

**ANALISIS SENTIMEN *REVIEW SKINCARE* LOKAL PADA *TWITTER*
MENGUNAKAN PENDEKATAN BERT**

SKRIPSI

Diajukan Untuk Melengkapi Tugas Dan Memenuhi Syarat Memperoleh Ijazah
Sarjana Komputer

AYU AFREZA SIREGAR

181401014



**PROGRAM STUDI S-1 ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA**

MEDAN

2024

**ANALISIS SENTIMEN *REVIEW SKINCARE* LOKAL PADA *TWITTER*
MENGUNAKAN PENDEKATAN BERT**

SKRIPSI

Diajukan Untuk Melengkapi Tugas Dan Memenuhi Syarat Memperoleh Ijazah
Sarjana Komputer

AYU AFREZA SIREGAR

181401014



**PROGRAM STUDI S-1 ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA**

MEDAN

2024

UNIVERSITAS SUMATERA UTARA

PERSETUJUAN

Judul : Analisis Sentimen *Review Skincare* Lokal pada
Twitter Menggunakan Pendekatan BERT

Kategori : Skripsi

Nama : Ayu Afreza Siregar

Nomor Induk Mahasiswa : 181401014

Program Studi : SARJANA (S-1) ILMU KOMPUTER

Fakultas : ILMU KOMPUTER DAN TEKNOLOGI
 INFORMASI UNIVERSITAS SUMATERA
 UTARA

Komisi Pembimbing :

Pembimbing I



Dr. Amalia S.T., M.T

NIP.197812212014042001

Pembimbing II



Dr. T. Henny Febriana Harumy, S.Kom., M.Kom

NIP.198802192019032016

Diketahui/ Disetujui oleh

Program Studi S-1 Ilmu Komputer

Ketua,



Dr. Amalia S.T., M.T

NIP.197812212014042001



PERNYATAAN**ANALISIS SENTIMEN *REVIEW SKINCARE* LOKAL PADA *TWITTER*
MENGUNAKAN PENDEKATAN BERT****SKRIPSI**

Saya mengakui bahwa skripsi ini adalah hasil karya saya sendiri, kecuali beberapa kutipan dan ringkasan yang masing-masing telah disebutkan sumbernya.

Medan, Desember 2023



Ayu Afreza Siregar

181401014

KATA PENGANTAR

Segala puji bagi Allah SWT yang memberikan kenikmatan berupa Iman, Islam, berkah, rahmat, kesehatan serta kekuatan kepada penulis, sehingga penulis bisa menyelesaikan tugas akhir untuk mendapatkan gelar Sarjana Komputer di Program Studi S-1 Ilmu Komputer, Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Sumatera Utara. Penulis juga menyampaikan shalawat dan salam kepada junjungan Nabi Besar Muhammad SAW beserta keluarga, dan para sahabat.

Dengan penuh rasa hormat dan terima kasih penulis ingin menyampaikan kepada:

1. Bapak Dr. Muryanto Amin, S.Sos., M.Si. selaku Rektor Universitas Sumatera Utara
2. Ibu Dr. Maya Silvi Lydia B.Sc., M.Sc. selaku Dekan Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara.
3. Ibu Dr. Amalia ST., M.T. selaku Ketua Program Studi S-1 Ilmu Komputer Universitas Sumatera Utara dan juga selaku Dosen Pembimbing I yang sudah memberikan banyak saran, kritik, dukungan, dan motivasi pada pengerjaan skripsi ini.
4. Ibu Dr. T. Henny Febriana Harumy, S.Kom., M.Kom. selaku Dosen Pembimbing II yang sudah memberikan saran, kritik, dukungan, dan motivasi pada pengerjaan skripsi ini.
5. Dosen dan Staff Pegawai Program Studi S-1 Ilmu Komputer Fasilkom-TI USU.
6. Kedua orangtua tersayang, Ibunda Rohana dan Ayahanda Indra Siregar, yang telah senantiasa memberi dukungan, doa, dan motivasi anak-anaknya.
7. Kepada kakak kandung tersayang, Tia Novita Siregar, dan sepupu tersayang, Devy Syafriyanti yang selalu memberi dukungan, doa, serta semangat untuk penulis.
8. Sahabat penulis. Annisa Novia Larasati, Putri Natasya, dan Iswanti yang sudah memberikan motivasi, semangat, serta doa untuk penulis.
9. Sahabat seperjuangan penulis Kania Ulfa Lubis, yang telah bersama sejak awal perkuliahan sampai sekarang dan selalu membantu, mendoakan, menyemangati penulis dalam suka maupun duka.
10. Seluruh teman satu stambuk 2018 prodi Ilmu Komputer yang telah bersama selama masa perkuliahan.

11. Seluruh pihak yang terlibat baik secara langsung ataupun tidak langsung yang sudah banyak memberikan bantuan, yang tidak bisa disebutkan satu per satu.
12. Terima kasih kepada diri sendiri, Ayu Afreza Siregar yang sudah berhasil bertahan dan tidak menyerah sehingga skripsi dapat selesai.

Medan, Desember 2023

A handwritten signature in black ink, appearing to read 'Ayu -af' with a stylized flourish at the end.

Penulis

ABSTRAK

Skincare menjadi isu penting bagi wanita sejak jaman dahulu. Banyaknya produk lokal menjadikan konsumen lebih selektif dalam memilih produk yang digunakan. *Review* produk menjadi acuan dalam membeli *skincare*. *Review* yang diambil yaitu berasal dari *twitter*. Penelitian ini bertujuan untuk mengetahui Analisis Sentimen *Review Skincare Lokal Pada Twitter Menggunakan Pendekatan Bert*. Penelitian ini juga mengatasi permasalahan kata negasi pada *review user* sehingga hasil menjadi lebih baik dengan metode yang digunakan yaitu dengan memberi tanda pada kata yang mengandung negasi. Penelitian ini menggunakan Pendekatan Bert, BERT adalah sebuah *pretrained language* model, sebuah arsitektur, dan juga bisa disebut sebagai sebuah *metode state of the art* dalam dunia *natural language processing* dan merupakan sebuah model yang dapat mempelajari teks secara kontekstual. Dengan populasinya adalah *tweet* yang berasal dari *Twitter* dengan pengambilan sampel yang memuat *review* dari *skincare* lokal MS Glow sebesar 1389 data *review skincare* MS Glow. Dengan hasil pengujian dengan tiga *epoch* 5,10, dan 16. *Epoch* 16 menghasilkan hasil terbaik, sehingga *epoch* 16 digunakan untuk menganalisis sentimen. Hasil pengujian menggunakan metode *Birectional Encoder Representations from Transfromer* (BERT) menghasilkan akurasi 80% dengan menggunakan *hyperparamters* yaitu *batch size* sebesar 16, *learning rate* 5e-6, dan *epoch* 16. Hasil akurasi terbaik didapatkan oleh *epoch* 16 yaitu 80%

Kata Kunci : *Analisis Sentimen , Review, Skincare lokal, MS GLOW, BERT*

ABSTRACT

Skincare has been an important issue for women for a long time. The large number of local products makes consumers more selective in choosing the products they use. Product reviews are a reference when buying skincare. The reviews taken are from Twitter. This research aims to determine the sentiment analysis of local skincare reviews on Twitter using the Bert approach. This research also addresses the problem of negated words in user reviews so that the results are better with the method used, namely by marking words that contain negation. This research uses the Bert Approach, BERT is a pretrained language model, an architecture, and can also be called a state of the art method in the world of natural language processing and is a model that can study text contextually. The population is tweets originating from Twitter with sampling using tweet data containing reviews of local MS GLOW skincare of 1389 MS Glow skincare review data. With test results with three epochs 5, 10, and 16. Epoch 16 produces the best results, so epoch 16 is used to analyze sentiment. Test results using the Bidirectional Encoder Representations from Transfromer (BERT) method produced 80% accuracy using hyperparamters, namely batch size of 16, learning rate 5e-6, and epoch 16. The best accuracy results were obtained by epoch 16, namely 80%

Keywords: *Sentiment Analysis, Review, Local Skincare, MS GLOW, BERT*

DAFTAR ISI

PERNYATAAN	ii
KATA PENGANTAR	iii
ABSTRACT	vi
DAFTAR ISI	vii
DAFTAR TABEL	x
DAFTAR GAMBAR	xi
DAFTAR LAMPIRAN	xi
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	4
1.3 Batasan Masalah	4
1.4 Tujuan Penelitian	5
1.5 Manfaat Penelitian	5
1.6 Metodologi Penelitian	5
BAB II LANDASAN TEORI	7
2.1. Analisis Sentimen	7
2.2 Skincare	8
2.3 <i>Twitter</i>	8
2.4 Pre-processing Dataset	9
2.5 <i>Deep Learning</i>	10
2.6 Metode BERT	11
2.7 Penelitian yang Relevan	17
BAB III ANALISIS DAN PERANCANGAN	20
3.1 Arsitektur Umum	20
3.1.1 <i>Scraping Twitter</i>	20
3.1.2 Labelisasi <i>Dataset</i>	21

3.1.3 <i>Pre-processing Dataset</i>	21
3.1.3.1 <i>Case Folding</i>	21
3.1.3.2 <i>Data cleaning</i>	22
3.1.2.3 <i>Tokenisasi</i>	23
3.1.2.4 <i>Negation Handling</i>	24
3.1.2.5 <i>Stopwords removal</i>	25
3.1.2.6 <i>Stemming</i>	26
3.1.2.7 <i>Normalisasi</i>	27
3.2 <i>BERT Sentiment Classifier</i>	28
3.5 <i>Evaluasi</i>	32
BAB IV IMPLEMENTASI DAN PENGUJIAN	34
4.1 <i>Implementasi Sistem</i>	34
4.4.1 <i>Spesifikasi perangkat keras</i>	34
4.1.2 <i>Perangkat Lunak</i>	34
4.2 <i>Implementasi Scraping Data Tweet</i>	34
4.3 <i>Labelisasi dataset</i>	35
4.4 <i>Preprocessing dataset</i>	35
4.4.1 <i>Case Folding</i>	36
4.4.2 <i>Data Cleaning</i>	37
4.4.3 <i>Tokenize</i>	37
4.4.4 <i>Negation Handling</i>	38
4.4.5 <i>Stopword Removal</i>	38
4.4.6 <i>Stemming</i>	39
4.4.7 <i>Normalisasi</i>	39
4.5 <i>Split Dataset</i>	39
4.6 <i>BERT</i>	40

4.7 Evaluasi	47
BAB V KESIMPULAN DAN SARAN	50
5.1 Kesimpulan.....	50
5.2 Saran.....	50
DAFTAR PUSTAKA.....	51
LAMPIRAN	54

DAFTAR TABEL

Tabel 1. Contoh <i>Dataset</i>	21
Tabel 2. <i>Pseudocode Case Folding</i>	21
Tabel 3. Perbandingan Hasil <i>Case Folding</i>	22
Tabel 4. <i>Pseudocode Data Cleaning</i>	22
Tabel 5. Perbandingan Hasil <i>Data Cleaning</i>	23
Tabel 6. Tabel <i>Pseudocode Tokenisasi</i>	23
Tabel 7. Perbandingan Hasil Tokenisasi	24
Tabel 8. <i>Pseudocode Negation Handling</i>	24
Tabel 9. Perbandingan Hasil <i>Negation Handling</i>	25
Tabel 10. <i>Pseudocode Stopwords removal</i>	25
Tabel 11. Contoh Stopword Kamus Tala	26
Tabel 12. <i>Pseudocode Stemming</i>	27
Tabel 13. Perbandingan Hasil <i>Stemming</i>	27
Tabel 14. <i>Pseudocode Normalisasi</i>	27
Tabel 15. Perbandingan Hasil Normalisasi	28
Tabel 16. <i>Pseudocode BERT</i>	31
Tabel 17. <i>Confusion Martix</i>	32
Tabel 18. Hasil Pengujian	44
Tabel 19. Tabel Prediksi	46
Tabel 20. Perbandingan <i>Batch Size</i>	49

DAFTAR GAMBAR

Gambar 2. 1	Arsitektur <i>Encoder</i> dan <i>Decorder</i> pada <i>Transformer</i>	11
Gambar 2. 2	Perbedaan Ukuran <i>BERT_{BASE}</i> dan <i>BERT_{LARGE}</i>	12
Gambar 2. 3	Arsitektur <i>BERT</i>	13
Gambar 2. 4	Proses <i>Pre-Training</i> dan <i>Fine-tuning</i>	14
Gambar 2. 5	Proses <i>Masked Language Modelling (Masked LM)</i>	15
Gambar 2. 6	Representasi Input pada <i>BERT</i>	15
Gambar 3. 1	Arsitektur Umum.....	20
Gambar 3. 2	Proses Tokenisasi Dengan <i>Indobert</i>	28
Gambar 3. 3	Proses <i>Token Embeddings</i>	29
Gambar 3. 4	Proses Pemberian <i>Padding</i>	29
Gambar 3. 5	Tahap Substitusi Token dengan IDnya	30
Gambar 3. 6	Tahap <i>Sentence Embedding</i>	30
Gambar 3. 7	Tahap <i>Positional Embedding</i>	30
Gambar 4. 1	Hasil <i>Scraping</i>	35
Gambar 4. 2	Labelisasi <i>Dataset</i>	35
Gambar 4. 3	<i>Dataset</i>	36
Gambar 4. 4	Hasil <i>Case Folding</i>	36
Gambar 4. 5	Hasil <i>Data Cleaning</i>	37
Gambar 4. 6	Hasil <i>Tokenize</i>	37
Gambar 4. 7	Hasil <i>Negation Handling</i>	38
Gambar 4. 8	Hasil <i>Stopword Removal</i>	38
Gambar 4. 9	Hasil <i>Stemming</i>	39
Gambar 4. 10	Hasil Normalisasi.....	39
Gambar 4. 11	Proses <i>Spilitting Dataset</i>	40
Gambar 4. 12	Akurasi dengan 5 <i>Epoch</i>	41
Gambar 4. 13	Akurasi dengan 10 <i>Epoch</i>	41
Gambar 4. 14	Akurasi dengan 16 <i>Epoch</i>	41
Gambar 4. 15	Proses <i>Training dan Evaluasi</i>	42
Gambar 4. 16	Kurva Hasil Performa <i>Training dan Validasi Epoch 5</i>	42

Gambar 4. 17 Kurva Hasil Performa <i>Training</i> dan Validasi <i>Epoch</i> 10	43
Gambar 4. 18 Kurva Hasil Performa <i>Training</i> dan Validasi <i>Epoch</i> 16	43
Gambar 4. 19 Diagram <i>Confusion Matrix</i>	47
Gambar 4. 20 Hasil Evaluasi	49

DAFTAR LAMPIRAN

Lampiran <i>Listing</i> Program	A-1
---------------------------------------	-----

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perawatan kulit telah menjadi isu penting bagi wanita sejak zaman dahulu. Saat ini perawatan kulit atau yang sering disebut dengan *skin care* sudah semakin *universal* dan dibutuhkan tidak hanya oleh wanita namun juga oleh pria. Seiring berjalannya waktu, *skin care* mengalami perkembangan yang sangat pesat dalam hal perawatan. Salah satu brand produk perawatan kulit lokal adalah MS Glow. MS Glow menawarkan banyak varian produk antara lain perawatan wajah, perawatan tubuh, dan kosmetik. Pada tahun 2013, Maharani Kemala dan Shandy Purnamasari mendirikan MS Glow. MS Glow telah memantapkan posisinya yang unik di masyarakat dengan cabang MS Glow yang tersebar di beberapa kota di Indonesia (GLOW, 2013)

Berbagai jenis produk *skincare* dibuat dengan metode tradisional dengan memakai berbagai bahan alami hingga yang diolah secara modern. Meningkatnya jumlah produk *skincare* berarti semakin banyak pilihan bagi pelanggan. Untuk mempromosikan *skincare*, produsen produk berlomba-lomba mengembangkan produknya. Konsep kecantikan pada remaja putri menggambarkan tubuh perempuan yang menarik bagi dirinya dan orang lain. Hal ini dapat menimbulkan perasaan puas terhadap diri sendiri sehingga dapat mempengaruhi percaya diri seseorang. Keinginan remaja untuk tampil cantik membuat mereka menjadi konsumtif dan mudah tergiur dengan informasi berbagai produk kecantikan, termasuk produk yang memberikan hasil instan. Kurangnya pengetahuan mengenai kesalahan dalam memilih produk *skincare* produk yang mengandung produk ilegal atau berbahaya. Produk perawatan kulit yang mengandung bahan berbahaya dapat menimbulkan risiko kesehatan, antara lain iritasi kulit seperti kemerahan, mengelupas, dan terbakar, kerusakan organ dalam seperti otak dan ginjal, serta risiko kanker (Khafida & Hadiyati, 2020). Maka diperlukannya informasi ulasan pentingnya pemilihan suatu produk *skincare* yang tepat sesuai dengan jenis kulit.

Menurut (Zukhrufillah, 2018), berbagai jenis ulasan yang diberikan konsumen terhadap merek yang ada menjadi tolak ukur keputusan pembelian produk. Oleh karena itu, perlu dilakukan analisis *review* terhadap produk MS Glow untuk mengukur

reaksi masyarakat terhadap produk MS Glow. Di tahun 2018, menurut data survei Asosiasi Penyelenggara Jasa Internet Indonesia (APJII), di Indonesia terdapat 171,1 juta pengguna Internet. Angka ini menunjukkan peningkatan sebanyak 27,9 juta orang dibandingkan dengan tahun sebelumnya, yaitu 2017, yang mencatat jumlah pengguna sebanyak 143,2 juta orang. Jack Dorsey di bulan Maret 2006 mendirikan *Twitter*, yang kini sudah menjadi salah satu *platform* media sosial terbesar di Indonesia. *Twitter* memungkinkan penggunanya untuk membaca serta mengirim pesan teks dengan batasan sampai 140 karakter, yang dikenal sebagai *tweet*.

Munculnya MS Glow sebagai produk *skincare* lokal yang peminatnya tinggi membuat MS Glow harus terus berbenah agar tetap mampu bersaing dan mempertahankan posisinya di masyarakat. Dengan banyaknya produk lokal, konsumen dapat memilih produk yang sesuai dengan jenis kulitnya. Klaim merek seringkali tidak sesuai dengan hasil yang dicapai. Pengusaha harus mampu berinteraksi dengan konsumen dan calon konsumen dengan memberikan informasi mengenai produk yang ditawarkan. *Review* produk memberikan manfaat bagi mereka yang ingin membeli dan menggunakan suatu produk. Keputusan pembelian merupakan tahapan dimana konsumen memutuskan membeli sebuah produk (Kotler, 2016)

Banyaknya jumlah pengguna *Twitter* menghasilkan data *tweet* yang beragam. *Tweet* menjadi wadah bagi penggunanya untuk memberikan berbagai informasi mengenai isu-isu sosial. Pendapat pengguna tentang produk dapat digunakan sebagai evaluasi. Dengan berbagai data *tweet*, Anda dapat menggunakan analisis sentimen untuk menilai apakah suatu produk mendapat reaksi positif, negatif, maupun netral. Analisis sentimen adalah subbidang pemrosesan bahasa alami. Analisis sentimen digunakan untuk mengetahui apakah opini dan komentar terhadap sebuah topik dapat dijadikan acuan untuk meningkatkan kualitas layanan dan produk (Nurjanah *et al.*, 2017)

Oleh karena itu, bisa disimpulkan bila fokus analisis sentimen pada penelitian ini berfokus pada ekstraksi data dari platform *Twitter*. Data *tweet* yang telah diekstraksi kemudian diklasifikasi dilakukan dengan menentukan opini produk. *Tweet* terkait MS Glow tersedia banyak dengan *tweet* yang beragam. Saat ini banyak netizen berkomentar bahkan menghujat pemilik MS Glow karena kasus yang menimpa pemilik MS Glow. Dengan demikian, batasan masalah dalam penelitian ini terfokus

review produk skincare MS Glow. Data *tweet* diambil berfokus di beberapa kata kunci seperti “MS Glow whitening”, “MS Glow Acne”, “Night Cream MS Glow”, “Sunscreen MS Glow”, “Facewash MS Glow”, “Skincare MS Glow” sehingga data yang dihasilkan yaitu berupa *review* produk MS Glow.

Dapat disimpulkan bahwa fokus analisis sentimen pada penelitian ini adalah pada ekstraksi data dari platform *Twitter*. Data *tweet* yang diekstraksi kemudian diklasifikasikan dengan menentukan opini produk. Ada banyak *tweet* terkait MS Glow dengan *tweet* berbeda-beda. Saat ini banyak netizen yang berkomentar dan menghina pemilik MS Glow karena kasus yang menimpa pemilik MS Glow. Maka dari itu, batasan masalah pada penelitian ini fokus pada *review* produk perawatan kulit MS Glow. Data *tweet* yang fokus pada beberapa kata kunci seperti “MS Glow Whitening” dan “MS Glow Acne” dikumpulkan dalam data tersebut berupa *review* produk MS Glow.

Pemilihan media sosial *Twitter* sebagai sumber menimbulkan beberapa tantangan. Tantangan di *Twitter* antara lain *tweet* yang berisi kata-kata informal, menggunakan bahasa asing atau daerah, menghilangkan huruf vokal, mengulang huruf, dan mengubah posisi huruf. Misalnya, sebuah *Tweet* mungkin menyertakan kata "kusem", yang merupakan ulasan produk dan bukan kata formal. Kata “kusem” tersebut bermaksud mengacu pada artian yang sama dengan “kusam” sesuai dengan bentuk formal. *Tweet review* yang beragam tersebut dapat menentukan suatu produk tersebut bagus atau tidaknya produk tersebut.

Menurut (Ibrohim & Budi, 2018), ada beberapa kata informal yang biasa digunakan oleh orang Indonesia, yaitu kata-kata yang mengungkapkan emosi, pengulangan huruf untuk menekankan makna, penggunaan bahasa gaul, dan konversi vokal menjadi angka.

BERT adalah sebuah model bahasa *pretrained*, sebuah arsitektur, dan mampu dianggap sebagai metode *state of the art* dalam dunia *natural language processing*. BERT bekerja dalam konteks kalimat, bukan hanya berupa kata per kata. Dalam penggunaan metode BERT terdapat *Transformer* yakni, suatu mekanisme *attention* yang memahami hubungan kontekstual antara kata-kata dalam suatu teks. Didalam *Transformer* ada dua mekanisme yaitu *encoder* yang berguna untuk membaca teks masukan dan decoder sebagai penghasil urutan dari *output* yang berupa prediksi. Pada tahun 2018 para peneliti di Google AI *Language* mengembangkan model

representasi bahasa terlatih disebut metode *Bidirectional Encoder Representations from Transformers* atau disingkat BERT (Devlin, 2019).

Analisis sentimen dengan memakai pendekatan BERT memiliki kelebihan yaitu memudahkan komputer memahami bahasa selayaknya manusia. Sedangkan, kelemahan metode BERT dibatasi hanya untuk memahami dokumen singkat. Karena jika setiap teks yang panjang dibutuhkan waktu dalam mencocokkan pemahaman yang ekstra (Yansens, n.d.)

Pada penelitian (Mas et al., 2021) yang berjudul “Analisis Sentimen Customer Review Aplikasi Ruang Guru dengan Metode BERT (*Bidirectional Encoder Representations from Transformers*)” Penelitian ini mencakup 5437 data yang akan disiapkan untuk pelatihan dan pengujian. dengan perbandingan 70:30, di mana 70% data yang akan digunakan untuk pelatihan dan 30% untuk pengujian. Hasil Akurasi sebesar 99%, F1 score mencapai 98.9% , Presisi sebesar 64.13%, dan Recall sebesar 60.51% .

Penelitian dengan topik “Analisis Sentimen Terhadap Film Indonesia dengan Pendekatan BERT”. Dari komentar sebanyak 9.484 dan hasil *dataset* yang digunakan berjumlah 10652 kalimat. *Epoch* yang digunakan yaitu 10 *epoch* dikarenakan tingkat akurasi mencapai 74%. Dari 3 Percobaan menghasilkan akurasi 66% (Fimoza, 2021)

1.2 Rumusan Masalah

Karena banyaknya pengguna produk perawatan kulit MS Glow, *review* produk banyak dimuat di media sosial khususnya *Twitter*. Dengan memanfaatkan berbagai data *review tweet*, calon konsumen dapat menilai apakah suatu produk baik atau buruk. Sebagai media sosial, *Twitter* merupakan pilihan yang sulit karena *tweet* dapat memuat bahasa informal. Oleh karena itu, analisis sentimen *review* perawatan kulit regional dilakukan dengan memakai pendekatan BERT (*Bidirectional Encoder Representations from Transformers*). Metode BERT masih tergolong baru dan belum banyak digunakan sebagai metode penelitian. Analisis sentimen *review skincare* di *Twitter* menggunakan pendekatan BERT untuk mengklasifikasikan review positif, negatif, atau netral terhadap produk *skincare* MS Glow.

1.3 Batasan Masalah

1. *Dataset* memakai bahasa Indonesia
2. Data yang digunakan berasal dari *tweet review* pengguna produk Ms Glow.

3. Produk yang diteliti hanya untuk produk whitening dan acne.
4. Bahasa Pemrograman yang dipakai adalah bahasa pemrograman Python 3.6
5. Menggunakan analisis sentimen dengan menggunakan pendekatan BERT
6. Klasifikasi opini terbagi menjadi positif, negatif dan netral

1.4 Tujuan Penelitian

Penelitian ini memiliki tujuan untuk menghasilkan penilaian terhadap produk MS Glow dengan data *tweet* yang berisi ulasan pengguna dengan memakai BERT (*Bidirectional Encoder Representation from Transformers*).

1.5 Manfaat Penelitian

Manfaat dari penelitian ini ialah memperoleh informasi *review* produk MS Glow yang berasal dari opini pengguna *twitter* sehingga sentimen yang dihasilkan dapat menjadi suatu kritik, saran dan apresiasi suatu produk.

1.6 Metodologi Penelitian

Berikut merupakan metodologi penelitian yang akan dijalankan di penelitian ini:

1. Studi Pustaka

Dalam tahapan ini, diawali dengan mencari referensi dan sumber-sumber kredibel serta melakukan peninjauan pustaka melalui buku-buku, *e-book*, jurnal, makalah, artikel ilmiah, ataupun situs internet yang berhubungan dengan sentiment analysis, BERT, dan lainnya.

2. Identifikasi Masalah

Dalam tahap ini, penulis melakukan identifikasi masalah dalam analisis sentimen. Identifikasi masalah yang akan dilakukan pada *skincare* lokal dengan *brand* bernama MS Glow.

3. Analisis Sistem

Dalam tahap ini, pengumpulan data *tweet* dilakukan dengan memakai *scraping data tweet* dari *twitter*.

4. Perancangan dan Implementasi Sistem

Dalam tahap ini, perancangan dilakukan dengan menggunakan hasil analisis sistem dan implementasi sistem dilakukan menggunakan perancangan yang sudah dibuat. Implementasi dijalankan memakai bahasa pemrograman *Python*.

5. Pengujian Sistem

Dalam tahap ini, dilakukan pengujian terhadap sistem yang sudah dibuat dalam bentuk program dan memperoleh hasil yang sesuai.

6. Dokumentasi

Dalam tahap ini, dokumentasi serta penulisan laporan dilakukan, dimulai dari tahap analisa hingga pengujian berbentuk skripsi.

7. Sistematika Penyusunan

Berikut merupakan sistematika penulisan skripsi :

BAB I : Pendahuluan

Pada Bab satu, terdiri dari tahapan awal penelitian yaitu latar belakang, rumusan masalah, batasan masalah, tujuan dan manfaat penelitian, metodologi penelitian, serta sistematika penulisan.

BAB II : Landasan Teori

Pada Bab kedua, terdiri dari berbagai teori dan konsep yang berhubungan dengan penelitian ini yaitu Analisis Sentimen, *Skincare*, *Twitter*, *Deep Learning*, *BERT (Bidirectional Encoder Representations from Transformers)*.

BAB III : Analisis dan Perancangan

Pada Bab ketiga, berisi mengenai analisis sentimen terhadap *Review Skincare* Lokal pada *Twitter* Menggunakan Pendekatan BERT yakni sentimen dikategorikan ke dalam tiga kategori yaitu negatif, netral, serta positif. Analisis sistem yakni berupa arsitektur umum sistem dan perancangan sistem.

BAB IV : Implementasi dan Pengujian

Pada Bab keempat, berisi hasil penelitian yang sudah dijalankan dengan implementasi sistem yang telah dibangun sesuai dengan analisis masalah penelitian serta pembahasan pengujian sistem yang sudah dibuat.

BAB V: Kesimpulan dan Saran

Pada Bab kelima, berisi kesimpulan akhir dari penelitian yang sudah dijalankan, dan saran bagi penelitian mendatang.

BAB II

LANDASAN TEORI

2.1. Analisis Sentimen

Menurut (Mehta & Pandya, 2020), analisis sentimen digunakan untuk mengidentifikasi, mengenali, dan mengklasifikasikan perasaan dan opini pengguna terhadap suatu layanan, seperti film, isu produk, acara, atau fitur lainnya, sebagai positif, dijelaskan sebagai langkah-langkahnya. atau negatif, atau netral. Analisis sentimen (juga dikenal sebagai penambangan opini) adalah proses pemrosesan, pemahaman, dan penggalian data secara otomatis untuk memperoleh informasi emosional dalam teks yang mengungkapkan opini positif atau negatif (Sudiantoro & Zuliarso, 2018). Analisis sentimen bertujuan untuk mengevaluasi perasaan, sikap, pendapat, dan evaluasi yang disampaikan oleh masyarakat dan para ahli melalui berbagai media mengenai produk, merek, layanan, kebijakan, dan agensi (Cahyani, 2020).

Natural Language Processing (NLP) adalah cabang ilmu komputer dan linguistik yang mempelajari interaksi antara komputer dan bahasa manusia (bahasa alami). Penelitian dalam NLP meliputi berbagai aspek, termasuk segmentasi suara, segmentasi tuturan (*speech segmentation*), segmentasi teks (*text segmentation*), penandaan kelas kata (*partofspeech*), dan disambiguasi makna kata (*word sense disambiguation*) (Hoppe et al., 2018).

Karena dampak dan manfaat analisis sentimen yang sangat besar, penelitian dan aplikasi di bidang analisis sentimen berkembang pesat, seperti mengidentifikasi orang-orang yang mengemukakan pendapat yang memancing banyak reaksi dari pengguna *Twitter* (Rahman, 2020). Analisis sentimen bermanfaat untuk mengungkapkan opini publik pada sebuah topik (Kurniawan & Apriliani, 2020). Selain itu, analisis sentimen merupakan metode pengumpulan opini masyarakat di jejaring sosial, termasuk layanan publik dan topik terkini (Syarifuddin, 2020).

Dari beberapa pendapat sebelumnya, dapat disimpulkan bahwa analisis sentimen sebagai berfungsi sebagai alat pemrosesan yang mampu mengidentifikasi suatu opini, pendapat, dan emosi dari sebuah teks dengan mengklasifikasikan ke dalam kategori negatif, positif, atau netral.

2.2 Skincare

Istilah *Skincare* berasal dari bahasa Inggris dan mengacu pada berbagai tahapan Perawatan kulit atau kegiatan yang bertujuan untuk menjaga kesehatan kulit. Untuk sebagian orang, terutama kaum hawa jenis kegiatan perawatan kulit untuk mempercantik penampilan. *Skincare* bisa dilakukan dengan berbagai cara, termasuk menggunakan produk yang mengandung bahan-bahan yang disesuaikan dengan kondisi kulit Anda. Bagi sebagian orang, produk *skincare* sudah menjadi barang rutin dalam kehidupan sehari-hari. Meski tidak terbatas pada wanita, kini banyak produk perawatan kulit yang menyasar pria karena pria dan wanita memiliki kebutuhan dan dosis yang berbeda pada setiap produknya (Shinta Dewi Kusumaningrum, 2021).

Masyarakat Indonesia, khususnya remaja putri, sudah sangat bergantung pada layanan dan produk perawatan kulit yang menjanjikan wajah pelanggannya menjadi lebih cerah, bersih, dan menawan. Salah satu cara remaja tampil menarik adalah dengan menggunakan produk kecantikan (Azmi, 2019).

Produk *skincare* Indonesia telah mengalami perkembangan yang pesat. Banyaknya brand Indonesia salah satunya yaitu MS Glow. MS Glow adalah salah satu merek lokal yang mempunyai beberapa produk yaitu diantaranya *skincare*, *bodycare*, serta kosmetik. Maharani Kemala dan Shandy Purnamasari mendirikan MS Glow pada tahun 2013. Salah satu fokus produk *skincare* yang dimilikinya yaitu pada bidang *whitening* dan *acne*.

2.3 Twitter

Twitter adalah platform *Mikroblog* yang dimanfaatkan penggunaanya untuk melakukan pembaharuan status atau kiriman singkat secara regular. *Twitter* menjadi salah satu layanan mikroblog yang dapat dipakai untuk melakukan pembaharuan dalam bentuk teks dengan batasan maksimal 140 karakter. Pembaruan ini biasa disebut dengan tweets (cuitan). *Twitter* sendiri didirikan oleh Jack Dorsey pada Maret 2006 di San Fransisco. *Twitter* mulai dikembangkan sebagai proyek pengembangan dan penelitian di perusahaan Odeo.

Berdasarkan data dari Kementrian dan Informatika (Kominfo, 2023), Saat ini Indonesia memiliki pengguna internet sebanyak 215,63 juta pengguna, dimana 25,5 juta diantaranya adalah pengguna *Twitter*. Hal ini menjadikan Indonesia sebagai pengguna *Twitter* terbesar keempat di dunia yang menggunakan *Twitter* karena fitur-fitur yang ditawarkan sederhana dan mudah digunakan. Pengguna dapat memperoleh

akun hanya dengan mendaftar menggunakan email atau nomor telepon mereka. *Twitter* memiliki beberapa fitur seperti *Tweet*. *Tweet* adalah pesan yang diunggah ke *Twitter*, termasuk teks, foto, video, dan GIF. Dalam pemanfaatannya, *tweet* berfungsi sebagai wadah opini, kritik, saran, bahkan sebagai sumber berita (informasi). *Tweet* dapat diunggah secara bebas, sehingga tersedia data dalam jumlah yang banyak. Data *Tweet* tersebut mampu untuk menghasilkan informasi yang bermanfaat bagi individu maupun bisnis.

2.4 Pre-processing Dataset

Pre-processing dataset merupakan tahap di mana format data diubah agar lebih terstruktur sejalan dengan kebutuhan proses data *mining*. Proses ini umumnya mengubah data menjadi representasi numerik (Ashari *et al.*, 2020). Proses ini adalah tahapan awal untuk mengetahui pesan hoax sebelum dipakai dalam model. Proses ini dikelompokkan ke dalam berbagai tahapan. Supaya dataset yang terkumpul bisa dipakai oleh algoritma yang dipilih. Tahapan *Pre-processing* dataset yakni:

2.4.1 Case Folding

Case Folding adalah proses penyamaan case dalam sebuah dokumen. Hal ini dilakukan untuk mempermudah pencarian. Tidak semua dokumen teks konsisten dalam penggunaan huruf kapital. Oleh karena itu peran case folding dibutuhkan dalam mengkonversi keseluruhan teks dalam dokumen menjadi suatu bentuk standar (dalam hal ini huruf kecil atau *lowercase*). (Putranti & Winarko, 2014) dalam penelitian (Rahman, 2020)

2.4.2 Data Cleaning

Data cleaning adalah kegiatan menganalisa kualitas data dengan cara memodifikasi, mengubah, atau menghapus data-data yang dianggap tidak perlu, tidak lengkap, data tidak akurat, atau memiliki format data atau file yang salah dalam basis data guna menghasilkan data berkualitas tinggi. Data cleansing juga biasa disebut data cleaning atau data scrubbing (Alita et al., 2020)

2.4.3 Tokenisasi

Tokenisasi atau *tokenizing* menggambarkan proses pembagian teks menjadi kata-kata dengan memakai spasi memiliki tujuan agar setiap kata bisa berdiri sendiri tanpa terikat kata lain sehingga hanya ada kata dasar (Nata dan Yudiastra, 2017).

2.4.4 Stopwords Removal

Stopword removal melibatkan penghapusan beberapa kata dalam sebuah kalimat yang tidak lagi memiliki arti terkait, seperti "ini" atau "itu". Kata-kata yang biasa muncul namun berdampak rendah tidak disertakan. Penghapusan ini membantu meningkatkan kinerja disambiguasi (Indraloka dan Santosa, 2017).

2.4.5 Negation Handling

Menurut Dian Noviani Syafar (2016) dalam Bahasa Indonesia terdapat empat kata negasi yang lazim digunakan yaitu tak atau tidak, bukan, jangan, dan belum. Penggunaan kata negasi, dapat digunakan pada jenis kalimat deklaratif, interogatif, dan ekslamatif.

2.4.6 Stemming

Stemming merupakan proses mengubah sebuah kata ke bentuk dasarnya dengan menghilangkan imbuhan yang terdapat dalam kata tersebut seperti “di-“, “-nya”, dan lainnya. *Stemming* bertujuan untuk menghemat waktu saat melakukan proses klasifikasi (Violos, dkk. 2018).

2.4.7 Normalisasi

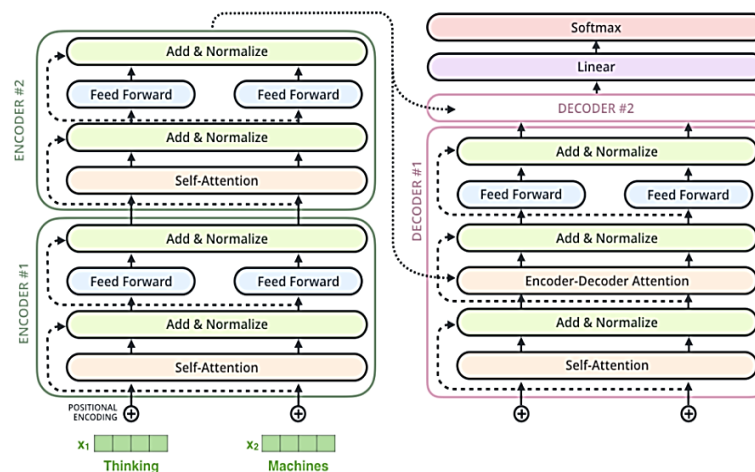
Normalisasi adalah sebuah proses merubah kata-kata non baku menjadi bentuk baku. Tahap ini didukung oleh penggunaan kamus yang dibuat oleh (Aliyah Salsabila et al., 2018) yang berisi kumpulan pasangan kata tidak baku beserta bentuk bakunya. Jika dalam data ada kata-kata yang tidak baku, maka diganti dengan kata-kata baku yang terdapat dalam kamus.

2.5 Deep Learning

Deep Learning merupakan bagian *Machine Learning* yang terdiri dari algoritma pemodelan abstraksi pada tingkat tinggi. Pembelajaran mendalam dapat diterapkan melalui *supervised learning*, *unsupervised learning* serta *semi-supervised learning* (Cholissodin et al., 2020). *Deep Learning* adalah bagian dari *machine learning* yang memakai jaringan saraf tiruan dalam mengimplementasikan masalah dengan data dalam jumlah besar. Karena manfaat yang ditawarkannya, *deep learning* semakin banyak dipakai dalam komunitas riset dan industri dalam memecahkan banyak permasalahan data besar seperti *Computer vision*, *Speech recognition*, serta *Natural Language Processing*.

2.6 Metode BERT

BERT merupakan model *Deep Learning* yang dipakai untuk representasi konteks kata pada *Natural Language Processing* (NLP). Google mengembangkan model ini dan dirilis di tahun 2018. Selama proses pelatihan, kata-kata diadaptasi memakai *Masked Language Model* (MLM) dan transformator dua arah (*bidirectional*). Seperti namanya, BERT berfokus terhadap proses pengkodean dan pembuatan model bahasa. Model BERT terdiri atas struktur *transencoder* dua arah multilayer. BERT hanya memakai tumpukan *encoder transformator* dan bukan tumpukan *decoder* (Rothman, D. 2021)



Gambar 2. 1 Arsitektur Encoder dan *Decorder* pada Transformer

Sumber: (Vaswani *et al.*, 2017)

Pendekatan BERT menggunakan transformator, suatu mekanisme yang mempelajari hubungan kontekstual antar kata dalam teks (Vaswani *et al.*, 2017). Pada gambar 2.1 terdapat Transformer yang mencakup dua mekanisme, yaitu:

a. Encoder

Encoder dibentuk oleh sejumlah tumpukan (*stack*) yang berjumlah $N = 6$ layers yang identik. Fungsi dari *encoder* adalah untuk membaca sekaligus seluruh input teks.

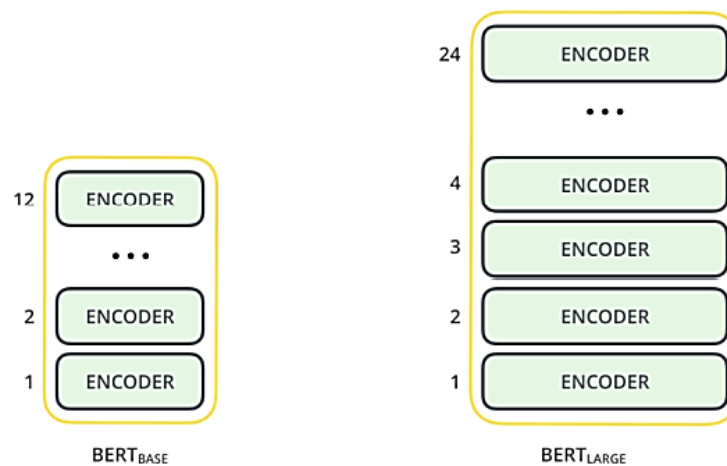
b. Decoder

Fungsi dari *decorder* yaitu sebagai penyusun urutan dari *output* yang berbentuk prediksi. *Decoder* juga dilengkapi dengan tumpukan (*stack*) yang terdiri dari $N = 6$ lapisan yang seragam.

2.7 Arsitektur BERT

Tujuan BERT adalah untuk membuat model bahasa yang menganggap mekanisme encoder penting. Dalam arsitektur model BERT, yaitu transformator dua arah *multilayer*, pemrosesan ini hanya dilakukan hingga *encoder*. Seperti terlihat pada Gambar 2.2, *encoder* Transformer membaca seluruh urutan kata sekaligus, sedangkan *encoder directional* hanya membaca teks masukan secara berurutan (kiri ke kanan atau kanan ke kiri). Hal ini memungkinkan BERT mengaktifkan model untuk mempelajari konteks kata berdasarkan lingkungan. BERT memiliki dua ukuran yaitu $BERT_{BASE}$ dan $BERT_{LARGE}$.

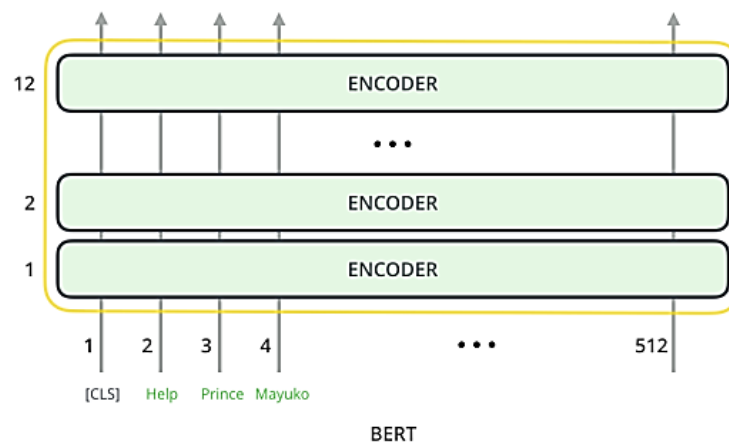
Arsitektur model BERT dibentuk dari *multilayer bi-directional* Transformer *encoder*. ada 2 bentuk model :



Gambar 2.2 Perbedaan Ukuran $BERT_{BASE}$ dan $BERT_{LARGE}$

Sumber: (Devlin et al., 2019)

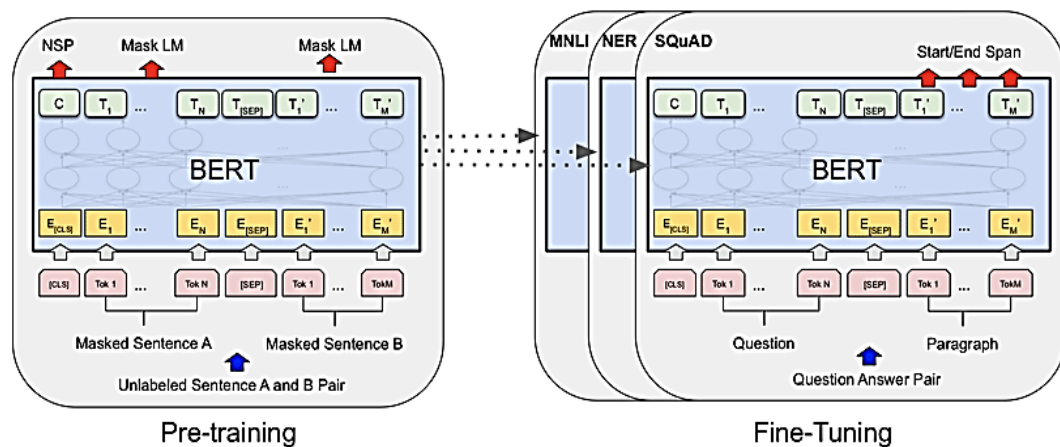
1. $BERT_{BASE}$: model ini dibangun dari 12 Transformer *block*, 12 *Attention layer* serta 768 *hidden layer*.
2. $BERT_{LARGE}$: model ini mempunyai *layer* dan *attention layer* yang lebih banyak dibandingkan $BERT_{BASE}$ berfungsi untuk menghasilkan hasil yang lebih baik yaitu 24 Transformer *block*, 16 *attention head* serta 1024 *hidden layer*.



Gambar 2.3 Arsitektur BERT

Sumber: (Devlin et al., 2019)

BERT memakai *WordPiece embeddings* dengan kosakata 30.000 token. Token pertama dari masing-masing urutan diberi token *spesial class classification* ([CLS]). Token *separate* ([SEP]) ditempatkan di akhir setiap kalimat untuk membedakannya. Selanjutnya, tambahkan *positional embedding* pada kalimat untuk menandai posisinya. Input ke encoder BERT merupakan jumlah dari Token *Embedding*, *Segment embedding*, serta *Positional Embedding*. BERT dapat dilatih untuk memahami bahasa dan juga dapat disesuaikan dalam mempelajari tugas tertentu. Pelatihan BERT terbagi dalam dua fase: *pra-training* serta *tuning*. Fase pertama, atau *pra-training* merupakan fase di mana BERT diciptakan untuk memahami serta mempelajari bahasa ataupun konteksnya. BERT bisa dipahami melalui *training*. Dua tugas tanpa *unsupervised* secara bersamaan dilakukan: *Masked Language Model* dan *Next Sentence Prediction*.



Gambar 2.4 Proses *Pre-Training* dan *Fine-tuning*

Sumber: (Devlin et al., 2019)

Training BERT memiliki dua fase: *pre-training* dan *fine-tuning*. Hal ini dapat dilihat pada 3.5. Tujuan *Pre-training* adalah untuk memahami dan mempelajari bahasa dan konteksnya. Dalam tahap *fine-tuning*, model BERT dilengkapi dengan parameter terlatih dan sudah mempunyai data. BERT memiliki dua tugas *unsupervised* dalam proses *pre-training*, yaitu :

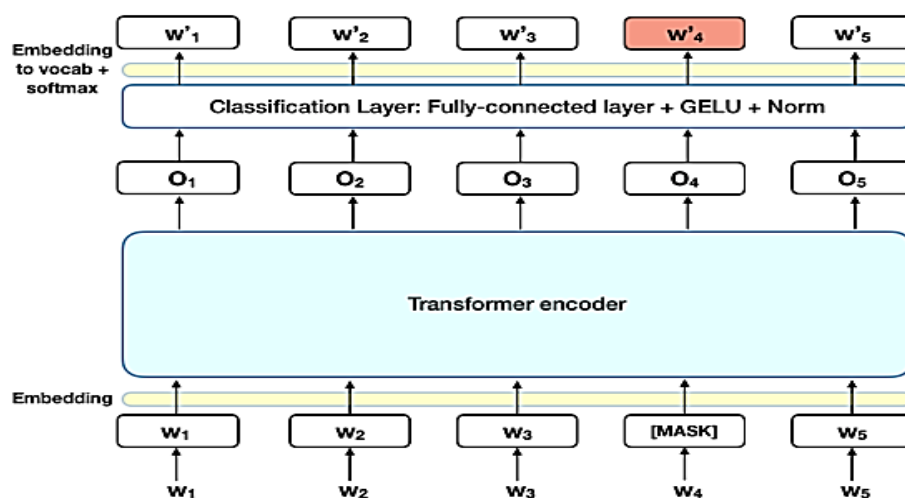
1. *Pre-Training*

Pre-training BERT bisa dipakai dalam menyelesaikan tugas *unsupervised* seperti *Masked model language* serta *next sentence prediction*.

1.1.1 *Masked Language Modelling* (Masked LM)

Masked Language Modelling bertujuan untuk menutupi kata atau mask secara acak dalam sebuah kalimat dengan probabilitas rendah. Dalam model BERT, kata-kata yang diterima dimodifikasi menggunakan token [MASK]. Harus mengubah 15% jumlah kata di setiap urutan. Model yang ditentukan oleh [MASK] kemudian memprediksi nilai asli dengan menyesuaikan konteks kata lain dalam urutan tersebut. Kata output yang diprediksi adalah :

- Tambahkan lapisan klasifikasi di atas output *encoder*.
- Kalikan vektor output dengan matriks *embedding* dan ubah menjadi *vocabulary dimension*.
- Hitung probabilitas masing-masing kata dalam *vocabulary* menggunakan *softmax*

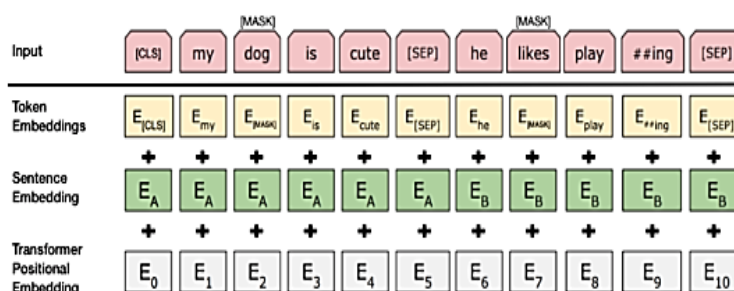


Gambar 2.5 Proses *Masked Language Modelling* (Masked LM)

Sumber: (Devlin *et al.*, 2019)

1.1.2 Next Sentence Prediction

BERT melakukan proses *pre-training* untuk melatih model yang bisa memahami hubungan kalimat dan memprediksi kalimat berikutnya. Model dapat menerima pasangan kalimat masukan. Pada proses *training*, 50% masukannya berupa pasangan kalimat, dimana kalimat kedua menjadi kalimat berikutnya. Sisanya 50% merupakan kalimat yang secara acak diambil dari korpus dan membentuk kalimat kedua.



Gambar 2.6 Representasi Input pada BERT

Sumber: (Devlin *et al.*, 2019)

Ada tiga *embedding layers*, antara lain:

1. Lapisan pertama (*Token embeddings*), merupakan token klasifikasi khusus (CLS). Di layer ini, token memasuki layer ini yang merupakan representasi vektor dari setiap token.
2. Lapisan kedua (*Sentence embedding*), adalah lapisan yang dirancang untuk membedakan kalimat jika terdapat tiga kalimat atau lebih. Representasi untuk lapisan ini adalah : A adalah tanda apakah kalimat pertama disertakan dan B adalah tanda apakah kalimat kedua disertakan, dipakai untuk membedakan kalimat bila tersedia lebih dari dua kalimat.
3. Lapisan ketiga (*Positional embedding*), adalah lapisan yang ditambahkan pada masing-masing token dalam menyimpan informasi posisi kata sesuai dengan urutan. Implementasi dan konsep *positional embedding* ditunjukkan di Transformer.

d. Fine-tuning

Saat *fine-tuning*, fase pelatihan terjadi dan perubahan kecil dilakukan pada model BERT. Dalam *fine-tuning*, Transformer memiliki mekanisme *self-attention* yang dapat membuat model dengan berbagai tugas, termasuk tugas satu kalimat dan berpasangan kalimat.

Model BERT yang telah mengalami tahap *pre-training* dapat diterapkan untuk menangani berbagai tugas NLP lainnya dengan menambahkan lapisan yang sesuai dengan kebutuhan tugas yang hendak diselesaikan. BERT bisa dipakai untuk melakukan klasifikasi teks, *question-answering*, *sentiment analysis*, *named entity recognition* dan tugas *natural language* lainnya.

2.1.1 Transfer Learning

Transfer learning merupakan studi tentang *machine learning* yang berfokus terhadap pelestarian pengetahuan yang didapatkan melalui data pelatihan. Pengetahuan ini dipakai untuk memecahkan permasalahan tertentu dan juga bisa dipakai dalam memecahkan masalah lain yang terkait. Sebagai contoh, ilmu yang diperoleh dari belajar mengenal mobil bisa dipakai untuk mengenal truk. Umumnya *transfer learning* dipakai untuk beberapa tugas yang data pelatihannya sedikit (Devlin et al., 2019).

2.1.2 Confusion Matrix

Confusion Matrix merupakan metode pengujian dalam menghitung keakuratan sebuah model dan menetapkan apakah model tersebut benar atau salah. *Confusion Matrix* memiliki rumus untuk menghitung presisi, presisi, serta *recall*. Nilai akurasi merupakan sejauh mana keakuratan prediksi data terhadap seluruh data. Presisi merupakan tingkat keakuratan antara informasi yang diminta dengan respon yang sistem berikan. *Recall* merupakan tingkat keberhasilan sistem dalam mengambil informasi (A, Tharwat. 2018).

$$\text{Akurasi} = \frac{TP+TN}{TP+TN+FP+FN} \dots\dots\dots(1)$$

$$\text{Presisi} = \frac{TP}{TP+FP} \dots\dots\dots(2)$$

$$\text{Recall} = \frac{TP}{TP+FN} \dots\dots\dots(3)$$

Sesudah klasifikasi bisa diukur kebenarannya, kemudian dilakukan perhitungan nilai kombinasi untuk mendapatkan nilai pengukuran, yang disebut sebagai *F1-score*. Rumus berikut digunakan untuk menghitung *F1-score*:

$$\text{F1-Score} = \frac{2 \times \text{Presisi} \times \text{Recall}}{\text{Presisi} + \text{Recall}} \dots\dots\dots(4)$$

2.7 Penelitian yang Relevan

Berikut merupakan penelitian terdahulu yang relevan dengan penelitian penulis:

1. Berdasarkan penelitian terdahulu yang dijalankan oleh (Adhine Salsabila et al., 2019) yang berjudul “Analisis Sentimen Terhadap Produk The Body Shop Tea Tree Oil “ (2019), menggunakan metode Naïve baiyes. Data yang digunakan yaitu data *training* dan *testing* sebanyak 1974 (70 %) dan 846 (30%). Hasil yaitu komentar positif lebih tinggi berjumlah 454 positif dan 391 komentar negatif.
2. Berdasarkan penelitian yang dilakukan oleh Raden Mas Rizqi Wahyu Panca Kusuma Atmaja dan Wiyli Yustanti dengan judul “Analisis Sentimen Customer Review Aplikasi Ruang Guru dengan Metode BERT (*Bidirectional Encoder Representations from Transformers*)” (2021). Terdapat 5437 data yang akan dilatih serta diuji memakai proporsi 70:30 dimana 70% data akan dilatih serta 30% data yang akan diuji. Hasil Akurasi bernilai 99%, F1 score :98.9% , Presisi :64.13%, dan Recall :60.51% .

3. Berdasarkan penelitian yang dijalankan oleh (Fimoza, Dwi., 2021) dengan judul “Sentiment Analysis for Movie Review in Bahasa Indonesia Using BERT”. Dari komentar sebanyak 9.484 dan hasil *dataset* yang digunakan berjumlah 10652 kalimat. *Epoch* yang digunakan yaitu 10 *epoch* dikarenakan tingkat akurasi mencapai 74%. Dari 3 Percobaan menghasilkan akurasi 66%, 68% dan 66 secara berturut. Nilai yang diperoleh tidak lebih tinggi dibandingkan dengan dengan penggunaan metode *Multinomial Naïve bayes* yaitu 72%.
4. Berdasarkan penelitian yang dijalankan oleh (Cindy Alifia Putri *et al.*, 2020) dalam penelitiannya yang berjudul “Analisis Sentimen Review Film Berbahasa Inggris Dengan Pendekatan *Bidirectional Encoder Representations from Transformers*”. Pada penelitian ini memaki *dataset* cornelldu berbahasa Inggris. Hasil dari penelitian ini memiliki hasil akurasi 73%. Dalam penelitian ini, sentimen dikelompokkan ke dalam sentimen positif dan negatif saja.
5. Berdasarkan penelitian yang dilakukan oleh (Kusnadi *et al.*, 2021) dengan judul “Analisis Sentimen Terhadap Game Genshin Impact Menggunakan Bert”. Hasil dari penelitian ini memiliki akurasi diatas 74%.
6. Berdasarkan penelitian yang dilakukan oleh (Cai *et al.*, 2020) yang berjudul “Sentimen Analysis About Investors and Consumers in Energy Market Based on BERT-BiLSTM”. Penelitian ini menggunakan kombinasi metode BERT dan BERT-BiLSTM untuk memprediksi orientasi sentimen investor dan konsumen dengan dengan menggunakan *dataset* berasal dari Chinese Internet . Hasil Penelitian memiliki akurasi diatas 80% .
7. Berdasarkan penelitian yang dijalnan oleh (Rahate *et al.*, 2022) dengan judul “Review Sentimental Analysis”. Penelitian ini menggunakan data lebih dari 50 judul film dengan masing-masing 10 ulasan setiap judul. Polaritas ulasan dikategorikan dalam positif atau negatif.
8. Berdasarkan penelitian yang dijalankan oleh (Hoang *et al.*, 2019) yang berjudul “Aspect-Based Sentiment Analysis using BERT”. Penelitian ini menggunakan metode BERT menghasilkan model gabungan klasifikasi aspek dan sentimen memiliki akurasi diatas 78% .
9. Berdasarkan penelitian yang dilakukan oleh (Dankhara, 2022) yang berjudul “Sentiment Analysis of Tweets”. Pada penelitian ini *dataset* yang berasal dari *twitter* diambil dan menggunakan beberapa metode klasifikasi dan memberikan

hasil yang cukup baik. Metode BERT cenderung lebih bekerja lebih baik dalam klasifikasi sentimen.

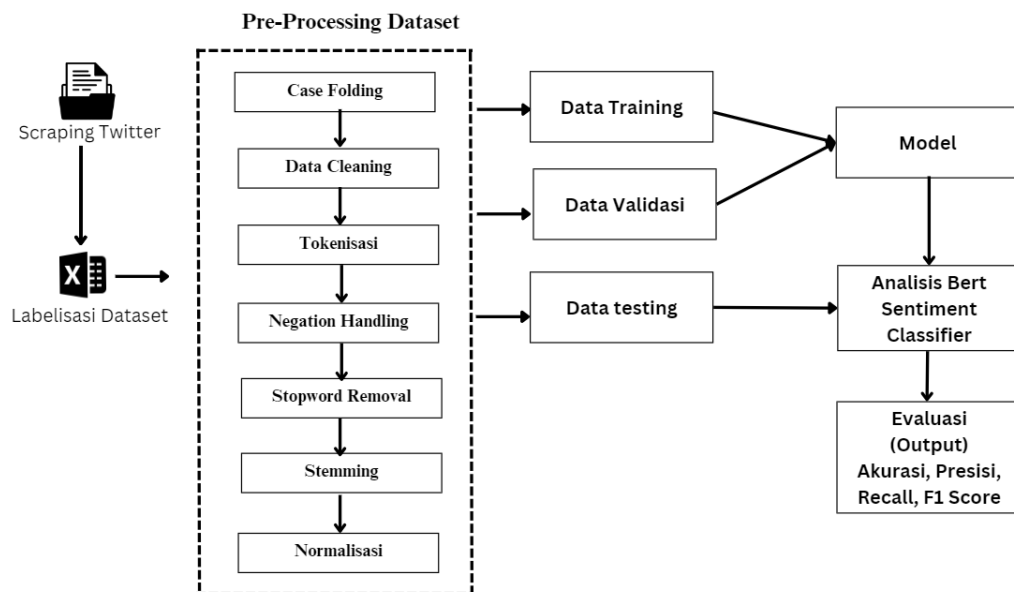
10. Berdasarkan penelitian yang dijalankan oleh (Wardani et al., 2022) yang berjudul *“Aspect Based Sentiment Analysis on Beauty Product Review Using Random Forest Method and Chi-Square”*. Penelitian ini dilakukan dengan menganalisis review produk kecantikan berdasarkan aspek. Hasil dari penelitian ini mendapatkan akurasi 81,75%.

BAB III

ANALISIS DAN PERANCANGAN

3.1 Arsitektur Umum

Arsitektur umum yang disertakan dalam penelitian ini dimulai dengan mengambil *scraping* media sosial Twitter sebagai *dataset*. *Dataset* yang diperoleh dari *scraping* diberi label positif, negatif, dan netral. Selanjutnya *dataset* memasuki tahap *preprocessing*. Tahap *preprocessing* meliputi *case folding*, *data cleaning*, *tokenisasi*, *negation handling*, *stop removal*, *stemming* serta normalisasi bahasa tidak baku. *Dataset* yang diperoleh pada tahap *preprocessing* kemudian ataupun menggunakan BERT dan diklasifikasikan menjadi tiga kategori yaitu positif, negatif, dan normal. Hasil data yang diklasifikasikan kemudian dievaluasi. Arsitektur umum analisis sentimen ditunjukkan pada gambar berikut:



Gambar 3. 1 Arsitektur Umum

3.1.1 Scraping Twitter

Scraping Twitter adalah proses pengambilan data dari *Twitter*. Dalam penelitian ini *scraping* dilaksanakan dengan menggunakan *tweet harvest* pada *Google Collab*. Data *tweet* yang diambil berdasarkan kata kunci dan hashtag *review* produk MS Glow seperti “*Whitening MS Glow*”.

3.1.2 Labelisasi *Dataset*

Labelisasi *dataset* dilakukan bertujuan untuk menentukan komentar memiliki sentimen negatif, positif, dan netral. *Dataset* negatif diberi nilai 0, positif diberi nilai 2, dan netral diberi nilai 1. Pelabelan *dataset* dilakukan secara manual. Berikut merupakan contoh labelisasi *dataset*:

Tabel 1. Contoh *Dataset*

Komentar	Sentimen
Cocok cocokan sih, tp klo aku personally ms glow itu skincare pertama aku, life saver bgtt. Bikin cerah glowing.	2
aku barengin sama toner dan krim malam ms glow yg luminous	1
Mengsedih gara” muka ga cocok sama ms glow	0

3.1.3 *Pre-processing Dataset*

Merupakan teknik data mining untuk mentransformasi data mentah ke dalam format yang lebih mudah untuk dimengerti. Tahap *Pre-processing* dalam penelitian ini meliputi:

3.1.3.1 Case Folding

Case folding merupakan proses yang dipakai untuk mengubah seluruh huruf di sebuah kalimat atau dokumen ke huruf kecil. Berikut adalah tabel *pseudocode* untuk case folding pada tabel 2 dan hasil contoh perbandingan *case folding* pada tabel 3.

Tabel 2. *Pseudocode Case Folding*

<i>Pseudocode Case Folding</i>
<pre># Fungsi case folding def casefolding(review): review_casefolded = review.lower() return review_casefolded # Input teks input_text = "Cantik" # Memanggil fungsi casefolding output_text = casefolding(input_text)</pre>

```
# Menampilkan hasil case folding
print(output_text)
```

Tabel 3. Perbandingan Hasil *Case Folding*

Komentar	Hasil <i>Case Folding</i>
EH SUMPAH HONEST REVIEW MS GLOW BAGUS BANGET GUE SMPE SHOCKKK. cm takut kata rangorang stlh bbrapa lama bakal breakout gtu but kita buktiin aja kali ya. soalnya gue bebal akwoakwka. lagian muka gue tergolong muka badak krna ga pernah breakout grgr skincare manapun sih	eh sumpah honest review ms glow bagus banget gue smpe shockkk. cm takut kata rangorang stlh bbrapa lama bakal breakout gtu but kita buktiin aja kali ya. soalnya gue bebal akwoakwka. lagian muka gue tergolong muka badak krna ga pernah breakout grgr skincare manapun sih

3.1.3.2 Data cleaning

Data Cleaning merupakan proses ketika *dataset* dibersihkan dengan menghapus karakter, kata berulang, *hashtag*, *symbol*, angka, tanda baca, username, dan link yang dapat mempengaruhi hasil analisis. Berikut adalah tabel *pseudocode* untuk data cleaning dan hasil contoh perbandingan *data cleaning*.

Tabel 4. *Pseudocode Data Cleaning*

<i>Pseudocode Data Cleaning</i>
<pre># Fungsi filtering Fungsi filtering(review): # Input: review (teks yang akan difilter) # Output: review_filtered (teks yang telah difilter) # Menghapus web links review_filtered = regex_substitusi(review, r'http\S+', '') # Menghapus @username review_filtered = regex_substitusi(review_filtered, '@[^\s]+', '') # Menghapus #tagger</pre>

```

review_filtered = regex_substitusi(review_filtered, r'#([^\s]+)', '')
# Menghapus angka, termasuk angka dalam string
review_filtered = regex_substitusi(review_filtered, r'[0-9]+', '')
# Menghapus karakter non-ASCII
review_filtered = regex_substitusi(review_filtered, r'^\x00-\x7f', '')
# Menghapus \uXXXX
review_filtered = regex_substitusi(review_filtered, r'(\u[0-9A-Fa-f]+)', '')
# Menghapus karakter selain huruf, angka, tanda baca tertentu
review_filtered = regex_substitusi(review_filtered, r"^[A-Za-z0-9^,!\./'+-="], " ")
# Menghapus \u\w\w\w\w
review_filtered = regex_substitusi(review_filtered, r'\u\w\w\w\w', '')
# Menghapus simbol, karakter aneh, dan karakter aneh
review_filtered = regex_substitusi(review_filtered, r"[!-
'\"$%&'\(\)\*|+,\.\/\:\;\<|=\>|?[\[\]\^\_\`{\|\}]\~|\d\Y~£\©|-|¤|¶|»|²|~]", " ")
Kembalikan review_filtered

```

Tabel 5. Perbandingan Hasil *Data Cleaning*

Komentar	Hasil <i>Data Cleaning</i>
Review jujur MS Glow Acne. Emang bener bikin putih, beneneer banget. Tapi untuk jerawat... Masih butuh kesabaran ekstra. Suka hilang, suka nimbul juga nih acne.	review jujur ms glow acne. emang bener bikin putih bener banget tapi untuk jerawat Masih butuh kesabaran ekstra suka hilang suka nimbul juga nih acne.

3.1.2.3 Tokenisasi

Tokenisasi merupakan proses yang dilakukan untuk memisahkan suatu rangkaian karakter berdasarkan spasi, tanda baca, dan ekspresi bermakna lainnya. Berikut merupakan pseudocode pada tabel 6 dan contoh hasil perbandingan tokenisasi pada tabel 7.

Tabel 6. Tabel *Pseudocode Tokenisasi*

<i>Pseudocode Tokenisasi</i>

```
# Fungsi untuk tokenisasi
Fungsi tokenize(review):

# Input: review (teks yang akan di-tokenisasi)
# Output: tokens (daftar token)
# Menggunakan NLTK untuk tokenisasi
tokens = nltk.word_tokenize(review)

# Kembalikan daftar token
Kembalikan tokens
```

Tabel 7. Perbandingan Hasil *Tokenisasi*

Komentar	Hasil Tokenisasi
percaya gak percaya aku pake facewash msglow serum scarlett, sumpah muka aku jadi lebih putih dari sebelumnya	'percaya' 'gak' 'percaya' 'aku' 'pake' 'facewash' 'msglow' 'serum' 'scarlett' 'sumpah' 'muka' 'aku' 'jadi' 'lebih' 'putih' 'dari' 'sebelumnya'

3.1.2.4 Negation Handling

Negasi dalam bahasa Indonesia dapat meniadakan seluruh kalimat atau hanya sebagian kalimat saja. Dalam beberapa kasus, kata-kata yang mengandung angka tak tentu (misalnya “sedikit”) tidak membalikkan nilai makna secara keseluruhan. Misalnya, frasa yang berisi penanda negasi konvensional "tidak nakal" membalikkan nilai sentimen keseluruhan dari kata "nakal", sedangkan frasa yang berisi numeral tak tentu "sedikit nakal" membalikkan nilai sentimen keseluruhan dari kata "nakal". tidak membalikkan, melainkan mengurangi atau menurunkan nilai sentimen secara keseluruhan nilai sentimennya. Berikut adalah contoh *negasi handling*, ditunjukkan pada Tabel 9 di bawah ini.

Tabel 8. *Pseudocode Negation Handling*

<i>Pseudocode Negation Handling</i>
<pre># Fungsi untuk mengatasi negasi Fungsi negation(review, negation_words=['tak', 'tidak', 'bukan', 'jangan', 'belum']): # Input: review (teks yang akan diatasi negasi), negation_words (kata-kata negasi)</pre>

```

# Output: review yang telah diatasi negasi
words = split(review)

# Inisialisasi daftar untuk menyimpan kata-kata yang telah diatasi negasi
negated_words = []

# Iterasi melalui setiap kata dalam review
for word in words:
    if any(neg_word in toLowerCase(word) for neg_word in negation_words):
        word = f"{word}_"
    negated_words.append(word)
review

Kembalikan gabungan(negated_words, ' ')

```

Tabel 9. Perbandingan Hasil *Negation Handling*

Komentar	Hasil <i>Negation Handling</i>
ms glow yg acne, jejak breakout sudah tidak bersisa bekasnya	ms glow yg acne, jejak breakout sudah tidak_ bersisa bekasnya

3.1.2.5 *Stopwords removal*

Stopwords removal merupakan proses yang dilakukan untuk menghapus berbagai kata yang tidak memiliki arti. Pada penelitian ini memakai *stopwords* pada kamus *Tala*. Berikut contoh kamus *Tala* pada tabel 10.

Tabel 10. *Pseudocode Stopwords removal*

<i>Pseudocode Stopwords removal</i>
<pre> # Fungsi untuk menghapus stopword Fungsi removeStopword(review): # Input: review (teks yang akan dihapus stopword) # Output: review yang telah dihapus stopword # Kata-kata tambahan yang ingin dihapus additional_stop = ['wts', 'deh', 'dia'] # Ambil daftar stopword dari stopwords tala </pre>


```

stopwords = load_stopwords(stopword_path)

# Gabungkan stopword dengan kata-kata tambahan (additional)
stopwords += additional_stop

# Inisialisasi list untuk menyimpan hasil filter
content = []

# Filter kata-kata stopwords dari review
filteredtext = [word for word in review.split() if word.lower() not in stopwords]

# Gabungkan kata-kata yang telah difilter kembali menjadi review
content.append(" ".join(filteredtext))

# Ambil elemen pertama dari list content
review = content[0]

# Kembalikan review yang telah dihapus stopwords
Kembalikan review

```

Tabel 11. Contoh *Stopword* Kamus *Tala*

Stopwords Tala			
ada	aku	Bagai	Beberapa
cara	cukup	Dan	Dapat
dia	disebutkan	Entah	Enggak
hari	harus	Ingat	Jauh
jelas	juga	Kalua	Kira

3.1.2.6 *Stemming*

Stemming merupakan tahapan yang dilakukan untuk mengubah kata menjadi kata dasar dengan menghilangkan imbuhan kata seperti prefix, sufiks, serta konfiks. Berikut contoh kamus hasil perbandingan *stemming* pada tabel 13.

Tabel 12. *Pseudocode Stemming*

<i>Pseudocode Stemming</i>
<pre> # Fungsi untuk stemming Fungsi stemText(review): # Input: review (teks yang akan di-stemming) # Output: review yang telah di-stemming # Membuat objek stemmer dari library Sastrawi factory = StemmerFactory() stemmer = factory.create_stemmer() # Melakukan stemming pada review stemmed_review = stemmer.stem(review) # Kembalikan review yang telah di-stemming Kembalikan stemmed_review </pre>

Tabel 13. Perbandingan Hasil *Stemming*

Komentar	Hasil <i>Stemming</i>
Cuma make red jelly nya aja sii. Sejauh ini, alhamdulillah better si.. Cuma aku make nya pencampuran 3 merek skincare, salah 1 nya ms glow.	<pre> 'make' 'red' 'jelly' 'alhamdulillah' 'better' 'campur' 'merek' 'skincare' 'salah' 'ms glow' </pre>

3.1.2.7 Normalisasi

Normalisasi merupakan proses yang dijalankan untuk merubah kata menjadi baku dan sesuai dengan ejaan.

Tabel 14. *Pseudocode Normalisasi*

<i>Pseudocode Normalisasi</i>
<pre> # Fungsi untuk stemming Fungsi stemText(review): # Input: review (teks yang akan di-stemming) </pre>

```
# Output: review yang telah di-stemming

# Membuat objek stemmer dari library Sastrawi
factory = StemmerFactory()
stemmer = factory.create_stemmer()

# Melakukan stemming pada review
stemmed_review = stemmer.stem(review)

# Kembalikan review yang telah di-stemming
Kembalikan stemmed_review
```

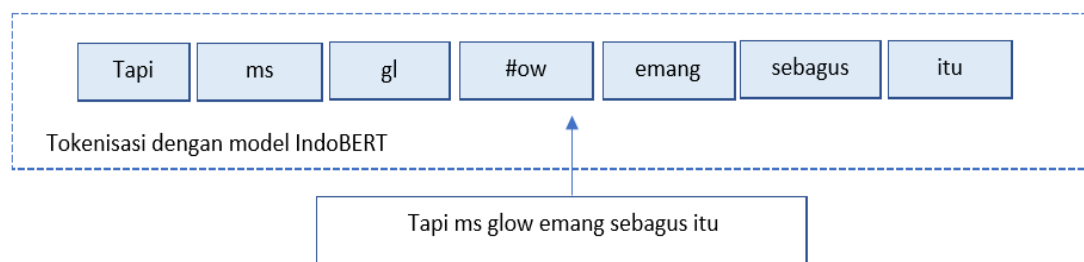
Tabel 15. Perbandingan Hasil Normalisasi

Komentar	Hasil Normalisasi
gapernah coba ms glow nder, tp siapa tau cocok coba cari review trusted dulu	Gak pernah coba ms glow sender tapi siapa tau cocok coba cari review trusted dulu

3.2 BERT *Sentiment Classifier*

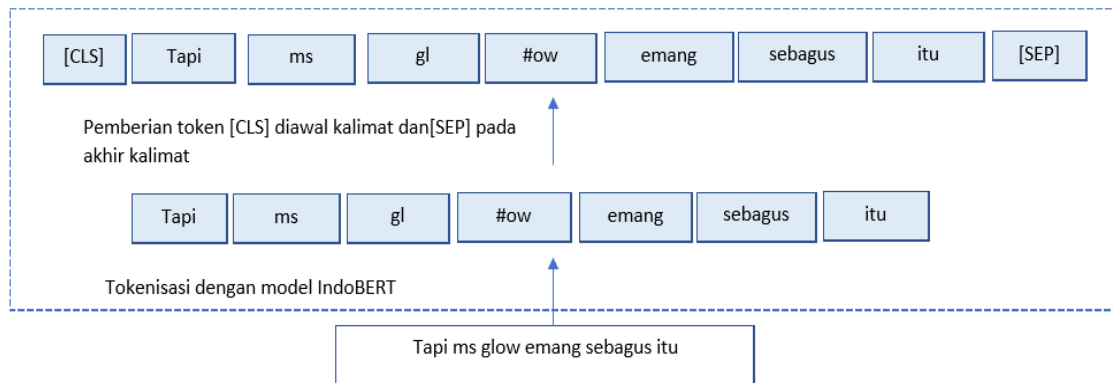
Pada tahap ini *dataset* akan di proses menggunakan metode BERT. Kalimat akan diubah sesuai dengan representasi input pada BERT yang dilakukan oleh *tokenizer*. Berikut tahapan metode BERT

1. Kalimat dipisah menjadi kata per kata atau sub dengan model indobert. Jika sebuah kata tidak terdapat pada vocabulary maka kata tersebut diganti dengan token [UNK] atau unknown. Namun pada BERT untuk menghindari informasi yang hilang maka kata dipecah menjadi sub kaya dengan symbol ##. Dataset yang berisi kalimat “Tapi ms glow emang bagus itu”.



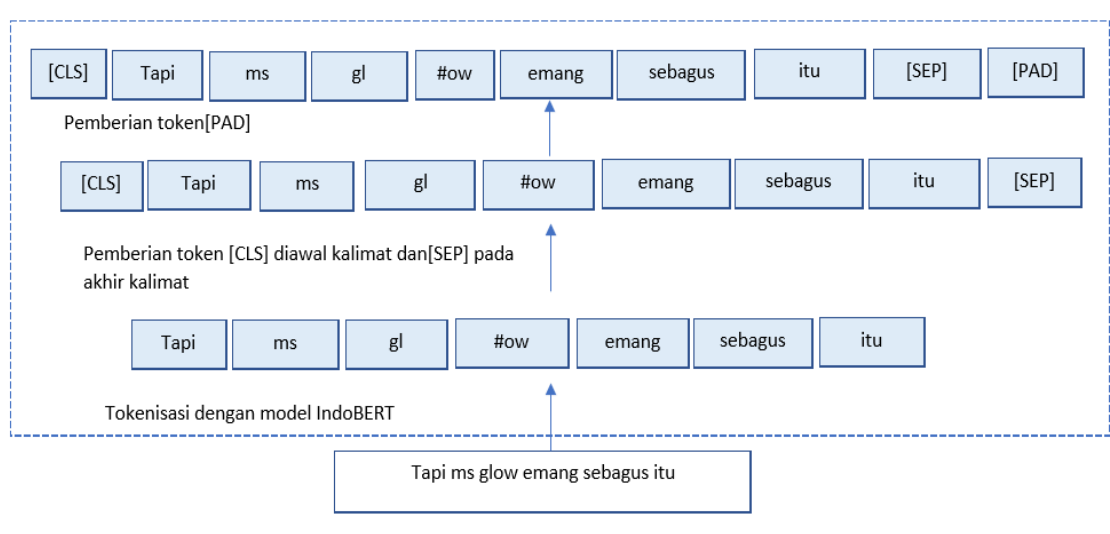
Gambar 3. 2 Proses Tokenisasi Dengan Indobert

2. Kalimat diberikan token khusus yakni token [CLS] untuk awal kalimat serta [SEP] untuk akhir kalimat.



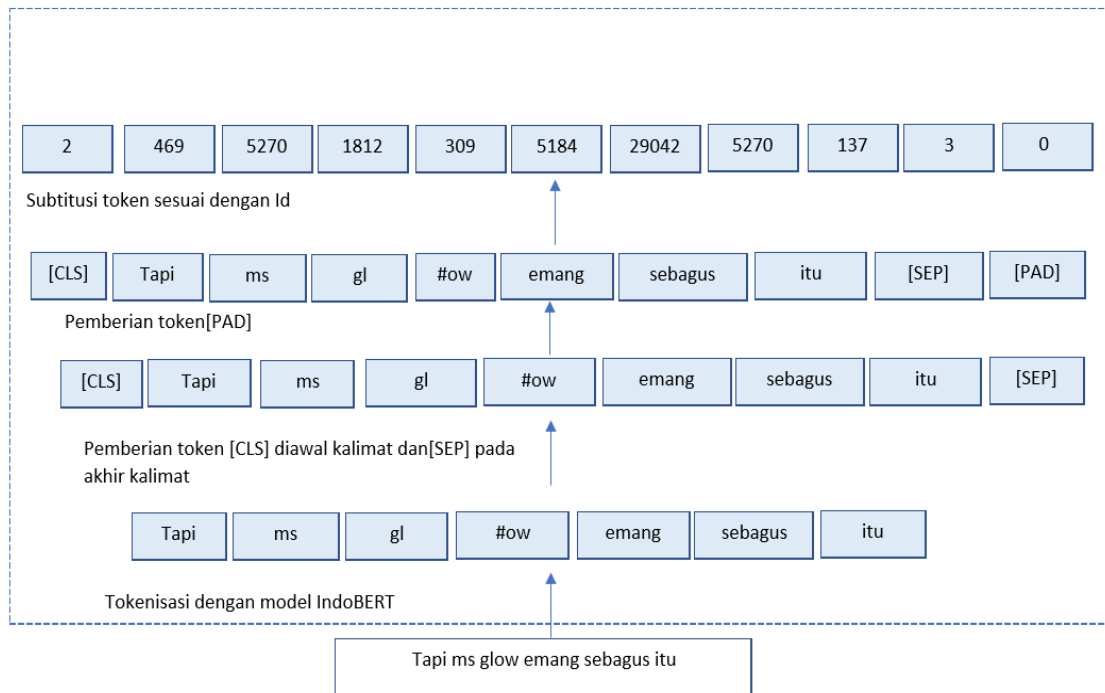
Gambar 3. 3 Proses Token *Embeddings*

3. Panjang maksimum yang telah di tentukan disesuaikan dengan kalimat dan memberi padding dengan token [PAD].



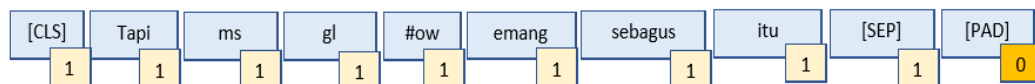
Gambar 3. 4 Proses Pemberian *Padding*

4. Kalimat akan disesuaikan menggunakan ID atau bilangan unik yang diperoleh berdasarkan indeks kata pada *vocabulary*. Kemudian akan disimpan sebagai *token id*.



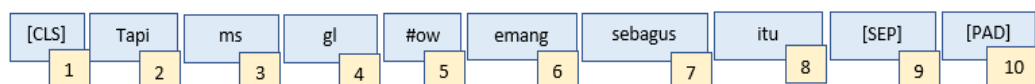
Gambar 3. 5 Tahap Substitusi Token dengan IDnya

5. Dalam tahap ini membedakan kalimat dengan pertama dan kalimat kedua atau yang disebut *sentence embedding*. *Sentence embedding* pada token ini memberikan angka pada 0 pada *padding*.



Gambar 3. 6 Tahap *Sentence Embedding*

6. *Positional embedding* untuk menunjukkan posisi setiap kata pada kalimat dengan penambahan tiap token.



Gambar 3. 7 Tahap *Positional Embedding*

Berikut merupakan *pseudocode* Alur Pengerjaan BERT pada tabel 16:

Tabel 16. *Pseudocode BERT*

```

# Fungsi untuk tokenisasi menggunakan IndoBERT
Fungsi tokenize_with_indobert(text, max_length=10):
    # Input: text (teks yang akan ditokenisasi), max_length (panjang maksimum token)
    # Output: tokens (daftar token), token_ids (daftar ID token)

    # Inisialisasi tokenizer menggunakan model 'indobenchmark/indobert-base-p1'
    tokenizer = BertTokenizer.from_pretrained('indobenchmark/indobert-base-p1')

    # Tokenisasi menggunakan WordPiece
    tokens = tokenizer.tokenize(text)

    # Tambahkan token [CLS] di awal kalimat dan [SEP] di akhir kalimat
    tokens = ['[CLS]'] + tokens + ['[SEP]']

    # Jika panjang tokens kurang dari panjang maksimum, tambahkan padding [PAD]
    if len(tokens) < max_length:
        padding_length = max_length - len(tokens)
        tokens += ['[PAD]'] * padding_length

    # Konversi token ke ID token
    token_ids = tokenizer.convert_tokens_to_ids(tokens)

    # Kembalikan hasil tokenisasi
    Kembalikan tokens, token_ids

# Kata-kata yang ingin ditokenisasi
text = "Tapi ms glow emang bagus itu"
# Panggil fungsi tokenisasi dengan IndoBERT
tokens_result, token_ids_result = tokenize_with_indobert(text)
# Tampilkan hasil tokenisasi
print("Tokens:", tokens_result)
print("Token IDs:", token_ids_result)

```

3.3 Training data

Dataset dikategorikan ke dalam tiga jenis yaitu *training*, validasi serta *testing*. Proses *training* data BERT di *fine-tuning* menggunakan *hyperparameter* untuk menyesuaikan data BERT guna mendukung proses pembelajaran. *Hyperparameter* dalam *training* BERT adalah ukuran Batch size, *Epoch*, *Learning rate*.

3.5 Evaluasi

Evaluasi merupakan tahapan yang bertujuan untuk melihat dari hasil analisis sentimen terhadap *dataset*. Nilai akurasi yang diperoleh dari tahapan sebelumnya akan menjadi nilai akurasi model apabila berada pada nilai tertinggi. pada tahap evaluasi ini akan mendapatkan prediksi dari model, *confusion matrix* yang dipakai dan terlihat sebagai berikut:

Tabel 17. *Confusion Martix*

		KELAS BENAR		
		POSITIF	NETRAL	NEGATIF
Prediksi Kelas	POSITIF	Benar Positif (FP)	Salah Positif (Fnt)	Salah Positif (Fp)
	NETRAL	Salah Netral (FNt)	Benar Netral (TNt)	Salah Netral (FNt)
	NEGATIF	Salah Negatif (FN)	Salah Negatif (FNt)	Benar Negatif (TN)

Han dan Kamber (2017), berpendapat bila *Confusion Matrix* bisa didefinisikan menjadi alat yang memiliki kemampuan menganalisis apakah sebuah pengklasifikasi bagus dalam mendeteksi tupel dari kelas yang berbeda. Nilai Positif Benar dan Benar Negatif memberikan informasi pada saat *classifier* mengelompokkan data dengan nilai yang benar, sedangkan Salah Positif dan Salah Negatif memberikan informasi pada saat melakukan klasifikasi.

Dengan Keterangan :

1. TP (Benar Positif) → Jumlah data dengan nilai sebenarnya positif dan nilai prediksi positif

2. FP (Salah Positif) → Jumlah data dengan nilai sebenarnya negatif dan nilai prediksi positif
3. FN (Salah Negatif) → Jumlah data dengan nilai sebenarnya positif dan nilai prediksi negatif
4. TN (Salah Negatif) → Jumlah data dengan nilai sebenarnya negatif dan nilai prediksi negatif

BAB IV

IMPLEMENTASI DAN PENGUJIAN

Dalam Bab IV ini berisikan implementasi serta pengujian penelitian dari analisis sentimen dengan metode BERT yang sebelumnya telah dijelaskan pada Bab III .

4.1 Implementasi Sistem

Implementasi sistem adalah tahapan diterapkannya perangkat lunak yang telah dijalankan. Dengan kata lain telah dapat diterapkan/atau didesain untuk selanjutnya dapat dijalankan sepenuhnya. Tahapan ini adalah tahap dimana sistem telah siap untuk dioperasikan.

4.4.1 Spesifikasi perangkat keras

Agar mampu diimplementasikan serta dilakukannya pengujian terhadap analisis sentimen pada Review Skincare Lokal pada *Twitter*, penelitian ini memakai spesifikasi perangkat keras berikut:

1. ACER Swift 3
2. Memory RAM 4 GB
3. Intel(R) Core(TM) i5-8250U CPU @ 1.60GHz (8 CPUs) 1.8 GHz
4. Memory 819 GB

4.1.2 Perangkat Lunak

Spesifikasi perangkat lunak yang dipakai pada penelitian ini antara lain:

1. Sistem Operasi Windows 10 Pro 64-bit (10.0, Buid 14393)
2. Google Collab
3. Library BERT: transformers. Torch, nltk, numptk, re, matplotlib, scabon, sklearn, sastrawi.
4. Library *Scraping* : Pandas, node.js

4.2 Implementasi *Scraping Data Tweet*

Dataset review skincare MS Glow yang diambil berasal dari kolom pencarian *search bar*. Review yang diambil yaitu untuk produk Acne dan whitening. Penulis melakukan *scraping* dengan menggunakan phyton dengan untuk mengumpulkan datta public dari platform media social *Twitter*. Pada penelitian ini penulis mengambil 1389 data

review product whitening dan *acne* yang berasal dari *tweet search bar* kata kunci. Data yang diambil hanya menggunakan Bahasa Indonesia.

96	Aku pake paket acne ms glow plus serum scarlet acne alhamdulillahh membaik walupun skrg lg ga make tetap seger mukanya
97	dips!ms glow acne apa whitening yang lebih ngaruh yang mana
98	kalo hanasui nya dipake bareng ama ms glow yg acne boleh ga sih kak?
99	Sebelumnya boleh pake toner acne ms glow gk ka? Ada kandungan niacinamide soalnya
100	jerawat hilang, bruntusan hilang, bekas' nya hilang, dan alhamdulillah gua udah mau setahun pake itu dan alhamdulillah cocok' ajaa, anw gu
101	jujurly aku nyoba nyoba yg dari twt nggak ngaruh di aku, akhirnya iseng nyoba ke ms glow trus pake acne premiumnya(cream malam), se
102	Kalo aku baru kali ini ngerasa agak mendingan karna rajin double cleansing nder.. aku pake micellar waternya garnier, fash wash ms glow.
103	2019 breakout parah, akhirnya 2021 nyobain MS GLOW ACNE gataunya sampe sekarang cocoknya sih, bahkan pernah hampir 3minggu ga
104	dips!ms glow acne sama ms glow yang whitening lebih ngaruh yang mana guys?
105	Temenku yg oily acne prone pada ga cocok ms glow malah jd breakout
106	Oh ya mau nnya kalo make ms glow acne yg dluan ss nya atau ms glow
107	Aku pernah breakout parah nder pas SMP. Terus pas kls 10 aku coba pake ms glow sampe skrng, udah setahunan. Di aku cocokkk banget!
108	Dear! kalau pakai paket acne ms glow, tapi day cream nya di ganti sunscreen bisa gak ya? trus kalau bisa bagusnya pakai sunscreen apa? '
109	Jerawatku hilang pake ms glow di aku cocok muka jadi lumayan cerah, tapi sekarang udah ga pake ms glow aku pake yang paket acne

Gambar 4. 1 Hasil Scraping

4.3 Labelisasi dataset

Hasil *scraping dataset* yang berformat scv tersebut dilakukannya labelisasi *dataset* dengan penentuan kategori data yakni, negatif, netral, dan positif.

1	review	category
2	Aku pake serum acne MS glow ampuh sih	positive
3	acne spot nya skintific sama dark spot nya ms glow ni bener2 perpaduan yg mantul bgt????	positive
4	wt's facial wash ms glow sama night cream ms glow acne masih baru,baru dipake 2x aja. 60k aja 2 ya ?? #zonauang	neutral
5	wt's facial wash ms glow sama night cream ms glow acne masih baru,baru dipake 2x aja. 60k aja 2 ya ??	neutral
6	Aku lagi make acne sopt MS GLOW. Best banget, jerawat mendem pada muncul matanya terus betus, bekasnya juga nggak ada, te	positive
7	Aku pake ms glow yang acne untuk type wajah berminyak dan beruntusan	negative
8	nder maaf oot kalo skincare nya ms glow acne terus mau eksfo itu mending pake produk apa ya?	neutral
9	st oily acne sensi, pake acne cream sama fw, muka jadi bo parah pdhl cuma sekali beli bukan efek pemakaian lama	negative
10	penyesalan gue seumur hidup itu beli ms glow	negative
11	kak mjb mau tnya kalo habis exfo bisa pake toner glowingnya ms glow nggak ya? atau harus toner for acne? yg trmasuk hydratin	neutral
12	Pernah gini deh cmn aku suka ganti2 dulu?? kitoderm acne sepaket, maggie glow acne sama ms glow acne cmn pake cream serui	neutral

Gambar 4. 2 Labelisasi Dataset

Pada Gambar 4.2 dataset review skincare MS GLOW yang telah dilabelisasi secara manual sesuai dengan kategori data sebelum masuk ke tahap *pre-processing*.

4.4 Preprocessing dataset

Dalam penelitian ini, tahapan *preprocessing* terdiri atas berbagai tahapan, antara lain: *case folding*, data cleaning, tokenisasi, stopwords removal, stemming, serta normalisasi.

	review_text	category
0	Aku pake serum acne MS glow ampuh sih	positive
1	acne spot nya skintific sama dark spot nya ms ...	positive
2	wt's \nfacial wash ms glow sama night cream ms ...	neutral
3	wt's \nfacial wash ms glow sama night cream ms ...	neutral
4	Aku lagi make acne sopt MS GLOW. Best banget, ...	positive
...
1385	Paket MS GLOW whitening,flek dan acne ready ya...	neutral
1386	PAKET CREAM MS GLOW WHITENING SERIS PEMULA DAN...	neutral
1387	Ready ms glow paket whitening untuk agne sma u...	neutral
1388	Siapa sih yg ga kenal Ms Glow...ud meluber lar...	positive
1389	MS GLOW , ADA PAKET WHITENING , PAKET ACNE DAN...	neutral

Gambar 4. 3 *Dataset*

Dataset yang akan di proses bisa dilihat dalam gambar 4.3. Dataset review tersebut akan melalui tahapan *pre-processing*.

4.4.1 Case Folding

Hasil Case Folding:

```

0          aku pake serum acne ms glow ampuh sih
1      acne spot nya skintific sama dark spot nya ms ...
2      wt's \nfacial wash ms glow sama night cream ms ...
3      wt's \nfacial wash ms glow sama night cream ms ...
4      aku lagi make acne sopt ms glow. best banget, ...
      ...
1385     paket ms glow whitening,flek dan acne ready ya...
1386     paket cream ms glow whitening seris pemula dan...
1387     ready ms glow paket whitening untuk agne sma u...
1388     siapa sih yg ga kenal ms glow...ud meluber lar...
1389     ms glow , ada paket whitening , paket acne dan...
Name: case_folded_text, Length: 1390, dtype: object

```

Gambar 4. 4 Hasil *Case Folding*

Dataset review yang mengandung huruf kapital pada tahap *case folding*, Oleh karena itu, akan diubah menjadi huruf kecil menggunakan fungsi *lower* pada dataset. Hasil case folding bisa dilihat dalam Gambar 4.4.

4.4.2 Data Cleaning

```

Hasil Filtering:
0          aku pake serum acne ms glow ampuh sih
1      acne spot nya skintific sama dark spot nya ms ...
2      wts facial wash ms glow sama night cream ms g...
3      wts facial wash ms glow sama night cream ms g...
4      aku lagi make acne sopt ms glow best banget ...
...
1385     paket ms glow whitening flek dan acne ready ya...
1386     paket cream ms glow whitening seris pemula dan...
1387     ready ms glow paket whitening untuk agne sma u...
1388     siapa sih yg ga kenal ms glow ud meluber lar...
1389     ms glow ada paket whitening paket acne dan...
Name: filtered_text, Length: 1390, dtype: object

```

Gambar 4. 5 Hasil *Data Cleaning*

Dalam Gambar 4.5 didapatkan hasil proses *Data Cleaning*. *Data review* yang sebelumnya mengandung karakter, kata berulang, *hashtag*, *symbol*, angka, tanda baca akan dihapus.

4.4.3 Tokenize

```

Hasil Tokenize:
0      [aku, pake, serum, acne, ms, glow, ampuh, sih]
1      [acne, spot, nya, skintific, sama, dark, spot,...
2      [wts, facial, wash, ms, glow, sama, night, cre...
3      [wts, facial, wash, ms, glow, sama, night, cre...
4      [aku, lagi, make, acne, sopt, ms, glow, best, ...
...
1385     [paket, ms, glow, whitening, flek, dan, acne, ...
1386     [paket, cream, ms, glow, whitening, seris, pem...
1387     [ready, ms, glow, paket, whitening, untuk, agn...
1388     [siapa, sih, yg, ga, kenal, ms, glow, ud, melu...
1389     [ms, glow, ada, paket, whitening, paket, acne,...
Name: tokenized_text, Length: 1390, dtype: object

```

Gambar 4. 6 Hasil *Tokenize*

Selanjutnya kalimat dipisahkan untuk suatu rangkaian karakter berdasarkan spasi, tanda baca, dan ekspresi bermakna lainnya melalui tahap *tokenize*. Hasil *tokenize* bisa dilihat dalam Gambar 4.6.

4.4.4 Negation Handling

```
[11] Hasil Negation:
0          aku pake serum acne ms glow ampuh sih
1      acne spot nya skintific sama dark spot nya ms ...
2      wts facial wash ms glow sama night cream ms gl...
3      wts facial wash ms glow sama night cream ms gl...
4      aku lagi make acne sopt ms glow best banget je...
...
1385     paket ms glow whitening flek dan acne ready ya...
1386     paket cream ms glow whitening seris pemula dan...
1387     ready ms glow paket whitening untuk agne sma u...
1388     siapa sih yg ga kenal ms glow ud meluber laris...
1389     ms glow ada paket whitening paket acne dan jug...
Name: negation text, Length: 1390, dtype: object
```

Gambar 4. 7 Hasil *Negation Handling*

Dataset review yang mengandung huruf kata ('tak', 'tidak', 'bukan', 'jangan', 'belum') pada tahap *negation handling*. Maka kata yang mengandung kata pada list *negation handling* akan diberi penanda (“_”) sehingga kata tidak dihapus pada stopword removal Gambar 4.7.

4.4.5 Stopword Removal

```
Hasil Stopword Removal:
0          pake serum acne ms glow ampuh sih
1      acne spot nya skintific dark spot nya ms glow ...
2      facial wash ms glow night cream ms glow acne d...
3      facial wash ms glow night cream ms glow acne d...
4      make acne sopt ms glow best banget jerawat men...
...
1385     paket ms glow whitening flek acne ready yaaaa ...
1386     paket cream ms glow whitening seris pemula acn...
1387     ready ms glow paket whitening agne sma ultimat...
1388     sih yg ga kenal ms glow ud meluber larisnya kh...
1389     ms glow paket whitening paket acne paket flek ...
Name: stopword_removed_text, Length: 1390, dtype: object
```

Gambar 4. 8 Hasil *Stopword Removal*

Pada Gambar 4.8 didapatkan hasil proses *stopword removal*. Tahapan ini termasuk dalam proses preprocessing teks dengan tujuan untuk mengeliminasi kata yang tidak relevan pada sebuah kalimat berdasarkan *list stopwords* pada kamus tala.

4.4.6 Stemming

```

Hasil Stemming:
0          pake serum acne ms glow ampuh sih
1      acne spot nya skintific dark spot nya ms glow ...
2      facial wash ms glow night cream ms glow acne d...
3      facial wash ms glow night cream ms glow acne d...
4      make acne sopt ms glow best banget jerawat men...
...
1385     paket ms glow whitening flek acne ready yaaaa ...
1386     paket cream ms glow whitening seris mula acne ...
1387     ready ms glow paket whitening agne sma ultimat...
1388     sih yg ga kenal ms glow ud luber laris khasiat...
1389     ms glow paket whitening paket acne paket flek ...
Name: stemmed_text, Length: 1390, dtype: object

```

Gambar 4. 9 Hasil Stemming

Dalam tahapan ini, kata yang mengandung imbuhan awalan, akhiran, dan gabungan keduanya akan di hapus untuk mengambil kata dasar dari sebuah kata. Hasil *stemming* bisa dilihat dalam Gambar 4.9

4.4.7 Normalisasi

Dataset hasil *scraping* memiliki kata-kata tidak sesuai standar atau tata ejaan yang benar dalam Bahasa Indonesia. Dikarenakan data pada *twitter* banyak menggunakan slang, sehingga data yang tidak sesuai standar akan sesuai standar.

```

Teks Normalisasi:
0          pakai serum acne ms glow ampuh sih
1      acne spot nya skintific dark spot nya ms glow ...
2      facial wash ms glow night cream ms glow acne d...
3      facial wash ms glow night cream ms glow acne d...
4      memakai acne sopt ms glow best banget jerawat ...
...
1385     paket ms glow whitening flek acne ready ya dea...
1386     paket cream ms glow whitening seris pemula acn...
1387     ready ms glow paket whitening agne sama ultima...
1388     sih yang enggak kenal ms glow sudah meluber la...
1389     ms glow paket whitening paket acne paket flek ...
Name: normalized_text, Length: 1390, dtype: object

```

Gambar 4. 10 Hasil Normalisasi

4.5 Split Dataset

Pada tahapan ini, Dataset dikelompokkan ke dalam *dataset training*, *dataset validasi*, serta *dataset testing*. *Training* data dipakai untuk melatih model. Validasi berfungsi dalam meminimalisir *overfitting* sehingga tidak dapat melakukan generalisasi dengan

baik pada data baru. Proporsi data pada penelitian ini adalah 60% *train set*, 24% *validation set*, 16% *test set*. Dapat dilihat pada gambar 4.11.

```
[ ] from sklearn.model_selection import train_test_split

# Memisahkan DataFrame df_prepos menjadi train set dan val_test set
train_set, val_test_set = train_test_split(df_prepos, test_size=0.4, stratify=df_prepos['category'], random_state=42)

# Memisahkan val_test set menjadi validation set dan test set
val_set, test_set = train_test_split(val_test_set, test_size=0.4, stratify=val_test_set['category'], random_state=42)
```

Gambar 4. 11 Proses *Spilitting Dataset*

4.6 BERT

Setelah *splitting* data, implementasi BERT menggunakan model IndoBERT-base-p1 yang dikembangkan oleh IndoNLU dengan menggunakan teknik *fine-tuning*. Arsitektur berbasis BERT ini dirancang khusus untuk menangani bahasa Indonesia. IndoBERT memanfaatkan transformator, yaitu mekanisme yang mempelajari terkait hubungan kontekstual antar kata pada teks, sehingga meningkatkan akurasi hasil klasifikasi setiap label (Putra et al., 2021).

Penulis juga menjalankan *fine-tuning* IndoBERT dengan memakai *hyperparameters* berikut:

1. Batch size: 16
2. *Epoch*: 16
3. Learning rate: 5e-6 (0.000003)

Hypermaters yang digunakan pada penelitian ini ditentukan oleh beberapa alasan. Batch size 16 dipilih dikarenakan batch size yang semakin besar akan berdampak pada semakin lama waktu yang diperlukan dalam menyelesaikan satu batch (Osinga, 2018). Learning rate 5e-6 dipilih karena untuk melanjutkan pelatihan dan pembelajaran untuk memperoleh tingkat pengenalan kesalahan yang lebih baik (Liu et al., 2023).

Pada penelitian ini dilakukan pengujian dengan 3 *epoch* yang terdiri dari 5 *epoch* (Putra, Eza Ananda., 2021), 10 *epoch* dan 16 *epoch* (Fimoza, Dwi., 2021). Ketiga *epoch* diuji coba dan memberikan masing-masing hasil. Diperoleh *epoch* 16 menghasilkan hasil yang lebih baik diantara ketiganya.

```
[ ] print(classification_report(test_real, test_pred, target_names=['positive', 'neutral', 'negative']))
```

	precision	recall	f1-score	support
positive	0.50	0.08	0.14	25
neutral	0.80	0.95	0.87	119
negative	0.76	0.75	0.75	79
accuracy			0.78	223
macro avg	0.69	0.59	0.59	223
weighted avg	0.75	0.78	0.75	223

Gambar 4. 12 Akurasi dengan 5 *Epoch*

```
✓ [72] print(classification_report(test_real, test_pred, target_names=['positive', 'neutral', 'negative']))  
0d
```

	precision	recall	f1-score	support
positive	0.70	0.28	0.40	25
neutral	0.84	0.91	0.87	119
negative	0.73	0.78	0.76	79
accuracy			0.79	223
macro avg	0.76	0.66	0.68	223
weighted avg	0.79	0.79	0.78	223

Gambar 4. 13 Akurasi dengan 10 *Epoch*

```
[ ] print(classification_report(test_real, test_pred, target_names=['positive', 'neutral', 'negative']))
```

	precision	recall	f1-score	support
positive	0.70	0.28	0.40	25
neutral	0.87	0.90	0.88	119
negative	0.72	0.82	0.77	79
accuracy			0.80	223
macro avg	0.76	0.67	0.68	223
weighted avg	0.80	0.80	0.79	223

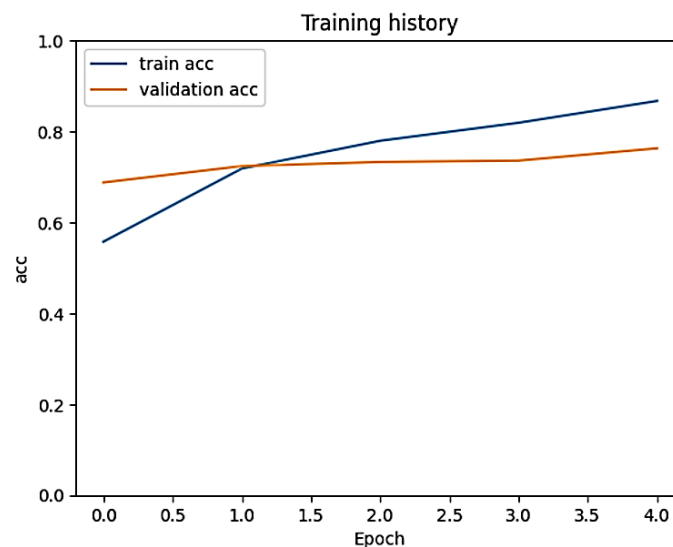
Gambar 4. 14 Akurasi dengan 16 *Epoch*

Dari penelitian yang dilakukan dengan 3 epoch tersebut. Masing-masing memiliki hasil yang berbeda. Penggunaan *epoch* 5 pada gambar 4.12 didapatkan hasil 78% dan *epoch* 10 pada gambar 4.13 didapatkan hasil akurasi 79%. *Epoch* 16 pada gambar 4.14 diuji didapatkan hasil akurasi bernilai 80 %. Sehingga dari ketiga *epoch* yang diuji maka *epoch* 16 menghasilkan hasil yang lebih baik diantara ketiganya.

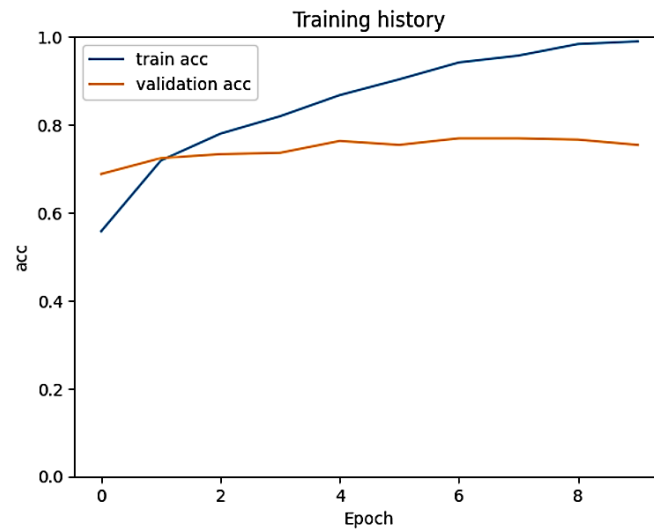

```
[ ] (Epoch 14) TRAIN LOSS:0.0248 LR:0.00000500: 100%|██████████| 53/53 [00:11<00:00, 4.61it/s]
(Epoch 14) TRAIN LOSS:0.0248 ACC:1.00 F1:1.00 REC:1.00 PRE:0.99 LR:0.00000500
0%|          | 0/21 [00:00<?, ?it/s]/usr/local/lib/python3.10/dist-packages/torch/utils/data/dataloader.py:557: I
warnings.warn(_create_warning_msg(
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344: UndefinedMetricWarning: Precision
_warn_prf(average, modifier, msg_start, len(result))
VALID LOSS:0.9806 ACC:0.76 F1:0.69 REC:0.68 PRE:0.70: 100%|██████████| 21/21 [00:03<00:00, 6.94it/s]
(Epoch 14) VALID LOSS:0.9806 ACC:0.76 F1:0.69 REC:0.68 PRE:0.70
0%|          | 0/53 [00:00<?, ?it/s]/usr/local/lib/python3.10/dist-packages/torch/utils/data/dataloader.py:557: I
warnings.warn(_create_warning_msg(
(Epoch 15) TRAIN LOSS:0.0153 LR:0.00000500: 100%|██████████| 53/53 [00:11<00:00, 4.70it/s]
(Epoch 15) TRAIN LOSS:0.0153 ACC:1.00 F1:1.00 REC:1.00 PRE:1.00 LR:0.00000500
0%|          | 0/21 [00:00<?, ?it/s]/usr/local/lib/python3.10/dist-packages/torch/utils/data/dataloader.py:557: I
warnings.warn(_create_warning_msg(
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344: UndefinedMetricWarning: Precision
_warn_prf(average, modifier, msg_start, len(result))
VALID LOSS:1.0415 ACC:0.76 F1:0.66 REC:0.65 PRE:0.70: 100%|██████████| 21/21 [00:03<00:00, 6.42it/s]
(Epoch 15) VALID LOSS:1.0415 ACC:0.76 F1:0.66 REC:0.65 PRE:0.70
0%|          | 0/53 [00:00<?, ?it/s]/usr/local/lib/python3.10/dist-packages/torch/utils/data/dataloader.py:557: I
warnings.warn(_create_warning_msg(
(Epoch 16) TRAIN LOSS:0.0212 LR:0.00000500: 100%|██████████| 53/53 [00:12<00:00, 4.40it/s]
(Epoch 16) TRAIN LOSS:0.0212 ACC:0.99 F1:0.99 REC:0.99 PRE:0.99 LR:0.00000500
0%|          | 0/21 [00:00<?, ?it/s]/usr/local/lib/python3.10/dist-packages/torch/utils/data/dataloader.py:557: I
warnings.warn(_create_warning_msg(
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344: UndefinedMetricWarning: Precision
_warn_prf(average, modifier, msg_start, len(result))
VALID LOSS:1.0708 ACC:0.75 F1:0.69 REC:0.69 PRE:0.69: 100%|██████████| 21/21 [00:02<00:00, 7.09it/s](Epoch 16) VA
```

Gambar 4. 15 Proses *Training* dan *Evaluasi*

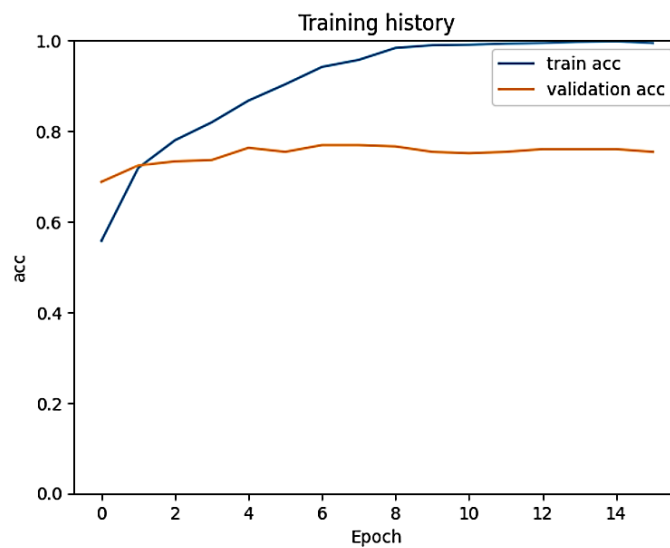
Proses training data dan evaluasi dapat dilihat dari gambar 4.15. Digunakannya *epoch* 16 menghasilkan nilai akurasi.



Gambar 4. 16 Kurva Hasil Performa *Training* dan Validasi *Epoch* 5



Gambar 4. 17 Kurva Hasil Performa *Training dan Validasi Epoch 10*



Gambar 4. 18 Kurva Hasil Performa *Training dan Validasi Epoch 16*

Dari hasil pengamatan gambar 4. 16, gambar 4.18, gambar 4.18 dari eksperimen tersebut, diketahui bahwa akurasi pada saat pelatihan lebih baik dibandingkan dengan hasil pada saat validasi. Kurva menunjukkan akurasi hasil yang diperoleh meningkat selama pelatihan. Tetapi, hasil dari validasi lebih rendah.

Tabel 18. Hasil Pengujian**Hasil Pengujian**

No.	Teks Review	Kategori	Prediksi
1.	sekarang pakai ms glow acne kak	Netral	Netral
2.	contoh btw gue pakai produk ms glow whitening yang malem yang red jelly nama saja sudah whitening bikin putih ya enggak putih	Negatif	Negatif
3.	pakai ms glow paket whitening serum luminous	Netral	Netral
4.	banget sempat memakai belum ms glow seviral jerawat memakai ms glow yang acne benaran bagus sih enggak gantung juga	Positif	Positif
5.	susah beri paket whitening ms glow yang cream malam dobel karena suka cpet habis walopun enggak susah bismillah notiv	Netral	Negatif
6.	pakai masker nya ms glow yang acne jerawat kempes malam	Positif	Positif
7.	pakai ms glow sama sabar enggak langsung lihat pakai paket rangkai nya benar konsultasi biar enggak menyesal kaya aku pakai yang whitening kusam	Negatif	Negatif
8.	coba pakai ms glow acne gua begitu jerawat parah pakai cocok bukan jual ya	Positif	Negatif
9.	ms glow acne whitening ngaruh	Netral	Positif
10.	ms glow acne spot treatment bikin	Negatif	Positif

	jerawat jadi mendem kalo ken jadi mateng pakai wardah acne derm		
11.	ms glow pakai yang whitening premium minggu sudah lihat sih cerah bekas jerawat hilang	Positif	Positif
12.	ms glow yang acne jejak breakout tidak sisa bekas	Positif	Positif
13.	beauty gaiss review ms glow dong teman ku pakai jadi ken pakee tapi takut bo huhuu btw oily acne prone	Netral	Netral
14.	hai nder sori mjb jual pteloved fw nya pakai whitening kalo mencoba banget nderr btw ms glow bagus itu	Positif	Positif
15.	minggu pakai ms glow paket acne bagus banget benar perlahan lahan muluss percaya karena sudah review dari richard lee enggak gantung aku kadang suka enggak sempat pakenya muka enggak kenpaa	Positif	Positif
16.	jerawat hilang bruntusan hilang bekas nya hilang alhamdullilah gua sudah tahun pakai alhamdullilah cocok saja anw gua pakai ms glow acne gua sudah pindah ms glow whitening	Positif	Positif
17.	dear parah banget sampai bekas huhu sudah pakai saja tetap sunscrean bersih sampai serum tetap saja pakai ms glow acne maju mundur	Netral	Positif
18.	pakai ms glow paket yang	Netral	Netral

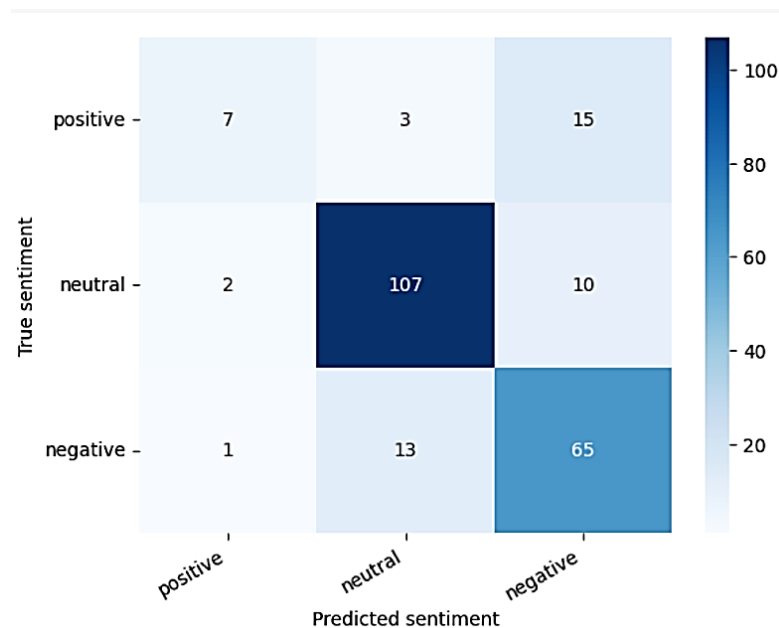
	whitening kalo serum amp essence nya dari loreal revitalift		
19.	pakai ms glow whitening ngaruh banget ratain warna kulit habis biasa pakai sunscreen nder	Positif	Positif
20.	posisi tetap pakai paket ms glow whitening benar ku pakai dua alhamdulillah sih asli alus banget muka dari brandon kece sih harga segitu alhamdulillah cocok	Positif	Positif

Berdasarkan hasil pengujian dengan menggunakan *dataset testing* yang berjumlah 223 data, maka diperoleh hasil sebagai berikut:

Tabel 19. Tabel Prediksi

Kategori	Jumlah	Prediksi Benar
Positif	25	7
Negatif	79	65
Netral	119	107

4.7 Evaluasi



Gambar 4. 19 Diagram *Confusion Matrix*

Pada penelitian ini menggunakan *confusion matrix* pada gambar 4.19 untuk mengetahui performa dari model yang telah didapatkan dari proses *training*. Untuk pengukuran performa model, menggunakan nilai *precision*, akurasi, *F1 Score*, dan *recall*. Berdasarkan hasil yang didapatkan dari data *testing* ada beberapa teks review yang tidak diprediksi dengan tepat. Terdapat 223 data *testing* yang menghasilkan 179 data diprediksi dengan benar. Akurasi yang dicapai yaitu 80%. Adapun rumus yang dipakai untuk menghitung nilai akurasi yaitu:

Akurasi = Jumlah hasil benar/ total data * 100%

$$= 179/223 * 100\%$$

$$= 80\%$$

Selanjutnya nilai *precision*, *recall*, *F1 Score* juga diperoleh. Rumus yang digunakan yaitu:

1. *Precision*

$$\text{Positif} = TP / (TP + FP)$$

$$= 7/(7+3) = 0.70$$

$$\text{Negatif} = TP / (TP + FP)$$

$$= 65/(65+25)$$

$$= 0.72$$

$$\text{Netral} = \text{TP} / (\text{TP} + \text{FP})$$

$$= 107 / (107 + 16)$$

$$= 0.87$$

2. *Recall*

$$\text{Positif} = \text{TP} / (\text{TP} + \text{FN}) = 7 / (7 + 18)$$

$$= 0.28$$

$$\text{Negatif} = \text{TP} / (\text{TP} + \text{FN}) = 65 / (65 + 14)$$

$$= 0.82$$

$$\text{Netral} = \text{TP} / (\text{TP} + \text{FN}) = 107 / (107 + 12)$$

$$= 0.90$$

3. *F1 Score*

$$\text{Positif} = 2 * (\text{Recall} * \text{Precision}) / (\text{Recall} + \text{Precision})$$

$$= 2 * (0.28 * 0.70) / (0.28 + 0.70)$$

$$= 2 * (0.19 / 0.98)$$

$$= 2 * 0.19$$

$$= 0.40$$

$$\text{Negatif} = 2 * (\text{Recall} * \text{Precision}) / (\text{Recall} + \text{Precision})$$

$$= 2 * (0.82 * 0.72) / (0.82 + 0.72)$$

$$= 2 * (0.59 / 1.54)$$

$$= 2 * 0.38$$

$$= 0.77$$

$$\text{Netral} = 2 * (\text{Recall} * \text{Precision}) / (\text{Recall} + \text{Precision})$$

$$= 2 * (0.90 * 0.87) / (0.90 + 0.87)$$

$$= 2 * (0.78 / 1.77)$$

$$= 2 * 0.44$$

$$= 0.88$$

	precision	recall	f1-score	support
positive	0.70	0.28	0.40	25
neutral	0.87	0.90	0.88	119
negative	0.72	0.82	0.77	79
accuracy			0.80	223
macro avg	0.76	0.67	0.68	223
weighted avg	0.80	0.80	0.79	223

Gambar 4. 20 Hasil Evaluasi

Hasil evaluasi ditunjukkan dalam gambar 4.20, melalui tabel tersebut nilai *presicion* yang cukup tinggi didapatkan pada kategori netral, sedangkan positif mendapatkan nilai terendah. Kemudian nilai *F1 score* tertinggi didapatkan pada kategori netral dan yang terendah yaitu kategori positif. Akurasi yang berjumlah 80% menghasilkan error dengan nilai 20% dari total data yang telah proses.

$$\begin{aligned}
 \text{Error Rate} &= \frac{\text{Number of Errors}}{\text{Total Observations}} \\
 &= \frac{44}{223} \\
 &= 20\%
 \end{aligned}$$

Tabel 20. Perbandingan Batch Size

<i>Epoch</i>	<i>Batch Size 16</i>	<i>Batch Size 32</i>
5	78	77
10	79	79
16	80	79

Percobaan menggunakan 2 Batch Size pada tabel 20 yakni, *batch size* 16 dan 32. Berdasarkan percobaan tersebut didapatkan akurasi data paling tinggi yaitu menggunakan batch size 16 dengan akurasi sejumlah 80 %.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan dari data-data hasil pengujian Metode BERT dalam analisis sentimen, didapati hasil sebagai berikut ;

1. Hasil pengujian dengan tiga *epoch* 5,10, dan 16. *Epoch* 16 menghasilkan hasil terbaik, sehingga *epoch* 16 digunakan untuk menganalisis sentimen.
2. Hasil pengujian menggunakan pendekatan *Birectional Encoder Representations from Transfromer* (BERT) dengan model IndoBERT menghasilkan akurasi 80% dengan menggunakan *hyperparamters* yaitu *batch size* sebesar 16, learning rate $5e-6$, dan *epoch* 16.
3. Dari pengujian yang dilakukan model IndoBERT dapat digunakan untuk *dataset* berskala kecil.

5.2 Saran

Berdasarkan hasil penelitian serta kesimpulan diatas, saran yang bisa dipertimbangkan apabila ingin melakukan penelitian untuk selanjutnya yakni ;

1. Semakin banyak *dataset* yang digunakan dan dikumpulkan maka nilai akurasi yang didapat akan semakin baik.
2. Masih minimnya sumber informasi yang memuat informasi mengenai metode BERT oleh karena itu, semoga penelitian ini bisa menjadi salah satu sumber informasi untuk peneliti berikutnya yang ingin memakai metode BERT dalam proses penelitiannya.

DAFTAR PUSTAKA

- Adhine Salsabila, K. D., Ghiffarie, A., Baistama, R. P., Variadi, M. I., & Rhajendra, M. D. (2019). Analisis Sentimen Terhadap Produk The Body Shop Tea Tree Oil. *Jurnal Teknologi Dan Manajemen Informatika*, 5(2). <https://doi.org/10.26905/jtmi.v5i2.3684>
- Alita, D., Fernando, Y., & Sulistiani, H. (2020). Implementasi Algoritma Multiclass Svm Pada Opini Publik Berbahasa Indonesia Di Twitter. *Jurnal Tekno Kompak*, 14(2), 86. <https://doi.org/10.33365/jtk.v14i2.792>
- Aliyah Salsabila, N., Ardhitto Winatmoko, Y., Akbar Septiandri, A., & Jamal, A. (2018). Colloquial Indonesian Lexicon. *Proceedings of the 2018 International Conference on Asian Language Processing, IALP 2018, August*, 226–229. <https://doi.org/10.1109/IALP.2018.8629151>
- Cai, R., Qin, B., Chen, Y., Zhang, L., Yang, R., Chen, S., & Wang, W. (2020). Sentiment analysis about investors and consumers in energy market based on BERT-BILSTM. *IEEE Access*, 8, 171408–171415. <https://doi.org/10.1109/ACCESS.2020.3024750>
- Cholissodin, I., Soebroto, A. A., Hasanah, U., & Febiola, Y. I. (2020). *AI, MACHINE LEARNING & DEEP LEARNING*. Fakultas Ilmu Komputer Universitas Brawijaya. https://doi.org/https://www.academia.edu/44806315/Buku_Ajar_AI_Machine_Learning_and_Deep_Learning
- Cindy Alifia Putri, Adiwijaya, S. A. F. (2020). Analisis Sentimen Review Film Berbahasa Inggris Dengan Pendekatan Bidirectional Encoder Representations from Transformers. *JATISI (Jurnal Teknik Informatika Dan Sistem Informasi)*, 6(2), 181–193. <https://doi.org/10.35957/jatisi.v6i2.206>
- Dankhara, D. (2022). *A Review of Sentiment Analysis of Tweets*. <https://doi.org/10.13140/RG.2.2.14283.67366/1>
- Devlin, J. M. W. C. & K. L. & K. T. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *Wettbewerb in Recht Und Praxis*, 69(7), 796–804. <https://doi.org/https://arxiv.org/pdf/1810.04805.pdf>
- Fimoza, D. (2021). *Analisis Sentimen Terhadap Film Indonesia dengan Pendekatan*

- BERT*. Universitas Sumatera Utara.
<https://doi.org/https://repositori.usu.ac.id/handle/123456789/30445>
- GLOW, M. (2013). *About MS GLOW*. <https://msglowid.com/about>
- Hoang, M., Alija Bihorac, O., & Rouces, J. (2019). Aspect-Based Sentiment Analysis Using BERT. *Proceedings of the 22nd Nordic Conference on Computational Linguistics*, 187–196. <https://aclanthology.org/W19-6120/>
- Ibrohim, M. O., & Budi, I. (2018). A Dataset and Preliminaries Study for Abusive Language Detection in Indonesian Social Media. *Procedia Computer Science*, 135, 222–229. <https://doi.org/10.1016/j.procs.2018.08.169>
- Khafida, A. A., & Hadiyati, F. N. R. (2020). Hubungan Antara Koformitas Teman Sebaya dengan Perilaku Konsumtif Pembelian Skincare di Marketplace Pada Remaja Putri SMA N 1 Kendal. *Jurnal EMPATI*, 8(3), 588–592. <https://doi.org/10.14710/empati.2019.26501>
- Kotler, P. & A. (2016). *Dasar-Dasar Pemasaran* (9th ed.). Erlangga. <https://doi.org/https://onsearch.id/Record/IOS3774.JAKPU000000000104021>
- Kusnadi, R., Yusuf, Y., Andriantony, A., Ardian Yaputra, R., & Caintan, M. (2021). Analisis Sentimen Terhadap Game Genshin Impact Menggunakan Bert. *Rabit : Jurnal Teknologi Dan Sistem Informasi Univrab*, 6(2), 122–129. <https://doi.org/10.36341/rabit.v6i2.1765>
- Liu, L., Chen, G., Yu, F., Du, C., Gong, Y., Yuan, H., & Dai, Z. (2023). Research on fault diagnosis method of electromechanical transmission system based on one-dimensional convolutional neural network with variable learning rate. *Journal of Vibroengineering*, 25(5), 873–894. <https://doi.org/10.21595/jve.2023.22973>
- Mas, R., Panca, R. W., Atmaja1, K., & Yustanti2, W. (2021). Analisis Sentimen Customer Review Aplikasi Ruang Guru dengan Metode BERT (Bidirectional Encoder Representations from Transformers). *Jeisbi*, 02(3), 55–62.
- Mehta, P., & Pandya, S. (2020). A review on sentiment analysis methodologies, practices and applications. *International Journal of Scientific and Technology Research*, 9(2), 601–609.
- Nurjanah, W. E., Setya Perdana, R., & Fauzi, M. A. (2017). Analisis sentimen terhadap tayangan televisi berdasarkan opini masyarakat pada media sosial twitter menggunakan metode k-kearest neighbor dan pembobotan jumlah retweet. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 1(12),

- 1750–1757. <http://j-ptiik.ub.ac.id>
- Putranti, N. D., & Winarko, E. (2014). Analisis Sentimen Twitter untuk Teks Berbahasa Indonesia dengan Maximum Entropy dan Support Vector Machine. *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)*, 8(1), 91. <https://doi.org/10.22146/ijccs.3499>
- Rahate, S., Dehanka, V., Teppalwar, T., & Surjuse, V. R. (2022). Review Sentimental Analysis. *International Journal of Computer Science and Mobile Computing*, 11(3), 37–41. <https://doi.org/10.47760/ijcsmc.2022.v11i03.005>
- Rahman, D. A. & A. (2020). Pendeteksian Sarkasme pada Proses Analisis Sentimen Menggunakan Random Forest Classifier. *Jurnal Komputasi*, 8(2), 50–58. <https://doi.org/10.23960/komputasi.v8i2.2615>
- Wardani, A. P. P., Adiwijaya, A., & Purbolaksono, M. D. (2022). Sentiment Analysis on Beauty Product Review Using Modified Balanced Random Forest Method and Chi-Square. *Journal of Information System Research (JOSH)*, 4(1), 1–7. <https://doi.org/10.47065/josh.v4i1.2047>
- Yansens, C. (n.d.). *Apa itu Algoritma Smith dan Keunggulannya dari Algoritma Google BERT?*. Tanyadigital.Com. https://doi.org/https://tanyadigital.com/apa-itu-algoritma-smith-dan-keunggulannya-dari-algoritma-google-bert/#google_vignette
- Zukhrufillah, I. (2018). Gejala Media Sosial Twitter Sebagai Media Sosial Alternatif. *Al-I'lam: Jurnal Komunikasi Dan Penyiaran Islam*, 1(2), 102. <https://doi.org/10.31764/jail.v1i2.235>

LAMPIRAN

Lampiran *Listing* Program

1. *Scraping Twitter*

```
#Twitter Auth Token
twitter_auth_token = ''

# Untuk Menginstal phyton
!pip install pandas

# Install Node.js (tweet harvest untuk mengkstraksi data twitter dengan menginstall
node.js)
!sudo apt-get update
!sudo apt-get install -y ca-certificates curl gnupg
!sudo mkdir -p /etc/apt/keyrings
!curl -fsSL https://deb.nodesource.com/gpgkey/nodesource-repo.gpg.key | sudo gpg --
dearmor -o /etc/apt/keyrings/nodesource.gpg

!NODE_MAJOR=20 && echo "deb [signed-by=/etc/apt/keyrings/nodesource.gpg]
https://deb.nodesource.com/node_${NODE_MAJOR}.x nodistro main" | sudo tee
/etc/apt/sources.list.d/nodesource.list

!sudo apt-get update
!sudo apt-get install nodejs -y

!node -v

filename = 'msglowwhitening.csv'
search_keyword = 'ms glow whitening lang:id'
limit = 1000
!npx --yes tweet-harvest@2.2.8 -o "{filename}" -s "{search_keyword}" -l {limit} --
token {twitter_auth_token}
```

```

import pandas as pd

# Specify the path to your CSV file
file_path = f'tweets-data/{filename}'

# Read the CSV file into a pandas DataFrame
df = pd.read_csv(file_path, delimiter=";")

# Display the DataFrame
display(df)

# Simpan ke csv
tweets_df.to_csv(NAMA_FILE_CSV, index=False)

```

2. Pre-Processing

```

# Fungsi case folding
def casefolding(review):
    review_casefolded = review.lower()
    return review_casefolded

# Fungsi filtering
def filtering(review):
    review_filtered = re.sub(r'http\S+', '', review) # Remove web links
    review_filtered = re.sub('@[\s]+', '', review_filtered) # Remove @username
    review_filtered = re.sub(r'#([\s]+)', '', review_filtered) # Remove #tagger
    review_filtered = re.sub(r'[0-9]+', '', review_filtered) # Remove numbers including
    numbers within strings
    review_filtered = re.sub(r'^[\x00-\x7f]', '', review_filtered) # Remove non-ASCII
    characters
    review_filtered = re.sub(r'(\u[0-9A-Fa-f]+)', '', review_filtered)
    review_filtered = re.sub(r'^A-Za-z0-9^,!\./'+-='', '', review_filtered)
    review_filtered = re.sub(r'\u\w\w\w\w', '', review_filtered)

```

```

review_filtered = re.sub(r"[!-
|'\"$%&'\(\)|*|+|,|\.|\/|:|;|<|=|>|?|[\|\\|]\^|_|`|{|\||}|~|đŸ~£|©|_|-|¸|¶|»|¿|~]", " ",
review_filtered) # Remove symbols, strange characters, and odd characters

return review_filtered

# Fungsi untuk tokenisasi
def tokenize(review):
    tokens = nltk.word_tokenize(review)
    return tokens

# Fungsi untuk mengatasi negasi
def negation(review, negation_words=['tak', 'tidak', 'bukan', 'jangan', 'belum']):
    words = review.split()
    negated_words = []
    for word in words:
        if any(neg_word in word.lower() for neg_word in negation_words):
            word = f"{word}_"
            negated_words.append(word)
    return ' '.join(negated_words)

def removeStopword(review):
    additional_stop = ['wts', 'deh', 'dia']
    stopwords = pd.read_csv(stopword_path, names=["stopword"],
header=None)["stopword"].tolist()

    # Gabungkan stopwords dengan kata-kata tambahan (additional)
    stopwords += additional_stop

    content = []
    filteredtext = [word for word in review.split() if word.lower() not in stopwords]
    content.append(" ".join(filteredtext))
    review = content[0] # Ambil elemen pertama dari list content
    return review

```

```

def stemText(review):
    factory = StemmerFactory()
    stemmer = factory.create_stemmer()
    stemmed_review = stemmer.stem(review)
    return stemmed_review

def normalize_text(text):
    kamus_slangword = { }

    with open(kamus_path, "r") as file:
        reader = csv.reader(file)
        for row in reader:
            kamus_slangword[row[0]] = row[1]

    normalized_text = []
    for word in text.split():
        normalized_word = kamus_slangword.get(word.lower(), word)
        normalized_text.append(normalized_word)

    normalized_text = " ".join(normalized_text)
    return normalized_text

# Path ke file dan membaca DataFrame dari file TSV
df_path = 'ayu.tsv'

df = pd.read_csv(df_path, sep='\t', names=['review_text', 'category'])

# Case Folding
df['case_folded_text'] = df['review_text'].apply(casefolding)

# Filtering
df['filtered_text'] = df['case_folded_text'].apply(filtering)

```



```
# Tokenisasi
df['tokenized_text'] = df['filtered_text'].apply(tokenize)

# Negation Handling
df['negation_text'] = df['filtered_text'].apply(negation)

# Remove Stopword
df['stopword_removed_text'] = df['negation_text'].apply(removeStopword)

# Stemming
df['stemmed_text'] = df['stopword_removed_text'].apply(stemText)

# Normalization
df['normalized_text'] = df['stemmed_text'].apply(normalize_text)

# Menampilkan hasil tiap tahapan preprocessing
print("Hasil Case Folding:")
print(df['case_folded_text'])

print("Hasil Filtering:")
print(df['filtered_text'])

print("Hasil Tokenize:")
print(df['tokenized_text'])

print("Hasil Negation:")
print(df['negation_text'])
print("Hasil stopwords:")
print(df['stopword_removed_text'])

print("Hasil Stemming:")
print(df['stemmed_text'])
```

```
# Menampilkan hasil normalisasi
print("Teks Normalisasi:")
print(df['normalized_text'])
```

3. *Split Dataset*

```
from sklearn.model_selection import train_test_split

# Memisahkan DataFrame df_prepos menjadi train set dan val_test set
train_set, val_test_set = train_test_split(df_prepos, test_size=0.4,
stratify=df_prepos['category'], random_state=42)

# Memisahkan val_test set menjadi validation set dan test set
val_set, test_set = train_test_split(val_test_set, test_size=0.4,
stratify=val_test_set['category'], random_state=42)
```

4. *Set Random Seed*

```
# Fungsi Umum yang dibuat dalam melakukan pemodelan BERT

def set_seed(seed):
    # fungsi random yang menghasilkan nilai acak atau random
    random.seed(seed)
    np.random.seed(seed)
    torch.manual_seed(seed)
    torch.cuda.manual_seed(seed)

def count_param(module, trainable=False):
    if trainable:
        return sum(p.numel() for p in module.parameters() if p.requires_grad)
    else:
        return sum(p.numel() for p in module.parameters())

def get_lr(optimizer):
    for param_group in optimizer.param_groups:
```

```

    return param_group['lr']

def metrics_to_string(metric_dict):
    string_list = []
    for key, value in metric_dict.items():
        string_list.append('{ }:{:.2f}'.format(key, value))
    return ' '.join(string_list)

```

5. BERT

```

from torch import nn
from transformers import BertForSequenceClassification, BertConfig
# Load Tokenizer and Config
tokenizer = BertTokenizer.from_pretrained('indobenchmark/indobert-base-p1')
config = BertConfig.from_pretrained('indobenchmark/indobert-base-p1')
config.num_labels = DocumentSentimentDataset.NUM_LABELS

# Instantiate model
model = BertForSequenceClassification.from_pretrained('indobenchmark/indobert-
base-p1', config=config)
model.dropout = nn.Dropout(0.3)

```

6. *Training dan evaluasi*

```

# Tentukan optimizer
import torch
from torch import nn
import torch.optim as optim

optimizer = optim.AdamW(model.parameters(), lr=5e-6, weight_decay=1e-4)
model = model.cuda()

# Train
n_epochs = 5
history = defaultdict(list)
for epoch in range(n_epochs):

```

```

model.train()
torch.set_grad_enabled(True)

total_train_loss = 0
list_hyp_train, list_label = [], []

train_pbar = tqdm(train_loader, leave=True, total=len(train_loader))
for i, batch_data in enumerate(train_pbar):
    # Forward model
    loss, batch_hyp, batch_label = forward_sequence_classification(model,
batch_data[:-1], i2w=i2w, device='cuda')

    # Update model
    optimizer.zero_grad()
    loss.backward()
    optimizer.step()

    tr_loss = loss.item()
    total_train_loss = total_train_loss + tr_loss

    # Hitung skor train metrics
    list_hyp_train += batch_hyp
    list_label += batch_label

    train_pbar.set_description("(Epoch {}) TRAIN LOSS:{:.4f}
LR:{:.8f}".format((epoch+1),
    total_train_loss/(i+1), get_lr(optimizer)))

metrics = document_sentiment_metrics_fn(list_hyp_train, list_label)
print("(Epoch {}) TRAIN LOSS:{:.4f} {} LR:{:.8f}".format((epoch+1),
    total_train_loss/(i+1), metrics_to_string(metrics), get_lr(optimizer)))

# save train acc for learning curve

```

```

history['train_acc'].append(metrics['ACC'])

# Evaluate di validation set
model.eval()
torch.set_grad_enabled(False)

total_loss, total_correct, total_labels = 0, 0, 0
list_hyp, list_label = [], []

pbar = tqdm(valid_loader, leave=True, total=len(valid_loader))
for i, batch_data in enumerate(pbar):
    batch_seq = batch_data[-1]
    loss, batch_hyp, batch_label = forward_sequence_classification(model,
batch_data[:-1], i2w=i2w, device='cuda')

    # Hitung total loss
    valid_loss = loss.item()
    total_loss = total_loss + valid_loss

    # Hitung skor evaluation metrics
    list_hyp += batch_hyp
    list_label += batch_label
    metrics = document_sentiment_metrics_fn(list_hyp, list_label)

    pbar.set_description("VALID LOSS:{:.4f} {}".format(total_loss/(i+1),
metrics_to_string(metrics)))

    metrics = document_sentiment_metrics_fn(list_hyp, list_label)
    print("(Epoch {}) VALID LOSS:{:.4f} {}".format((epoch+1),
total_loss/(i+1), metrics_to_string(metrics)))

# save validation acc for learning curve
history['val_acc'].append(metrics['ACC'])

```

```

plt.plot(history['train_acc'], label='train acc')
plt.plot(history['val_acc'], label='validation acc')
plt.title('Training history')
plt.ylabel('acc')
plt.xlabel('Epoch')
plt.legend()
plt.ylim([0, 1]);
# Simpan Hasil Prediksi Validation Set
val_df = pd.read_csv(valid_dataset_path, sep='\t', names=['review_text', 'category'])
val_df['pred'] = list_hyp
val_df.head()
val_df.to_csv('val result.csv', index=False)
# Prediksi test set
model.eval()
torch.set_grad_enabled(False)

total_loss, total_correct, total_labels = 0, 0, 0
pred, list_label = [], []

pbar = tqdm(test_loader, leave=True, total=len(test_loader))
for i, batch_data in enumerate(pbar):
    _, batch_hyp, _ = forward_sequence_classification(model, batch_data[:-1],
i2w=i2w, device='cuda')
    pred += batch_hyp
    # Simpan prediksi test set
test_df = pd.read_csv(test_dataset_path, sep='\t', names=['review_text', 'category'])
test_df['pred'] = pred
test_df.head()
test_df.to_csv('test result.csv', index=False)
val_real = val_df.category
val_pred = val_df.pred
test_real = test_df.category
test_pred = test_df.pred

```

```

def show_confusion_matrix(confusion_matrix):
    hmap = sns.heatmap(confusion_matrix, annot=True, fmt="d", cmap="Blues")
    hmap.yaxis.set_ticklabels(hmap.yaxis.get_ticklabels(), rotation=0, ha='right')
    hmap.xaxis.set_ticklabels(hmap.xaxis.get_ticklabels(), rotation=30, ha='right')
    plt.ylabel('True sentiment')
    plt.xlabel('Predicted sentiment');

cm = confusion_matrix(val_real, val_pred)
df_cm = pd.DataFrame(cm, index=['positive', 'neutral', 'negative'], columns=['positive',
'neutral', 'negative'])
show_confusion_matrix(df_cm)
print(classification_report(val_real, val_pred, target_names=['positive', 'neutral',
'negative']))

def show_confusion_matrix(confusion_matrix):
    hmap = sns.heatmap(confusion_matrix, annot=True, fmt="d", cmap="Blues")
    hmap.yaxis.set_ticklabels(hmap.yaxis.get_ticklabels(), rotation=0, ha='right')
    hmap.xaxis.set_ticklabels(hmap.xaxis.get_ticklabels(), rotation=30, ha='right')
    plt.ylabel('True sentiment')
    plt.xlabel('Predicted sentiment');

cm = confusion_matrix(test_real, test_pred)
df_cm = pd.DataFrame(cm, index=['positive', 'neutral', 'negative'], columns=['positive',
'neutral', 'negative'])
show_confusion_matrix(df_cm)
print(classification_report(test_real, test_pred, target_names=['positive', 'neutral',
'negative']))

```