahaan. Namun, setiap aplikasi novel online pasti memiliki keunggulan dan kelemahan tersendiri. Pengguna aplikasi novel online akan mengeluarkan ulasan atau opini mereka sebagai pengguna aplikasi yang dituangkan pada kolom komentar di media sosial, baik itu ulasan positif, netral atau negatif.

Ulasan yang terdapat di media sosial memiliki rating dari 1 sampai 5. Namun, tidak sedikit pengguna yang hanya memberikan rating tidak sesuai atau tanpa adanya ulasan komentar atau hanya berupa komentar tanpa adanya rating sehingga hal tersebut tidak cukup menggambarkan bagusnya aplikasi. Ulasan yang berupa kalimat dan terdapat rating lebih dapat menggambarkan kualitas dari aplikasi yang diberikan oleh pengguna. Hal ini dapat memungkinkan bahwa ulasan yang tertulis dapat mempengaruhi pengguna baru untuk memilih antara mengunduh aplikasi novel online atau tidak [2]. Sentimen positif, netral atau negatif masyarakat bisa merujuk ke beberapa aspek pada aplikasi. Misalnya, masyarakat senang dengan aplikasi novel online karena mempunyai tampilan yang millennial walaupun terdapat banyak iklan yang mengganggu. Maka dengan adanya permasalahan tersebut dapat di bangun sebuah sistem yang dapat dilakukan yaitu analisis sentimen. Analisis sentimen merupakan sebuah cara riset komputasional dari opini, sentimen dan emosi yang diekspresikan secara tekstual [3].

Salah satu upaya untuk mengetahui kelebihan dan kekurangan biasa disebut dengan *Aspect-Based Sentiment Analysis* (ABSA). Penelitian terdahulu yang dilakukan oleh Tun Thura Thet, Jin-Cheon Na dan Christopher S.G. Khoo

pada tahun 2010 tentang data ulasan film pada forum diskusi [4]. Penelitian yang dilakukan oleh [4] menjelaskan tentang sentimen dan keseluruhan aspek film seperti sutradara, pemeran, jalan cerita, dan lain-lain. Contoh kalimat yang menjadi penelitian adalah “Saya suka musiknya tetapi tidak dengan jalan ceritanya”. Pada kalimat tersebut mengandung dua sentimen dan dua aspek yang berbeda yaitu jalan cerita dan musik. Kemudian dilakukan klasifikasi sentimen dengan mengelompokkan kata negatif dan positif, dengan cara melihat skor rata-rata dari semua kata yang diproses menggunakan teknik SentiWordNet. Proses klasifikasi aspek film menggunakan tag semantik yang diperjelas oleh anotasi semantik.

Penelitian serupa juga dilakukan oleh Reza Amalia Priyantina dan Riyanarto Sarno pada tahun 2019 tentang analisis sentimen dari ulasan Hotel [5]. Penelitian ini menentukan sentimen ulasan berdasarkan aspek hotel yaitu lokasi, makan, pelayanan, kenyamanan dan kebersihan. Pertama, mereka menggunakan *Latent Dirichlet Allocation* (LDA) menentukan topik tersembunyi dari daftar istilah, kemudian *Semantic Similarity* mengkategorikan daftar istilah berdasarkan topik yang dihasilkan oleh *Latent Dirichlet Allocation* (LDA) menjadi lima aspek dari sebuah hotel. Kemudian dalam menghitung ukuran kemiripan, istilah daftar diperluas dengan menggunakan metode TF-ICF (*Term Frequency-Inverse Cluster Frequency*). Kedua, klasifikasi sentimen dilakukan dengan menggunakan kombinasi *Word Embedding* dan *Long-short Term Memory*. Hasil penelitian menunjukkan bahwa metode yang digunakan dapat mengklasifikasikan ulasan ke dalam lima aspek hotel. Performa klasifikasi aspek menggunakan LDA + TF-ICF + *Semantic Similarity* sebesar 85%; klasifikasi sentimen terhadap analisis sentimen berbasis aspek menggunakan *Word Embedding* + LSTM sebesar 93%. Penelitian yang dilakukan oleh Fu Xianghua, Liu Guo, Guo Yanyan, Wang Zhiqiang dengan objek penelitian ulasan Blog berbahasa China [6]. Mereka menggunakan *Latent Dirichlet Allocation* (LDA) untuk menentukan global topik, setelah itu mencari *local topic* atau topik yang lebih terperinci dengan menggunakan metode *Sliding Windows* yang dihitung menggunakan *HowNet Lexicon*. Proses sentimen dari *local topic* menggunakan metode BMW-DASM (*Benchmark Weighted Method + Direct*

Average Sum Method). Hasil akurasi yang didapat pada penelitian ini sebesar 92.15%.

Pada penelitian ini peneliti akan melakukan dua tahapan. Pertama, peneliti melakukan clustering topik/aspek menggunakan metode LDA seperti yang dilakukan pada penelitian [5] untuk menentukan jumlah topik yang digunakan dan menjadi referensi pada saat pelabelan manual. Kedua, melakukan klasifikasi sentiment dan aspek dengan menggunakan metode *Bidirectional Encoder Representations from Transformers* (BERT). Metode ini dipilih karena pada saat *fine-tuning* model telah dilatih sebelumnya dan hanya belajar sedikit untuk mendapat titik optimal pada *task* yang baru. Model BERT telah dilatih menggunakan 4 miliar kata dengan sekitar 250 juta Bahasa Indonesia dengan kalimat formal dan non formal atau bahasa sehari-hari [7]. Pelabelan pada ulasan dilakukan dengan cara melihat *rating* yang diberikan pengguna dan pada pemberian label aspek dilakukan dengan cara manual. Karena data yang digunakan tidak seimbang antara ulasan positif, netral dan negatif, maka peneliti melakukan resampling data dengan cara teknik oversampling dan undersampling. Lalu, hasil sentimen setiap aspek direpresentasikan menggunakan *wordcloud* dan *wordlink*. Maka penelitian ini akan melakukan penelitian tentang Analisis Sentimen Berbasis Aspek Pada Ulasan Aplikasi Novel *Online* di Media Sosial Menggunakan *Latent Dirichlet Allocation*) dan *Bidirectional Encoder Representations from Transformers*.

1.2. Rumusan Masalah

Berdasarkan latar belakang yang telah dijabarkan diatas, maka didapatkan perumusan masalah sebagai berikut:

1. Apa saja aspek dominan yang termuat di dalam ulasan aplikasi novel *online* di media sosial?
2. Seberapa optimal model ABSA yang digunakan pada metode *Bidirectional Encoder Representations from Transformers* untuk mengklasifikasi teks berbahasa Indonesia pada aplikasi novel *online* di media sosial.

3. Apa saja kelebihan dan kekurangan di setiap aspek yang terdapat pada aplikasi novel online?
4. Bagaimana hasil analisis trend per bulan pada tiap aspek dominan?

1.3. Batasan Masalah

Dalam penelitian ini, penulis memberikan batasan masalah agar lebih terarah. Batasan masalah tersebut sebagai berikut:

1. Data ulasan yang digunakan adalah data ulasan teks berbahasa Indonesia dari aplikasi novel *online* di media sosial dengan cara teknik web scraping dari tanggal 16 September 2020 06:49 sampai 14 Maret 2021 15:34.
2. Data ulasan aplikasi novel *online* dibagi menjadi tiga jenis klasifikasi sentimen, yaitu: positif, netral dan negatif.
3. Urutan data yang diambil berdasarkan yang paling bermanfaat.
4. Model BERT yang digunakan pada penelitian ini adalah model BERT_{BASE}.

1.4. Tujuan Masalah

Tujuan penulisan penelitian ini antara lain:

1. Mengetahui aspek dominan yang termuat di dalam ulasan aplikasi novel *online* di media sosial.
2. Mengetahui seberapa optimal model ABSA yang digunakan pada metode *Bidirectional Encoder Representations from Transformers* untuk mengklasifikasi teks berbahasa Indonesia pada aplikasi novel *online di media sosial*.
3. Mengetahui kelebihan dan kekurangan di setiap aspek yang terdapat pada aplikasi novel online.
4. Mengetahui hasil analisis trend per bulan pada tiap aspek dominan.

1.5. Manfaat Masalah

Melalui penelitian ini diharapkan bermanfaat untuk masyarakat yang akan mengunduh aplikasi untuk meninjau aplikasi novel *online* sebelum mengunduhnya,

dapat dijadikan bahan evaluasi untuk perusahaan aplikasi novel *online* tentang hal apa saja yang menjadi kelemahan dan kelebihan agar dapat meningkatkan kualitas aplikasi dan mempromosikan aplikasi yang lebih baik lagi.

BAB II

LANDASAN TEORI

2.1. Aspect-Based Sentiment Analysis (Analisis Sentimen Berbasis Aspek)

Analisis sentimen adalah salah satu cabang penelitian dari *text mining* yang bertujuan untuk mengklasifikasikan opini seseorang yang berupa dokumen *text* berdasarkan sentimen. Para peneliti terdahulu telah menemukan penelitian terbaru berupa *aspect-based sentiment analysis* yang merupakan analisis sentimen yang lebih mendalam pada sebuah *text* ulasan. Misalnya pada saat melihat ulasan dari sebuah hotel, opini yang dikeluarkan oleh masyarakat tidak hanya menyeluruh tentang sentimen melainkan terdapat pula aspek yang dikeluarkan tentang lokasi, pelayanan, makanan, kenyamanan dan lain-lainnya [5]. Pada dasarnya analisis sentimen berbasis aspek dapat mengidentifikasi aspek dan mengidentifikasi sentimen dari aspek tersebut [8]. Pada penelitian ini, peneliti akan melakukan dua klasifikasi, yaitu klasifikasi sentimen dan aspek.

2.2. Neural Network

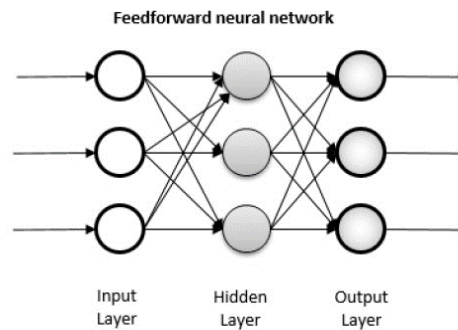
Neural Network atau Jaringan syaraf tiruan merupakan teknik pembelajaran mesin yang terinspirasi dari cara kerja mekanisme sistem saraf manusia. Jaringan ini dikenal sebagai arsitektur di mana lapisan yang berbeda saling terhubung. Lapisan antara *input* dan *output* disebut *hidden layers* (lapisan tersembunyi), dan *output* dari *hidden layers* (lapisan tersembunyi) adalah *hidden units* [9].

Inti dari *neural network* adalah *hidden layer* yang terdiri dari *hidden units*, di mana setiap *hidden unit* adalah unit saraf yang mengambil jumlah pembobot dari *inputnya* dan kemudian menerapkan nonlinier. Setiap unit di tiap lapisan akan menerima *input* dan *output* dari lapisan sebelumnya, dan hubungan antar tiap pasang unit di dua lapisan saling berdekatan, sehingga setiap lapisan akan saling terhubung. Setiap *hidden units* akan merangkum semua *input unit* [10].

Secara umum, ada dua jenis arsitektur jaringan saraf tiruan, yaitu:

1. *Feed-forward Network*

Feed-forward Network merupakan jaringan yang paling sederhana yang di mana informasi mengalir hanya dalam satu jalur dari node *input* menuju node *output* tanpa adanya *loop* [11]. Hal ini memungkinkan jaringan untuk beroperasi pada *input* yang berukuran tetap atau *input* dengan panjang variabel yang dapat diabaikan pada urutan elemen. *Hidden layer* dan lapisan *output* saling terhubung dengan lapisan sebelumnya. Arsitektur *feed-forward network* terdiri dari beberapa layer seperti: *input layer*, *hidden layer* dan *output layer* yang dapat di ilustrasikan sebagai berikut:



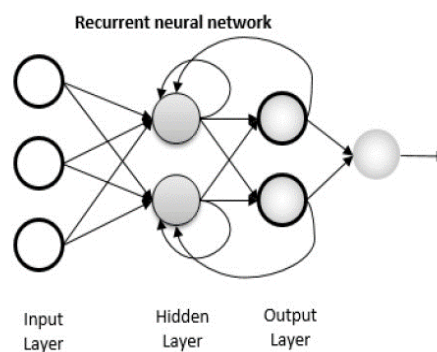
Gambar 2. 1 Arsitektur Feed-forward Network [11]

Contoh jenis jaringan dari *feed-forward network* adalah *Convolutional Neural Network* yang sering digunakan untuk pengenalan gambar. Misalkan Pada penelitian [12] tentang pengenalan dan deteksi citra makanan menggunakan metode *Convolutional Neural Network* (CNN) yang menggunakan 1234 gambar makanan dan 1980 gambar non-makanan dibagi menjadi 10 kelompok. Pembagian data dengan 8 kelompok digunakan untuk pelatihan, 1 untuk validasi dan 1 untuk pengujian. Menghasilkan nilai akurasi sebesar 93,8%.

2. *Recurrent Neural Network* (RNN)

Recurrent Neural Network merupakan jenis jaringan pada *neural network* dimana terdapat *loop* sebagai koneksi umpan balik dalam jaringan. Pola *looping* pada RNN berguna untuk menyimpan ingatan sementara yang nantinya akan dipakai pada proses selanjutnya atau nilai dari neuron-neuron pada hidden layer sebelumnya akan digunakan sebagai input [13]. Jenis data

yang bisa yang biasa digunakan pada metode RNN adalah data teks atau suara [14]. RNN mengambil kembali *hidden unit* pada lapisan sebelumnya sebagai *input* dan akan disimpan pada *hidden layer* yang akan terus di *update* hingga menghasilkan *vector* yang ukuran tetap yang merangkum urutan. Jaringan ini lebih kuat dan canggih daripada CNN dan lebih efisien dalam analisis sentimen. Dalam setiap pemrosesan, *output* yang dikeluarkan tidak hanya fungsi dari unit itu sendiri, tetapi juga dari hasil pemrosesan sampel sebelumnya (atau sampel berikutnya dalam *bidirectional* RNN)



Gambar 2. 2 Arsitektur *Recurrent Neural Network* [11]

[14].

Contoh pada penelitian [15] tentang metode pembelajaran untuk mengenali emosi dari sinyal EEG menggunakan *Long-Short Term Memory* (LSTM). *Long-Short Term Memory* (LSTM) digunakan untuk mempelajari fitur-fitur dari sinyal EEG kemudian lapisan akan mengklasifikasikan fitur-fitur ini ke dalam gairah rendah/tinggi, valensi, dan kesukaan. Dataset DEAP digunakan untuk memverifikasi metode ini yang memberikan akurasi rata-rata untuk kelas gairah sebesar 85,65%, kelas valensi sebesar 85,45%, dan kelas kesukaan sebesar 87,99%.

2.3. Fungsi Aktivasi

Setiap lapisan diikuti oleh sebuah fungsi aktivasi, yang merupakan suatu fungsi matematika yang dapat digunakan untuk memetakan *output* dari satu lapisan ke lapisan berikutnya. Contoh dari fungsi aktivasi adalah fungsi softmax. Fungsi aktivasi softmax harus memastikan bahwa *vector output* berjumlah 1. Kemudian

simpul *output* dengan probabilitas tertinggi dipilih sebagai label prediktif untuk kalimat *input* [16]. Fungsi aktivasi softmax dapat digunakan pada jaringan yang membutuhkan *output* dengan label yang telah ditentukan. Rumus untuk fungsi softmax dirumuskan sebagai berikut:

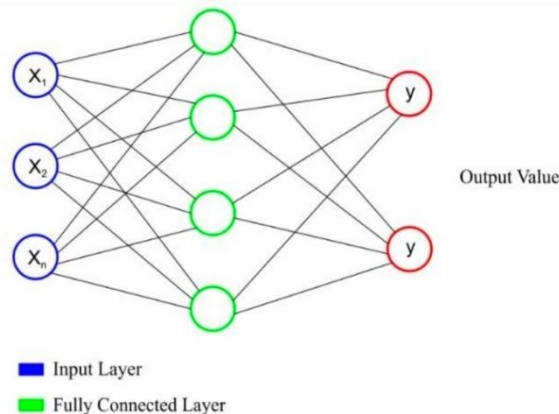
$$\text{softmax}(z_i) = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}} \text{ dengan } i = 1, 2, \dots, k \quad (2.1)$$

dimana:

1. $z = (z_1, z_2, \dots, z_k) \in \mathbb{R}^K$ ialah vektor *input* ke fungsi softmax atau vektor *output* dari lapisan terakhir yang biasa disebut dengan *logits*.
2. e^{z_i} ialah eksponensial dari tiap elemen dari *input* vektor.
3. $\sum_{j=1}^K e^{z_j}$ ialah proses normalisasi untuk semua *output* softmax berjumlah tepat 1 dan masing-masing nilai probabilitasnya berada pada interval (0,1).

2.4. Lapisan Fully Connected

Lapisan *fully connected* ialah lapisan terakhir dalam arsitektur *Deep Learning* atau lapisan yang di mana semua neuron aktif di lapisan sebelumnya terhubung ke semua neuron yang ada di lapisan berikutnya. Sebelum dihubungkan dengan semua neuron, setiap aktivasi di lapisan sebelumnya harus ditransformasi menjadi bentuk data satu dimensi [17], seperti Gambar 2. 3 *Lapisan Fully Connected*. Ada beberapa parameter yang dimiliki *fully connected*, yaitu jumlah fungsi *unit activation*, dan fungsi *loss*. Penelitian ini menggunakan jumlah *unit activation* sebanyak 3 karena ada penilaian untuk sentimen positif, netral, dan negatif.



Gambar 2. 3 Lapisan *Fully Connected* [17]

2.5. *Loss Function*

Ketika proses *training* selesai, jaringan akan menampilkan hasil nilai *loss*. Nilai *loss* akan terus berkurang setiap iterasi, jika tidak berarti jaringan tidak mempelajari proses yang telah dilalui. Untuk menghindari itu, diperlukan suatu *loss function* yang berguna untuk mengatur *output* dari jaringan saraf tiruan agar sesuai dengan yang diinginkan (optimisasi) penelitian. Hasil dari fungsi *loss* digunakan sebagai sinyal umpan balik yang menyesuaikan nilai bobot untuk mengurangi *loss score*. Salah satu fungsi *loss* yang biasa digunakan adalah *Cross-entropy loss*, yang dikombinasikan dengan *softmax* [18]. *Cross-entropy loss* digunakan untuk mengukur kinerja model klasifikasi *multiclass* yang *outputnya* berupa nilai probabilitas antara 0 dan 1 [19].

2.6. *Adaptive Moment Estimation Algorithm (ADAM)*

Adam atau *Adaptive Moment Estimation Algorithm* adalah salah satu algoritma Adam umum digunakan. Fungsi dari Algoritma Adam adalah untuk menghitung estimasi atau perkiraan momen pada saat mengoptimalkan fungsi [20]. Algoritma Adam dihasilkan dari dua kombinasi algoritma yaitu, AdaGrad dan RMSProp. AdaGrad mempertahankan *learning rate* (kecepatan pembelajaran) per parameter untuk meningkatkan kinerja masalah gradien yang renggang dan RMSProp juga mempertahankan *learning rate* (kecepatan pembelajaran) per parameter yang disesuaikan berdasarkan ukuran gradien untuk rata-rata bobot. Adam menghitung rata-rata eksponensial dari *weighted* dan kemudian mengkuadratkan gradien yang dihitung. Adam dapat digunakan dalam model dan kumpulan data yang besar, dan Adam dapat secara efektif memecahkan masalah-masalah *deep learning* yang praktis [20]. Hal ini membuat *Adam Optimizer* menjadi metode komputasi yang efisien, yang membutuhkan memori yang sangat sedikit tetapi tidak seperti gradient diagonal, dan metode ini cocok untuk sejumlah besar data dan masalah parameter [20].

2.7. Teknik Pengambilan Data

Teknologi yang begitu maju pada saat ini akan memudahkan para pengguna internet untuk mengungkapkan pendapat atau pengalaman mereka saat menggunakan aplikasi. Salah satu tempat yang dijadikan sarana untuk menyampaikan ungkapan adalah di kolom *review* pada media sosial, yang kemudian dapat digunakan untuk bahan penelitian. Untuk mendapatkan ulasan tersebut diperlukan teknik *web scraping*. *Web scraping* ialah teknik untuk mengambil dokumen semi-terstruktur dari internet dalam ukuran yang beragam.

2.8. *Preprocessing*

Data ulasan yang berasal dari media sosial, umumnya menggunakan kata-kata tidak terstruktur yang memerlukan *preprocessing* seperti menghapus simbol, angka, emotikon, dan mengubah kata singkatan. Proses *preprocessing* yang dilakukan pada penelitian ini sebagai berikut:

1. *Case Folding*

Case Folding ialah mengubah semua karakter menjadi bentuk yang sama, contohnya mengubah huruf kapital menjadi huruf kecil [21]. Penggunaan huruf kapital seringkali tidak konsisten dalam data ulasan, sehingga memerlukan proses *case folding*. Untuk data string, mengubah penggunaan huruf kapital menjadi huruf kecil menggunakan fungsi `string.lower()` atau sebaliknya menggunakan fungsi `string.upper()`.

2. Mengubah kata slang

Slang ialah melakukan konversi untuk menormalkan kata atau mengubah kata ke bentuk dasar [22]. Pada data ulasan sering kali memuat kata slang atau singkatan, jadi sebaiknya ganti singkatan menjadi bahasa standar atau kata baku yang mudah dipahami.

3. Menghapus *Stopword*

Stopword ialah kata umum yang biasa ditemukan tetapi tidak memiliki arti atau tidak berhubungan dengan informasi yang dipelajari, biasanya berupa kata depan atau subjek [22]. Maka dari itu diperlukan proses ini

dengan memuat membuat kamus untuk menyesuaikan dengan data yang diteliti. Contoh kata *stopword* ialah “abang”, “atau”, “datang” dan sebagainya.

4. Menghapus simbol, angka, dan emotikon

Dalam data penelitian ini, banyak pengguna yang menggunakan simbol atau angka pada tiap ulasan. Ada juga ulasan yang mengandung emotikon sebagai bentuk ekspresi penulis. Namun, ketiga faktor tersebut tidak penting dalam penelitian ini dan harus dihilangkan untuk memudahkan melakukan analisis data.

2.9. *Vector Space Model*

Pada penelitian ini menggunakan data teks, yang di mana data teks merupakan data yang tidak terstruktur. Sistem komputer hanya dapat memproses data terstruktur dalam bentuk tabel. Maka dari itu kita perlu mengkonversi data menjadi angka agar komputer dapat memproses.

Fungsi dari VSM adalah untuk mengubah dari format dokumen menjadi nilai vektor. Ada beberapa cara yang digunakan untuk menghasilkan vector untuk merepresentasikan dokumen yaitu seperti *Bag of Words* (BoW). *Bag of Words* adalah model yang mempelajari atau memeriksa kosakata seluruh dokumen dan memodelkan setiap dokumen dengan cara menghitung jumlah kemunculan setiap kata [23]. *Bag of Words* (BoW) akan mempresentasikan kalimat atau dokumen sebagai kumpulan multiset dari kata-kata yang didalamnya. Salah satu fungsi yang digunakan untuk membangun BoW adalah doc2bow.

Fungsi dari BoW adalah menghitung jumlah kata unik yang muncul dalam dokumen yang terdiri atas *word.id* dan *word.frequency*. Setelah itu, mengubahnya menjadi format array dan mengembalikannya menjadi format vector. Pada saat pembentuk BoW menggunakan fungsi doc2bow terdapat pembuatan *dictionary*, proses yang harus dilakukan adalah dataset harus melalui proses perubahan menjadi bentuk dasar dan menanggalkan masing-masing kata. Berikut merupakan contoh dari pembuatan *dictionary* dengan fungsi doc2bow:

Tabel 2. 1 Pembuatan Dictionary dengan Fungsi doc2bow

	Token						
	1	2	3	4	5	6	7
Documen1	nina	pergi	sekolah	naik	mobil	antar	ayah
Documen2	botol	minum	tisya	tinggal	mobil	nina	

Pada Tabel 2. 1, menunjukkan hasil dari penggalan kata-kata untuk pembuatan *dictionary*. Langkah selanjutnya pada table 2. 2 menunjukkan pemberian identitas atau ID untuk setiap kata dan kata-kata yang sama diberi identitas sama.

Tabel 2. 2 Proses doc2bow

	Token						
	1	2	3	4	5	6	7
Documen1	1	2	3	4	5	6	7
Documen2	8	9	10	11	5	1	

2.10. *Hyperparameters*

Hyperparameter ialah parameter yang tidak ditentukan oleh mesin dan tanpa melakukan pengujian. *Hyperparameter* akan diatur secara manual sesuai dengan kebutuhan studi untuk membantu memperkirakan parameter model. *Hyperparameter* yang digunakan dalam penelitian ini adalah:

1. *Batch Size*

Batch Size ialah *hyperparameter* yang digunakan untuk menentukan jumlah sampel yang dikerjakan sebelum *weight* disesuaikan. Untuk memproses satu *batch* akan membutuhkan waktu yang lebih lama, jika ukuran *batch* lebih besar [9].

2. *Epoch*

Epoch ialah *hyperparameter* yang digunakan untuk menentukan berapa kali jaringan akan bekerja mengelola *data training*. Satu *epoch* menandakan bahwa sebuah algoritma *deep learning* telah belajar dari *training dataset* secara keseluruhan [24].

3. *Learning rate*

Learning rate ialah *hyperparameter* yang digunakan untuk menghitung nilai korelasi bobot pada waktu proses training. Semakin besar nilai *learning rate*, maka proses training akan berjalan semakin cepat.

2.11. *WordPiece Tokenization*

WordPiece Tokenization pertama kali diperkenalkan oleh Google. *WordPiece Tokenization* adalah sebuah metode tokenisasi yang bekerja dengan cara memisahkan kata-kata menjadi bentuk lengkap (misalnya satu kata menjadi satu token) atau menjadi potongan kata (yang di mana satu kata dapat dipecah menjadi beberapa token). *WordPiece* dibuat untuk mempelajari unit kata dari sejumlah besar data secara otomatis dan tidak menghasilkan OOV apa pun. *Out-of-vocabulary* (OOV) adalah Unit linguistik atau token yang tidak terdapat dalam kosakata atau dokumen pada saat training, yang berakibat mengurangi nilai akurasi.

Vocabulary dari *WordPiece Tokenization* model berjumlah 30.000 kata yang sering ditemukan dalam *corpus* pada model latih. *Vocabulary* tersebut terdiri atas:

1. Keseluruhan kata
2. Sub Kata yang berada di depan kata atau terpisah, misalnya kata “pengusaha”, memiliki “usaha” yang vektor tidak sama dengan “usaha” yang terdapat pada kalimat “usaha mandiri”
3. Subkata yang tidak terdapat di depan kata, akan diawali dengan tanda ##
4. karakter individu seperti ‘a’, ‘b’, ‘c’, ‘d’, ‘e’ dan seterusnya.

Misalnya terdapat kalimat:

```
text = 'wattpad adalah aplikasi novel online yang saya sukai'

tokens = tokenizer.tokenize(text)
token_ids = tokenizer.convert_tokens_to_ids(tokens)

print(f' Sentence: {text}')
print(f' Tokens: {tokens}')
print(f'Token IDs: {token_ids}')

Sentence: wattpad adalah aplikasi novel online yang saya sukai
Tokens: ['watt', '##pad', 'adalah', 'aplikasi', 'novel', 'online', 'yang', 'saya', 'sukai']
Token IDs: [12469, 7274, 154, 1339, 5601, 663, 34, 209, 10960]
```

Gambar 2. 4 Contoh Kalimat yang Menggunakan Tokenisasi Wordpiece

Berdasarkan gambar 2. 4 *Contoh Kalimat yang Menggunakan Tokenisasi Wordpiece*, Perhatikan bagaimana kata “wattpad” ditokenisasi menggunakan Wordpiece menjadi “watt” dan “##pad”. Kata asli ‘wattpad’ telah dipecah menjadi subkata dan karakter yang lebih kecil. Dua *hashtag* (##) yang mendahului beberapa subkata ini hanyalah cara tokenizer kami untuk menunjukkan bahwa subkata atau karakter ini adalah bagian dari kata yang lebih besar dan didahului oleh subkata lain. Jadi, misalnya, token ‘##pad’ terpisah dari token ‘pad’; yang pertama digunakan setiap kali subkata ‘pad’ muncul dalam kata yang lebih besar dan yang kedua digunakan secara eksplisit ketika token mandiri ‘pendapatan asli daerah’ muncul.

Untuk tokenisasi sebuah kata menggunakan model ini, pertama-tama tokenizer akan memeriksa apakah seluruh kata terdapat ada dalam kosa kata. Jika tidak, ia mencoba untuk memecah kata menjadi subkata terbesar yang mungkin terkandung dalam kosakata, dan sebagai upaya terakhir akan menguraikan kata menjadi karakter individu.

2.12. *Principal Component Analysis*

Principal Component Analysis (PCA) adalah metode yang digunakan untuk menyederhanakan data dengan cara transformasi linear untuk membentuk sistem koordinat baru dengan tetap mempertahankan variansi maksimum. Fungsi dari PCA adalah melakukan ekstraksi factor yang memiliki kolerasi tinggi antar faktor tanpa mengurangi karakteristik dari data tersebut [25]. Prinsip dasar dari *principal component analysis* adalah mereduksi dimensi suatu data dengan cara menghilangkan korelasi diantara variabel bebas dan melakukan transformasi linear variabel bebas menjadi variabel baru yang tidak memiliki korelasi satu sama lain atau biasa disebut *principal component*. *Principal component* adalah bentuk proyeksi transformasi linear dari variabel data. *Principal component* tidak berkorelasi satu sama lain dan diurutkan sedemikian sehingga *Principal component* pertama mengandung varinsi ternyak dari kumpulan data.

BAB III

METODE PENELITIAN

3.1. Sumber Data

Penelitian ini menggunakan data sekunder yang berasal dari data ulasan aplikasi novel *online* pada media sosial yang ditulis oleh pengguna aplikasi novel *online*. Pengambilan data dilakukan dengan cara *scraping* dengan total 8000 data ulasan dari tanggal 16 September 2020 06:49 sampai 14 Maret 2021 15:34 dari versi 6.44.1 sampai 9.4.0.

Data yang diperoleh menggunakan bahasa Indonesia dan lokasi pengulas berada di negara Indonesia. Ulasan yang didapatkan dari hasil *scraping* berdasarkan ulasan yang paling membantu (*helpful*) yang dibuat oleh pengguna aplikasi novel *online*. Data disimpan dalam bentuk CSV (*Comma Separated Values*). Berikut beberapa contoh data awal dari hasil *scraping*:

Tabel 3. 1 Tabel Awal Hasil Scrapping

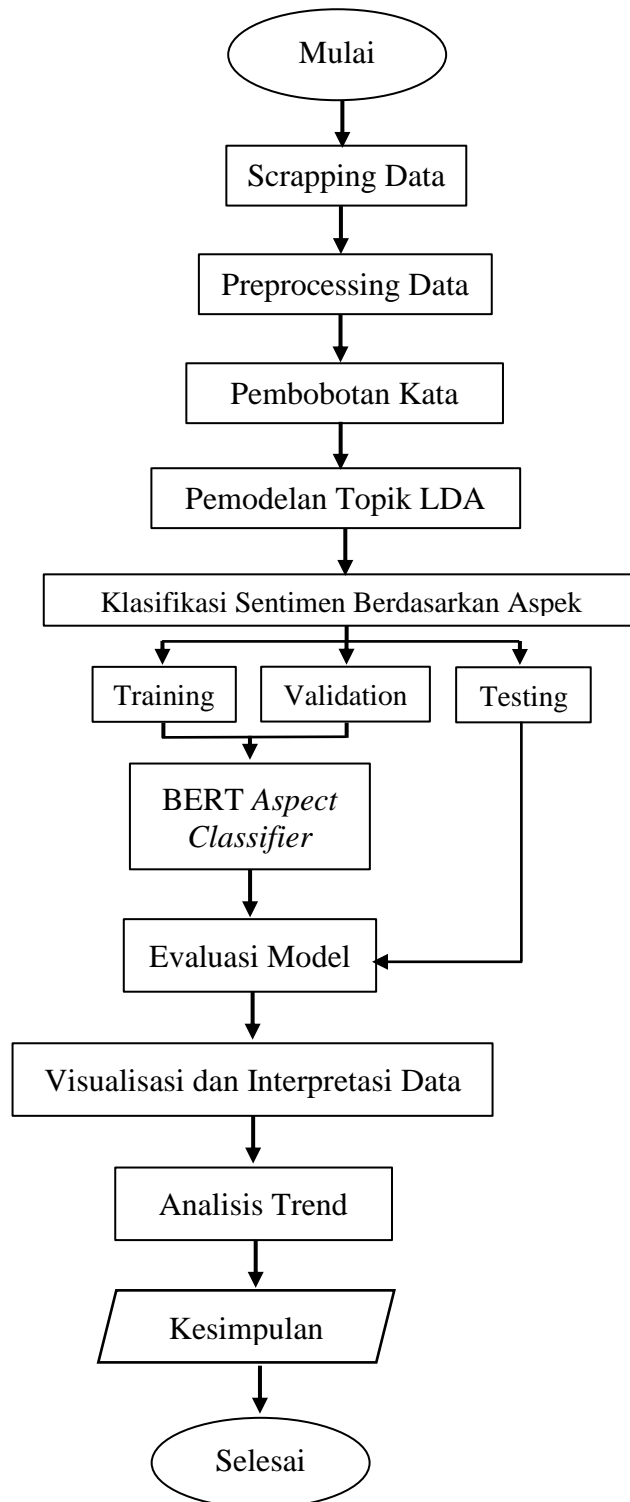
<i>Date</i>	<i>Version</i>	<i>User</i>	<i>Review</i>	<i>Rating</i>
14/03/2021 12:23:48	9.4.0	Pipit Prajitno	Cerita nya seru seru walau pun ada yg berbayar	5
14/03/2021 12:03	9.4.0	Annisa After	Tidak bisa masuk ke akun saya, padahal sudah konfirmasi email. AKhirnya cerita yg pernah saya buat dulu dan simpan di draft tidak bisa lagi di akses	2
01/03/2021 15:24:05	9.1.1	Lian Tan	Ga bisa masuk ke akun lama padahal e-mail dan sandi nya udh bener Udah jutaan pembaca yg baca karya saya Gk tau lg harus kaya gimana	1

30/01/2021 06:11	8.98.0	Erika Natahsya	Aplikasi novel <i>online</i> sangat bagus untuk menyalurkan hobi menulis dan membaca. Namun, aplikasi novel <i>online</i> sering error di bagian tag dan bagian notifikasinya. Sudah beberapa kali saya update cerita dan notifnya tidak masuk/muncul padahal jaringan baik-baik saja. Tag cerita juga sering error muncul dan hilang. Tolong lebih diperhatikan dan diperbaiki lagi. Terima kasih.	4
---------------------	--------	-------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---

Berdasarkan hasil scrapping, karakteristik yang diambil sebagai berikut *date*, *version*, *user*, *review*, dan *rating*. Pada saat scrapping, kolom *date* atau tanggal pada saat ulasan dibuat oleh pengguna tidak terurut berdasarkan waktu, maka peneliti mengurutkan di *Microsoft Excel* berdasarkan waktu terlama untuk bisa melihat perkembangan sentiment dari waktu ke waktu. Kolom *version* yang merupakan versi aplikasi novel *online* yang digunakan oleh pengguna, *user* yang merupakan nama pengguna yang memberi ulasan, *review* yang merupakan kalimat ulasan dari pengguna tentang aplikasi novel *online* yang digunakan dan *rating* yang diberikan oleh pengguna tentang aplikasi novel *online*.

3.2. Alur Penelitian

Alur penelitian yang dilakukan adalah sebagai berikut:



Gambar 3. 1 Alur Penelitian

Pada penelitian ini penulis menggunakan perangkat lunak Python 3.8 dan Google Colab dengan bantuan beberapa *library* yaitu *sklearn*, *pandas*, *matplotlib*, *pyLDAvis*, *numpy*, *transformasi*, *torch*, dan *re*. berdasarkan Gambar 3.1 *Alur penelitian*, langkah pertama yang dilakukan adalah melakukan scraping data di Media sosial. Selanjutnya data yang telah didapat terdiri dari *date*, *version*, *user*, *review*, dan *rating* dan disimpan ke bentuk format CSV. Tetapi terdapat beberapa data yang tidak sah karena tidak mengandung *rating* atau *review* yang hanya berisi simbol, maka data seperti itu dihapus. Data yang tersimpan merupakan data mentah dan menggunakan kata-kata yang tidak terstruktur dengan baik. Maka dari itu perlu dilakukannya *pre-processing* data dengan cara *case-folding*, mengubah kata singkatan/*slang*, dan menghapus *stopword*, emotikon dan simbol. Kata-kata yang kasar, tidak penting, dan yang sering muncul juga dihapus untuk mempermudah pengolahan data dengan melihat kata kata penting saja.

Kemudian, data diubah ke bentuk angka agar komputer dapat mengolah dengan cara VSM (*Vector Space Model*), yang dimana kata- kata diberi bobot yang sesuai dengan kemunculannya pada seluruh data. Selanjutnya dilakukan *clustering topic* menggunakan LDA untuk menentukan jumlah topik yang digunakan dan menjadi referensi pada saat pelabelan manual. Pada saat mendeteksi aspek terdapat ulasan yang memiliki *double* aspek yang dilihat dari ulasan secara manual. Pada pelabelan sentimen dilakukan dengan cara melihat *rating* yang diberikan pengguna, yaitu negatif, netral dan positif. Setelah itu, dilakukan pembagian data menjadi data training, data validation dan data testing. Kemudian data diolah ke dalam klasifikasi sentimen berbasis aspek dengan menggunakan model klasifikasi yaitu *Bidirectional Encoder Representations from Transformers* (BERT). Karena data yang digunakan tidak seimbang antara ulasan positif, netral dan negatif, maka peneliti melakukan *resampling* data untuk mengatasi masalah data tidak seimbang dengan cara teknik SMOTE. Setelah itu dilakukan visualisasi dan interpretasi data pada tiap sentimen dan aspek, dan direpresentasikan menggunakan *wordcloud* dan *wordlink*. Lalu dilakukan pembentukan analisis *trend* per bulan untuk mengetahui perkembangan aplikasi novel online berdasarkan ulasan pengguna. Terakhir, menarik kesimpulan dari apa yang telah dilakukan pada penelitian ini.

3.3. Latent Dirichlet Allocation (LDA)

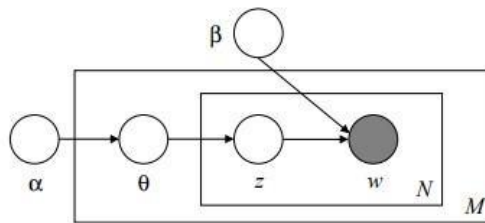
Ide pokok dari LDA adalah dokumen yang ditujukan sebagai campuran acak topik yang tersembunyi, di mana setiap topik yang ada dicirikan sebagai distribusi dari kata-kata [26]. Fungsi dari LDA adalah untuk menetapkan jumlah topik dari suatu corpus dan distribusi kata dalam setiap topiknya. Notasi berikut secara resmi didefinisikan sebagai berikut:

- Sebuah kata ialah unit dasar dari kumpulan data diskrit, yang didefinisikan sebagai item dari kosakata yang diindeks oleh $\{1, \dots, V\}$.
- Sebuah dokumen ialah urutan N kata yang dilambangkan dengan $w = (k_1, k_2, \dots, k_N)$, di mana k_n adalah urutan kata ke- n .
- Korpus ialah kumpulan M dokumen yang dilambangkan dengan $D = \{w_1, w_2, \dots, w_M\}$.

LDA memperkirakan proses di setiap dokumen w pada korpus D :

- Pilih N sebagai Poisson (ξ).
- Pilih θ sebagai Dirichlet (α).
- Untuk setiap N kata pada w_n :
 - Pilih topik z_n sebagai Multinomial (θ)
 - Pilih kata w_n dari $p(w_n|z_n, \beta)$, sebagai peluang multinomial yang dikondisikan pada topik z_n .

Proses kerja LDA ditunjukkan pada gambar representasi model LDA berikut [26]:



Gambar 3. 2 Representasi Model LDA [26]

Berdasarkan tampilan model pada Gambar 3. 2 *Representasi Model LDA*, bentuk persegi panjang menunjukkan tempat untuk mewakili pengulangan. Kotak terluar dilambangkan dengan M mewakili dokumen, sedangkan kotak terdalam

dilambangkan N mewakili pilihan berulang dari topik dan kata-kata dalam dokumen. Parameter α dan β adalah parameter distribusi topik pada tingkat korpus. Variabel θ adalah variabel pada tingkat dokumen. Variabel Z dan W adalah variabel pada tingkat kata.

Parameter α adalah parameter yang digunakan untuk mendistribusikan topik dalam dokumen, dan β adalah parameter yang digunakan untuk mendistribusikan kata dalam topik. Semakin tinggi nilai α , maka semakin besar juga campuran topik yang dibahas dalam dokumen. Semakin besar nilai β , maka semakin besar juga setiap topik mengandung kata-kata, maksudnya ada banyak kata yang membedakan topik satu dengan topik lainnya. θ adalah variabel distribusi topik untuk setiap dokumen. Semakin besar θ , maka semakin banyak pula topik yang terkandung dalam dokumen. Variabel z mempresentasikan penetapan topik untuk setiap kata-kata dalam dokumen. Variabel w adalah mempresentasikan spesifik kata yang berkaitan pada topik yang ada dalam dokumen.

Output model untuk setiap dokumen akan memunculkan kata-kata di setiap cluster topik. Kumpulan kata-kata di setiap cluster topik kemudian diinterpretasikan untuk memberikan informasi tentang topik apa saja yang dibahas di setiap dokumen. Oleh karena itu, total probabilitas model LDA berdasarkan pada representasi model LDA dapat ditulis dengan rumus [27]:

$$p(D|\alpha, \beta) = \prod_{d=1}^M \int p(\theta_d|\alpha) \left(\prod_{n=1}^N \sum_{Z_{dn}} p(z_{dn}|\theta_d) p(w_{dn}|z_{dn}, \beta) \right) d\theta_d \quad (3.1)$$

Langkah-langkah dalam pengerjaan LDA diberikan seperti contoh berikut ini:

1. Menentukan nilai untuk α , β dan jumlah topik k .
2. Kategorikan dokumen menjadi bentuk kata atau biasa dikenal dengan tokenisasi, lalu berikan nilai untuk setiap kata. Seperti yang ditunjukkan pada di bawah ini:

Tabel 3. 2 Contoh LDA untuk Dokumen, Token dan Kata Unik

No.	Dokumen	Hasil Token	Kata Unik	Nilai setiap kata
1.	nina pergi ke sekolah naik mobil	'nina'	'nina'	1
		'pergi'	'pergi'	2
		'ke'	'ke'	3
		'sekolah'	'sekolah'	4
		'naik'	'naik'	5
		'mobil'	'mobil'	6
2.	Botol minum tisyia tertinggal di mobil nina	'botol'	'botol'	7
		'minum'	'minum'	8
		'tisyia'	'tisyia'	9
		'tertinggal'	'tertinggal'	10
		'di'	'di'	11
		'mobil'		
		'nina'		

Dari 2 contoh dokumen yang diberikan pada Tabel 3. 2 *Contoh LDA untuk Dokumen, Token dan Kata Unik*, setiap kalimat akan di tokenisasi menjadi kata per kata. Kata-kata unik diambil dari hasil kemudian diberikan nomor yang digunakan untuk menginisialisasi setiap kata unik.

3. Selanjutnya, menentukan topik dari setiap kata uniknya, misalkan pada contoh dokumen ini ditetapkan jumlah topik sebanyak 2, ditunjukkan seperti tabel berikut:

Tabel 3. 3 Hasil Sebaran Kata Unik di Setiap Topik

		Kata Unik										
		1	2	3	4	5	6	7	8	9	10	11
Topik	1	2	0	0	0	1	2	1	1	1	1	1
	2	0	1	1	1	0	0	0	0	0	0	0

Setelah menentukan banyaknya jumlah topik, buat matriks dan kata-kata unik seperti pada Tabel 3. 3 *Hasil Sebaran Kata Unik di Setiap Topik*. Jumlah kolom yang digunakan adalah jumlah total dari banyaknya kata unik yang muncul di semua dokumen. Misal pada kata unik 1 “nina”, muncul 2 kali pada semua dokumen dan kata tersebut masuk ke dalam topik 1, sedangkan pada kata unik 2 “pergi” hanya terdapat 1 kata di semua dokumen, dan kata tersebut termasuk topik 2, dan begitu seterusnya.

4. Menentukan topik dari setiap kata yang ada pada tiap dokumen.

Tabel 3. 4 Topik Setiap Kata Unik dalam Masing-Masing Kalimat

Kalimat 1		Kalimat 2	
Kata Unik	Topik	Kata Unik	Topik
'nina'	1	'botol'	1
'pergi'	2	'minum'	1
'ke'	2	'tisyah'	1
'sekolah'	2	'tertinggal'	1
'naik'	2	'di'	1
'mobil'	1	'mobil'	1
		'nina'	1

Setelah ditentukan topik untuk setiap kata unik, hasil tersebut kemudian dimasukkan dalam setiap dokumen, yang nantinya akan dihitung berapa kata unik yang terdapat di setiap topik.

5. Menghitung jumlah kata yang terdapat di setiap topik pada masing-masing dokumen.

Tabel 3. 5 Jumlah Kata Unik di Setiap Topik pada Masing-Masing Dokumen

Dokumen	Topik	
	1	2
1	2	4
2	7	0

Nilai terdapat pada Tabel 3. 5 *Jumlah Kata Unik di Setiap Topik pada Masing-Masing Dokumen* menunjukkan jumlah kata unik yang terdapat di topik 1 dan 2 pada setiap dokumennya. Pada dokumen 1, memiliki 2 kata yang masuk ke topik 1 dan 4 kata yang masuk ke topik 2. Pada dokumen 2, memiliki 7 kata yang masuk ke topik 1 dan 0 kata yang masuk ke topik 2.

6. Menghitung probabilitas topik dari dokumen, menggunakan persamaan berikut:

$$p(\theta|\alpha) = \frac{\Gamma(\sum_{k=1}^K \alpha_k)}{\prod_{k=1}^K \Gamma(\alpha_k)} \prod_{k=1}^K \theta_k^{\alpha_k-1} \quad (3.2)$$

Hasilnya seperti pada matriks dibawah ini:

Tabel 3. 6 Peluang Topik di setiap Dokumen

Dokumen	Topik	
	1	2
1	0.375	0.625
2	0.889	0.111

Tabel 3. 6 *Peluang Topik di Setiap Dokumen*, menunjukkan probabilitas kemunculan topik 1 sebesar 0.375 dan topik 2 sebesar 0.625 di Dokumen 1, sedangkan probabilitas topik 1 sebesar 0.889 dan sebesar 0.111 pada topik 2 di Dokumen 2. Akibatnya, rasio topik pada dokumen 1 adalah lebih besar di topik 2, dan pada dokumen 2 lebih besar topik 1 daripada topik 2.

7. Terakhir, probabilitas distribusi kata di setiap topik diberikan sebagai seperti berikut:

Tabel 3. 7 Peluang Kata di setiap Topik

Topik	Kata Unik										
	nina	pergi	ke	sekolah	naik	mobil	botol	minum	Tisya	tertinggal	di
1	0.2220	0.0001	0.0001	0.0001	0.1111	0.2220	0.1111	0.1111	0.1111	0.1111	0.1111
2	0.0002	0.2496	0.2496	0.2496	0.0002	0.0002	0.0002	0.0002	0.0002	0.0002	0.0002

Nilai terdapat di Tabel 3. 7 *Peluang Kata di setiap Topik*, adalah nilai probabilitas kata pada setiap topik berdasarkan tabel pada Langkah ke 3. Setiap kata dapat berhubungan dengan beberapa topik, hanya saja proporsinya berbeda. Misalnya, kata nina lebih cenderung muncul di topik 1 daripada di topik 2.

3.4. BERT (*Bidirectional Encoder Representations from Transformers*)

Dalam penelitian ini, penulis menggunakan BERT_{BASE} dengan model ‘[indobenchmark/indobert-base-p1](#)’ yang mempunyai *Vocabulary* sebanyak 30521 kata [7] [28]. Model ini sebelumnya telah dilatih di Wikipedia berbahasa Indonesia dan surat kabar Indonesia yang berguna untuk pemodelan *masked language modeling* (MLM) [7]. Peneliti menggunakan library Transformers yang telah disediakan oleh HuggingFace. *Library* ini menyediakan pre-trained model yang dirancang untuk melakukan tugas klasifikasi, tanya jawab, *summarization*, *text generation*, dan sebagainya. Transformers didukung oleh dua *library* yaitu pyTorch dan TensorFlow. Kelebihan yang dimiliki BERT dibandingkan metode-metode lain:

1. BERT mempunyai parameter yang sangat besar untuk berbagai bahasa. Contohnya untuk model ‘indobert’ mempunyai total parameter sebesar 124,5 milion [28].
2. BERT merupakan model representasi berbasis *fine-tuning* pertama yang mencapai kinerja canggih pada ukuran besar dalam tingkatan kalimat dan tingkat token, dan banyak membentuk tugas arsitektur spesifik. [29]
3. BERT memanfaatkan encoder sebagai input, bukan sebagai urutan sekuensial. Dimana dengan karakteristik tersebut, BERT dapat memahami hubungan kontekstual untuk setiap token dengan baik. Cara kerja BERT membaca input yaitu dengan *positional embedding* yang membuat BERT dapat membaca input lebih panjang dari pada RNN [29].

4. Tidak seperti metode OpenAI GPT dan ELMo, arsitektur dari BERT untuk setiap encoder akan memiliki akses atau terhubung dengan setiap token, yang artinya model dapat mengerti arti token dari dua arah secara bersama-sama.

3.4.1. Penjelasan BERT

Pada tahun 2018, peneliti ahli di Google AI Language mengembangkan model representasi bahasa terlatih, yang biasa dengan dengan BERT atau *Bidirectional Encoder Representations from Transformers* [29]. Inovasi Model arsitektur BERT yaitu representasi *multi-layer* dua arah (*bidirectional*) *Transformer encoder* [30]. Transformasi merupakan suatu mekanisme yang berfungsi untuk mempelajari adanya hubungan antar kata dalam teks [30]. Mekanisme yang terdapat pada transformer, yaitu:

1. *Encoder*

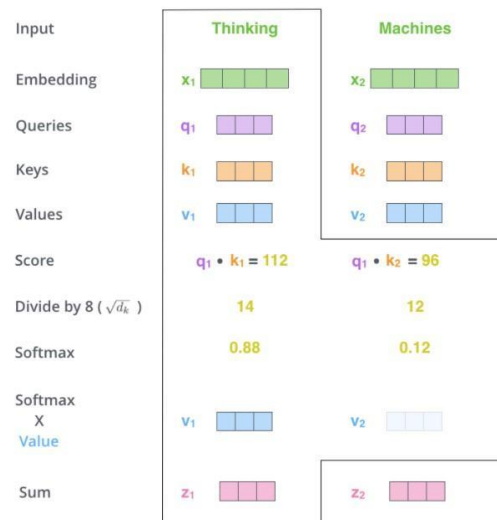
Encoder diperlukan pada saat membaca keseluruhan *input* teks. *Encoder* terdiri dari tumpukan 6 lapisan identik. Setiap tingkat memiliki 2 sublapisan, yaitu lapisan *self-attention* dan *feed-forward neural network*. Fungsi dari *self-attention layer* yaitu *encoder* membantu node untuk fokus pada kata yang terlihat, serta mendapatkan konteks semantik dari kata tersebut. Lapisan *Self-attention* memungkinkan setiap posisi *encoder* menangani semua posisi lapisan sebelumnya dan posisi saat ini

2. *Decoder*

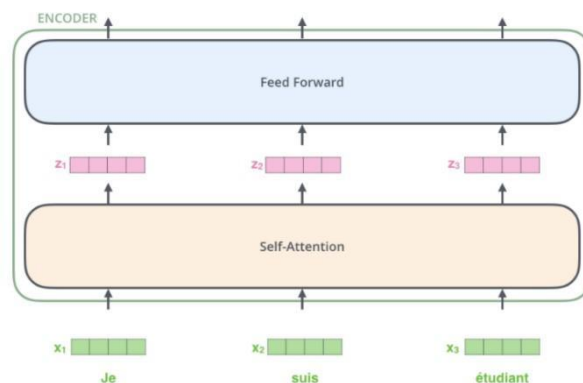
Decoder dapat menghasilkan urutan *output* yang dapat diprediksi. *Decoder* juga terdiri dari tumpukan 6 lapisan identik. Setiap lapisan terdiri dari dua sublapisan, yaitu *self-attention layer* dan *feed-forward neural network*, tetapi diantara dua lapisan ini terdapat lapisan *attention layer* yang membantu *node* saat ini mendapatkan *key content* yang membutuhkan perhatian dengan cara melakukan *multi-head attention* dari *output* di *encoder* [31].

Langkah-langkah transformasi sebagai berikut [32]:

1. Semua *input* yang masuk ke proses *encoder* akan dikonversi ke bentuk *list vector* menggunakan *embeddings*. Kemudian melewati *positional encoding* yang berguna untuk menandai posisi setiap kata. Masing-masing vektor *input* memiliki ukuran yang sama yaitu 512. Proses ini hanya dilakukan pada *encoder* yang berada di paling awal, sehingga *encoder* selanjutnya mendapatkan *output* dari *encoder* yang pertama.
2. Vektor *input* kemudian akan dilewati dua lapisan dari setiap *encoder*, yaitu lapisan *self-attention* dan *feed-forward neural network*. Pada *self-attention layer*, tiga vektor dihasilkan dari masing-masing vektor *input* yang ada, yaitu *query*, *key* dan *value vector*. 3 vektor tersebut dihasilkan dari perkalian *embedding*, yang masing-masing berdimensi 64. Kemudian, nilai dari *self-attention* setiap kata dihitung dengan mengalikan vektor *query* dengan vektor *key* seperti pada Gambar 3. 3 *Proses Self Attention Layer*. Kemudian bagi hasilnya dengan 8 yang merupakan akar dimensi dari setiap vektor. Nilai *self-attention* juga dihitung menggunakan *softmax*, dan kemudian tiap vektor *value* dikalikan dengan nilai dari hasil *softmax*. Lapisan *self-attention* akan menghasilkan *output* berupa jumlah *weight* dari vektor *value*, yang kemudian akan masuk ke tahap *feed-forward* untuk setiap posisi seperti pada Gambar 3. 4. *Proses Encoder*.



Gambar 3. 3 Proses Self Attention Layer [32]



Gambar 3. 4 Proses Encoder [32]

3. Kemudian *output* dari *encoder* terdiri dari vektor *key* dan vektor *value* yang menuju ke *decoder*. Setiap *input* dan *output* dari lapisan *self-attention* dan *feed-forward neural network* di *encoder* akan di proses di *decoder* menjadi lapisan *add & norm* yang mengandung struktur residual dan lapisan normalisasi. Proses tahap *decoder* sama dengan tahap *encoder*, perbedaannya terdapat lapisan *attention* yang membantu *decoder* fokus pada bagian kata yang relevan. *Output* dari setiap tahap terus menerus masuk ke dalam *decoder* dan memberikan hasil yang sama seperti tahap *encoder*. Akhirnya, tumpukan *decoder* akan menghasilkan vektor nilai *float*. lapisan *fully connected* dan

lapisan *softmax* diperlukan sebagai lapisan tambahan untuk mengubah vektor nilai *float* menjadi kata-kata.

Arsitektur dari model BERT adalah *multi-layer bidirectional transformer* tetapi hanya menggunakan *encoder* saja dan WordPiece *embedding* dengan 30.000 token *vocabulary*. Penelitian ini menggunakan ukuran model BERT_{BASE} dan *encoder* yang memiliki 12 *layers*, 12 *self-attentions heads*, 768 *hidden size*, dan 110M parameters. Model bahasa BERT terlatih secara *bidirectional* yang memiliki pemahaman konteks yang lebih mendalam karena dua arah dibandingkan model bahasa terlatih satu arah. Token khusus [CLS] menjadi indikator urutan awal dan token [SEP] menjadi memisahkan antar kalimat. Pada tahap *training* BERT terdiri dari dua tahapan, yaitu *pre-training* dan *fine tuning*. Tahap *pre-training* ialah tahap yang membawa BERT untuk mempelajari bahasa *input* dan konteksnya. BERT dapat melakukan tahap *training* secara bersamaan menggunakan dua tugas, yaitu *Masked Language Modelling (Masked LM)* dan *Next Sentence Prediction*.

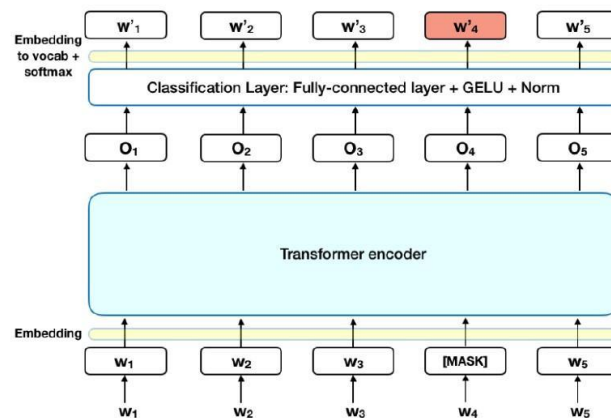
1. *Masked Language Modelling (Masked LM)*

Masked Language Modelling (Masked LM) berfungsi untuk memberikan *mask* atau menutupi kata-kata acak dalam kalimat dengan probabilitas rendah yang bertujuan untuk mengoptimalkan weight (bobot) pada BERT. Sebelum memasuki proses BERT sebanyak 15% dari setiap kata di semua urutan, diganti menggunakan token [MASK]. Kemudian token [MASK] digunakan model untuk memprediksi nilai asli kata, berdasarkan kata-kata lain yang tidak ditutup oleh token [MASK] di setiap frasa. Cara BERT menjawab token [MASK] yaitu dengan cara mengingat pola linguistic dan konteks kata-kata Tujuan dari token [MASK] adalah untuk pemahaman yang lebih baik tentang gaya bahasa yang digunakan.

Output dari prediksi kata - kata meliputi:

- a. Lapisan klasifikasi diperlukan setelah *output encoder*.

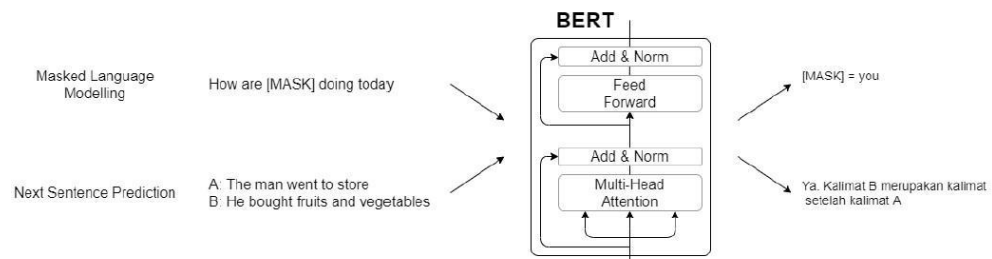
- b. Mengalikan vektor *output* dengan matriks *embedding*, dan kemudian ubah ke dimensi *vocabulary*.
- c. Gunakan *softmax* untuk menentukan probabilitas setiap kata dalam *vocabulary*.



Gambar 3. 5 Proses Masked Language Modelling [29]

2. Next Sentence Prediction

Selama proses *training* berjalan, model dapat mengambil pasangan kalimat sebagai *input* yang akan kemudian akan dilatih untuk memprediksi apakah kalimat kedua dari pasangan tersebut merupakan kalimat lanjutan atau kalimat tunggal dalam dokumen aslinya. 50% dari *input* merupakan pasangan kalimat, di mana kalimat ke dua merupakan kalimat selanjutnya dari dokumen asli. Sisanya 50% lainnya merupakan kalimat yang diambil secara acak pada *corpus*.



Gambar 3. 6 Proseses Training BERT [29]

Terdapat tiga *layer embedding* pada representasi *input* BERT, yaitu:

1. *Token embeddings*

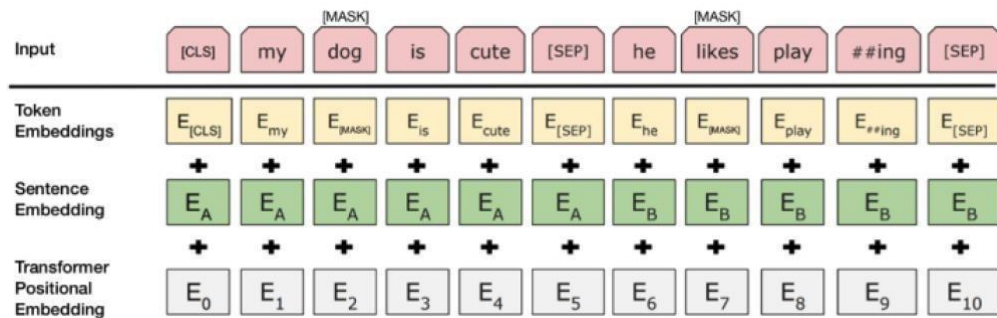
Token embeddings ialah lapisan pertama yang dimasuki oleh token, yang berfungsi sebagai representasi bentuk vektor di setiap token. Setiap token yang berada di dalam urutan kalimat akan dipetakan ke representasi vektor multidimensi. Setiap token diubah menjadi ID berdasarkan *vocabulary* model.

2. Sentence embeddings

Sentence embeddings berfungsi untuk menandai kalimat pertama atau kedua dan untuk membedakan antar kalimat jika terdapat lebih dari dua kalimat. Ada dua representasi yang ditampilkan pada contoh ini, yaitu A untuk token yang terdapat dalam kalimat pertama, dan B untuk token yang terdapat dalam kalimat kedua.

3. Positional embedding

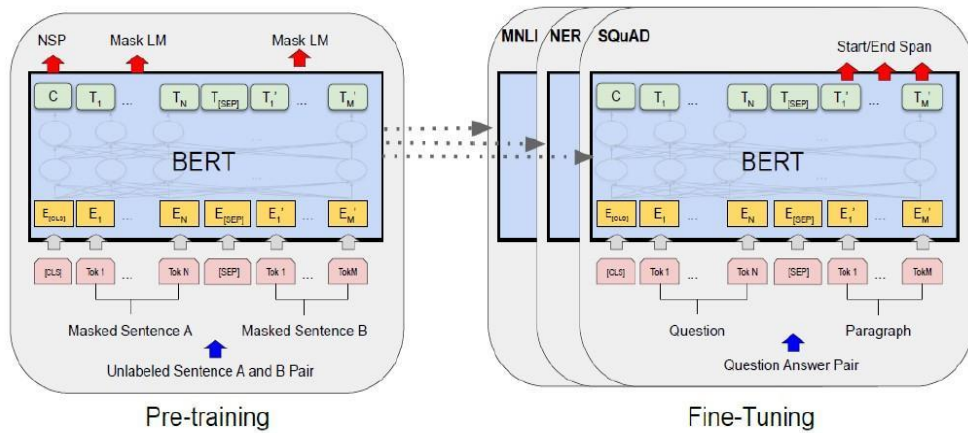
Positional embedding berfungsi untuk menyimpan informasi tentang posisi kata dalam urutan kalimat. *Positional embedding* ditunjukkan ketika proses Transformer yang dimana BERT akan mempelajari posisi lapisan *embedding* sejak proses pre-training.



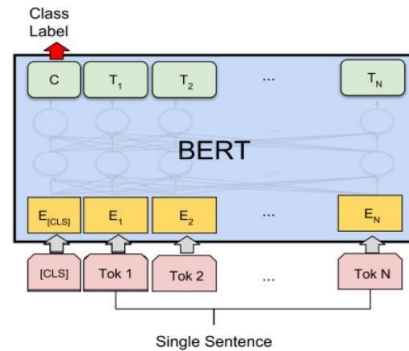
Gambar 3. 7 Representasi Input BERT [29]

Pada saat *training* sebuah model bahasa ada yang perlu dilatih, yaitu klasifikasi dengan beberapa perubahan kecil pada model BERT pada saat tahap *training* yang disebut *fine-tuning*. Ada beberapa rekomendasi *hyperparameters* saat *fine-tuning* untuk menghasilkan hasil yang maksimal [29], yaitu:

- Batch size: 16, 32
- Learning rate* (Adam): 2e-5, 3e-5, 5e-5
- Epoch*: 2, 3, 4



Gambar 3. 9 Ilustrasi proses *Pre-training* dan *Fine-tuning* [29]



Gambar 3. 8 Ilustrasi Fine-tuning pada *Single Sentence* [29]

Dengan dilakukannya mekanisme *self-attention* pada Transformer, *Fine-tuning* akan mudah dilakukan karena memungkinkan BERT untuk membuat model pada kalimat tunggal atau pada kalimat berpasangan, dan menukar *input* dan *output* yang sesuai.

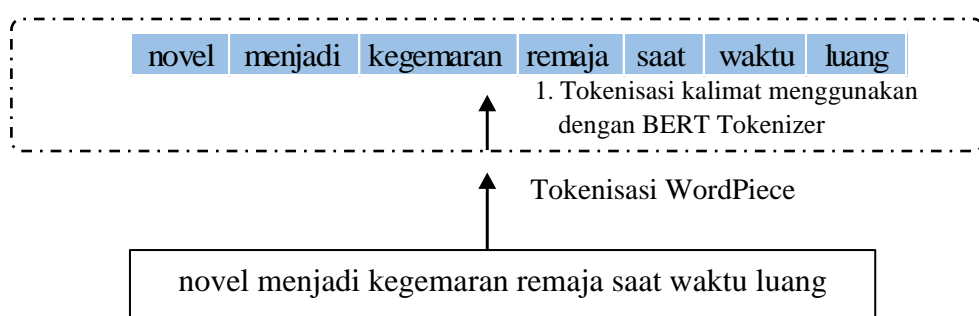
3.4.2. Representasi *Input* BERT

Sebelum dilakukan training, kumpulan data harus diubah terlebih dahulu menjadi bentuk representasi *input* yang dapat diterima oleh BERT. Dengan cara *tokenizer* yang bertujuan untuk melakukan tokenisasi pada kalimat kalimat yang ada pada dataset untuk menghasilkan *input* yang sesuai. Hal ini dilakukan oleh BERT dengan menggunakan *vocabulary* tertentu tergantung pada model yang digunakan. *Vocabulary* yang dibuat dengan menggunakan model *WordPiece*.

BERT akan menerima urutan kalimat dengan panjang yang sama pada setiap *input*. Panjang urutan kalimat maksimum yang ditentukan oleh BERT adalah 512 karena *encoder* pada Transformer hanya dapat menghasilkan *output* dengan berdimensi 512. Jika panjang kalimat yang diterima kurang dari panjang kalimat maksimum yang ditentukan, kalimatnya ditambah. Sebaliknya, jika panjang hukuman yang diterima melebihi panjang maksimal kalimat yang ditentukan, maka akan dipotong. Panjang maksimum urutan kalimat juga dapat ditentukan oleh peneliti sesuai kebutuhan.

Langkah-langkah yang dilakukan untuk mempersiapkan kalimat menjadi representasi *input* di BERT dilakukan dengan cara tokenisasi sebagai berikut:

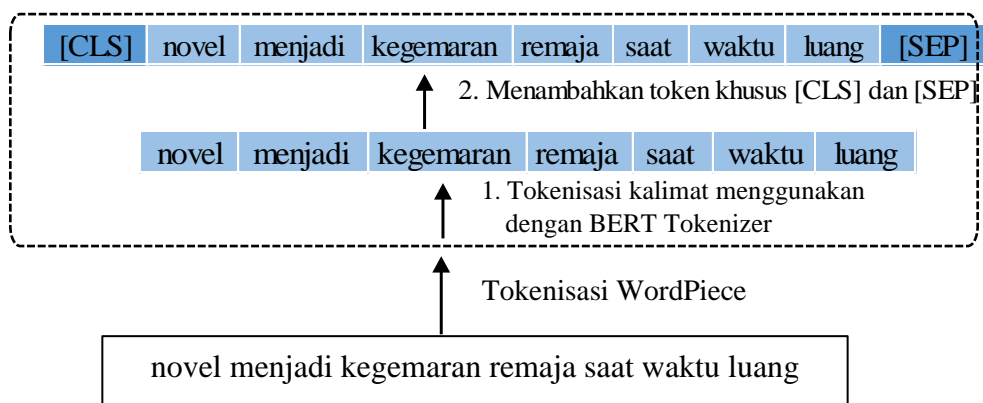
1. Semua kalimat akan di tokenisasi menggunakan WordPiece menjadi kata per kata atau sub kata. Setiap kata yang di tokenisasi diperiksa apakah kata tersebut termuat di *vocabulary*, jika tidak termuat di dalam *vocabulary* tokenizer akan membagi kata tersebut menjadi subkata yang kemungkinan munculnya paling besar dalam *vocabulary*. Kata-kata yang tidak termuat dalam *vocabulary* akan diubah menjadi token *unknown* [UNK]. Namun, akan banyak informasi yang akan hilang jika semua kata kata yang tidak terdapat di *vocabulary* di ubah menjadi token [UNK]. Oleh karena itu, kata tersebut dapat dipecah menjadi dua sub kata dimana token pertama berisi *vocabulary* yang dari sub kata dan token berikutnya yang diawali simbol ## menunjukkan bahwa token tersebut adalah sufiks yang diikuti subkata lainnya. BERT melakukan ini dengan untuk mempercepat *preprocessing*, untuk mengurangi jumlah parameter yang dilatih, dan untuk memecahkan masalah *out-of-vocabulary*.



Gambar 3. 10 Tahap Tokenisasi Menggunakan WordPiece

Gambar 3. 10 *Tahap Tokenisasi Menggunakan WordPiece*. menunjukkan tahap awal ketika sebuah kalimat dipecah menjadi token kata dan sub kata. Misalnya, kalimat yang di *input* adalah "novel menjadi kegemaran remaja saat waktu luang". Saat dilakukan tokenisasi kalimat berubah menjadi subkata seperti "novel" "menjadi" "kegemaran" dan seterusnya.

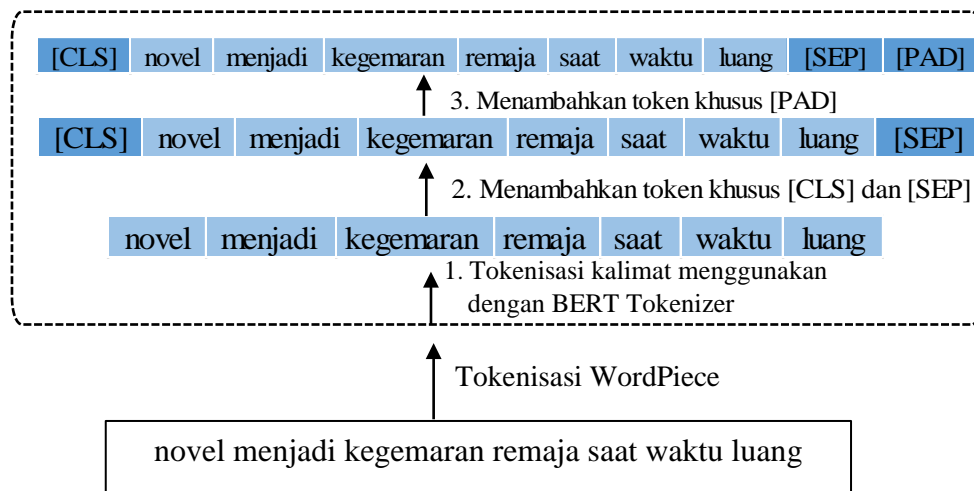
2. Setiap kalimat diberi tanda khusus di awal berupa token [CLS] dan di akhir kalimat berupa token [SEP]. Token [CLS] adalah suatu token yang menjadi indikator sebuah awal kalimat dan digunakan untuk proses analisis pada saat dilakukan klasifikasi sentimen karena token [CLS] dianggap melakukan pengumpulan rata-rata dari token kata untuk mendapatkan vektor dari kalimat. Fungsi Token [SEP] adalah untuk memisahkan kalimat satu dengan kalimat berikutnya. Kalimat yang sudah menerima token [CLS] dan [SEP] dinamakan token *embeddings*. Gambar 3.11 *Tahap menjadi Token Embedding*. Menunjukkan proses pemberian token khusus menjadi token *embeddings*.



Gambar 3. 11 Tahap menjadi Token Embedding

3. Kemudian, kalimat tersebut disesuaikan dengan panjang urutan maksimal yang sudah ditentukan, dengan mengurangi token atau memberikan *padding* dengan token khusus, yaitu token [PAD]. Misalkan pada Gambar 3. 12 *Tahapan Pemberian Token Padding*. menunjukkan tahapan

pemberian *padding* dengan panjang maksimum yang telah ditentukan yaitu 15. Karena kalimat yang *input* panjangnya kurang dari panjang urutan maksimal, sehingga token khusus [PAD] perlu ditambahkan di akhir kalimat setelah token [SEP].



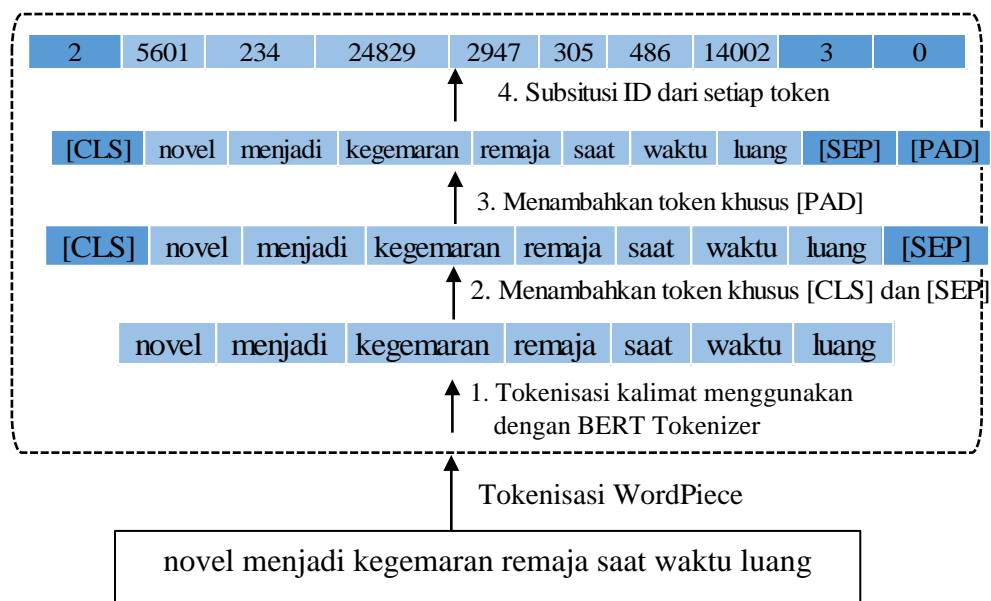
Gambar 3. 12 Tahapan Pemberian Token Padding

4. Nomor unik atau ID diperoleh berdasarkan indeks kata dalam *vocabulary* yang diurutkan berdasarkan kemunculannya dan setiap kata memiliki nomor unik berbeda. Kata dan sub kata harus diubah kedalam ID dikarenakan pada saat *pre-training* model hanya dapat memproses ID pada token. Misalkan *vocabulary* pada model '*indobenchmark/indobert-base-p1*', token [UNK] mempunyai ID 1 setelah itu token [CLS] mempunyai ID 2, token [SEP] mempunyai ID 3 seperti pada Gambar 3. 13 *Indeks pada Vocabulary Model*.

```
(0, '[PAD]'),
(1, '[UNK]'),
(2, '[CLS]'),
(3, '[SEP]'),
(4, '[MASK]'),
```

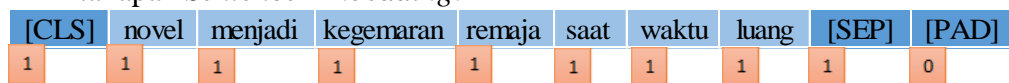
Gambar 3. 13 Indeks pada *Vocabulary Model*

Penomoran ID ke token sudah menjadi ketetapan, sehingga jika terdapat kata "novel", ID dari token "novel" adalah 5601. Pada Gambar 3. 14 *Tahap Substitusi ID dari Setiap Token*, menunjukkan tahapan substitusi id ke setiap token.



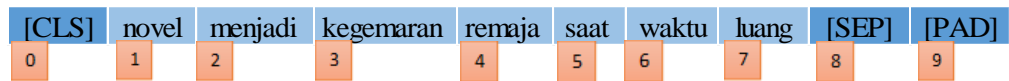
Gambar 3. 14 Tahap Substitusi ID dari Setiap Token

5. Kemudian, untuk membedakan kalimat pertama, kalimat selanjutnya atau *padding* ditandai dengan *Sentence embedding*. Pada contoh ini dilakukan pemberian angka 1 untuk kalimat pertama dan angka 0 untuk token [PAD]. *Tokenizer* akan membedakan kalimat pertama, kalimat selanjutnya, atau *padding* dengan melihat token [SEP] yang berguna sebagai pembatas antar kalimat. Gambar 3. 15 *Tahap Sentence Embedding*, menunjukkan Langkah Langkah di mana token-token diberikan angka 1 dan hanya token [PAD] yang diberi angka 0 untuk tahapan *Sentence Embedding*.



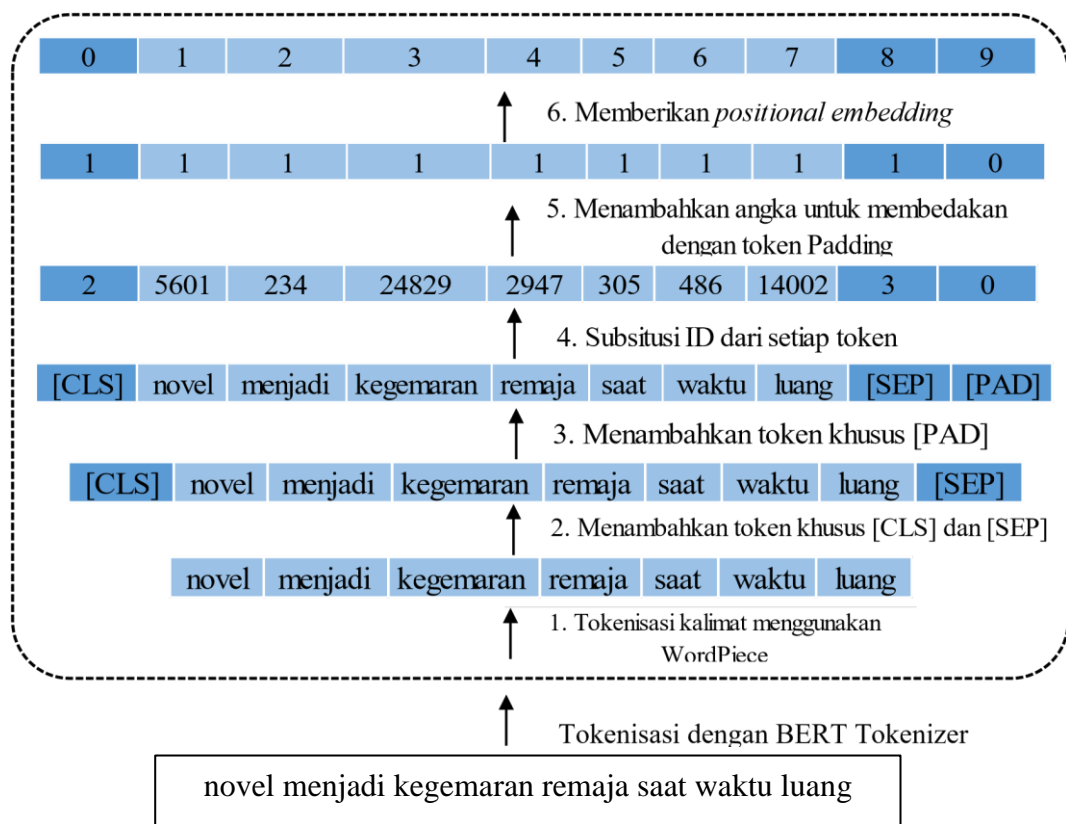
Gambar 3. 15 Tahap Sentence Embedding

6. Selanjutnya, pada tahapan *positional embedding* setiap token akan diberikan angka. Tahap ini berfungsi untuk mengetahui posisi tiap kata pada kalimat. Sehingga ketika di dalam suatu kalimat memiliki kata yang sama tetapi memiliki makna berbeda, maka BERT akan memberlakukan token tersebut dengan makna yang berbeda juga. Gambar 3. 16 *Tahap Positional Embedding*, menunjukkan tahapan positional embedding.

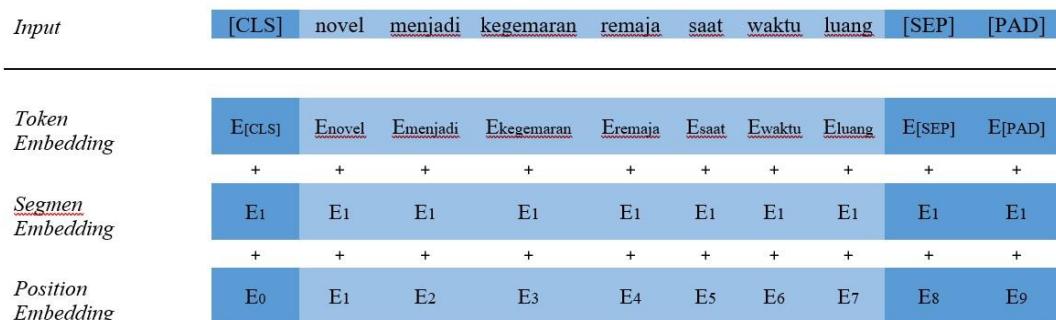


Gambar 3. 16 Tahap Positional Embedding

Pada Gambar 3. 17 *Proses Tokenisasi BERT*, menunjukkan tentang keseluruhan tahapan dalam mengubah kalimat menjadi representasi *input* BERT. Sehingga bentuk representasi *input* yang akan diolah oleh BERT dapat ditunjukkan pada Gambar 3. 18 *Representasi Input dan Output pada Tokenisasi BERT*.

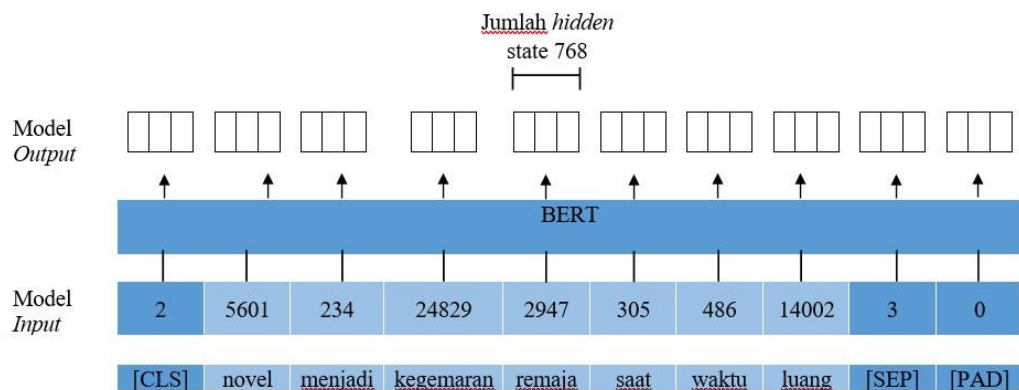


Gambar 3. 17 Proses Tokenisasi BERT



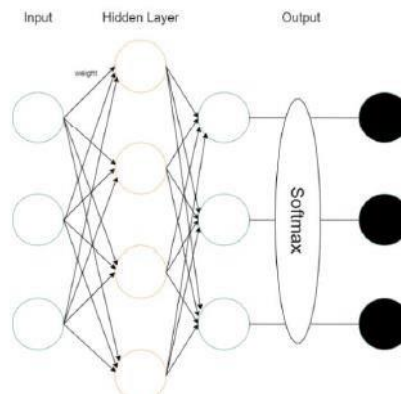
Gambar 3. 18 Representasi Input dan Output pada Tokenisasi BERT.

Setelah diubah menjadi bentuk representasi, BERT akan menerimanya sebagai *input* dan akan diteruskan melalui tumpukan *encoder*. Setiap *encoder* akan melalui *self-attention* dan *feed-forward network* dan *output* dari *encoder* akan berlanjut ke *encoder* selanjutnya. Pada penelitian ini menggunakan model BERT_{BASE}, maka akan melalui dua belas kali proses *encoder*. Setelah melalui tahapan *encoder*, setiap token akan memberikan *output* berupa vektor *hidden size* pada BERT_{BASE} berukuran 768 seperti pada Gambar 3. 19 *Input dan Output pada Model BERT*. Untuk proses analisis sentimen, *output* dari token [CLS] digunakan sebagai *input* untuk proses klasifikasi.



Gambar 3. 19 Input dan Output pada Model BERT

BERT akan mencapai hasil terbaik ketika menggunakan *neural network* tunggal pada proses klasifikasi. [29]. Dalam proses klasifikasi menggunakan *fully connected neural network* dengan fitur *softmax* seperti pada Gambar 3. 20 *Ilustrasi layer untuk Analisis Klasifikasi Sentimen*. Jadi *output* yang digunakan untuk klasifikasi berasal dari vektor [CLS]. vektor [CLS] diibaratkan dapat mengumpulkan rata-rata token kata untuk vektor pada kalimat. Lapisan terakhir dari lapisan klasifikasi akan menghasilkan *logits*. *Logits* ialah hasil prediksi *probability* kasar dari kalimat yang akan diklasifikasikan. *Softmax* dapat mengubah *logits* menjadi *probability* dengan mengambil eksponen untuk setiap nilai *logits* sehingga jumlah seluruh nilai probabilitasnya tepat 1 dan nilai probabilitasnya di antara 0 dan 1.



Gambar 3. 20 Ilustrasi layer untuk Analisis Klasifikasi Sentimen

Misalkan terdapat $logits = \begin{pmatrix} z_1 \\ z_2 \\ z_3 \end{pmatrix} = \begin{pmatrix} 8 \\ 5 \\ 2 \end{pmatrix}$ yang perlu diubah menjadi distribusi

probabilitas sehingga dapat diklasifikasi ke dalam kelas sentimen positif, netral, dan negatif menggunakan fungsi *softmax*. Tahapan untuk memperoleh nilai probabilitas dengan cara:

A. Hitung eksponensial di setiap elemen pada *logits*.

$$\begin{aligned} e^{z_1} &= e^8 = 2980.96 \\ e^{z_2} &= e^5 = 148.4 \\ e^{z_3} &= e^2 = 7.39 \end{aligned}$$

B. Normalisasikan nilai dengan menambahkan semua hasil eksponensial.

$$\sum_{j=1}^K e^{z_j} = 2980.96 + 148.4 + 7.39 = 3136.75$$

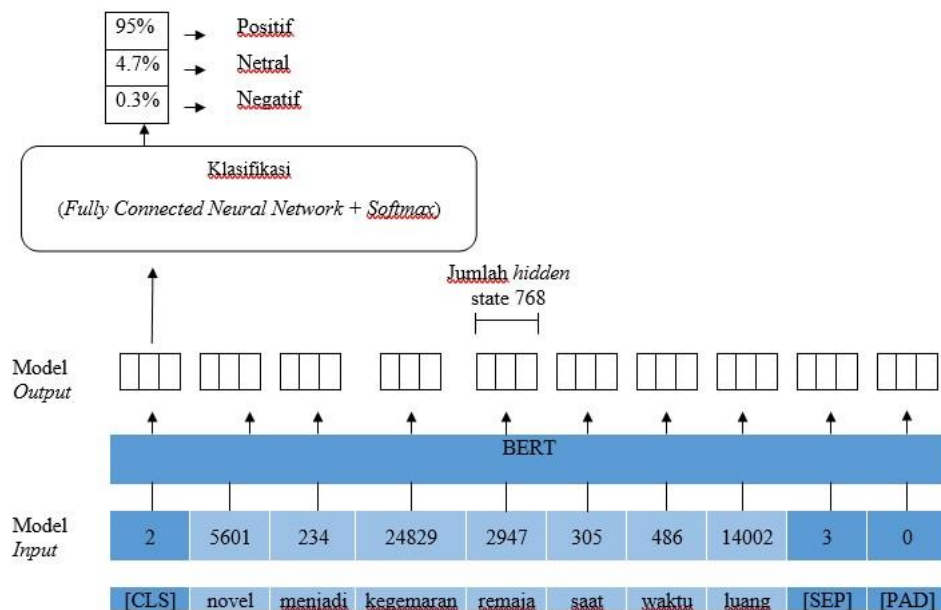
C. Bagilah tiap nilai eksponensial dengan normalitasnya untuk mendapatkan *output softmax* dari setiap elemen.

$$\begin{aligned} \sigma(\vec{z_1}) &= \frac{2980.96}{3136.75} = 0.95 \\ \sigma(\vec{z_2}) &= \frac{148.4}{3136.75} = 0.047 \\ \sigma(\vec{z_3}) &= \frac{7.39}{3136.75} = 0.003 \end{aligned}$$

Maka semua probabilitas dijumlahkan: $0.95 + 0.047 + 0.003 = 1$.

Dari Hasil probabilitas akan menunjukkan peluang sentimen dari komentar yang ada. Maka komentar tersebut memiliki probabilitas sebesar 0.95 untuk

sentimen positif, 0.047 untuk sentimen netral, dan 0.003 untuk sentimen negatif. Hal tersebut dilakukan ke semua komentar yang ada dalam dataset. Selain itu, dibutuhkan juga lapisan *dropout* yang berfungsi untuk mencegah jaringan agar tidak mengalami *overfitting*. Gambar 3. 21 *Ilustrasi Proses Klasifikasi BERT*, menunjukkan proses klasifikasi menggunakan klasifikasi BERT.



Gambar 3. 21 Ilustrasi Proses Klasifikasi BERT

3.5. Evaluasi Model

Evaluasi model berfungsi untuk mengukur seberapa baik model dalam mengklasifikasikan suatu kelas. Bentuk evaluasi model yang digunakan pada penelitian ini adalah confusion matrix. *Confusion matrix* ialah pengukuran performa kinerja model pada klasifikasi *machine learning* dengan output berupa dua kelas atau lebih [30]. Parameter yang digunakan untuk data uji adalah *true positive* (TP), *false positive* (FP), *true neutral* (TNt), *false neutral* (FNt), *true negative* (TN), dan *false negative* (FN). *True positive* ialah kalimat yang memiliki pandangan positif dan hasil prediksi menunjukkan sebagai sentimen positif. *False positive* (FP) ialah kalimat yang memiliki pandangan negatif dan netral tetapi hasil prediksi menunjukkan positif. *True neutral* (TNt) ialah kalimat yang memiliki pandangan netral dan hasil prediksi menunjukkan sebagai sentiment netral. *False*

neutral (FNt) ialah kalimat yang memiliki pandangan positif dan negatif tetapi hasil yang diprediksi menunjukkan sentimen netral. *True negative* (TN) ialah kalimat yang memiliki pandangan negatif dan hasil prediksi menunjukkan sentimen negatif. *False negative* (FN) ialah kalimat yang memiliki pandangan positif dan netral tetapi hasil yang diprediksi menunjukkan negatif.

Tabel 3. 8 Confusion matrix

Confusion matrix		Predicted		
		Positif	Netral	Negatif
Actual	Positif	TP	FNt	FN
	Netral	FP	TNt	FN
	Negatif	FP	FNt	FN

Dari *confusion matrix* tersebut dapat menghasilkan sebuah akurasi, *precision*, dan *recall*. Nilai akurasi bertujuan untuk mewakili presentasi input yang diprediksi dengan benar oleh sistem. Ketika nilai *loss* menurun maka nilai akurasi menjadi lebih baik atau lebih tinggi. *Precision* bertujuan menggambarkan akurasi antara data yang diminta dengan hasil prediksi yang diberikan oleh model. *Recall* bertujuan untuk menggambarkan keberhasilan model dalam menentukan kembali sebuah informasi yang ada. Rumus yang digunakan untuk menghitung akurasi *precision*, dan *recall* sebagai berikut:

Tabel 3. 9 Rumus Akurasi, *Precision*, dan *Recall*

Akurasi	$\frac{TP + TNt + TN}{TP + FP + TNt + FNt + TN + FN}$
$Precision_{positive}$	$\frac{TP}{TP + FP}$
$recall_{positive}$	$\frac{TP}{TP + FNt + FN}$

3.6. Kurva ROC dan AUC

Kurva *Receiving Operating Characteristic* (ROC) adalah kurva yang merepresentasikan hubungan antara TPR (*True Positive Rate*) / Recall / *Sensitivity* dan FPR (*False Positive Rate*) / 1- *Specificity*. TPR adalah presentase prediksi benar positif yang dibandingkan dengan keseluruhan data yang benar positif. Sedangkan

FPR adalah presentasi kesalahan memprediksi positif, pada saat data aslinya bernilai negatif, atau seberapa banyak proporsi kelas negatif yang salah diklasifikasikan oleh komputer. Kegunaan kurva ROC adalah untuk menggambarkan keakuratan diagnosis dan untuk mengatur kinerja klasifikasi tanpa melihat distribusi kelas. Semakin tinggi TNR dan FPR yang rendah, maka model akan dianggap semakin baik. *Specificity* berguna untuk melihat proporsi kelas negatif yang diklasifikasikan dengan benar. Nilai untuk TNR dan FPR didapat berdasarkan nilai dari *confusion matrix*. Rumusan untuk TNR dan FPR sebagai berikut [36]:

$$TPR/Recall = \frac{TP}{TP + FN} \quad (3.3)$$

$$FPR/1 - Specificity = \frac{FP}{TN + FP} \quad (3.4)$$

$$Specificity = \frac{TN}{TN + FP} \quad (3.5)$$

Area Under the Curve (AUC) adalah metode yang digunakan untuk menghitung nilai dibawah kurva ROC. AUC berguna untuk mebedakan antar kelas dan digunakan sebagai ringkasan dari kurva ROC. Oleh karena itu, semakin tinggi nilai AUC, maka semakin baik juga kinerja model dalam membedakan antar kelas positif dan negatif, nilai AUC dijadikan sebagai tujuan dalam pemaksimalan kinerja model. Nilai AUC selalu berada pada range 0 - 1, karena batas bagian luar persegi dengan sumbu x dan sumbu y memiliki nilai 0 sampai 1. Ketika nilai $0.5 < AUC < 1$, ada kemungkinan besar model akan dapat membedakan nilai kelas positif dari nilai kelas negatif. Hal ini terjadi karena model mampu mendeteksi lebih banyak jumlah *True positives* dan *True negatives* daripada *False negatives* dan *False positives*. Keakuratan hasil kualitas klasifikasi menggunakan nilai AUC dapat dilihat dari tabel berikut [33]:

Tabel 3. 10 Kategori Klasifikasian Model Berdasarkan Nilai AUC

Nilai AUC	Model Diklasifikasi Sebagai
0.90 – 1.00	<i>Excellent</i>
0.80 – 0.90	<i>Very Good</i>
0.70 – 0.80	<i>Good</i>
0.60 – 0.70	<i>Fair</i>
0.50 – 0.60	<i>Poor</i>

BAB IV

PEMBAHASAN

4.1. Kondisi Data

Berdasarkan hasil proses *scrapping* terdapat 8000 data ulasan yang diterima, tetapi terdapat 53 data ulasan yang tidak sah, karena hanya berisi simbol-simbol atau ulasan tanpa rating, sehingga data ulasan tersebut dihapus. Sehingga total data yang digunakan saat proses klasifikasi berjumlah 7947 data ulasan. Kondisi data ulasan yang dihapus ditunjukkan pada tabel berikut:

Tabel 4. 1 Contoh Beberapa Ulasan yang dihapus

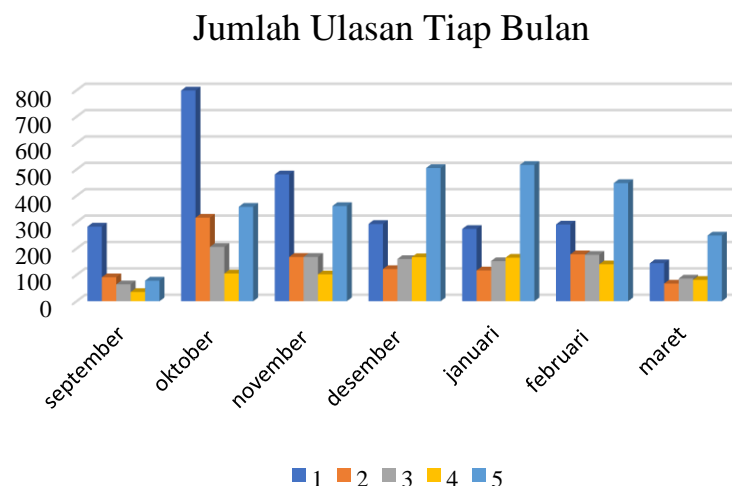
<i>No.</i>	<i>Date</i>	<i>Version</i>	<i>User</i>	<i>Review</i>	<i>Rating</i>
1	-	-	Selly Mrln	Lagi down kah? Mau baca book tapi katanya lagi offline padahal wifi aku baik baik aja dan arusnya kuat:(sekarang cuma bisa baca book yang sudah dipilih offline. Tolong ini kenapaaaâ€€)à®†Đ'à®†	-
2	-	-	x penggung	Kok sekarang berbayar ya ? Padahal kemarin enggak terus awalnya aku kan unistal karena penyimpanan penuh , terus aku download lagi kok berbayar	-
3	10/03/ 2021 09:14	9.3.0	Rappolem ba Fc	!!!!Â¡!!!!!!?!!?!!?!!?!!?!!?!!?!!?!! ?!!?!!?!!?!!?!!?!!?!!?!!?!!?!!?!! !!!!!!	5
4	09/02/ 2021 21:47	8.94.0	Rahmawat iririn Djabili	ðŸ§¡ ðŸ™ðŸ— ðŸ™ðŸ™ðŸ— ðŸ”ðŸ— ðŸŒðŸšðŸ¥ðŸ’ ðŸŸðŸ”ðŸ’ ðŸžðŸ—	2

Contoh data ulasan nomor 1 dan 2 pada Tabel 4.1 *Contoh Beberapa Ulasan yang dihapus*, menunjukkan bahwa kalimat ulasan hanya berisi *review* tanpa rating yang diberikan oleh pengguna. Pada data ulasan nomor 3 dan 4 menunjukkan bahwa kalimat ulasan yang tidak dapat dipahami karena berisi tentang simbol. Oleh karena itu, data ulasan seperti kalimat 1,2,3 dan 4 harus dihapus. Untuk melihat semua data dapat dilihat pada Lampiran I.

Tabel 4. 2 Jumlah Ulasan Tiap Rating

Rating	Jumlah Ulasan
1	2565
2	1059
3	1012
4	795
5	2516

Menurut Tabel 4.2 *Jumlah Ulasan Tiap Rating*, pada rating 1 sebagai rating tertinggi di antara rating lainnya yaitu sebanyak 2565 dan rating 4 sebagai jumlah rating paling sedikit yaitu sebanyak 795.

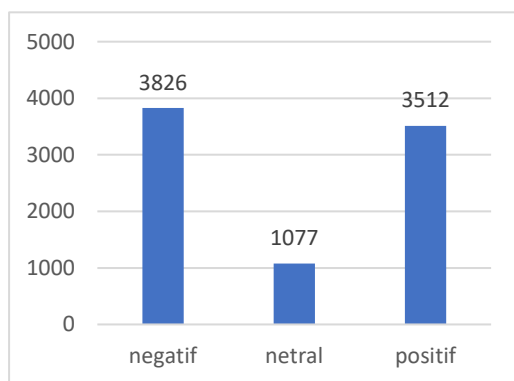


Gambar 4. 1 Grafik Jumlah Ulasan Tiap Bulan

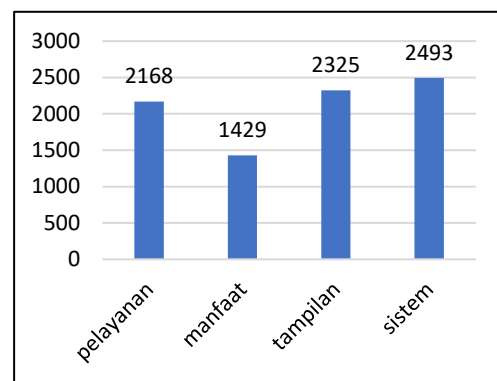
Gambar 4. 1 *Grafik Jumlah Ulasan Tiap Bulan*, menunjukkan perubahan jumlah ulasan aplikasi novel *online* setiap bulan berdasarkan rating, selama September 2020 sampai Maret 2021. Terlihat bahwa ulasan dengan rating 1 dan 5 mendominasi setiap bulannya. Ulasan rating 1 di bulan Oktober menjadi jumlah

terbanyak diantara bulan lainnya dan ulasan rating 5 di bulan Januari jumlah terbanyak diantara bulan lainnya.

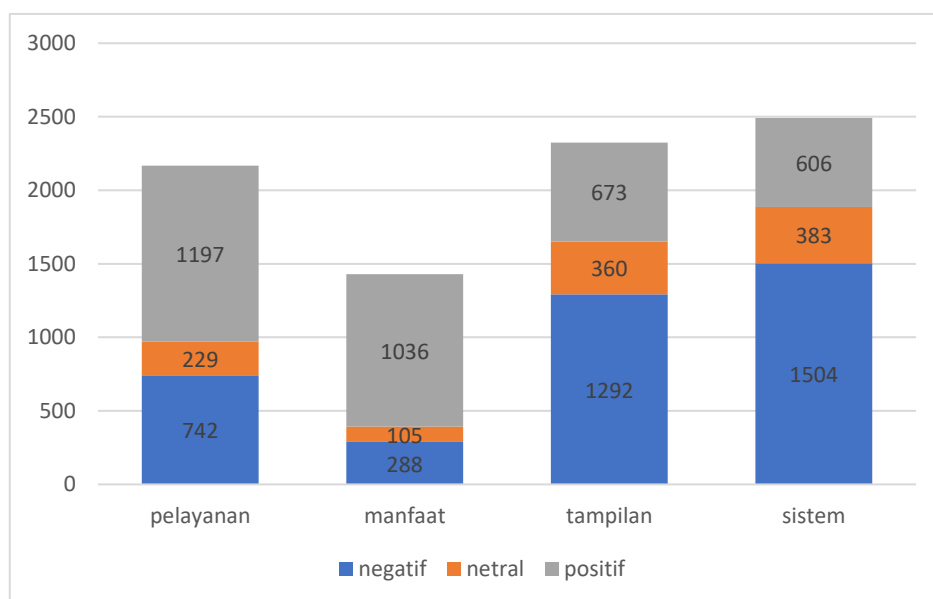
Label yang digunakan untuk klasifikasi sentimen pada penelitian ini menggunakan sentimen negatif, netral, dan positif. Pada klasifikasi aspek terdapat 4 aspek dominan yang didapatkan berdasarkan proses LDA dan pada saat mendeteksi aspek terdapat ulasan yang memiliki double aspek yang dilihat dari kata-kata yang terkandung dalam ulasan. Jumlah data yang digunakan untuk proses klasifikasi berjumlah 8415 ulasan yang dilihat berdasarkan aspek ulasan yang ditulis.



Gambar 4. 3 Jumlah Label Sentimen



Gambar 4. 4 Jumlah Label Aspek



Gambar 4. 2 Jumlah Sentimen Tiap Aspek

Pada Gambar 4. 3 *Jumlah Label Sentimen*, merupakan hasil rating dari media sosial dengan ulasan negatif (rating 1 dan rating 2) yang menjadi dominasi pada data ulasan aplikasi novel *online* yaitu sebanyak 3826 ulasan. Pada Gambar 4. 4 *Jumlah Label Aspek*, pada saat menentukan aspek menggunakan double jumlah aspek pelayanan menjadi dominasi pada ulasan aplikasi novel *online*. Sedangkan pada Gambar 4. 2 *Jumlah Sentimen Tiap Aspek*, menunjukkan jumlah sentimen untuk tiap aspek termasuk untuk ulasan yang double aspek. Karena jumlah label sentimen yang tidak seimbang, maka perlu dilakukan pengolahan seperti *resampling* data untuk mengatasi ketidakseimbangan data dengan teknik *SMOTE*.

4.2. Hasil Preprocessing

Data yang diperoleh selama proses *scrapping* harus dibersihkan sebelum diproses lebih lanjut. Proses penghapusan ini dinamakan preprocessing. Proses Preprocessing meliputi menghapus simbol, angka, emotikon, menghapus *stopword*, *case-folding*, dan mengubah kata slang. Namun peneliti tidak menghapus kata-kata negasi seperti “tidak” untuk menghindari perubahan makna atau salah menginterpretasikan ulasan topik. Hasil preprocessing akan disimpan menjadi bentuk CSV. Berikut contoh beberapa ulasan awal dan ulasan yang sudah di preprocessing:

Tabel 4. 3 Beberapa Ulasan Awal dan Hasil Processing

No.	Date	Version	Review	Cleaned_Review	Rating
1	22/09/2020 19:05	8.82.1	Banyak iklan ðŸ¥°ðŸ¥, terus jumlah perpustakaan offline dibatasi ðŸ˜ðŸ˜	iklan perpustakaan luar jaringan batasi	2
2	29/09/2020 21:34:00	8.82.1	Kecewa bngt sekarang sama aplikasi novel <i>online</i> . Udah 4thm aku bca aplikasi novel <i>online</i> tpi skrnag ko hrus bayar biar bisa baca offline. Dan daftar offline nya pun	kecewa sekarang aplikasi novel <i>online</i> baca aplikasi novel <i>online</i> sekarang bayar baca luar jaringan	2

			sedikit. Aku bner2. Maaf skrang mau beralih ke apk sebelahh	daftar luar jaringan sedikit sekarang alih aplikasi	
3	09/10/2020 19:27	8.75.1	Bagusan versi lama. Sekarang cuma bisa baca offline 2 novel. Itu pun kadang gk bisa kebuka kalau tidak pakai data dulu. Padahal udah masuk daftar offline. Tapi gk bisa d baca offline. Aneh. Makin ribet.	bagus versi lama baca luar jaringan novel tidak bisa buka tidak pakai data masuk daftar luar jaringan tidak bisa baca luar jaringan aneh ribet	1

Tabel 4. 3 *Beberapa Ulasan Awal dan Hasil Processing*, menunjukkan hasil scraping yang ditambah dengan ulasan yang telah di preprocessing yang dinamakan kolom *Cleaned_Review*.

4.3. BoW

Setelah melewati proses *preprocessing* akan dilakukan pembuatan *dictionary*. Dictionary berfungsi sebagai kata-kata yang telah ditokenisasi akan dipetakan ke dalam bentuk ID yang unik (berbeda-beda). Selanjutnya, *term weight* untuk mengetahui bobot untuk setiap kata unik dan menyaring kata-kata penting dari dokumen. Pada penellitian ini menggunakan nilai *min_df* yaitu mengabaikan kata-kata yang muncul di kurang dari 5 dokumen. Setelah proses *dictionary* dan pembobotan kata, langkah berikutnya adalah BoW menggunakan fungsi *doc2bow*.

Tabel 4. 4 Bentuk Bag of Words (BoW)

$[(0, 3), (1, 2), (2, 1), (3, 1), (4, 1), \dots]$

Berdasarkan Tabel 4. 4, dapat dilihat bahwa contoh bentuk *Bag of Words* (BoW). Misalnya $[(0, 3)]$ yang dapat diartikan sebagai id kata ke 0 muncul 3 kali, $[(1, 2)]$ yang dapat diartikan sebagai id kata ke 1 muncul 2 kali, dan seterusnya

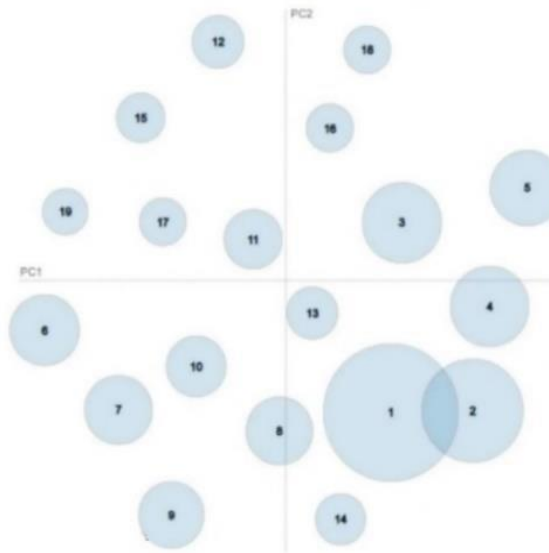
4.4. Penentuan Jumlah Aspek

Pada penelitian ini pemilihan jumlah aspek memanfaatkan nilai *coherence* tertinggi untuk setiap iterasi jumlah pembagian topik di pemodelan LDA. Pertama-tama jumlah aspek ditentukan dengan mengelompokkan seluruh data. Pada proses pengelompokkan akan menghasilkan jumlah aspek dan interpretasi yang diperoleh berdasarkan seluruh data. Pada saat menentukan jumlah aspek dilakukan dengan cara melakukan beberapa iterasi yaitu, 1 hingga 20 aspek, 1 hingga 10 aspek dan 1 hingga 5 aspek, setelah itu lihat nilai *coherence* dari masing – masing aspek. Pemilihan banyaknya aspek dilakukan dengan cara melihat nilai *coherence* yang tertinggi dari masing-masing iterasinya. Nilai *coherence* adalah tingkat kesesuaian antar kata pada topik yang diperoleh dengan cara menganalisis perbedaan/kesamaan semantik pada antar kata pada topik [34], atau suatu ukuran yang digunakan untuk mengevaluasi pemodelan topik, dimana model yang baik akan menghasilkan topik dengan nilai *coherence* yang tinggi.

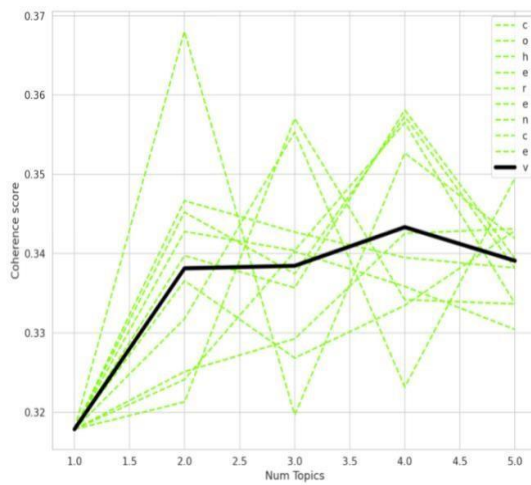
Tabel 4. 5 Penentuan jumlah Aspek

5 Aspek		10 Aspek		20 Aspek			
<i>Num</i> <i>Topic</i>	Nilai Coherence	<i>Num</i> <i>Topic</i>	Nilai Coherence	<i>Num</i> <i>Topic</i>	Nilai Coherence	<i>Num</i> <i>Topic</i>	Nilai Coherence
1	0.317856	1	0.317856	1	0.317856	11	0.346668
2	0.338146	2	0.331664	2	0.331935	12	0.344416
3	0.338456	3	0.336441	3	0.333646	13	0.350721
4	0.343332	4	0.340494	4	0.339609	14	0.350669
5	0.339101	5	0.338876	5	0.345159	15	0.353576
		6	0.34098	6	0.342356	16	0.352942
		7	0.341644	7	0.342642	17	0.349534
		8	0.34519	8	0.344792	18	0.355607
		9	0.341212	9	0.345495	19	0.353911
		10	0.346313	10	0.341255	20	0.35222

Berdasarkan Tabel 4. 5 *Penentuan jumlah Aspek*, didapatkan hasil dari masing masing iterasi topik. Pada iterasi 5 Aspek dihasilkan jumlah topik terbaik pada *num topic* 4 dengan nilai coherence sebesar 0.343332. Berikutnya pada iterasi 10 Aspek dihasilkan jumlah topik terbaik pada *num topic* 10 dengan nilai coherence sebesar 0.346313. Terakhir, pada iterasi 20 Aspek dihasilkan jumlah topik terbaik pada *num topic* 18 dengan nilai coherence sebesar 0. 355607. Berdasarkan hasil dari ketiga iterasi topik, pada iterasi 1-10 masih terlalu luas dan pada Gambar 4.5 *Visualisasi pyLDavis iterasi 1 - 20 Aspek*, masih terdapat *overlapping* yang bisa dibuat menjadi satu cluster, maka peneliti memilih iterasi 1-5 aspek dengan num topic 4 sebagai pemodelan topik dengan LDA.



Gambar 4. 5 Visualisasi Visualisasi pyLDavis
20 topik



Gambar 4. 6 Grafik Nilai Coherence untuk iterasi 1-5 Aspek



Gambar 4. 7 Visualisasi PyLDAvis num topic 4

Dapat dilihat pada Gambar 4. 6, terlihat garis hitam yang menandakan nilai coherence tertinggi berada di *num topic* 4. Pada Gambar 4. 7, menunjukkan pemetaan jarak antar topik dengan jumlah cluster topik sebanyak 4 cluster. Ukuran cluster topik menunjukkan tentang seberapa penting topik dalam corpus. Dalam visualisasi pyLDAvis terbagi menjadi dua sumbu yaitu PC1 (*Principal Components1*) dan PC2 (*Principal Components2*) yang merupakan nama lain dari titik koordinat sumbu x dan y [35]. Letak setiap *cluster* topik bergantung pada titik koordinat masing-masing cluster, yang dilihat berdasarkan PC (*Principal Components*). Nilai PC untuk setiap cluster topik antara lain:

Tabel 4. 6 Nilai *Principal Components* Untuk Setiap Cluster Topik

Topik	PC1	PC2
1	-0.055930	-0.000146
2	-0.083308	0.045723
3	0.030234	-0.111322
4	0.109005	0.065745

Pada Tabel 4. 6, menunjukkan hasil nilai PC untuk masing-masing cluster yang dipakai untuk visualisasi pyLDAvis. pyLDAvis merupakan library yang terdapat pada Python yang berguna untuk membantu peneliti menafsirkan topik yang sesuai pada kumpulan data. Letak cluster yang saling berdekatan mempunyai arti kata-kata yang terkandung terdapat kecenderungan sama. Model topic yang

baik adalah yang memiliki kata-kata yang berbeda disetiap cluster topiknya. Setiap Cluster akan mengantung berbagai macam distribusi kata dan selanjutnya akan diinterpretasikan berdasarkan tiap topiknya. Kata-kata yang bercetak tebal mewakili interpretasi topik daripada kata-kata lain di setiap cluster topik. Hasil distribusi kata setiap cluster ditunjukkan sebagai berikut:

Tabel 4. 7 Sebaran Kata Tiap Cluster

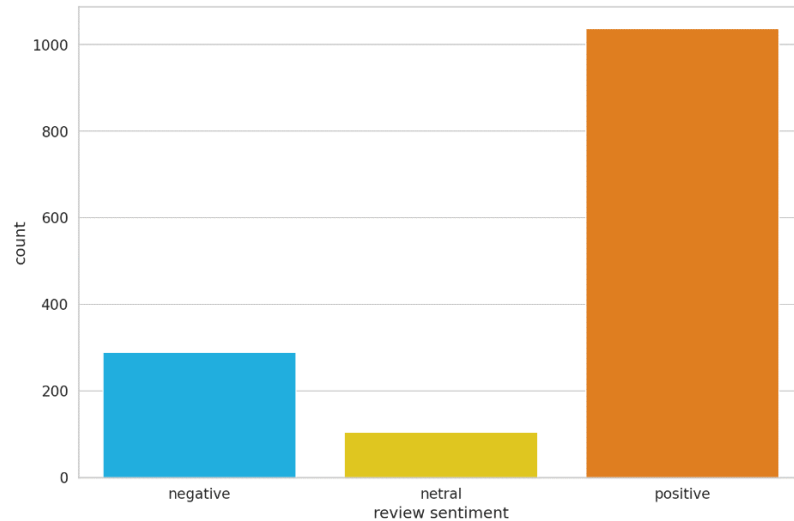
Topik	Top Word	Interpretasi
1	suka aplikasi baca tulisan halusinasi karya penulis hobi bintang novel komentar cocok	Manfaat
2	iklan keren hilang keluar nyaman premium batasi cari kabar baru	Tampilan
3	luar jaringan versi lama kecewa batasi dalam jaringan perbarui bayar premium ganggu perpustakaan versi baru	Sistem
4	bagus masuk seru akun eror cerita lambat unduh email lama perbarui	Pelayanan

4.5. Hasil Klasifikasi Sentimen Berbasis Aspek

Dalam penelitian ini, hasil klasifikasi sentimen dibagi menjadi berdasarkan aspeknya, yaitu manfaat, tampilan, sistem dan pelayanan. Metode yang digunakan pada penelitian ini adalah metode model BERT (*Bidirectional Encoder Representations from Transformers*). Hyperparameter yang digunakan pada pengujian ini adalah beberapa *learning rate* yaitu $[10^{-5}, 2 \cdot 10^{-5}, 3 \cdot 10^{-5}]$ [28] [29]. Berdasarkan rekomendasi penelitian terdahulu, *batch size* yang digunakan yaitu *batch size* 32 [29], berdasarkan rekomendasi penelitian terdahulu. Jika ukuran *batch* yang dipilih lebih besar, maka akan membutuhkan waktu yang lebih lama untuk memproses satu *batch size* [8]. Jumlah urutan maksimum kata setelah dilakukannya stopword adalah 64 kata, oleh karena itu peneliti menggunakan urutan maksimum kata yaitu 64 kata pada saat token embedding. *Layer* tambahan yang digunakan adalah *dropout* dengan probabilitas 0.1 [19]. Parameter lain yang membantu proses training yaitu *epoch* sebesar 4 berdasarkan *early stopping*. Hasil dari klasifikasi sebagai berikut:

4.5.1. Hasil Klasifikasi Sentimen Pada Aspek Manfaat

Data yang digunakan untuk klasifikasi sentiment pada aspek manfaat sebanyak 1429 data. Label kelas yang digunakan untuk klasifikasi sentimen berdasarkan aspek manfaat yaitu kelas sentimen negatif, netral, dan positif, berikut banyaknya jumlah ulasan per kelas pada aspek manfaat:

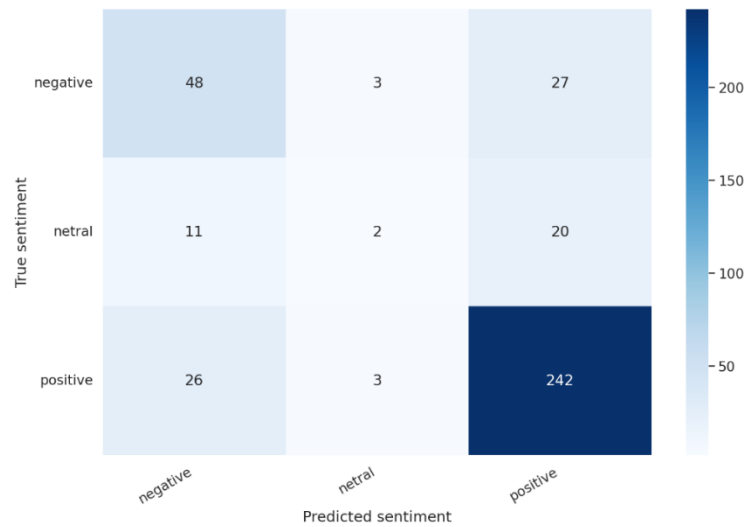


Gambar 4. 8 Jumlah Ulasan Per Kelas pada Aspek Manfaat

Berdasarkan Gambar 4. 8 menunjukkan bahwa kelas sentimen positif yang menjadi dominasi dengan jumlah ulasan sebanyak 1036 ulasan. Sedangkan untuk kelas sentimen netral yang menjadi minoritas dengan jumlah ulasan sebanyak 105 ulasan. Pada saat klasifikasi data dibagi menjadi 3 yaitu data training, data validation dan data testing. Data training adalah data yang dilatih untuk membangun model dengan ukuran 70% dari data yaitu 952 data. Data validation adalah data untuk proses validasi model dan untuk proses minimalisasi *overfitting* sebanyak 95 data, dan data testing adalah data untuk menguji performa model yang sudah dilatih sebanyak 382 data.

1. Hasil Klasifikasi Sentimen pada Aspek Manfaat dengan *Learning Rate* 10^{-5}

Berikut hasil *confusion matrix* dari hasil pengujian model menggunakan *learning rate* 10^{-5} .



Gambar 4. 9 Hasil *Confusion Matrix* Sentimen pada Aspek Manfaat dengan *learning rate* 10^{-5}

Berdasarkan Gambar 4. 9, hasil data test menunjukkan bahwa sebanyak 48 data terklasifikasi dengan benar sebagai ulasan negatif, untuk ulasan positif sebanyak 242 data. Sedangkan pada ulasan netral sebanyak 2 data. Berdasarkan hasil tabel *confusion matrix*, maka dapat dihitung nilai akurasi sebagai berikut:

$$\begin{aligned}
 \text{Akurasi} &= \frac{TN + TNt + TP}{TN + FN + TNt + FNt + TP + FP} \\
 &= \frac{48 + 3 + 27 + 11 + 2 + 20 + 26 + 3 + 242}{48 + 3 + 27 + 11 + 2 + 20 + 26 + 3 + 242} = 0.764
 \end{aligned}$$

Pada hasil tabel *confusion matrix* juga dapat melihat tingkat kepercayaan model dengan melihat nilai *precision*, *recall* dan *f1-score* yang bertujuan untuk mengetahui seberapa baik model dalam memprediksi kelas.

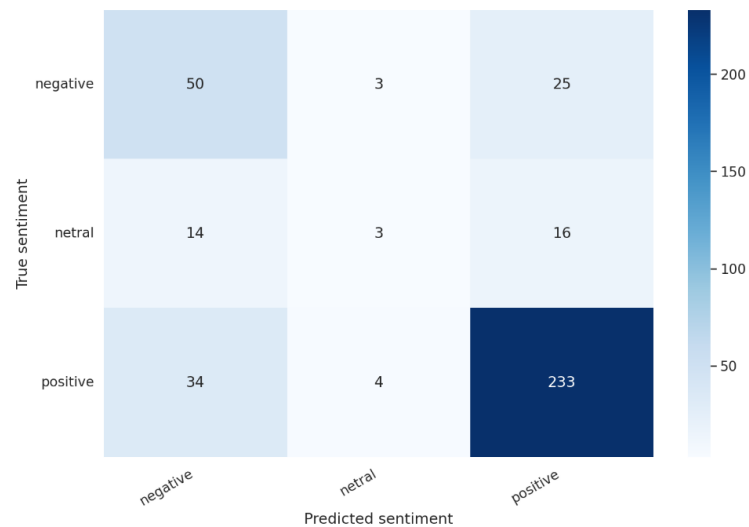
Tabel 4. 8 Hasil *Confusion Matrix* Sentimen pada Aspek Manfaat dengan *learning rate* 10^{-5}

	<i>Precision</i>	<i>Recall</i>	<i>F1-Score</i>
Negatif	0.56	0.62	0.59
Netral	0.25	0.06	0.10
Positif	0.84	0.89	0.86

Berdasarkan Tabel 4. 8, Hasil evaluasi kinerja klasifikasi model cukup baik, *recall* pada sentimen negatif dan positif cukup tinggi yang berarti sistem dapat memprediksi dengan benar untuk data bersentimen negatif dan positif. Sedangkan *recall* pada sentimen netral sangat rendah, yang berarti data pada sentimen netral tidak dapat terprediksi dengan benar oleh sistem.

2. Hasil Klasifikasi Sentimen pada Aspek Manfaat dengan *Learning Rate* $2 \cdot 10^{-5}$

Berikut hasil *confusion matrix* dari hasil pengujian model menggunakan *learning rate* $2 \cdot 10^{-5}$.



Gambar 4. 10 Hasil *Confusion Matrix* Sentimen pada Aspek Manfaat dengan *learning rate* $2 \cdot 10^{-5}$

Berdasarkan Gambar 4. 10, hasil data test menunjukkan bahwa sebanyak 50 data terklasifikasi dengan benar sebagai ulasan negatif, untuk ulasan positif sebanyak 233 data. Sedangkan pada ulasan netral sebanyak 3 data. Berdasarkan hasil tabel *confusion matrix*, maka dapat dihitung nilai akurasi sebagai berikut:

$$\begin{aligned}
 \text{Akurasi} &= \frac{TN + TNt + TP}{TN + FN + TNt + FNT + TP + FP} \\
 &= \frac{50 + 3 + 233}{50 + 3 + 25 + 14 + 3 + 16 + 34 + 4 + 233} = 0.748
 \end{aligned}$$

Pada hasil tabel *confusion matrix* juga dapat melihat tingkat kepercayaan model dengan melihat nilai precision, recall dan f1-score yang

bertujuan untuk mengetahui seberapa baik model dalam memprediksi kelas.

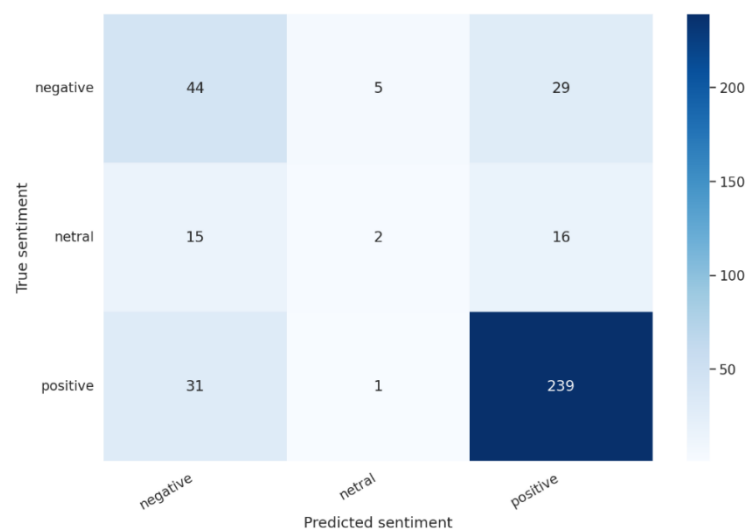
Tabel 4. 9 Evaluasi *Confusion Matrix* Sentimen pada Aspek Manfaat dengan *learning rate* $2 \cdot 10^{-5}$

	Precision	Recall	F1-Score
Negatif	0.51	0.64	0.57
Netral	0.30	0.09	0.14
Positif	0.85	0.86	0.86

Berdasarkan Tabel 4. 9, Hasil evaluasi kinerja klasifikasi model cukup baik, recall pada sentimen negatif dan positif cukup tinggi yang berarti sistem dapat memprediksi dengan benar untuk data bersentimen negatif dan positif. Sedangkan recall pada sentimen netral sangat rendah, yang berarti data pada sentimen netral tidak dapat terprediksi dengan benar oleh sistem.

3. Hasil Klasifikasi Sentimen pada Aspek Manfaat dengan *Learning Rate* $3 \cdot 10^{-5}$

Berikut hasil *confusion matrix* dari hasil pengujian model menggunakan *learning rate* $3 \cdot 10^{-5}$



Gambar 4. 11 Hasil *Confusion Matrix* Sentimen pada Aspek Manfaat dengan *learning rate* $3 \cdot 10^{-5}$

Berdasarkan Gambar 4. 11, hasil data test menunjukkan bahwa sejumlah 44 data terklasifikasi dengan benar sebagai ulasan negatif, untuk ulasan

positif sebanyak 239 data. Sedangkan pada ulasan netral sebanyak 2 data. Berdasarkan hasil tabel *confusion matrix*, maka dapat dihitung nilai akurasi sebagai berikut:

$$\begin{aligned}
 \text{Akurasi} &= \frac{TN + TNt + TP}{TN + FN + TNt + FNT + TP + FP} \\
 &= \frac{44 + 2 + 239}{44 + 5 + 29 + 15 + 2 + 16 + 31 + 1 + 239} = 0.746
 \end{aligned}$$

Pada hasil tabel *confusion matrix* juga dapat melihat tingkat kepercayaan model dengan melihat nilai precision, recall dan f1-score yang bertujuan untuk mengetahui seberapa baik model dalam memprediksi kelas.

Tabel 4. 10 Evaluasi *Confusion Matrix* Sentimen pada Aspek Manfaat dengan *learning rate* $3 \cdot 10^{-5}$

	Precision	Recall	F1-Score
Negatif	0.49	0.56	0.52
Netral	0.25	0.06	0.10
Positif	0.84	0.88	0.86

Berdasarkan Tabel 4. 10, Hasil evaluasi kinerja klasifikasi model cukup baik, recall pada sentimen negatif dan positif cukup tinggi yang berarti sistem dapat memprediksi dengan benar untuk data bersentimen negatif dan positif. Sedangkan recall pada sentimen netral sangat rendah, yang berarti data pada sentimen netral tidak dapat terprediksi dengan benar oleh sistem.

Dari 3 model yang telah diuji, selanjutnya akan dibandingkan nilai akurasi dari masing masing *learning rate* terlihat pada Tabel 4. 10.

Tabel 4. 11 Perbandingan Nilai Akurasi dari Setiap *Learning Rate* pada Aspek Manfaat

<i>Learning rate</i>	Accuracy	AUC
10^{-5}	0.764	0.893
$2 \cdot 10^{-5}$	0.748	0.881
$3 \cdot 10^{-5}$	0.746	0.877

Dapat dilihat dari Tabel 4. 11, bahwa *learning rate* 10^{-5} memiliki nilai akurasi terbesar dibandingkan dengan yang lainnya

yaitu sebesar 0.764 dan nilai AUC sebesar 0.893. Berdasarkan hasil test percobaan, didapatkan model dengan nilai parameter terbaik yaitu pada *learning rate* 10^{-5} .

Berdasarkan hasil tabel *confusion matrix* 3 kali percobaan *learning rate* dapat dilihat bahwa sistem mengalami kesulitan dalam mengklasifikasikan sentimen netral. Namun, saat pada ulasan positif dan negatif sistem cenderung memberikan klasifikasi dengan benar, hal ini terjadi karena jumlah ulasan yang tidak seimbang antara ulasan positif, negatif dan negatif atau biasa disebut dengan data *imbalanced*.

Untuk memperbaiki keseimbangan kelas peneliti menggunakan teknik SMOTE. Teknik *Synthetic Minority Over-sampling Technique* (SMOTE) adalah teknik yang digunakan untuk menangani ketidakseimbangan kelas. Teknik ini mereplikasikan data minoritas atau biasa dikenal dengan data sintesis (data buatan) pada kelas minoritas dengan cara menambah data kelas pada kelas minoritas untuk menyeimbangkan kelas mayoritas [36]. Data sintesis dilakukan berdasarkan *k-nearest neighbors* untuk setiap data di kelas minoritas.

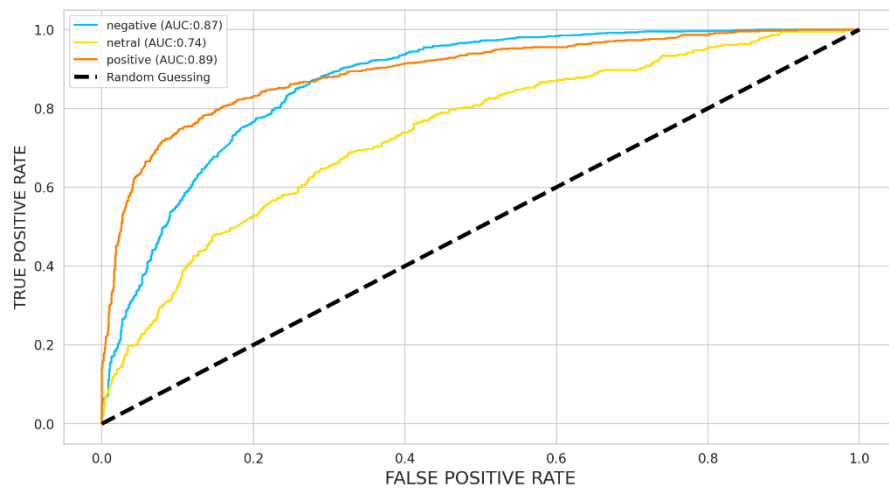
Tabel 4. 12 Evaluasi *Confusion Matrix* Sentimen pada Aspek Manfaat dengan Teknik SMOTE

	Precision	Recall	F1-Score
Negatif	0.76	0.82	0.79
Netral	0.78	0.17	0.28
Positif	0.93	0.98	0.96

Akurasi: 89%; AUC: 97%

Berdasarkan hasil Tabel 4. 12 *Evaluasi Confusion Matrix Teknik SMOTE*, dapat disimpulkan bahwa model klasifikasi sentiment pada aspek manfaat memperoleh nilai akurasi (89%) dan AUC (97%) lebih tinggi dibandingkan tanpa SMOTE dengan nilai akurasi (76%) dan AUC (89%). Hal tersebut menunjukkan bahwa model yang telah melalui teknik SMOTE yang dapat meningkatkan performa dan nilai akurasi. Kinerja model sudah cukup baik dengan *recall* pada sentimen negatif dan positif cukup tinggi,

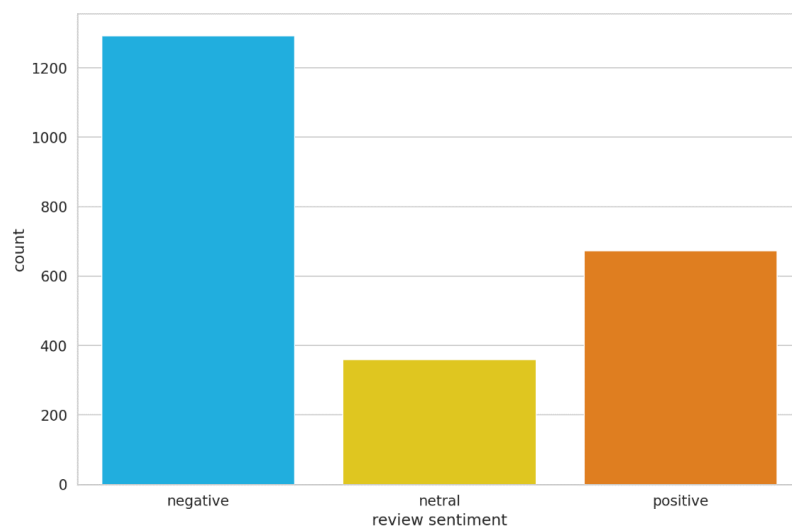
yang berarti data pada sentimen negatif dan positif dapat dideteksi secara benar, tetapi pada sentimen netral masih cukup rendah, yang berarti data pada sentimen netral tidak dapat dideteksi secara benar. Berikut kurva ROC dan AUC dari data testing:



Gambar 4. 12 kurva ROC dan AUC Aspek Manfaat menggunakan Teknik SMOTE

4.5.2. Hasil Klasifikasi Sentimen Pada Aspek Tampilan

Data yang digunakan untuk klasifikasi sentiment pada aspek tampilan sebanyak 2325 data. Label kelas yang digunakan untuk klasifikasi sentimen berdasarkan aspek manfaat yaitu kelas sentimen negatif, netral, dan positif, berikut banyaknya jumlah ulasan per kelas pada aspek tampilan:

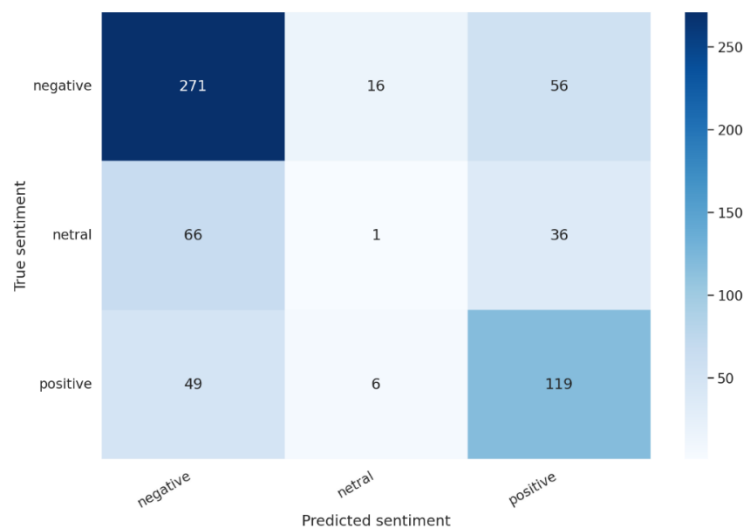


Gambar 4. 13 jumlah ulasan per kelas pada aspek tampilan

Berdasarkan Gambar 4. 13, menunjukkan bahwa kelas sentimen negatif yang menjadi dominasi dengan jumlah ulasan sebanyak 1292 ulasan. Sedangkan untuk kelas sentimen netral yang menjadi minoritas dengan jumlah ulasan sebanyak 360 ulasan. Pada saat klasifikasi data dibagi menjadi 3 yaitu data training, data validation dan data testing. Data training adalah data yang dilatih untuk membangun model dengan ukuran 70% dari data yaitu 1550 data. Data validation adalah data untuk proses validasi model dan untuk proses minimalisasi *overfitting* sebanyak 155 data, dan data testing adalah data untuk menguji performa model yang sudah dilatih sebanyak 620 data.

1. Hasil Klasifikasi Sentimen pada Aspek Tampilan dengan *Learning Rate* 10^{-5}

Berikut hasil *confusion matrix* dari hasil pengujian model menggunakan *learning rate* 10^{-5} .



Gambar 4. 14 Hasil Confusion matrix Sentimen pada Aspek Tampilan dengan *learning rate* 10^{-5}

Berdasarkan Gambar 4. 14, hasil data test menunjukkan bahwa sebanyak 271 data terklasifikasi dengan benar sebagai ulasan negatif, untuk ulasan positif sebanyak 119 data. Sedangkan pada ulasan netral sebanyak 1

data. Berdasarkan hasil tabel *confusion matrix*, maka dapat dihitung nilai akurasi sebagai berikut:

$$\begin{aligned}
 \text{Akurasi} &= \frac{TN + TNt + TP}{TN + FN + TNt + FNt + TP + FP} \\
 &= \frac{271 + 1 + 119}{271 + 16 + 56 + 66 + 1 + 36 + 49 + 6 + 119} = 0.630
 \end{aligned}$$

Pada hasil tabel *confusion matrix* juga dapat melihat tingkat kepercayaan model dengan melihat nilai precision, recall dan f1-score yang bertujuan untuk mengetahui seberapa baik model dalam memprediksi kelas.

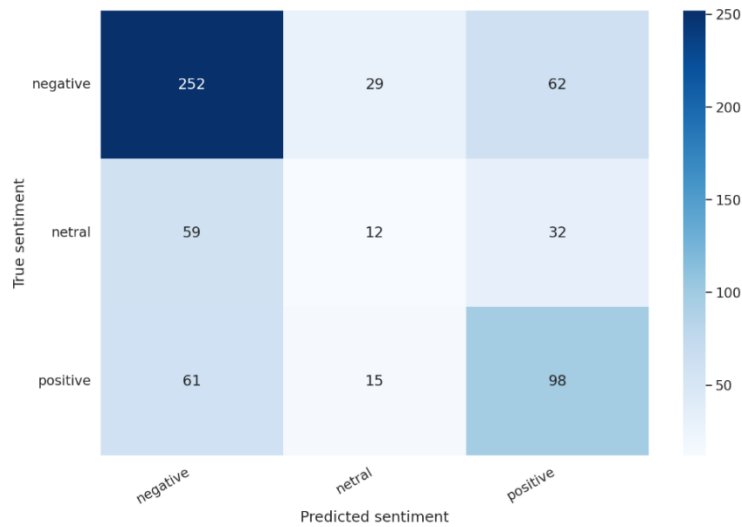
Tabel 4. 13 Evaluasi Confusion Matrix Sentimen pada Aspek Tampilan dengan learning rate 10^{-5}

	Precision	Recall	F1-Score
Negatif	0.70	0.79	0.74
Netral	0.04	0.01	0.02
Positif	0.56	0.68	0.62

Berdasarkan Tabel 4. 13, Hasil evaluasi kinerja klasifikasi model cukup baik, recall pada sentimen negatif dan positif cukup tinggi yang berarti sistem dapat memprediksi dengan benar untuk data bersentimen negatif dan positif. Sedangkan recall pada sentimen netral sangat rendah, yang berarti data pada sentimen netral tidak dapat terprediksi dengan benar oleh sistem.

2. Hasil Klasifikasi Sentimen pada Aspek Tampilan dengan *Learning Rate* $2 \cdot 10^{-5}$

Berikut hasil *confusion matrix* dari hasil pengujian model menggunakan *learning rate* $2 \cdot 10^{-5}$.



Gambar 4. 15 Hasil Confusion Matrix Sentimen pada Aspek Tampilan dengan learning rate $2 \cdot 10^{-5}$

Berdasarkan Gambar 4. 15, hasil data test menunjukkan bahwa sebanyak 252 data terklasifikasi dengan benar sebagai ulasan negatif, untuk ulasan positif sebanyak 98 data. Sedangkan pada ulasan netral sebanyak 12 data. Berdasarkan hasil tabel *confusion matrix*, maka dapat dihitung nilai akurasi sebagai berikut:

$$\begin{aligned}
 \text{Akurasi} &= \frac{TN + TNt + TP}{TN + FN + TNt + FNt + TP + FP} \\
 &= \frac{252 + 12 + 98}{252 + 29 + 62 + 59 + 12 + 32 + 61 + 15 + 98} = 0.584
 \end{aligned}$$

Pada hasil tabel *confusion matrix* juga dapat melihat tingkat kepercayaan model dengan melihat nilai precision, recall dan f1-score yang bertujuan untuk mengetahui seberapa baik model dalam memprediksi kelas.

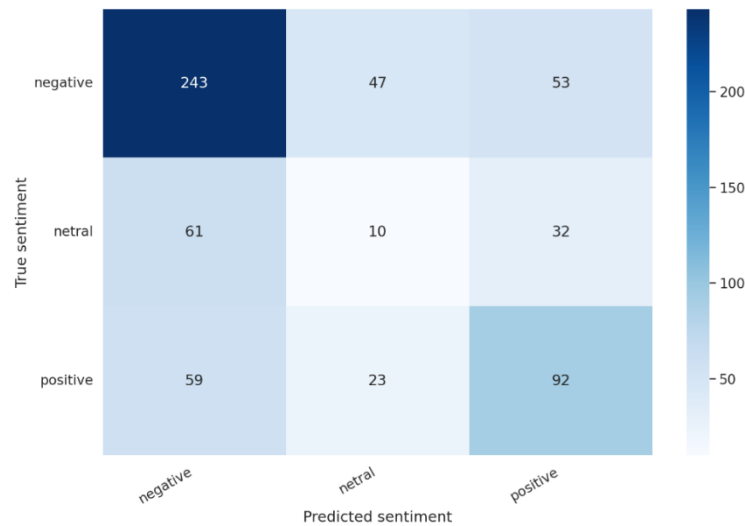
Tabel 4. 14 Evaluasi Confusion Matrix Sentimen pada Aspek Tampilan dengan learning rate $2 \cdot 10^{-5}$

	Precision	Recall	F1-Score
Negatif	0.68	0.73	0.70
Netral	0.21	0.12	0.15
Positif	0.51	0.56	0.54

Berdasarkan Tabel 4. 14, Hasil evaluasi kinerja klasifikasi model cukup baik, recall pada sentimen negatif dan positif cukup tinggi yang berarti sistem dapat memprediksi dengan benar untuk data bersentimen negatif dan positif. Sedangkan recall pada sentimen netral sangat rendah, yang berarti data pada sentimen netral tidak dapat terprediksi dengan benar oleh sistem.

3. Hasil Klasifikasi Sentimen pada Aspek Tampilan dengan *Learning Rate* $3 \cdot 10^{-5}$

Berikut hasil *confusion matrix* dari hasil pengujian model menggunakan *learning rate* $3 \cdot 10^{-5}$.



Gambar 4. 16 Hasil Confusion Matrix Sentimen pada Aspek Tampilan dengan learning rate $2 \cdot 10^{-5}$

Berdasarkan Gambar 4. 16, hasil data test menunjukkan bahwa sejumlah 243 data terklasifikasi dengan benar sebagai ulasan negatif, untuk ulasan positif sebanyak 92 data. Sedangkan pada ulasan netral sebanyak 10 data. Berdasarkan hasil tabel *confusion matrix*, maka dapat dihitung nilai akurasi sebagai berikut:

$$\begin{aligned}
 \text{Akurasi} &= \frac{TN + TNt + TP}{TN + FN + TNt + FNt + TP + FP} \\
 &= \frac{243 + 10 + 92}{243 + 47 + 53 + 61 + 10 + 32 + 59 + 23 + 92} = 0.556
 \end{aligned}$$

Pada hasil tabel *confusion matrix* juga dapat melihat tingkat kepercayaan model dengan melihat nilai precision, recall dan f1-score yang

bertujuan untuk mengetahui seberapa baik model dalam memprediksi kelas.

Tabel 4. 15 Evaluasi Confusion Matrix Sentimen pada Aspek Tampilan dengan learning rate $2 \cdot 10^{-5}$

	Precision	Recall	F1-Score
Negatif	0.67	0.71	0.69
Netral	0.12	0.10	0.11
Positif	0.52	0.53	0.52

Berdasarkan Tabel 4. 15, Hasil evaluasi kinerja klasifikasi model cukup baik, recall pada sentimen negatif dan positif cukup tinggi yang berarti sistem dapat memprediksi dengan benar untuk data bersentimen negatif dan positif. Sedangkan recall pada sentimen netral sangat rendah, yang berarti data pada sentimen netral tidak dapat terprediksi dengan benar oleh sistem.

Dari 3 model yang telah diuji, selanjutnya akan dibandingkan nilai akurasi dari masing masing *learning rate* terlihat pada Tabel 4. 15.

Tabel 4. 16 Perbandingan Nilai Akurasi dari Setiap Learning Rate pada Aspek Tampilan

<i>Learning rate</i>	Accuracy	AUC
10^{-5}	0.631	0.780
$2 \cdot 10^{-5}$	0.584	0.746
$3 \cdot 10^{-5}$	0.556	0.716

Dapat dilihat dari Tabel 4. 16, bahwa *learning rate* 10^{-5} memiliki nilai akurasi terbesar dibandingkan dengan yang lainnya yaitu sebesar 0.630 dan nilai AUC sebesar 0.780. Berdasarkan hasil test percobaan, didapatkan model dengan nilai parameter terbaik yaitu pada *learning rate* 10^{-5} .

Berdasarkan hasil tabel *confusion matrix* 3 kali percobaan *learning rate* dapat dilihat bahwa sistem mengalami kesulitan dalam mengklasifikasikan sentimen netral. Namun, saat pada ulasan positif dan negatif sistem cenderung memberikan klasifikasi dengan benar, hal ini

terjadi karena jumlah ulasan yang tidak seimbang antara ulasan positif, negatif dan negatif atau biasa disebut dengan data *imbalanced*.

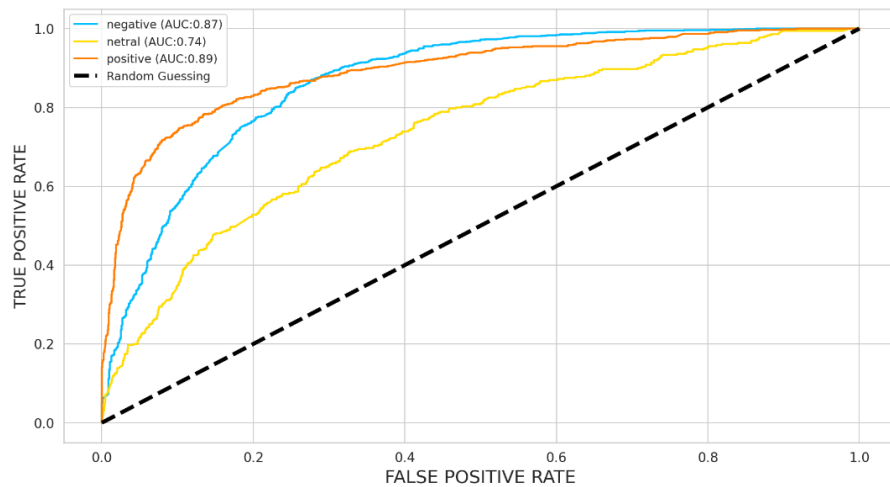
Untuk memperbaiki keseimbangan kelas peneliti menggunakan teknik SMOTE. Teknik *Synthetic Minority Over-sampling Technique* (SMOTE) adalah teknik yang digunakan untuk menangani ketidakseimbangan kelas. Teknik ini mereplikasikan data minoritas atau biasa dikenal dengan data sintetis (data buatan) pada kelas minoritas dengan cara menambah data kelas pada kelas minoritas untuk menyeimbangkan kelas mayoritas [36]. Data sintesis dilakukan berdasarkan *k-nearest neighbors* untuk setiap data di kelas minoritas.

Tabel 4. 17 Evaluasi Confusion Matrix Sentimen pada Aspek Tampilan dengan Teknik SMOTE

	Precision	Recall	F1-Score
Negatif	0.77	0.91	0.84
Netral	0.60	0.03	0.05
Positif	0.68	0.79	0.73

Akurasi: 74%; AUC: 88%

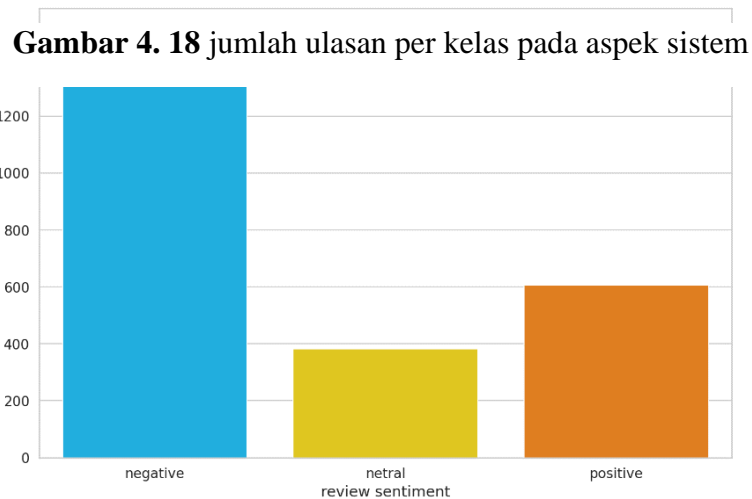
Berdasarkan hasil Tabel 4. 17, *Evaluasi Confusion Matrix Teknik SMOTE*, dapat disimpulkan bahwa model klasifikasi sentiment pada aspek tampilan memperoleh nilai akurasi (74%) dan AUC (88%) lebih tinggi dibandingkan tanpa SMOTE dengan nilai akurasi (63%) dan AUC (78%). Hal tersebut menunjukkan bahwa model yang telah melalui teknik SMOTE yang dapat meningkatkan performa dan nilai akurasi. Kinerja model sudah cukup baik dengan *recall* pada sentimen negatif dan positif cukup tinggi, yang berarti data pada sentimen negatif dan positif dapat dideteksi secara benar, tetapi pada sentimen netral masih cukup rendah yang berarti data pada sentimen netral tidak dapat dideteksi secara benar. Berikut kurva ROC dan AUC dari data testing:



Gambar 4. 17 kurva ROC dan AUC Aspek Tampilan menggunakan Teknik SMOTE

4.5.3. Hasil Klasifikasi Sentimen Pada Aspek Sistem

Data yang digunakan untuk klasifikasi sentiment pada aspek sistem sebanyak 2493 data. Label kelas yang digunakan untuk klasifikasi sentimen berdasarkan aspek manfaat yaitu kelas sentimen negatif, netral, dan positif, berikut banyaknya jumlah ulasan per kelas pada aspek sistem:



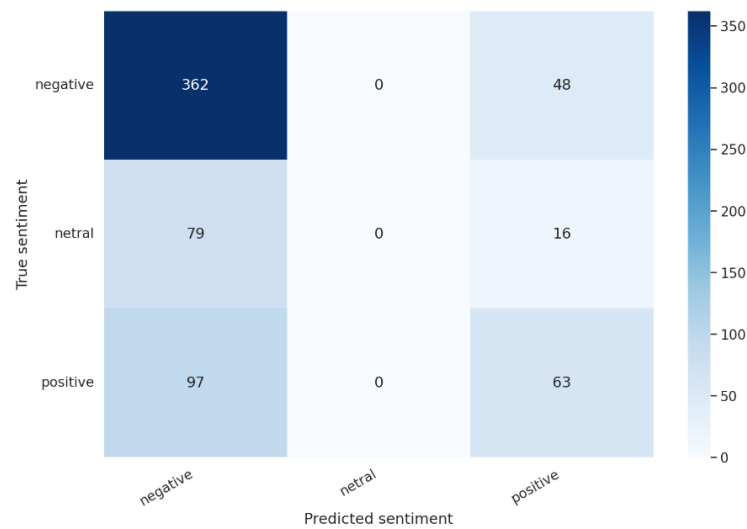
Gambar 4. 18 jumlah ulasan per kelas pada aspek sistem

Berdasarkan Gambar 4. 18, menunjukkan bahwa kelas sentimen negatif yang menjadi dominasi dengan jumlah ulasan sebanyak 1504 ulasan. Sedangkan untuk kelas sentimen netral yang menjadi minoritas dengan jumlah ulasan sebanyak 383 ulasan. Pada saat klasifikasi data dibagi menjadi 3 yaitu data training, data validation dan data testing. Data training adalah data yang dilatih untuk membangun model dengan ukuran 70% dari data

yaitu 1662 data. Data validation adalah data untuk proses validasi model dan untuk proses minimalisasi *overfitting* sebanyak 166 data, dan data testing adalah data untuk menguji performa model yang sudah dilatih sebanyak 665 data.

1. Hasil Klasifikasi Sentimen pada Aspek Sistem dengan *Learning Rate* 10^{-5}

Berikut hasil *confusion matrix* dari hasil pengujian model menggunakan *learning rate* 10^{-5} .



Gambar 4. 19 Hasil Confusion matrix Sentimen pada Aspek Sistem dengan *learning rate* 10^{-5}

Berdasarkan Gambar 4. 19, hasil data test menunjukkan bahwa sebanyak 362 data terklasifikasi dengan benar sebagai ulasan negatif, untuk ulasan positif sebanyak 63 data. Sedangkan pada ulasan netral sebanyak 0 data. Berdasarkan hasil tabel *confusion matrix*, maka dapat dihitung nilai akurasinya sebagai berikut:

$$\begin{aligned}
 \text{Akurasi} &= \frac{TN + TNt + TP}{TN + FN + TNt + Fnt + TP + FP} \\
 &= \frac{362 + 0 + 63}{362 + 0 + 48 + 79 + 0 + 16 + 97 + 0 + 63} = 0.639
 \end{aligned}$$

Pada hasil tabel *confusion matrix* juga dapat melihat tingkat kepercayaan model dengan melihat nilai precision, recall dan f1-score yang bertujuan untuk mengetahui seberapa baik model dalam memprediksi kelas.

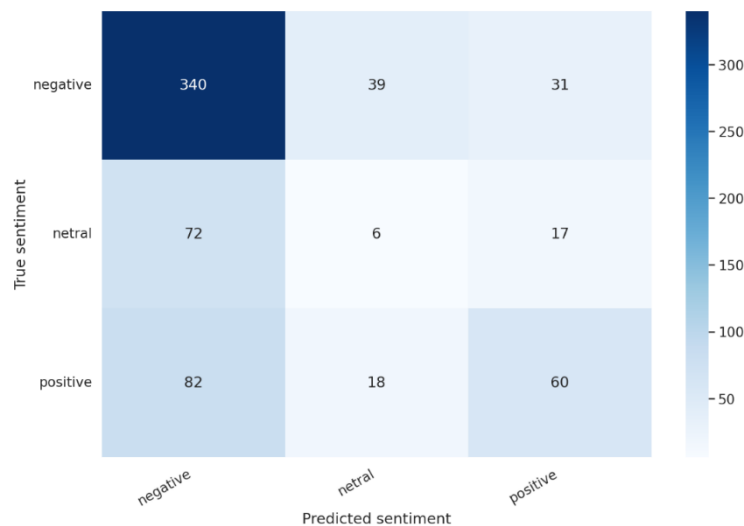
Tabel 4. 18 Evaluasi Confusion Matrix Sentimen pada Aspek Sistem dengan learning rate 10^{-5}

	Precision	Recall	F1-Score
Negatif	0.67	0.88	0.76
Netral	0.00	0.00	0.00
Positif	0.50	0.39	0.44

Berdasarkan Tabel 4. 18, Hasil evaluasi kinerja klasifikasi model cukup baik, recall pada sentimen negatif dan positif cukup tinggi yang berarti sistem dapat memprediksi dengan benar untuk data bersentimen negatif dan positif. Sedangkan recall pada sentimen netral sangat rendah, yang berarti data pada sentimen netral tidak dapat terprediksi dengan benar oleh sistem.

2. Hasil Klasifikasi Sentimen pada Aspek Sistem dengan *Learning rate* $2 \cdot 10^{-5}$

Berikut hasil *confusion matrix* dari hasil pengujian model menggunakan *learning rate* $2 \cdot 10^{-5}$



Gambar 4. 20 Hasil Confusion Matrix Sentimen pada Aspek Sistem dengan learning rate $2 \cdot 10^{-5}$

Berdasarkan Gambar 4. 20, hasil data test menunjukkan bahwa sebanyak 340 data terklasifikasi dengan benar sebagai ulasan negatif, untuk ulasan positif sebanyak 60 data. Sedangkan pada ulasan netral sebanyak 6

data. Berdasarkan hasil tabel *confusion matrix*, maka dapat dihitung nilai akurasi sebagai berikut:

$$\begin{aligned}
 \text{Akurasi} &= \frac{TN + TNt + TP}{TN + FN + TNt + FNt + TP + FP} \\
 &= \frac{340 + 6 + 60}{340 + 39 + 31 + 72 + 6 + 17 + 82 + 18 + 60} = 0.610
 \end{aligned}$$

Pada hasil tabel *confusion matrix* juga dapat melihat tingkat kepercayaan model dengan melihat nilai precision, recall dan f1-score yang bertujuan untuk mengetahui seberapa baik model dalam memprediksi kelas.

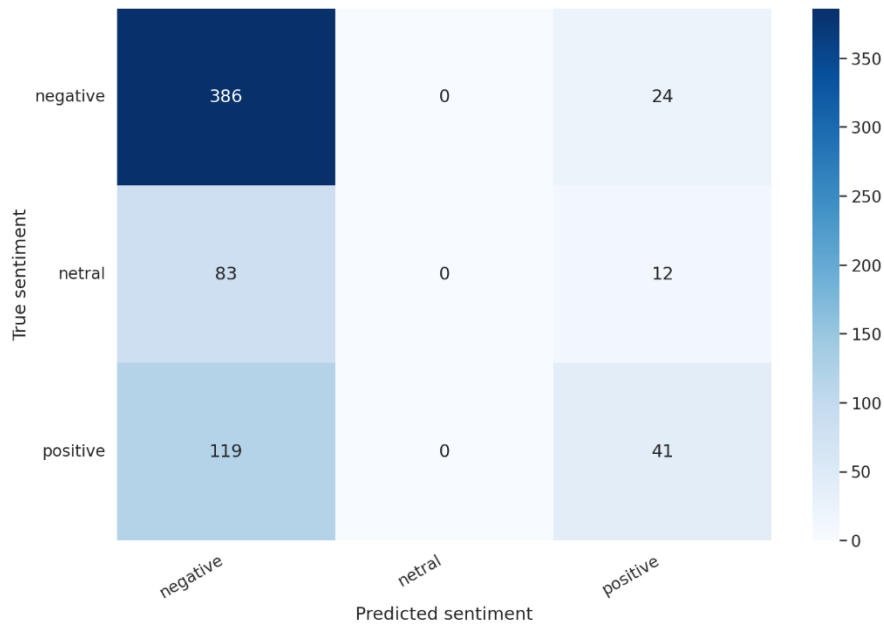
Tabel 4. 19 Evaluasi Confusion Matrix Sentimen pada Aspek Sistem dengan learning rate $2 \cdot 10^{-5}$

	Precision	Recall	F1-Score
Negatif	0.69	0.83	0.75
Netral	0.10	0.06	0.08
Positif	0.56	0.38	0.45

Berdasarkan Tabel 4. 19, Hasil evaluasi kinerja klasifikasi model cukup baik, recall pada sentimen negatif dan positif cukup tinggi yang berarti sistem dapat memprediksi dengan benar untuk data bersentimen negatif dan positif. Sedangkan recall pada sentimen netral sangat rendah, yang berarti data pada sentimen netral tidak dapat terprediksi dengan benar oleh sistem.

3. Hasil Klasifikasi Sentimen pada Aspek Sistem dengan *Learning Rate* $3 \cdot 10^{-5}$

Berikut hasil *confusion matrix* dari hasil pengujian model menggunakan *learning rate* $3 \cdot 10^{-5}$.



Gambar 4. 21 Hasil Confusion Matrix Sentimen pada Aspek Sistem dengan learning rate $3 \cdot 10^{-5}$

Berdasarkan Gambar 4. 21, hasil data test menunjukkan bahwa sejumlah 386 data terklasifikasi dengan benar sebagai ulasan negatif, untuk ulasan positif sebanyak 41 data. Sedangkan pada ulasan netral sebanyak 0 data. Berdasarkan hasil tabel *confusion matrix*, maka dapat dihitung nilai akurasi sebagai berikut:

$$\begin{aligned}
 \text{Akurasi} &= \frac{TN + TNt + TP}{TN + FN + TNt + FNt + TP + FP} \\
 &= \frac{386 + 0 + 41}{386 + 0 + 24 + 83 + 0 + 12 + 119 + 0 + 41} = 0.642
 \end{aligned}$$

Pada hasil tabel *confusion matrix* juga dapat melihat tingkat kepercayaan model dengan melihat nilai precision, recall dan f1-score yang bertujuan untuk mengetahui seberapa baik model dalam memprediksi kelas.

Tabel 4. 20 Evaluasi *Confusion Matrix* Sentimen pada Aspek Sistem dengan learning rate $3 \cdot 10^{-5}$

	Precision	Recall	F1-Score
Negatif	0.67	0.71	0.69
Netral	0.12	0.10	0.11
Positif	0.52	0.53	0.52

Berdasarkan Tabel 4. 20, Hasil evaluasi kinerja klasifikasi model cukup baik, recall pada sentimen negatif dan positif cukup tinggi yang berarti sistem dapat memprediksi dengan benar untuk data bersentimen negatif dan positif. Sedangkan recall pada sentimen netral sangat rendah, yang berarti data pada sentimen netral tidak dapat terprediksi dengan benar oleh sistem.

Dari 3 model yang telah diuji, selanjutnya akan dibandingkan nilai akurasi dari masing masing *learning rate* terlihat pada Tabel 4. 20.

Tabel 4. 21 Perbandingan nilai akurasi dari setiap *learning rate*

<i>Learning rate</i>	Accuracy	AUC
10^{-5}	0.639	0.783
$2 \cdot 10^{-5}$	0.610	0.768
$3 \cdot 10^{-5}$	0.642	0.763

Dapat dilihat dari Tabel 4. 21, bahwa *learning rate* $3 \cdot 10^{-5}$ memiliki nilai akurasi terbesar dibandingkan dengan yang lainnya yaitu sebesar 0.642 dan nilai AUC sebesar 0,763. Berdasarkan hasil test percobaan, didapatkan model dengan nilai parameter terbaik yaitu pada *learning rate* $3 \cdot 10^{-5}$.

Berdasarkan hasil tabel *confusion matrix* 3 kali percobaan *learning rate* dapat dilihat bahwa sistem mengalami kesulitan dalam mengklasifikasikan sentimen netral. Namun, saat pada ulasan positif dan negatif sistem cenderung memberikan klasifikasi dengan benar, hal ini terjadi karena jumlah ulasan yang tidak seimbang antara ulasan positif, negatif dan negatif atau biasa disebut dengan data *imbalanced*.

Untuk memperbaiki keseimbangan kelas peneliti menggunakan teknik SMOTE. Teknik *Synthetic Minority Over-sampling Technique* (SMOTE) adalah teknik yang digunakan untuk menangani ketidakseimbangan kelas. Teknik ini mereplikasikan data minoritas atau biasa dikenal dengan data sintetis (data buatan) pada kelas minoritas dengan cara menambah data kelas pada kelas minoritas untuk menyeimbangkan kelas mayoritas [36].

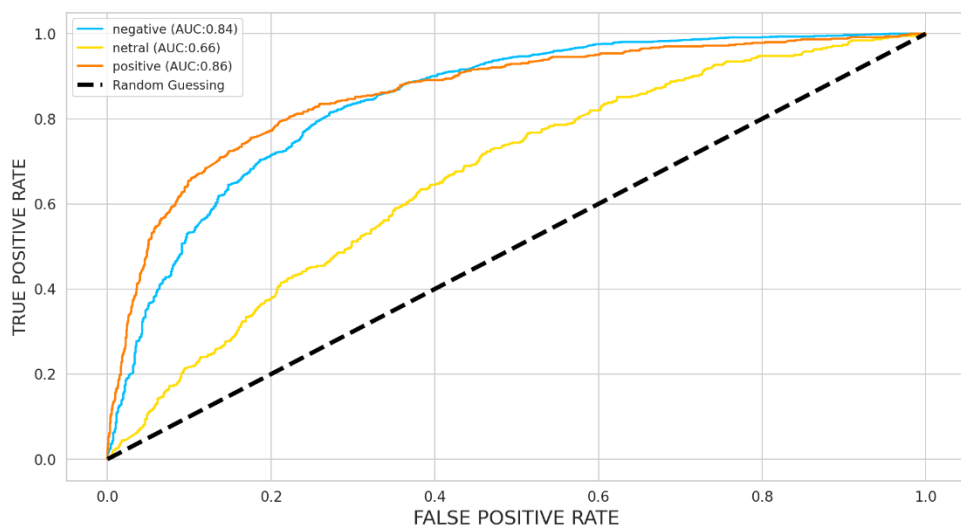
Data sintesis dilakukan berdasarkan *k-nearest neighbors* untuk setiap data di kelas minoritas.

Tabel 4. 22 Evaluasi Confusion Matrix Sentimen pada Aspek Sistem dengan Teknik SMOTE

	Precision	Recall	F1-Score
Negatif	0.74	0.95	0.83
Netral	0.00	0.00	0.00
Positif	0.68	0.64	0.66

Akurasi: 73%; AUC: 86%

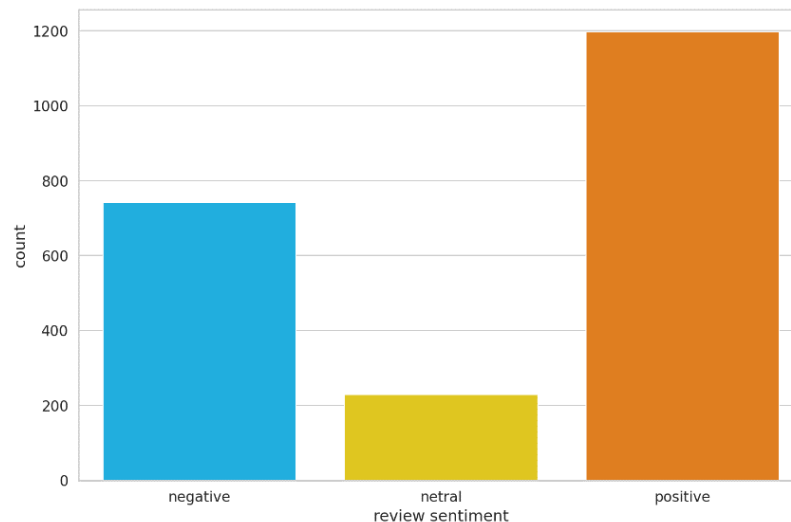
Berdasarkan hasil Tabel 4. 22, *Evaluasi Confusion Matrix Teknik SMOTE*, dapat disimpulkan bahwa model klasifikasi sentiment pada aspek sistem memperoleh nilai akurasi (73%) dan AUC (86%) lebih tinggi dibandingkan tanpa SMOTE dengan nilai akurasi (64%) dan AUC (76%). Hal tersebut menunjukkan bahwa model yang telah melalui teknik SMOTE yang dapat meningkatkan performa dan nilai akurasi. Kinerja model sudah cukup baik dengan *recall* pada sentimen negatif dan positif cukup tinggi, yang berarti data pada sentimen negatif dan positif dapat dideteksi secara benar, tetapi pada sentimen netral masih cukup rendah, yang berarti data pada sentimen netral tidak dapat dideteksi secara benar. Berikut kurva ROC dan AUC dari data testing:



Gambar 4. 22 kurva ROC dan AUC Aspek Sistem menggunakan Teknik SMOTE

4.5.4. Hasil Klasifikasi Sentimen Pada Aspek Pelayanan

Data yang digunakan untuk klasifikasi sentiment pada aspek pelayanan sebanyak 2168 data. Label kelas yang digunakan untuk klasifikasi sentimen berdasarkan aspek manfaat yaitu kelas sentimen negatif, netral, dan positif, berikut banyaknya jumlah ulasan per kelas pada aspek pelayanan:

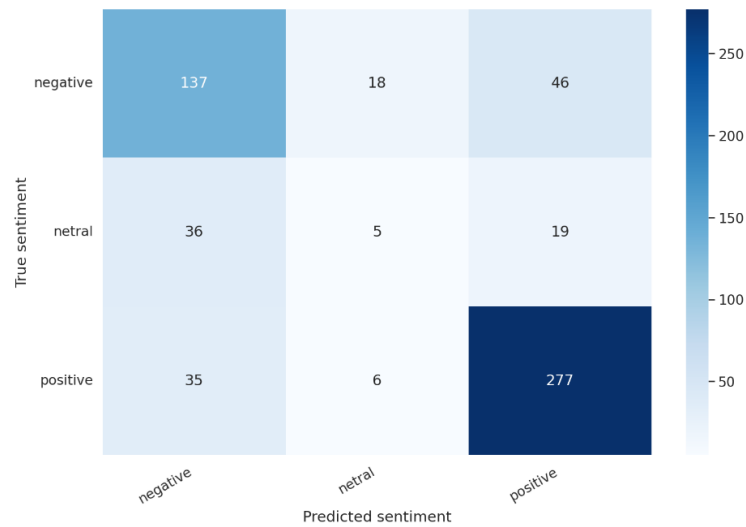


Gambar 4. 23 Jumlah Ulasan per Kelas pada Aspek Pelayanan

Berdasarkan Gambar 4. 23 menunjukkan bahwa kelas sentimen positif yang menjadi dominasi dengan jumlah ulasan sebanyak 1197 ulasan. Sedangkan untuk kelas sentimen netral yang menjadi minoritas dengan jumlah ulasan sebanyak 229 ulasan. Pada saat klasifikasi data dibagi menjadi 3 yaitu data training, data validation dan data testing. Data training adalah data yang dilatih untuk membangun model dengan ukuran 70% dari data yaitu 1445 data. Data validation adalah data untuk proses validasi model dan untuk proses minimalisasi *overfitting* sebanyak 144 data, dan data testing adalah data untuk menguji performa model yang sudah dilatih sebanyak 579 data.

1. Hasil Klasifikasi Sentimen pada Aspek Pelayanan dengan *Learning Rate* 10^{-5}

Berikut hasil *confusion matrix* dari hasil pengujian model menggunakan *learning rate* 10^{-5} .



Gambar 4. 24 Hasil Confusion Matrix Sentimen pada Aspek Pelayanan dengan learning rate 10^{-5}

Berdasarkan Gambar 4. 24, hasil data test menunjukkan bahwa sebanyak 137 data terklasifikasi dengan benar sebagai ulasan negatif, untuk ulasan positif sebanyak 277 data. Sedangkan pada ulasan netral sebanyak 5 data. Berdasarkan hasil tabel *confusion matrix*, maka dapat dihitung nilai akurasi sebagai berikut:

$$\begin{aligned}
 \text{Akurasi} &= \frac{TN + TNt + TP}{TN + FN + TNt + FNT + TP + FP} \\
 &= \frac{137 + 5 + 277}{137 + 18 + 46 + 36 + 5 + 19 + 35 + 6 + 277} = 0.724
 \end{aligned}$$

Pada hasil tabel *confusion matrix* juga dapat melihat tingkat kepercayaan model dengan melihat nilai precision, recall dan f1-score yang bertujuan untuk mengetahui seberapa baik model dalam memprediksi kelas.

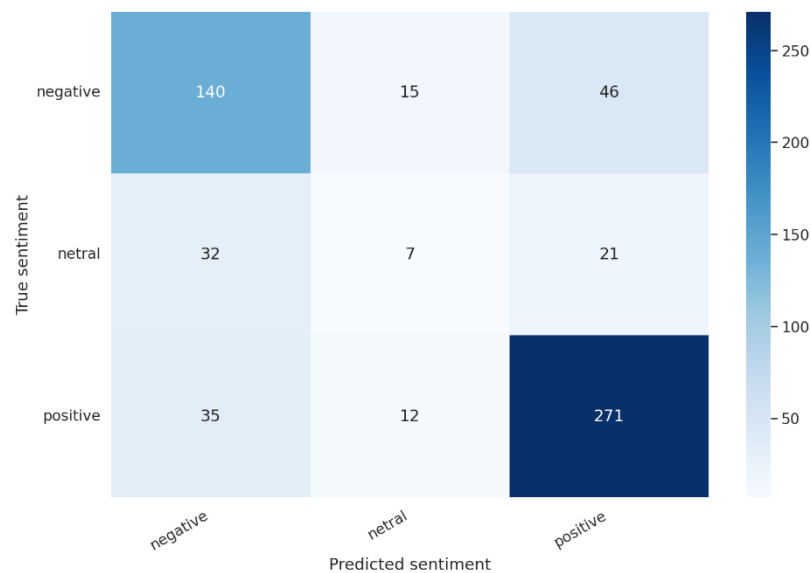
Tabel 4. 23 Evaluasi Confusion Matrix Sentimen pada Aspek Pelayanan dengan learning rate 10^{-5}

	Precision	Recall	F1-Score
Negatif	0.66	0.68	0.67
Netral	0.17	0.08	0.11
Positif	0.81	0.87	0.84

Berdasarkan Tabel 4. 23, Hasil evaluasi kinerja klasifikasi model cukup baik, recall pada sentimen negatif dan positif cukup tinggi yang berarti sistem dapat memprediksi dengan benar untuk data bersentimen negatif dan positif. Sedangkan recall pada sentimen netral sangat rendah, yang berarti data pada sentimen netral tidak dapat terprediksi dengan benar oleh sistem.

2. Hasil Klasifikasi Sentimen pada Aspek Pelayanan dengan *Learning Rate* $2 \cdot 10^{-5}$

Berikut hasil *confusion matrix* dari hasil pengujian model menggunakan *learning rate* $2 \cdot 10^{-5}$.



Gambar 4. 25 Hasil Confusion Matrix Sentimen pada Aspek Pelayanan dengan *learning rate* $2 \cdot 10^{-5}$.

Berdasarkan Gambar 4. 25, Berdasarkan hasil data test menunjukkan bahwa sebanyak 140 data terklasifikasi dengan benar sebagai ulasan negatif, untuk ulasan positif sebanyak 271 data. Sedangkan pada ulasan netral sebanyak 7 data. Berdasarkan hasil tabel *confusion matrix*, maka dapat dihitung nilai akurasinya sebagai berikut:

$$\begin{aligned}
 \text{Akurasi} &= \frac{TN + TNt + TP}{TN + FN + TNt + FNt + TP + FP} \\
 &= \frac{140 + 7 + 271}{140 + 15 + 46 + 32 + 7 + 21 + 35 + 12 + 271} = 0.722
 \end{aligned}$$

Berdasarkan hasil tabel *confusion matrix* juga dapat melihat tingkat kepercayaan model dengan melihat nilai precision, recall dan f1-score yang bertujuan untuk mengetahui seberapa baik model dalam memprediksi kelas.

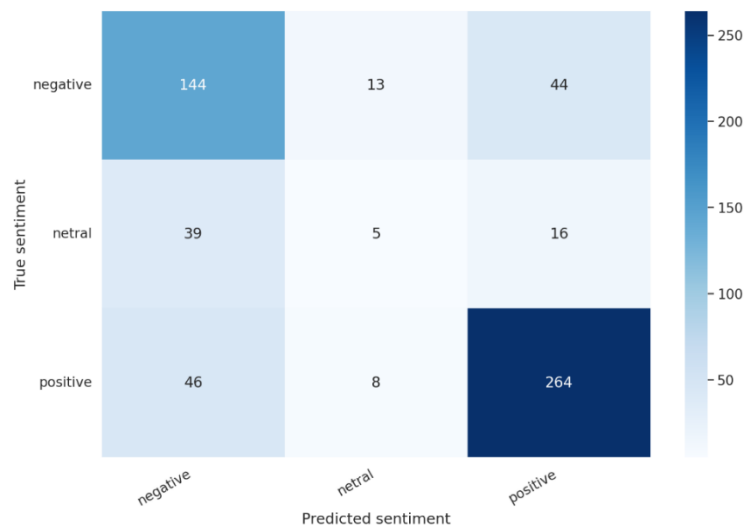
Tabel 4. 24 Evaluasi *Confusion Matrix* Sentimen pada Aspek Pelayanan dengan *learning rate* $2 \cdot 10^{-5}$

	Precision	Recall	F1-Score
Negatif	0.68	0.70	0.69
Netral	0.21	0.12	0.15
Positif	0.80	0.85	0.83

Berdasarkan Tabel 4. 24, Hasil evaluasi kinerja klasifikasi model cukup baik, recall pada sentimen negatif dan positif cukup tinggi yang berarti sistem dapat memprediksi dengan benar untuk data bersentimen negatif dan positif. Sedangkan recall pada sentimen netral sangat rendah, yang berarti data pada sentimen netral tidak dapat terprediksi dengan benar oleh sistem.

3. Hasil Klasifikasi Sentimen pada Aspek Pelayanan dengan *Learning Rate* $3 \cdot 10^{-5}$

Berikut hasil *confusion matrix* dari hasil pengujian model menggunakan *learning rate* $3 \cdot 10^{-5}$.



Gambar 4. 26 Hasil Confusion Matrix Sentimen pada Aspek Pelayanan dengan *learning rate* $3 \cdot 10^{-5}$

Berdasarkan Gambar 4. 26, Berdasarkan hasil data test menunjukkan bahwa sejumlah 144 data terklasifikasi dengan benar sebagai ulasan negatif, untuk ulasan positif sebanyak 264 data. Sedangkan pada ulasan netral sebanyak 5 data. Berdasarkan hasil tabel *confusion matrix*, maka dapat dihitung nilai akurasinya sebagai berikut:

$$\begin{aligned}
 \text{Akurasi} &= \frac{TN + TNt + TP}{TN + FN + TNt + FNt + TP + FP} \\
 &= \frac{144 + 5 + 264}{144 + 13 + 44 + 39 + 5 + 16 + 46 + 8 + 264} = 0.713
 \end{aligned}$$

Berdasarkan hasil tabel *confusion matrix* juga dapat melihat tingkat kepercayaan model dengan melihat nilai precision, recall dan f1-score yang bertujuan untuk mengetahui seberapa baik model dalam memprediksi kelas.

Tabel 4. 25 Evaluasi *Confusion Matrix* Sentimen pada Aspek Pelayanan dengan *learning rate* $3 \cdot 10^{-5}$

	Precision	Recall	F1-Score
Negatif	0.63	0.72	0.67
Netral	0.19	0.08	0.12
Positif	0.81	0.83	0.82

Berdasarkan Tabel 4. 25, Hasil evaluasi kinerja klasifikasi model cukup baik, recall pada sentimen negatif dan positif cukup tinggi yang berarti sistem dapat memprediksi dengan benar untuk data bersentimen negatif dan positif. Sedangkan recall pada sentimen netral sangat rendah, yang berarti data pada sentimen netral tidak dapat terprediksi dengan benar oleh sistem.

Dari 3 model yang telah diuji, selanjutnya akan dibandingkan nilai akurasi dari masing masing *learning rate* terlihat pada Tabel 4. 13.

Tabel 4. 26 Perbandingan Nilai Akurasi dari Setiap Learning Rate pada Aspek Pelayanan

<i>Learning rate</i>	Accuracy	AUC
10^{-5}	0.724	0.854
$2 \cdot 10^{-5}$	0.722	0.822
$3 \cdot 10^{-5}$	0.713	0.819

Dapat dilihat dari Tabel 4. 26, bahwa *learning rate* 10^{-5} memiliki nilai akurasi terbesar dibandingkan dengan yang lainnya yaitu sebesar 0.724 dan nilai AUC sebesar 0.854. Berdasarkan hasil test percobaan, didapatkan model dengan nilai parameter terbaik yaitu pada *learning rate* 10^{-5} .

Berdasarkan hasil tabel *confusion matrix* 3 kali percobaan *learning rate* dapat dilihat bahwa sistem mengalami kesulitan dalam mengklasifikasikan sentimen netral. Namun, saat pada ulasan positif dan negatif sistem cenderung memberikan klasifikasi dengan benar, hal ini terjadi karena jumlah ulasan yang tidak seimbang antara ulasan positif, negatif dan netral atau biasa disebut dengan data *imbalanced*.

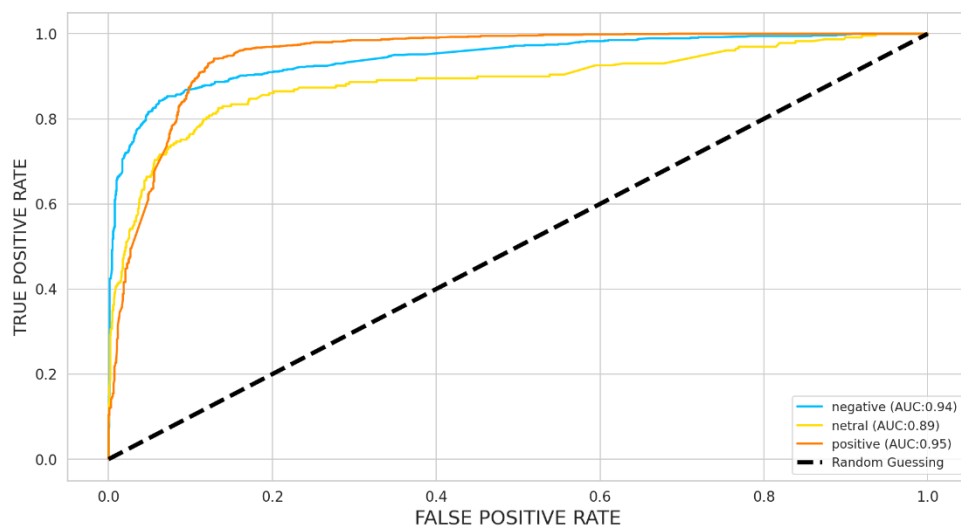
Untuk memperbaiki keseimbangan kelas peneliti menggunakan teknik SMOTE. Teknik *Synthetic Minority Over-sampling Technique* (SMOTE) adalah teknik yang digunakan untuk menangani ketidakseimbangan kelas. Teknik ini mereplikasikan data minoritas atau biasa dikenal dengan data sintetis (data buatan) pada kelas minoritas dengan cara menambah data kelas pada kelas minoritas untuk menyeimbangkan kelas mayoritas [36]. Data sintesis dilakukan berdasarkan *k-nearest neighbors* untuk setiap data di kelas minoritas.

Tabel 4. 27 Evaluasi Confusion Matrix Sentimen pada Aspek Pelayanan dengan Teknik SMOTE

	Precision	Recall	F1-Score
Negatif	0.85	0.85	0.85
Netral	0.78	0.41	0.53
Positif	0.88	0.96	0.92

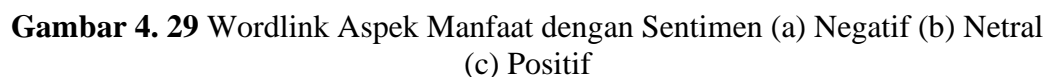
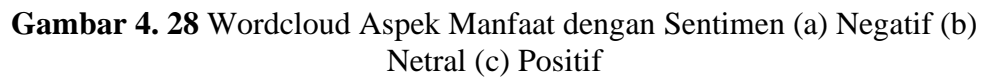
Akurasi: 86%; AUC: 95%

Berdasarkan hasil Tabel 4. 27, *Evaluasi Confusion Matrix Teknik SMOTE*, dapat disimpulkan bahwa model klasifikasi sentiment pada aspek pelayanan memperoleh nilai akurasi (86%) dan AUC (95%) lebih tinggi dibandingkan tanpa SMOTE dengan nilai akurasi (72%) dan AUC (85%). Hal tersebut menunjukkan bahwa model yang telah melalui teknik SMOTE yang dapat meningkatkan performa dan nilai akurasi. Kinerja model sudah cukup baik dengan *recall* pada sentimen negatif dan positif cukup tinggi, yang berarti data pada sentimen negatif dan positif dapat dideteksi secara benar, tetapi pada sentimen netral masih cukup rendah, yang berarti data pada sentimen netral tidak dapat dideteksi secara benar. Berikut kurva ROC dan AUC dari data testing:



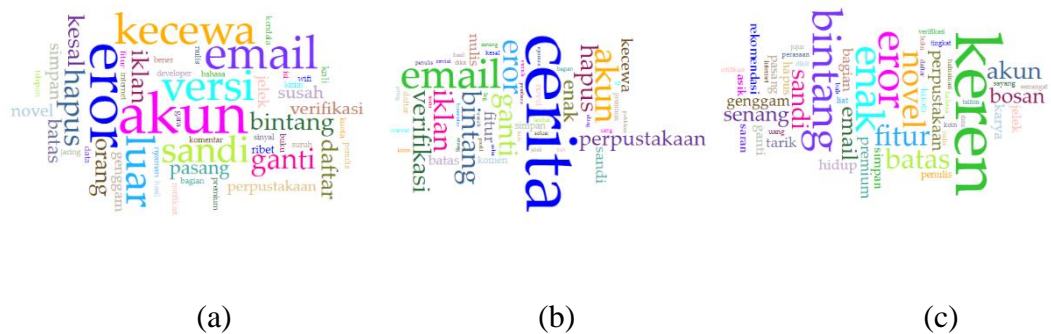
Gambar 4. 27 Kurva ROC dan AUC Aspek Pelayanan Menggunakan Teknik SMOTE

Pada penelitian ini visualisasi dan interpretasi data menggunakan teknik *wordcloud* dan *wordlink* untuk sentimen di setiap aspek seperti pada gambar dibawah ini:

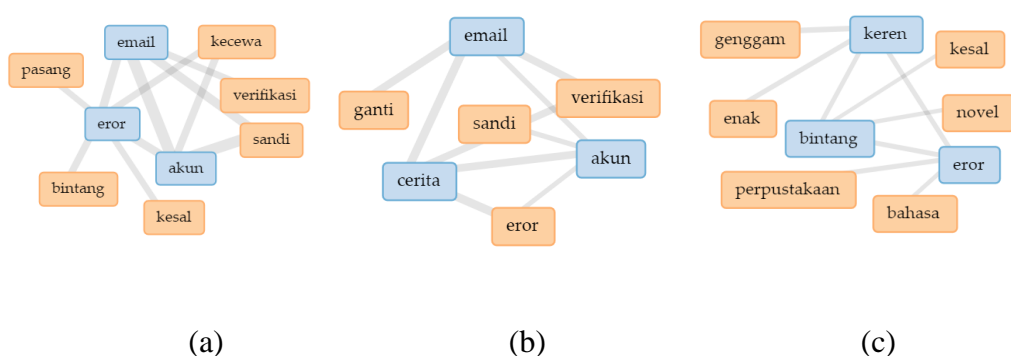


81

adanya premium membuat saya jarang membuka aplikasi ini”. Sedangkan sentimen positif pada aspek manfaat menunjukkan bahwa terlihat aplikasi baca cerita yang bagus dan menghilangkan bosan, pengguna merasa puas ketika membaca cerita karena terdapat berbagai macam genre, dan meningkatkan halusinasi pembaca. Cerita yang ditulis oleh pengarang berbentuk novel sehingga seru dan cocok untuk orang yang gemar membaca dan menulis. Salah satu contoh aspek manfaat yaitu “apk ini sangat bagus dan banyak cerita2 menarik juga bisa menjadi wadah bagi yang punya minat dan bakat dalam menulis atau merangkai kata sedikit pesan pengguna apk ini kebanyakan pelajar bisakah limit untuk cerita offline di hilangkan. jadi kita bisa lebih banyak menampung cerita, sekian”. Kesimpulan visualisasi untuk aspek manfaat, yaitu agar aplikasi novel *online* memperbaiki kualitas aplikasi dan kapasitas bacaan novel *online* untuk pembaca supaya pembaca merasa nyaman dan mudah pada saat membaca novel *online*.



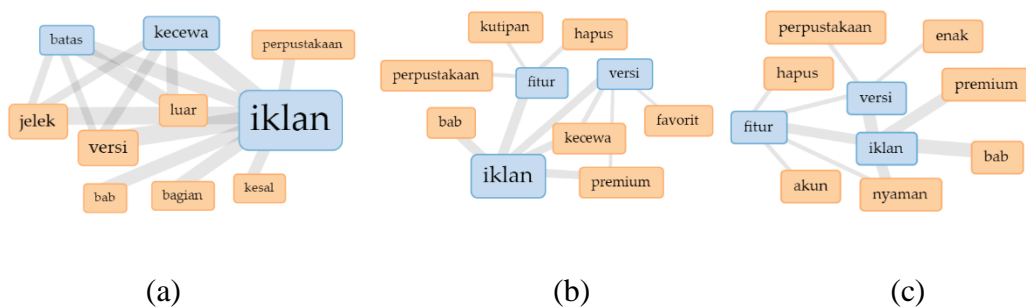
Gambar 4. 30 Wordcloud Aspek Pelayanan dengan sentimen (a) Negatif (b) Netral (c) Positif



Gambar 4. 31 Wordlink Aspek Pelayanan dengan sentimen (a) Negatif (b) Netral (c) Positif

Berdasarkan Gambar 4. 30 dan Gambar 4. 31, aspek pelayanan pada sentimen negatif pada aplikasi novel *online* yaitu sering terjadi eror, setelah itu pengguna tidak bisa masuk ke dalam aplikasi novel *online* dan banyak email yang digunakan tidak terdata. Contoh ulasan sentimen negatif pada aspek manfaat yaitu “Tolong ya udah Saya login gak bisa mulu passwordnya padahal sudah benar, Sekarang apk nya kalau dibuka malah kedap kedip gak jelas. Sumpah kecewa banget saya lebih suka Wattpad yang dulu”. Untuk sentimen netral pada aspek pelayanan menunjukkan bahwa aplikasi novel *online* merupakan aplikasi yang sering meminta verifikasi email dan mengganti sandi akun. Contoh ulasan sentimen netral pada aspek manfaat yaitu “aplikasinya sih bagus,tapi kenapa setiap saya mau pubhlis cerita saya,selalu disuruh verifikasi email,padahal saya udah verifikasi”. Sedangkan untuk sentimen positif pada aspek pelayanan menunjukkan cerita yang terdapat didalamnya bagus dan aplikasinya sangat seru untuk membaca novel atau menulis cerita dengan berbagai bahasa. Contoh ulasan sentimen positif pada aspek manfaat yaitu “Bagus. Di kasih tawaran cerita2 terbaik. Jadi nggak susah2 lagi nyari sendiri cerita nya.”. Kesimpulan visualisasi untuk aspek pelayanan, yaitu agar memperbaiki pelayan pada novel online dengan cara memperbaiki kualitas pelayanan pada aplikasi novel online agar pengguna tidak kesulitan saat masuk (*login*) ke dalam aplikasi atau error, mengecek kembali email yang tidak bisa masuk ke dalam aplikasi agar pengguna bisa masuk ke dalam aplikasi novel online kapan pun, dan jangan terlalu sering update aplikasi.

Gambar 4. 32 Wordcloud Aspek Tampilan dengan sentimen (a) Negatif (b) Netral (c) Positif

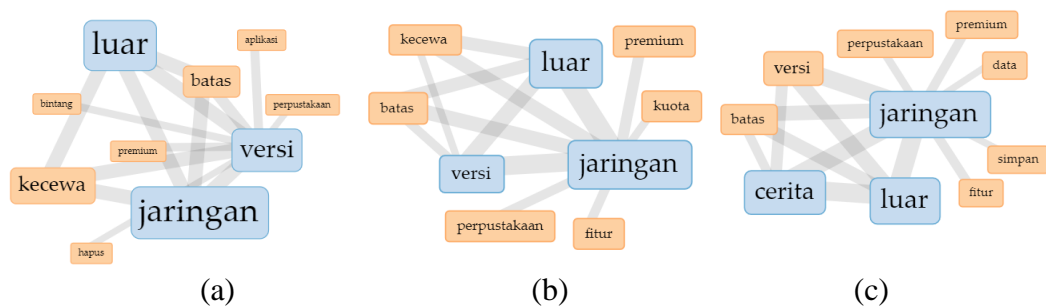


Gambar 4. 33 Wordlink Aspek Tampilan dengan sentimen (a) Negatif (b) Netral (c) Positif

Berdasarkan Gambar 4. 32 dan Gambar 4. 33, aspek tampilan pada aplikasi novel *online* menunjukkan bahwa aplikasi aplikasi novel *online* ini adalah aplikasi yang baca yang banyak fitur yang memudahkan pengguna untuk mencari cerita atau novel untuk dibaca. Namun pengguna yang merasa terganggu dengan adanya banyak iklan di setiap pergantian bab pada versi terbaru, dan pengguna meminta dikembalikan ke versi sebelum diperbarui ketika tidak ada iklan dan bisa membaca secara offline (luar jaringan). Pengguna juga merasa tampilan pada aplikasi novel *online* lebih nyaman dengan versi lama dari pada versi terbaru. Salah satu contoh aspek tampilan yaitu “Semakin gak jelas.. semakin banyak iklan,, beranda pemberitahuan juga hilang .. trus cerita baru yg aku baca padahal udah aq masukin "cerita yg dibaca" tapi mendadak hilang ... ini dari tahun ke tahun gak makin oke tp kenapa makin kayak gini sih.. padahal pengguna wattpad sekarang banyak”. Kesimpulan visualisasi untuk aspek tampilan, yaitu agar memperbaiki tampilan pada novel online dengan cara mengurangi iklan yang terdapat di setiap pergantian bab novel.



Gambar 4. 34 Wordcloud Aspek Sistem dengan sentimen (a) Negatif (b) Netral (c) Positif

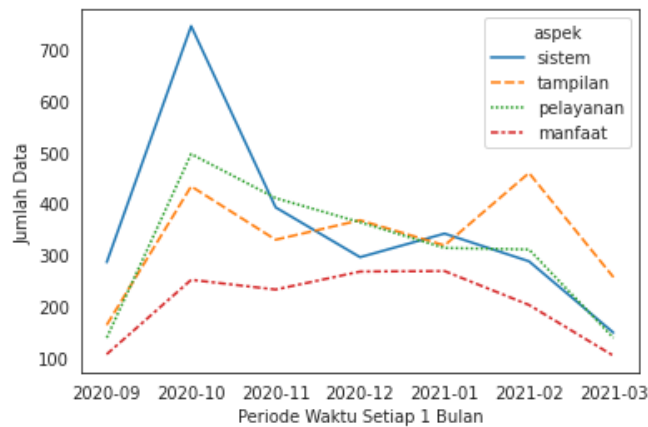


Gambar 4. 35 Wordlink Aspek Sistem dengan sentimen (a) Negatif (b) Netral (c) Positif

Berdasarkan Gambar 4. 34 dan Gambar 4. 35, aspek sistem pada aplikasi novel *online* menunjukkan bahwa cerita yang dibuat oleh penulis aplikasi novel *online* bagus untuk dibaca. Namun pengguna kecewa karena cerita dibatasi sehingga hanya terdapat beberapa cerita di perpustakaan. Terkait jaringan data pada saat membaca, pengguna mengeluh perubahan penggunaan data dari *offline* (luar jaringan) menjadi *online* (dalam jaringan) yang membuat tidak nyaman, dan ketika diperbarui aplikasi beberapa fitur yang dihilangkan seperti *newsfeed*. Salah satu contoh aspek sistem yaitu “Sekarang Wattpad semakin berubah ya, dulu tidak ada batasan untuk menyimpan book secara offline. Sekarang hanya sedikit book yang bisa saya simpan secara offline, please cuma 2 yg bisa disimpan, saya kecewa. Karena tidak semua orang bisa terus terusan baca secara online. Tolong kembalikan wattpad seperti dulu!”. Kesimpulan visualisasi untuk aspek sistem, yaitu agar memperbaiki sistem pada novel online dengan cara tidak membatasi bacaan pengguna, tingkatkan fitur yang ada tanpa di aplikasi tanpa menghapus fitur yang lain, dan mengembalikan penggunaan data offline pada saat membaca novel agar pengguna merasa nyaman tanpa gangguan pada saat membaca novel.

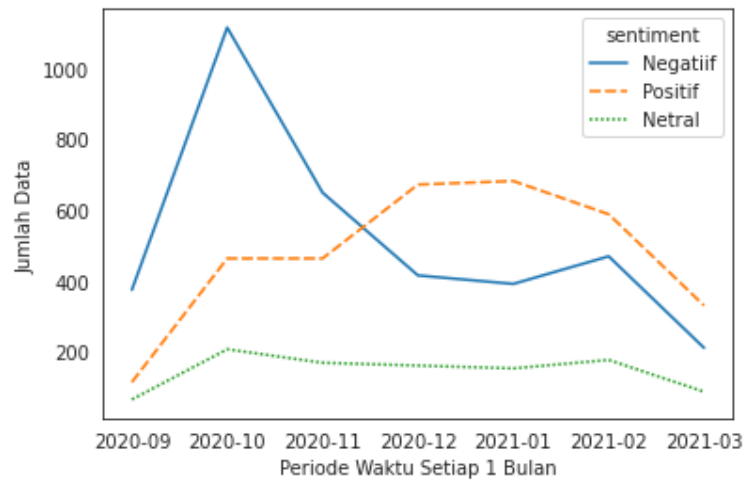
4.7. Analisis Trend per Bulan

Analisis trend adalah sebuah analisis yang digunakan untuk mengamati kecenderungan data secara menyeluruh dari waktu ke waktu. Pada penelitian ini untuk melihat seberapa banyak ulasan yang membicarakan setiap topik per bulannya dan untuk mengetahui perkembangan aplikasi novel online berdasarkan ulasan pengguna, disajikan dengan trend sebagai berikut:



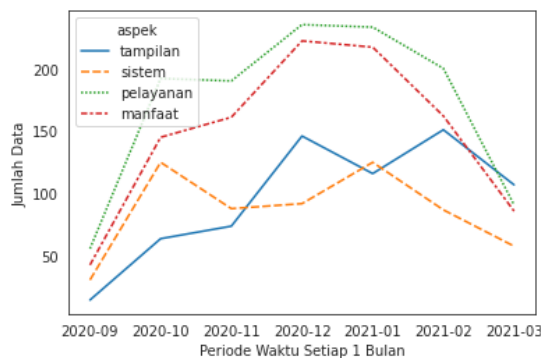
Gambar 4. 36 Trend Kelas Aspek per Bulan

Berdasarkan Gambar 4. 36, merupakan trend topik tiap bulan. Pada topik sistem membentuk pola trend menurun, terlihat bahwa dari bulan September 2020 sampai Oktober 2020 grafik mengalami kenaikan, tetapi bulan Oktober 2020 topik sistem terus menurun tiap bulannya sampai bulan Maret 2021, dan mengalami kenaikan sedikit dibulan Desember 2020 sampai Januari 2021. Untuk topik tampilan membentuk pola berulang, terlihat bahwa grafik naik dari bulan September 2020 sampai Oktober 2020, November 2020 sampai Desember 2020, dan Januari 2021 sampai Februari 2021, dan grafik turun dari bulan Oktober 2020 sampai bulan November 2020, Desember 2020 sampai Januari 2021, dan Februari 2021 sampai Maret 2021. Untuk topik pelayanan membentuk pola menurun, terlihat bahwa grafik naik dari bulan September 2020 sampai Oktober 2020 dan terus menurun tiap bulannya sampai bulan Maret 2021. Untuk topik manfaat terlihat bahwa pola grafik naik dari bulan September 2020 sampai Januari 2021, dan mengalami penurunan drastis dari bulan Februari 2021 sampai Maret 2021. Maka dapat disimpulkan bahwa naik nya grafik pada setiap aspek di bulan September 2020 sampai Oktober 2020, dikarenakan banyak jumlah ulasan yang tertulis dan aspek manfaat menjadi aspek yang sedikit ditulis oleh pengguna novel online.

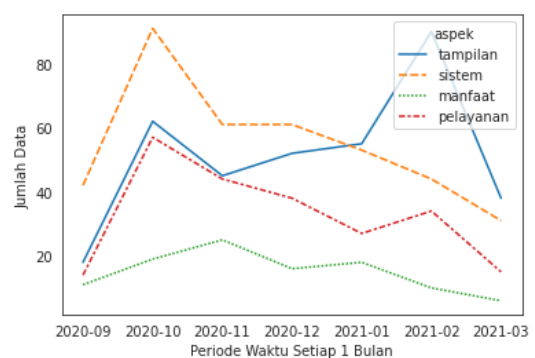


Gambar 4. 37 Trend Kelas Sentimen per Bulan

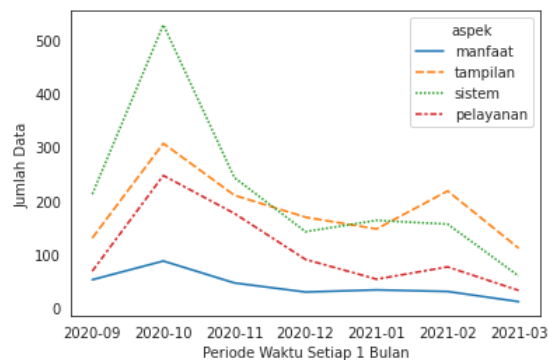
Berdasarkan Gambar 4. 18, merupakan trend sentimen per bulan. Pada sentimen negatif terlihat bahwa dari bulan September 2020 sampai Oktober 2020 grafik mengalami kenaikan, lalu setelah bulan Oktober 2020 sampai Januari 2021 mengalami penurunan. Pada bulan Januari 2021 sampai Februari 2021 grafik mengalami sedikit naik, lalu grafik turun dari bulan Februari 2021 sampai Maret 2021. Sedangkan pada sentimen positif grafik mengalami kenaikan dari September 2020 sampai Januari 2021. Lalu grafik turun setelah bulan Januari 2021 sampai Maret 2021. Maka dapat disimpulkan bahwa grafik sentimen positif dan negatif tidak seimbang di setiap bulannya. Lalu ulasan sentimen negatif meningkat drastic pada bulan September 2020 sampai bulan Oktober 2020.



(a)



(b)



(c)

Gambar 4. 38 Visualisasi Jumlah Data Kelas Sentimen Tiap Aspek (a) Positif (b) Netral dan (c) Negatif

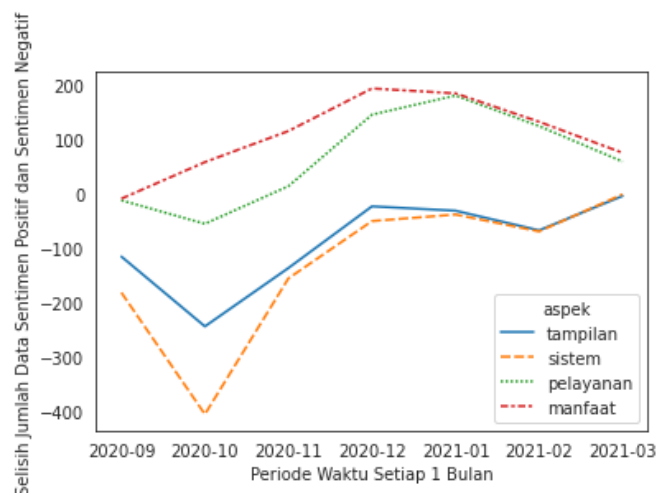
Pada Gambar 4. 38 *Visualisasi Jumlah Data Kelas Sentimen Tiap Aspek (a) Positif (b) Netral dan (c) Negatif*, merupakan grafik pola trend kelas sentimen positif tiap aspek per bulan. Untuk kelas sentimen positif pada aspek pelayanan, menunjukkan pola grafik menurun pada bulan Februari 2021 sampai bulan Maret 2021. Untuk kelas sentimen netral menurun pada bulan Februari 2021 sampai bulan Maret 2021. Untuk kelas sentimen negatif menurun pada bulan Februari 2021 sampai bulan Maret 2021. Maka dapat disimpulkan bahwa ulasan sentimen kelas positif, netral, dan negatif sama banyak untuk bulan Februari 2021 sampai bulan Maret 2021. Sehingga novel *online* perlu memperbaiki untuk meningkatkan kualitas pelayanan pada aplikasi novel *online* agar tidak terus menurun pada bulan selanjutnya.

Untuk kelas sentimen positif pada aspek manfaat, menunjukkan pola grafik menurun pada bulan Februari 2021 sampai bulan Maret 2021. Untuk kelas sentimen netral menurun pada bulan Februari 2021 sampai bulan Maret 2021. Untuk kelas sentimen negatif menurun pada bulan Februari 2021 sampai bulan Maret 2021. Maka dapat disimpulkan bahwa ulasan sentimen kelas positif, netral, dan negatif sama banyak untuk bulan Februari 2021 sampai bulan Maret 2021. Sehingga novel *online* perlu memperbaiki untuk meningkatkan kualitas manfaat pada aplikasi novel *online* agar tidak terus menurun pada bulan selanjutnya.

Untuk kelas sentimen positif pada aspek tampilan, menunjukkan pola grafik menurun pada bulan Februari 2021 sampai bulan Maret 2021. Untuk kelas

sentimen netral menurun pada bulan Februari 2021 sampai bulan Maret 2021. Untuk kelas sentimen negatif menurun pada bulan Februari 2021 sampai bulan Maret 2021. Maka dapat disimpulkan bahwa ulasan sentimen kelas positif, netral, dan negatif sama banyak untuk bulan Februari 2021 sampai bulan Maret 2021. Sehingga novel *online* perlu memperbaiki untuk meningkatkan kualitas tampilan pada aplikasi novel *online* agar tidak terus menurun pada bulan selanjutnya.

Untuk kelas sentimen positif pada aspek sistem, menunjukkan pola grafik menurun pada bulan Februari 2021 sampai bulan Maret 2021. Untuk kelas sentimen netral menurun pada bulan Februari 2021 sampai bulan Maret 2021. Untuk kelas sentimen negatif menurun pada bulan Februari 2021 sampai bulan Maret 2021. Maka dapat disimpulkan bahwa ulasan sentimen kelas positif, netral, dan negatif sama banyak untuk bulan Februari 2021 sampai bulan Maret 2021. Sehingga novel *online* perlu memperbaiki untuk meningkatkan kualitas sistem pada aplikasi novel *online* agar tidak terus menurun pada bulan selanjutnya.



Gambar 4. 39 Visualisasi Selisih Jumlah Data Sentimen Positif dan Negatif

Perdasarkan Gambar 4. 39 *Visualisasi Selisih Jumlah Data Sentimen Positif dan Negatif*, untuk lebih menjelaskan perbandingan sentimen pada sentimen analisis berbasis aspek. Terlihat bahwa aspek tampilan pada bulan Februari 2021 sampai Maret 2021 mengalami kenaikan grafik. Ini menjelaskan bahwa aspek tampilan bersentimen positif lebih banyak daripada yang bersentimen negatif. Sehingga pada aplikasi novel online perlu dijadikan acuan untuk meningkatkan

kualitas tampilan untuk bulan selanjutnya. Untuk aspek sistem pada bulan Februari 2021 sampai Maret 2021 mengalami kenaikan grafik. Ini menjelaskan bahwa aspek sistem bersentimen positif lebih banyak daripada yang bersentimen negatif. Sehingga pada aplikasi novel online perlu dijadikan acuan untuk meningkatkan kualitas sistem untuk bulan selanjutnya.

Untuk aspek pelayanan pada bulan Februari 2021 sampai Maret 2021 mengalami penurunan grafik. Ini menjelaskan bahwa aspek sistem bersentimen negatif lebih banyak daripada yang bersentimen positif. Sehingga pada aplikasi novel online perlu dijadikan acuan untuk meningkatkan kualitas pelayanan untuk bulan selanjutnya. Untuk aspek manfaat pada bulan Februari 2021 sampai Maret 2021 mengalami penurunan grafik. Ini menjelaskan bahwa aspek sistem bersentimen negatif lebih banyak daripada yang bersentimen positif. Sehingga pada aplikasi novel online perlu dijadikan acuan untuk meningkatkan kualitas manfaat untuk bulan selanjutnya.

BAB V

PENUTUP

5.1. Kesimpulan

Berdasarkan hasil pengujian dan penjelasan yang dipaparkan, maka dapat diperoleh kesimpulan sebagai berikut:

1. Penentuan jumlah aspek dominan dilakukan menggunakan dengan cara *clustering topic* menggunakan Latent Dirichlet Allocation (LDA) menghasilkan 4 aspek dominan yaitu aspek manfaat sebanyak 1429 ulasan, aspek pelayanan sebanyak 2168 ulasan, aspek tampilan sebanyak 2325 ulasan, dan aspek sistem 2493 sebanyak ulasan.
2. Analisis sentiment berbasis aspek menggunakan metode *Bidirectional Encoder Representations from Transformers* (BERT). Untuk klasifikasi sentiment pada aspek Manfaat dengan *learning rate* 10^{-5} memiliki nilai akurasi sebesar 76% dan nilai AUC sebesar 0.893, karena data imbalanced maka memerlukan bantuan teknik SMOTE diperoleh nilai akurasi (89%) dan AUC (97%). Klasifikasi sentiment pada aspek Tampilan dengan *learning rate* 10^{-5} memiliki nilai akurasi sebesar 63% dan nilai AUC sebesar 0.780, karena data imbalanced maka memerlukan bantuan teknik SMOTE diperoleh nilai akurasi (74%) dan AUC (88%). Klasifikasi sentiment pada aspek sistem dengan *learning rate* $3 \cdot 10^{-5}$ memiliki nilai akurasi sebesar 64% dan nilai AUC sebesar 0,763, karena data imbalanced maka memerlukan bantuan teknik SMOTE diperoleh nilai akurasi (73%) dan AUC (86%). Sedangkan, klasifikasi sentiment pada aspek sistem dengan *learning rate* 10^{-5} memiliki nilai akurasi sebesar 72% dan nilai AUC sebesar 0.854, karena data imbalanced maka memerlukan bantuan teknik SMOTE diperoleh nilai akurasi (86%) dan AUC (95%). Dalam Penelitian ini nilai akurasi yang dihasilkan tanpa teknik SMOTE kurang baik bisa disebabkan oleh beberapa kemungkinan antara lain yaitu data yang digunakan adalah data imbalance, penentuan sentimen yang dilakukan berdasarkan melihat rating yang diberikan oleh pengguna banyak

terdapat perbedaan seperti memberikan ulasan positif tetapi memberikan rating negatif, selain itu dalam hal penentuan hyperparameter yang digunakan untuk penelitian membutuhkan analisis yang baik oleh orang-orang yang benar ahlinya, karena dalam penelitian ini peneliti masih sangat pemula dalam melakukan maka bisa saja terjadi kesalahan.

3. Kelebihan aplikasi novel *online* dari aspek manfaat yaitu terdapat berbagai macam *genre*, dan meningkatkan halusinasi pembaca. Sedangkan kekurangannya yaitu batasan jumlah novel. Kelebihan dari aspek pelayanan yaitu novelnya bagus, aplikasi sangat seru. Sedangkan kekurangannya yaitu sering terjadi pembaruan secara otomatis dan email yang digunakan tidak terdata. Kelebihan dari aspek tampilan yaitu mudah untuk mencari novel yang direkomendasikan. Sedangkan kekurangannya yaitu iklan disetiap pergantian bab, dan fitur dihilangkan. Kelebihan dari aspek sistem yaitu cerita yang dibuat oleh penulis aplikasi novel *online* bagus untuk dibaca. Sedangkan kekurangannya yaitu perubahan penggunaan data dari *offline* (luar jaringan) menjadi *online* (dalam jaringan) pada saat membaca.
4. Visualisasi analisis trend jumlah data kelas sentimen pada tiap aspek terjadi fluktuasi setiap 1 bulan. Visualisasi Jumlah data Kelas Sentimen Tiap Aspek per bulan pada bulan Februari 2021 sampai Maret 2021 dapat menjadi acuan untuk meningkatkan kualitas tampilan, sistem, manfaat dan pelayanan pada aplikasi novel online. Untuk aspek dominan tampilan dan sistem pada bulan Februari 2021 sampai Maret 2021 mengalami kenaikan grafik. Sedangkan untuk aspek dominan pelayanan dan manfaat pada bulan Februari 2021 sampai Maret 2021 mengalami penurunan grafik.

5.2. Saran

Dalam penelitian ini, peneliti menyadari banyak terjadi kekurangan dan keterbatasan dalam penelitian. Adapun saran yang dapat dipertimbangkan untuk peneliti lain yang tertarik untuk melanjutkan penelitian terkait dengan ini diantaranya adalah memperbanyak data, menggunakan data yang balance antara sentimen positif, netral, dan negatif, jumlah data yang digunakan untuk penelitian

bisa lebih banyak lagi. Selain itu, dapat menggunakan model BERT dalam bahasa Indonesia atau *multilanguage* lainnya untuk mendapatkan hasil yang lebih baik.

Daftar Pustaka

- [1] M. E. Anindita, "Manfaat membaca e-book gramedia digital," Gramedia, 08 11 2018. [Online]. Available: <https://www.gramedia.com/blog/manfaat-membaca-e-book-gramedia-digital/#gref>. [Accessed 15 02 2021].
- [2] R. Potharaju, M. Rahman and B. Carbunar, "A Longitudinal Study of Google Play," *IEEE Transactions on Computational Social Systems*, vol. 4, pp. 135-149, 2017.
- [3] L. Bing, "Sentiment Analysis and Subjectivity," in *Handbook of natural language processing*, 2010, pp. 627-666.
- [4] T. T. Thet, J.-C. Na and C. S. Khoo, "Aspect-based sentiment analysis of movie reviews on discussion boards," *Journal of Information Science*, vol. 36, no. 6, pp. 823-848, 2010.
- [5] R. A. P. Sarno and Riyanarto, "Sentiment Analysis of Hotel Reviews Using Latent Dirichlet Allocation, Semantic Similarity and LSTM," *International Journal of Intelligent Engineering and Systems*, vol. 12, no. 4, 2019.
- [6] F. Xianghua, L. Guo, G. Yanyan and W. Zhiqiang, "Multi-aspect sentiment analysis for Chinese online social reviews based," *Knowledge-Based Systems*, vol. 37, p. 186–195, 2013.
- [7] B. Wilie, K. Vincentio, G. I. Winata, S. Cahyawijaya, X. Li, Z. Y. Lim, S. Soleman, R. Mahendra, P. Fung, S. Bahar and A. Purwarianti, "IndoNLU: Benchmark and Resources for Evaluating Indonesian Natural Language Understanding," *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pp. 843-857, 2020.
- [8] A. Alghunaim, "A vector space approach for aspect-based sentiment analysis," *Diss. Massachusetts Institute of Technology*, 2015.
- [9] D. Osinga, *Deep learning cookbook: practical recipes to get started quickly*, O'Reilly Media, Inc., 2018.
- [10] D. Jurafsky and J. H. Martin, "Speech and language processing," *In Prentice Hall, 3rd Edition*, 2019.

- [11] E. Pekel and S. S. Kara, "A Comprehensive Review for Artificial Neural Network Application to Public Transportation," *Sigma Journal of Engineering and Natural Sciences*, p. 157–179, 2017.
- [12] H. Kagaya, K. Aizawa and M. Ogawa, "Food Detection and Recognition Using Convolutional Neural Network," *MM '14: Proceedings of the 22nd ACM international conference on Multimedia*, p. 1085–1088, 2014.
- [13] A. A. Rizal and S. Soraya, "Multi Time Steps Prediction dengan Recurrent Neural Network Long Short Term Memory," *MATRIK : Jurnal Manajemen, Teknik Informatika Dan Rekayasa Komputer*, vol. 18, no. 1, pp. 115-124, 2018.
- [14] A. Ingolfsson and E. Sachs, "Recurrent neural network based language model," *J. Qual. Technol* 25.4, pp. 271-287, 1993.
- [15] S. Alhagry, A. A. Fahmy and R. A. El-Khoribi, "Emotion Recognition based on EEG using LSTM," (*IJACSA*) *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 10, pp. 355-358, 2017.
- [16] M. Munikar, S. Shakya and A. Shrestha, "Fine-grained sentiment classification using bert," *2019 Artificial Intelligence for Transforming Business and Society (AITB). IEEE*, vol. 1, 2019..
- [17] A. Santoso and G. Ariyanto, "Implementasi deep learning berbasis keras untuk pengenalan wajah," *Emitor: Jurnal Teknik Elektro* 18.1, pp. 15-21, 2018.
- [18] C. C. Aggarwal, "Neural networks and deep learning," *Springer* 10, pp. 978-3, 2018.
- [19] F. Chollet, *Deep Learning mit Python und Keras: Das Praxis-Handbuch vom Entwickler der Keras-Bibliothek*, MITP-Verlags GmbH & Co. KG, 2018.
- [20] D. P. Kingma and J. Ba., "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
- [21] F. Kyoomarsi, H. Khosravi, E. Eslami, P. K. Dehkordy and A. Tajoddin, "Optimizing text summarization based on fuzzy logic," *Seventh IEEE/ACIS International Conference on Computer and Information Science (icis 2008)*, 2008.
- [22] S. N. Kane, A. Mishra and A. Gaur, "PREFACE: International Conference on Recent Trends in Physics (ICRTP 2014)," *Journal of Physics Conference Series*, vol. 534, 2014.

- [23] D. S. P. Raj and a. S. Rajaeaajeswari, "A Framework for Text Analytics using the Bag of Words (BoW) Model for Prediction," *International Journal of Advanced Networking & Application (IJANA)*, pp. 320-323, 2016.
- [24] M. S. Wibawa, "Pengaruh Fungsi Aktivasi, Optimisasi dan jumlah *Epoch* Terhadap Performa Jaringan Saraf Tiruan," *Jurnal Sistem dan Informatika*, vol. 11, p. 2, 2017.
- [25] D. Cahyadi, Ekstrasi dan Kemiripan Fitur Mata pada Sistem Identifikasi Buron, Depok: FASILKOM UI, 2007.
- [26] D. M. Blei, A. Y. Ng and M. I. Jordan, "Latent Dirichlet Allocation," *Journal of Machine Learning Research* 3, vol. 3, pp. 993-1022, 2003.
- [27] I. M. K. B. Putra, "Analisis Topik Informasi Publik Media Sosial di Surabaya Menggunakan Pemodelan Latent Dirichlet Allocation (LDA)," *Undergraduate thesis, Institut Teknologi Sepuluh Nopember*, vol. 6, no. 2, 2017.
- [28] K. S. Nugroho, A. Y. Sukmadewa, H. W. DW, F. A. Bachtiar and N. Yudistira, "BERT Fine-Tuning for Sentiment Analysis on Indonesian Mobile Apps Reviews," *6th International Conference on Sustainable Information Engineering and Technology 2021*, 2021.
- [29] J. Devlin, M.-W. Chang, k. Lee and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference, 1(Mlm)*, p. 4171–4186, 2019.
- [30] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser and I. Polosukhin, "Attention is all you need," *Advances in Neural Information Processing Systems*, p. 5999–6008, 2017.
- [31] H. Schütze, C. D. Manning and P. Raghavan, "Introduction to information retrieval," *Cambridge: Cambridge University Press*, vol. 39, 2008.
- [32] J. Alammar, "The Illustrated Transformer," 2018. [Online]. Available: <https://jalammar.github.io/illustrated-transformer/>. [Accessed 07 09 2021].
- [33] M. Bekkar, H. Djema and T. Alitouche, "Evaluation measures for models assessment over imbalanced data sets," *Journal of Information Engineering and Applications*, vol. 10, no. 3, 2013.

- [34] E. Jonsson and J. Stolee, "An Evaluation of Topic Modelling Techniques for Twitter," 2016.
- [35] C. Sievert and K. E. Shirley, "LDAvis: A Method For Visualizing and Interpreting Topics," *ACL Workshop on Interactive Language Learning, Visualization, and Interface*, pp. 63-70, 2014.
- [36] R. A. Barro, I. D. Sulvianti and F. M. Afendi, "Penerapan Synthetic Minority Oversampling Technique (SMOTE) Terhadap Data Tidak Seimbang Pada Pembuatan Model Komposisi Jamu," *Xplore*, vol. 1, no. 1, pp. 1-6, 2013.

Lampiran

Lampiran I

Link untuk melihat data dan codingan:

https://drive.google.com/drive/folders/1JtrW3-G0VPaWERSQTI8NmRTTK_NzBfVZ?usp=sharing