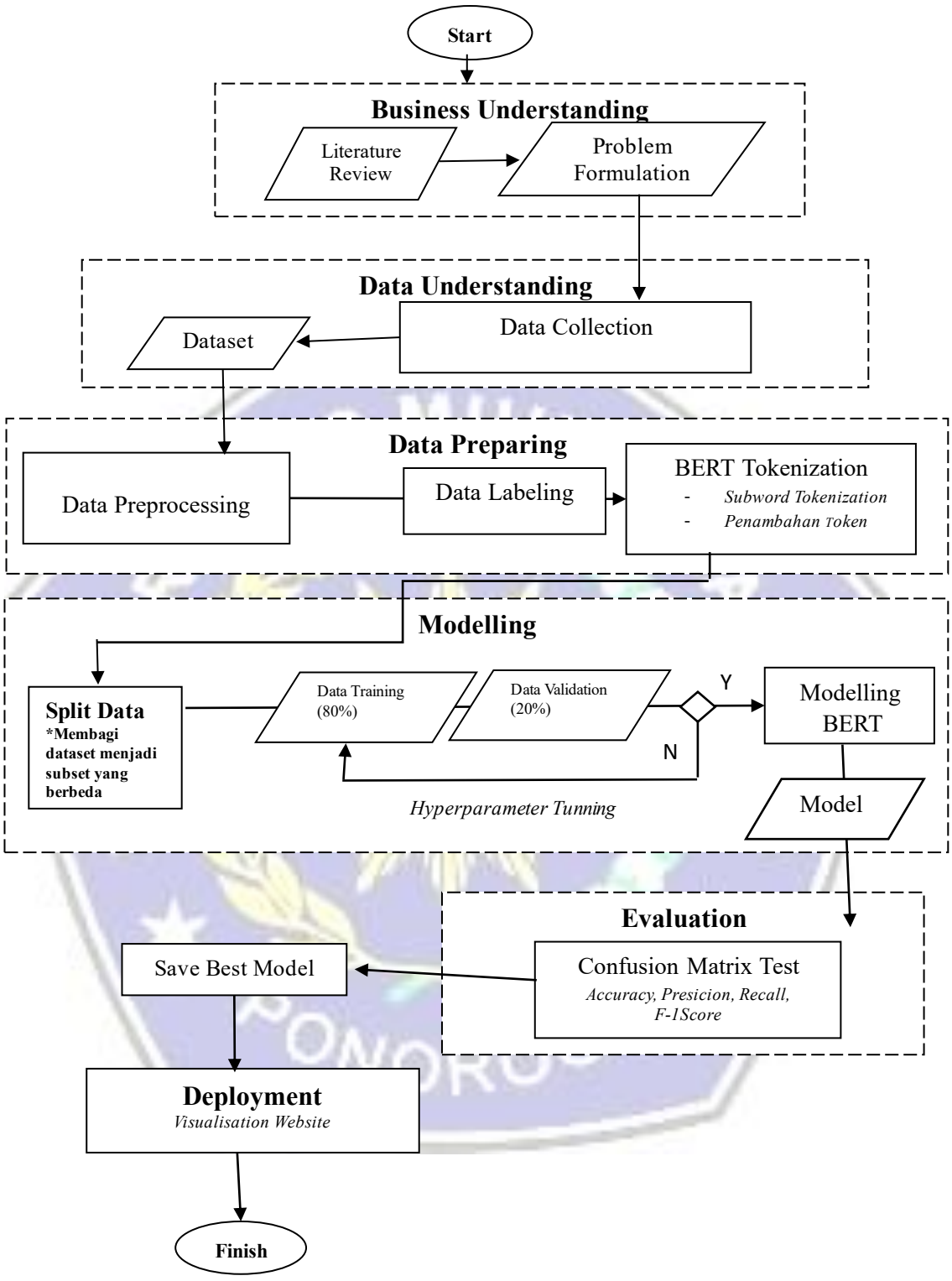
BAB III
METODE PENELITIAN

Dalam penelitian ini menerapkan metodologi CRISP-DM (*Cross-Industry Standard Process for Data Mining*). Menerapkan CRISP-DM dalam proyek analisis sentimen untuk penerapan *E-Parking* "PARKIR-GO" oleh Pemerintah Kabupaten Ponorogo dengan algoritma BERT menjadi pilihan peneliti karena menyediakan kerangka kerja terstruktur dan iteratif yang memastikan setiap tahap proyek dilaksanakan dengan hati-hati dan sistematis. CRISP-DM dimulai dengan pemahaman bisnis yang mendalam untuk memastikan relevansi hasil analisis, dilanjutkan dengan persiapan data yang komprehensif untuk menangani data tidak terstruktur dari media sosial.

Tahap pemodelan memungkinkan penggunaan dan fine-tuning BERT untuk analisis sentimen yang akurat, sementara evaluasi menyeluruh memastikan model memenuhi kriteria performa yang diinginkan. Pada tahap terakhir adalah tahap *deployment* dan *monitoring* memastikan model dapat digunakan secara efektif dan diperbarui sesuai kebutuhan, memberikan wawasan berharga bagi keputusan pemerintah. Kerangka kerja yang akan dilaksanakan dijelakan pada Gambar 3.1 Diagram Alur Metodologi CRISP-DM sebagai berikut**.**

.



Gambar 3. 1 Diagram Alur Metodologi Penelitian

3.1. Business Understanding

Pada tahapan ini dilakukan beberapa langkah untuk mendukung dan menjadi salah satu fondasi penelitian. Beberapa tahapan yang dilalui sebagai berikut:

- *Literature Review* (Tinjauan Pustaka)

  Penliti melakukan pengumpulan data dan informasi terkait berdasarkan jurnal ilmiah, buku, artikel, website dan sumber akurat lainnya. Setelah mengumpulkan beberapa referensi terkait dilakukan pengembangan ide berdasarkan masalah yang diamati di sosial media terkait penerapan *E-Parking* Kabupaten Ponorogo dan melakukan perbandingan untuk mendapatkan landasan teori pada penelitian. Seperti yang telah di paparkan pada Tabel 2.1 litertur yang dijadikan landasan memiliki topik serupa yakni sentimen analisis dengan metode NLP terkhusus metode BERT di berbagai macam sektor.

- *Problem Formulation* (Identifikasi Masalah)

  Setelah mengumpulkan landasan teori dan informasi dari beberapa sumber, peneliti kemudian merumuskan permasalahan yang dijadikan landasan penelitian ini dengan mempelajari penelitian terdahulu Tabel 2.1. yang berkaitan dengan BERT. Penelitian Rhini Fatmasaria,dkk(2023) dilakukan analisa terkait komentar sosial media twitter dan tiktok sebanyak 658 data hampir sama dengan jumlah sample yang dimiliki pada penelitian ini yakni sekitar 450 data menggunakan beberapa teknik yang diuji menunjukkan hasil BERT dengan akurasi sebesar 90% lebih tinggi dibanding yang lain[3]. Penelitian Nanang H (2023) menunjukkan algoritma BERT mencapai akurasi *training* sebesar 93% dan akurasi testing sebesar 92%, dengan marco avg dari f1 score sebesar 92%[2]. Dengan demikian, algoritma BERT terbukti efektif dalam mengklasifikasikan teks artikel berita, terutama dalam kasus dataset yang cukup besar dan tidak seimbang dibanding metode random forest yang memiliki akurasi *training* 81% dan Naïve Baiyes 78%. Kesimpulan dari beberapa penelitian tersebut menunjukan kemampuan yang mumpuni dari BERT dalam menganalisis

teks yang kompleks dalam analisis sentimen. Penelitian ini merumuskan masalah pada model BERT dengan hasil akhir sentimen analisis penerapan *E-Parking* Kabupaten Ponorogo dan tingkat akurasi dari model ini. Data pada penelitian ini adalah komentar sosial media dengan topik *E-Parking* Ponorogo pada jangka waktu januari-mei 2023. Pada penelitian ini dibutuhkan perencanaan kebutuhan seperti berikut.

a. Waktu Penelitian

Penelitian akan melewati beberapa proses sehingga perlu ditetapkan waktupenelitian diterangkan pada Tabel 3.1

Tabel 3. 1 Waktu Penelitian

Tahapan	*Tools*	*Objek*
Data Collection	*Apify IGCommentExport*	*Konten dengan topik E-Parking*
Data Preparation	*Python*	*Dataset*
Modeling	*Python*	*Dataset*

b. Perangkat Penelitian

Perangkat yang digunakan dalam proses pengujian penelitian ini sebagai berikut:

1) Data dikumpulkan dan diolah dengan menggunakan laptop berspesifikasi berikut:
   - Proccessor : Intel Core i3 10th Gen
   - RAM : 4 GB,
   - Storage : SSD 163 GB
2) Perangkat diatas dapat memanfaatkan layanan cloud gratis Google yakni Google Colaboratory dengan GPU yang tersedia sebagai akselerator perangkat.
3) Perangkat lunak yang digunakan dalam penelitian ini yakni sistem operasi window 11 64-bit, google colab, python, micrisoft edge.

3.2. Data Understanding

Setelah melalui tahapan bussines understanding langkah yang selanjutnya adalah data understanding atau memahami data yang akan digunakan. Pengumpulan data yang dilakukan menerapkan teknik web-scraping otomatis dengan tools IGCommentsExport dan Apify dari postingan akun terkait Tabel 3.2. Selanjutnya hasil scraping disusun menjadi dataset dengan format .csv ditampilkan pada Gambar 3.2 yang kemudian akan memasuki proses *data preparation* dengan *library* python seperti pandas. Berikut rincian pengumpulan data dari sosial media.

User Id	Username	Comment Id	Comment Text	Profile URL	Avatar URL	Date
3211367537	tatik_mujiarti	1.80731E+16	kalo bisa di persulit kenapa harus di pe	https://www.instagram.com/tatik_	https://scontent-cgk1-2.c	5/23/2023, 12:01:10 PM
47616636524	ramangabdulazis	1.7994E+16	Kampungan 😂	https://www.instagram.com/raman	https://instagram.ftir5-1.	5/23/2023, 12:03:20 PM
44103852692	marissasr_	1.78623E+16	@alistaavs bos berek	https://www.instagram.com/maris	https://scontent-cgk1-2.c	5/23/2023, 12:35:04 PM
1808322242	andika.dbh	1.80654E+16	Teknologi melebihi akal sehat, 2023 n	https://www.instagram.com/andik	https://scontent-cgk1-2.c	5/23/2023, 12:42:25 PM
2135889890	ragiel_mangku_l	1.79938E+16	Mugo ae nek khilangan di tmpt parkir	https://www.instagram.com/ragiel	https://scontent-cgk1-2.c	5/23/2023, 12:55:26 PM
9180409937	arin9499	1.79993E+16	Maleh gk minat dolan rno min . @por	https://www.instagram.com/arin94	https://scontent-cgk1-2.c	5/23/2023, 12:57:14 PM
43635174987	herysusantu	1.80119E+16	Ngurusi urusan sing gak urgent	https://www.instagram.com/herysu	https://scontent-cgk1-2.c	5/23/2023, 1:01:00 PM
5923339307	nadflix.jv	1.79666E+16	r ndwe e money	https://www.instagram.com/nadfli	https://scontent-cgk1-2.c	5/23/2023, 1:21:19 PM
7180642594	optik_tasikmalay	1.79891E+16	Lama lama kembali lagi ke manual	https://www.instagram.com/optik_	https://scontent-cgk1-2.c	5/23/2023, 1:45:06 PM
1654627163	erikwahyudy	1.78854E+16	Semangatt 😍	https://www.instagram.com/erikw	https://scontent-cgk1-2.c	5/23/2023, 1:45:26 PM
1781213145	eline_xiandhia	1.79868E+16	@yogiipramito nah tu dia.... 💧💧💧	https://www.instagram.com/eline_	https://scontent-cgk1-2.c	5/23/2023, 2:39:51 PM
1445963182	hanifahlayli	1.80127E+16	Aku luweh ikhlas duit sewu rongewu r	https://www.instagram.com/hanifa	https://scontent-cgk1-2.c	5/23/2023, 2:43:30 PM
2066042336	zazux_yeye	1.7967E+16	Pendapatku tetep penak parkir gratis.	https://www.instagram.com/zazux_	https://scontent-cgk1-2.c	5/23/2023, 3:08:36 PM
52763097857	dikaambarani_ba	1.7956E+16	Alhamdulilah Semangat Gak Belum B	https://www.instagram.com/dikaa	https://scontent-cgk1-2.c	5/23/2023, 4:07:39 PM
57408281609	adityaanakbaik_	1.80601E+16	Nyuen nyuen i lakon	https://www.instagram.com/aditya	https://scontent-cgk1-2.c	5/23/2023, 4:17:01 PM
6185169121	humairasyaharan	1.79826E+16	Pendapat ku apik lor mergo ben ora h	https://www.instagram.com/humai	https://scontent-cgk1-2.c	5/23/2023, 4:18:37 PM

Gambar 3. 2 Dataset Hasil Scrapping

Tabel 3. 2 Data Collection

Sosial Media	*Akun*	*Jumlah Konten terkait*	*Komentar Diambil*
Instagram	*@infoponorgo*	*2*	*200*
	@ponorogoupdate	*3*	*267*
	@ponorogopictures	*1*	*17*
Facebook	*Gemasurya*	*2*	*22*
Youtube	*KompasTV*	*1*	*32*
Total Data Komentar			*538*

3.3. Data Preparation

Tahapan ini merupakan tahap penting dalam penelitian karena mengubah data yang mentah menjadi data yang siap untuk dipakai dalam tahap *modelling*.

3.3.1 Data Preproccessing

Proses pembersihan data dilakukan agar dataset yang dimiliki menjadi data yang bersih dan akurat untuk proses analisa. Data yang bersih dan berkualitas akan mendukung pengambilan keputusan yang lebih baik dan akurat. Beberapa proses dilalui untuk mendapatkan data yang bersih sebagai bahan modeling di tahap selanjutnya.

1. Pembersihan Data Duplikat

   Proses ini menghilangkan data yang sama persis karena proses penggabungan data saat berada di tahap scraping. Metode yang dipakai yakni drop_duplicates() dengan python. Setelah itu dilakukan pemilihan kolom yang akan digunakan yakni kolom text karena berisi data komentar.

2. Pembersihan simbol dan *noise* (#, @, emoticon, simbol non alfabetik, dan spasi awal dan akhir kalimat)

   Proses ini memanfaatkan modul *regular expression*(re) yang dimiliki python. Modul ini menyediakan fungsi-fungsi yang mendukung penggunaan regular expression dalam pengolahan teks. Berikut contoh hasil proses ini.

Tabel 3. 3 Data Cleansing

Sebelum	*Sesudah*
text	text
0 Sip...buat ngurangi pungli	0 sipbuat ngurangi pungli
1 @ricardo_kneff kalau di jakarta e parking bias...	1 kalau di jakarta e parking biasanya pakai e mo...
2 Bayar pakek gopay,dana opo sopipay?	2 bayar pakek gopaydana opo sopipay
3 Pembayaran paling gampang Jane gae Qris	3 pembayaran paling gampang jane gae qris
4 Kalau gak di mulai tdk akan pernah mulai.. bis...	4 kalau gak di mulai tdk akan pernah mulai bismi...
... ...	... ...
441 Pokok aku padamu pak/bu semerintah, arep digaw...	441 pokok aku padamu pakbu semerintah arep digawe ...
442 Siji d kon ngno... Liane do ngaleh...	442 siji d kon ngno liane do ngaleh

3. Case Folding

   Setelah melewati data cleansing seperti Tabel 3.3, proses pengubahan huruf menjadi huruf kecil untuk membantu dalam membuat

teks lebih konsisten dengan menghilangkan variasi yang disebabkan oleh perbedaan kapitalisasi. Misalnya, kata "Hello" dan "hello" dianggap sama setelah *lowercase*, yang mengurangi kompleksitas dan ukuran kosakata yang perlu dipelajari oleh model. Proses ini juga menggunakan metode *text.lower()* untuk mengubah semua karakter dalam *string text* menjadi huruf kecil atau *lowercase* seperti pada Tabel 3.4 dibawah ini.

Tabel 3. 4 Case Folding

Sebelum diproses	***Lowercase***
Program Yang Harus Didukung Karena Positif	*program yang harus didukung karena positif*

4. Slang word Removal

Data yang sebelumnya kemudian di proses untuk menghilangkan kata-kata slang yang ada. Kamus yang dipakai dalam proses ini berasal dari https://github.com/nasalsabila/kamus-alay.git yang mengandung 3592 kumpulan kata.

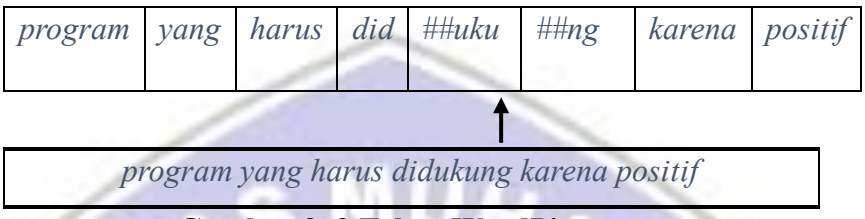
3.3.2 Data Labelling

Tahap *labelling* data menerapkan metode *lexicon* dengan kamus sentimen bahasa indonesia InSet dengan memberi bobot -5 hingga 5. Selanjutnya dikategorikan menjadi tiga sentimen yakni apabila bobot dibawah 0 maka negatif, lebih dari 0 hingga 5 positif, dan lainnya dilabeli dengan sentimen netral.

3.3.3 BERT Tokenization

Langkah dalam representasi *input* di BERT sebagai berikut:

1. Tokenisasi *Wordpiece* menjadi subkata. Beberapa sub kata dapat diawali dengan simbol ## sebagai penanda token tersebut adalah sufiks dan diikuti subkata lain.

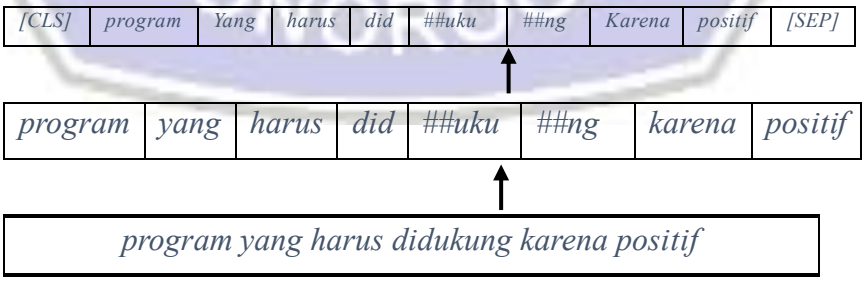


Gambar 3. 3 Token WordPiece

Seperti pada **Gambar 3.3** kalimat dibagi menjadi sub kata yang awalnya "program yang harus didukung karena positif" menjadi "program" "yang" "harus" "did" "##uku" "##ng" "karena" "positif". Kata didukung dipisah menjadi "did" "##uku" "##ng" karena tidak terdapat dalam *vocabulary* dari model pre-*trained* bert-base-*multilingual*.

2. Token Embedding

Setiap kalimat akan diberikan token khusus berupa [CLS] diawal dan akan digunakan pada proses klasisikasi sentimen karena token ini melakukan pengumpulan rata-rata token untuk mendapat vektor dari kalimat. Kemudian token [SEP] sebagai pemisah dengan kalimat selanjutnya.



Gambar 3. 4 Token Embedding

Pada Gambar 3.4. merupakan proses *embedding* dengan penambahan token pada kalimat yang sudah dipisah menjadi sub kata sesuai BERT *tokenize*.

3. Token Padding

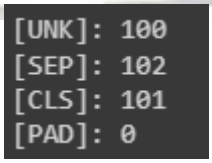
Karena dalam BERT panjang *input* harus dibuat sama, token [PAD] dapat difungsikan sebagai penambah *input* agar sesuai panjang yang ditentukan. Misalkan panjang maksimal 125 maka pada kalimat contoh membutuhkan token [PAD] hingga memenuhi panjangnya.

Gambar 3. 5 Token Padding

Proses ini dilakukan untuk penyesuaian panjang *input* , contoh pada Gambar 3.5 Menambahkan token *padding* [PAD] untuk menambah panjang token.

4. Subtitusi ID

Proses ini didapatkan dari indeks kata dalam *vocabulary* pada model bert-base-*multilingual* seperti Gambar 3.6, misalkan token [UNK] memiliki id 100, [CLS] memiliki id 101, [SEP] memiliki Id 102 dan [PAD] memiliki id 0.



Gambar 3. 6 Indeks ID token BERT

Dan untuk kata lain yang muncul akan mendapatkan id yang telah dimuat dalam vocabulary model.

Gambar 3. 7 Tokenisasi ID BERT

Proses pada Gambar 3.7. mengubah token menjadi token numerik yang kemudian akan dipakai pada proses *input* klasifikasi *sentimen* BERT.

5. Sentence Embeding

Sebagai pembeda kalimat satu dengan lainnya proses ini akan memberi angka penanda yang sama di setiap sub katanya seperi Gambar 3.8.

[CLS]	program	yang	harus	did	##uku	##ng	karena	positif	[SEP]	[PAD]
1	1	1	1	1	1	1	1	1	1	0

Gambar 3. 8 Sentence Embeding

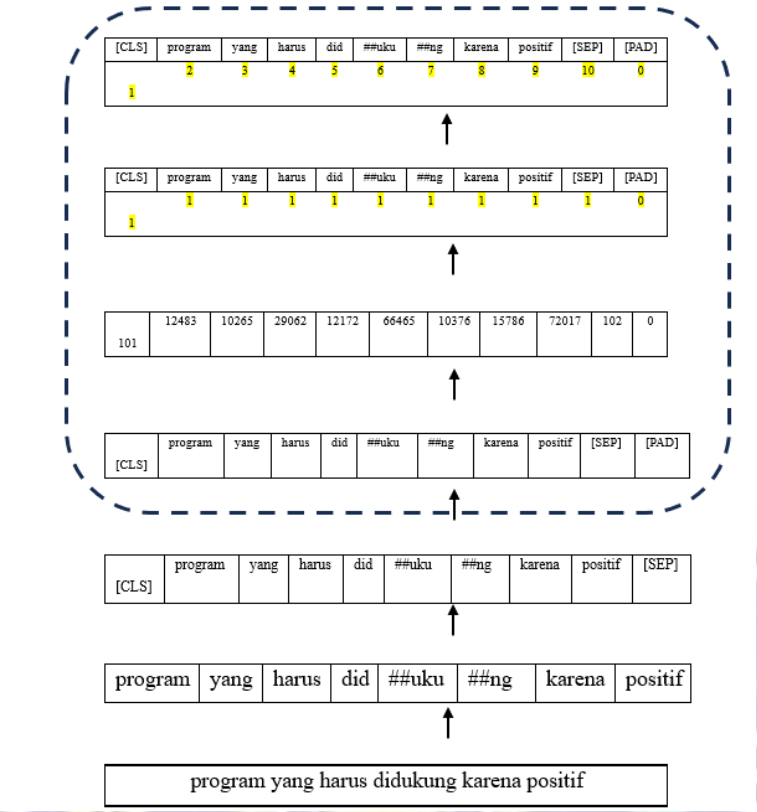
6. Positional embedding

Di dalam BERT diperlukan penomoran posisi tiap kata dalam kalimat agar dapat memahami makna dari kalimat. Pada Gambar 3. 9. menjelaskan tahap penomoran posisi masing-masing token dalam kalimat.

[CLS]	program	yang	harus	did	##uku	##ng	karena	positif	[SEP]	[PAD]
1	2	3	4	5	6	7	8	9	10	0

Gambar 3. 9 Positional Embeding

Pada Gambar 3.10. menunjukkan proses tokenisasi sebagai proses representasi *input*.



Gambar 3. 10 Proses Representasi *Input*

3.4. Modeling

3.4.1 Data Split

Data yang sudah dibersihkan dibagi menjadi data *train* untuk pelatihan model kemudian data *validation* untuk evaluasi. Dikarenakan data yang dimiliki cenderung kecil maka proposi data train lebih besar dengan perbandingan 80:20. Sebesar 80 persen data asli digunakan sebagai data train dan 20 persen sebagai data validasi. Pembagian kelas pada data *train* yakni 163 sampel kelas negatif(0), 156 sampel kelas positif(2) dan 100 sampel kelas netral(1). Pembagian kelas data validasi yakni 41 sampel kelas negatif(0), 38 sampel kelas positif(2), 26 sampel kelas netral(1).

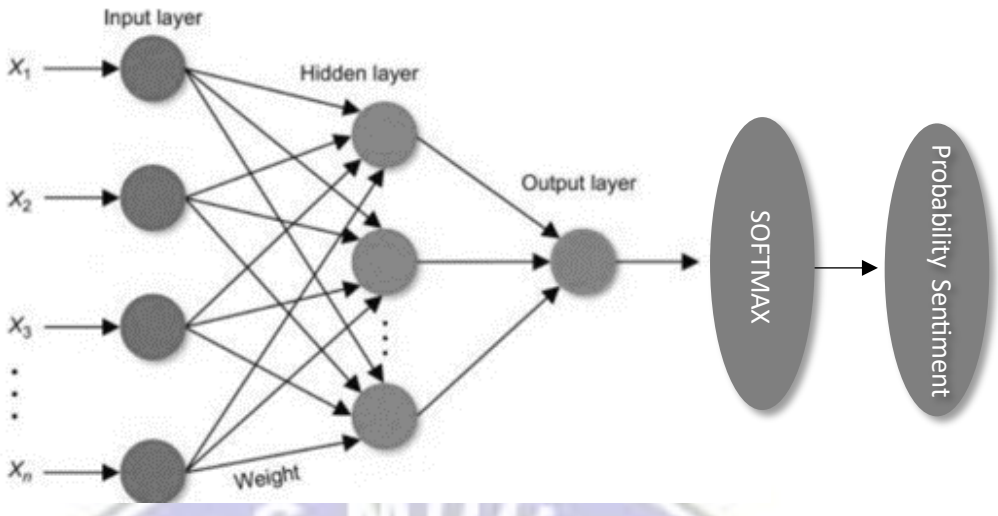
3.4.2 Load Pre-*Training* & Fine-Tuning BERT model

Pada BERT pelatihan tidak dilakukan dari awal tetapi memanfaatkan model yang telah dilatih sebelumnya(pre-trained) dengan data besar kemudian disesuaikan dengan sedikit pembelajaran untuk tugas baru atau disebut dengan teknik *fine-tune*.

3.4.3 Klasifikasi BERT

Pada model BERT ada beberapa proses yang dilalui mulai dari *input* representation untuk menyiapkan *input* an ke model hingga klasisifkasi. Peneliti pada penelitian ini akan menerapkan pre-trained bert-base-*multilingual* yang kemudian di *fine tuning*. Sebelumnya di tahap *data preparation* data sudah melalui proses sebagai representasi *input* model. BERT memiliki panjang maksimum kalimat *input* yakni 512 karena *encoder transformer* hanya mampu menghasilkan *output* dengan dimensi 512.

Setelah menjadi bentuk representasi *input* pada Gambar 3. 10 selanjutnya akan berlanjut ke proses klasifikasi sentimen. Pada layer model *output* akan menghasilkan vektor dari [CLS] yakni berupa logits. Logits ini yang akan diubah menjadi probability dengan metode aktivasi softmax. Probabilitas dengan *softmax* ini ketika dijumlah keseluruhan akan menjadi tepat 1 dan nilainya diantara 0 hingga 1.



Gambar 3. 11 Tahap Klasifikasi

Dari ilustrasi pada Gambar 3.11 *output* selanjutnya di proses dengan *softmax*. Berikut tahapan memperoleh nilai probabilitasnya:

- Menghitung eksponen elemen logits.

  Logits diperoleh dengan melakukan operasi linear pada *output* dari model sebelum diterapkan fungsi aktivasi. Secara matematis, ini dapat diwakili dengan persamaan (2.1)

  Misalkan hasil tokenisasi dan embedding untuk kalimat

  "program yang harus didukung karena positif"

  menghasilkan vektor representasi akhir X. Matriks bobot W akan memiliki dimensi yang sesuai dengan representasi akhir token [CLS] (misalnya, jika menggunakan BERT base, dimensi 768×3 untuk tiga kelas). Vektor bias b adalah vektor dengan panjang yang sama dengan jumlah kelas (misalnya, 3 untuk tiga kelas: positif, netral, negatif). Misalkan nilainya seperti berikut.

  X = [0.5, -0.3, 0.7, ...]

  W = [[w11, w12, w13], [w21, w22, w23], ..., [w768, w768, w768]]

  b = [0.1, -0.2, 0.3]

  Kemudian dihitung dengan persamaan (2.1).

  logit1 = X[0].W[0][0] + X[1].W[1][0] + ... + X[767].W[767][0] + b[0]

  logit2 = X[0].W[0][1] + X[1].W[1][1] + ... + X[767].W[767][1] + b[1]

logit3 = X[0].W[0][2] + X[1].W[1][2] + ... + X[767].W[767][2] + b[2]

Hasil logits adalah [logit1, logit2, logit3] atau $[Z_1, Z_2, Z_3]$

Misal hasil diatas adalah $Z = \begin{pmatrix} Z_1 \\ Z_2 \\ Z_3 \end{pmatrix} = \begin{pmatrix} 5 \\ 3 \\ 2 \end{pmatrix}$.

Selanjutnya untuk mendapatkan probabilitas langkah awalnya dengan menghitung eksponensial setiap elemen logits.

$e^{Z_1} = e^5 = 148.41$

$e^{Z_2} = e^3 = 20.09$

$e^{Z_3} = e^2 = 7.39$

- Menjumlahkan seluruh eksponen

  $\sum_{j=1}^{k} e^{z_j} = e^5 + e^3 + e^2 = 148.41 + 20.09 + 7.39 = 175.89$

- Mendapat probabilitas dengan *softmax* dengan persamaan (**2.2**)

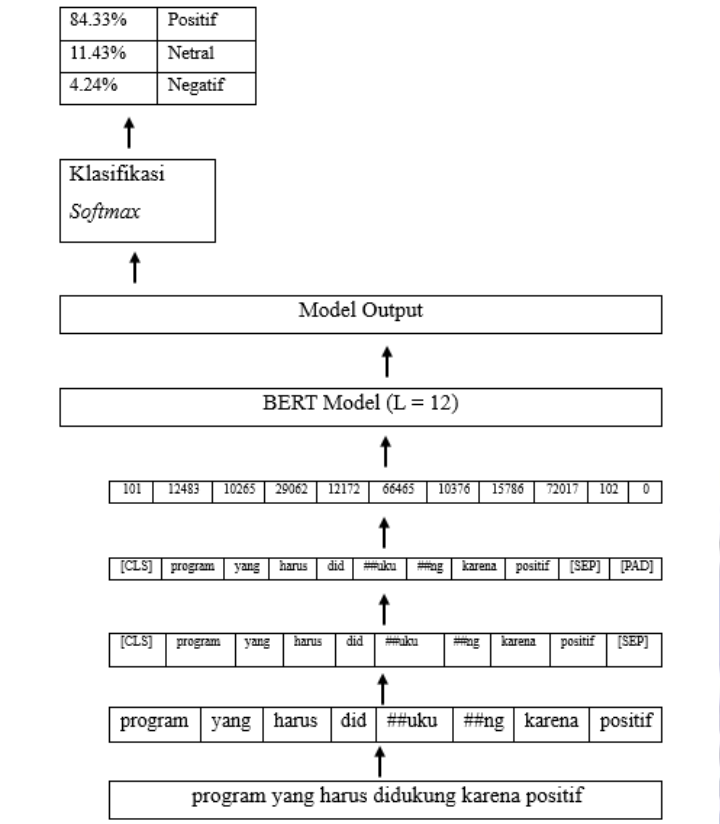
  $\text{Softmax}(z_1) = \frac{e^5}{\sum_{j=1}^{k} e^{z_j}} = \frac{148.41}{175.89} = 0.8433$

  $\text{Softmax}(z_2) = \frac{e^3}{\sum_{j=1}^{k} e^{z_j}} = \frac{20.09}{175.89} = 0.1143$

  $\text{Softmax}(z_3) = \frac{e^2}{\sum_{j=1}^{k} e^{z_j}} = \frac{7.39}{175.89} = 0.0424$

  Maka hasil probabilitas total 0.8433 + 0.1143 + 0.0424 = 1

  Pada contoh penerapan diatas maka *sentimen* dari kalimat tersebut dominan di kelas positif sebesar 0.8433. Gambar 3.12 menunjukkan ilustrasi klasifikasi sentimen BERT secara keseluruhan.



Gambar 3. 12 Ilustrasi Klasifikasi Sentimen BERT

Mekanisme di tahap modelling BERT Gambar 3.12 merupakan lapisan *encoder*berjumlah 12 karena pada penelitian ini menggunakan BERT base yang memiiki jumlah layer tersebut. Pada setiap layer *encoder*melakukan mekanisme *self-attention*seperti contoh mekanisme Gambar 2.3. Setelahnya di lapisan terakhir menghasilkan *output* untuk representasi teks dalam token [CLS] yang kemudian akan dilakukan klasifikasi dengan fungsi aktivasi *softmax* persamaan (2.2) untuk melihat probabilitas sentimen akhir masuk kelas apa.

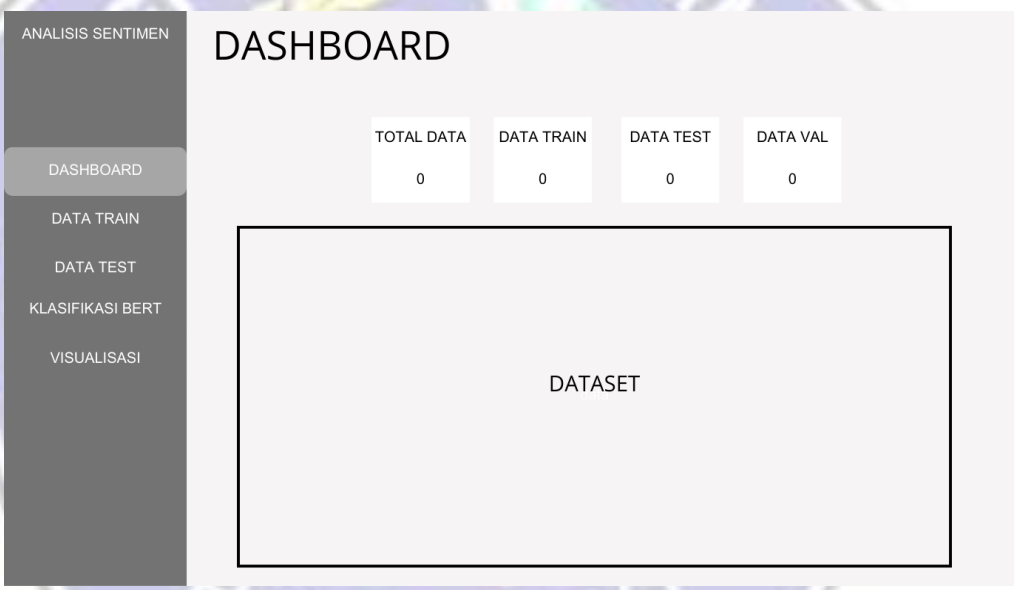
3.5 Evaluation

Evaluasi akurasi, presisi, *recall* dan F1-score adalah langkah penting dalam mengevaluasi performa model klasifikasi, terutama dalam konteks pemrosesan bahasa alami (NLP). Dengan menggunakan pesamaan (2.3)

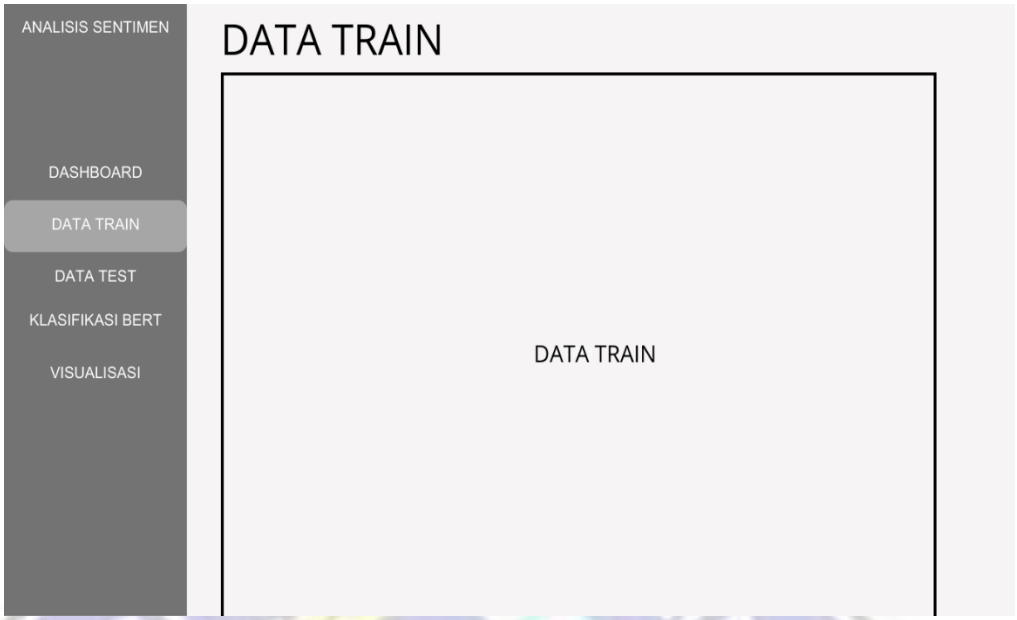
untuk akurasi, persamaan (2.4) untuk presisi, *recall* dengan persamaan (2.5) dan menghitung skor F1 dengan persamaan (2.6)

3.6 Deployment

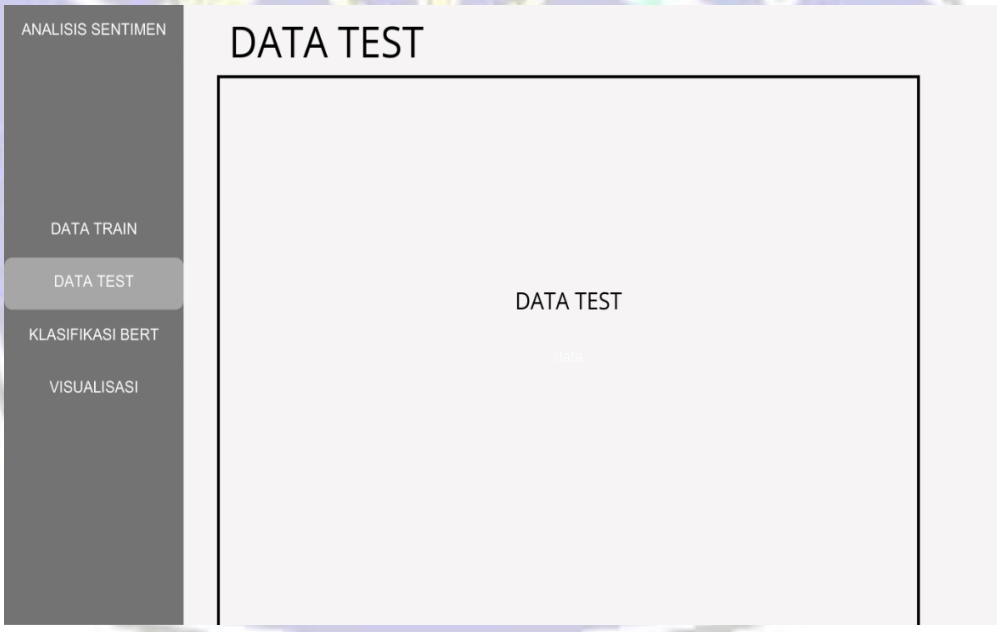
Tahap akhir dengan mengambil Kesimpulan dari hasil pemodelan menggunakan BERT. Selain merancancang GUI untuk melihat kemampuan analisis *sentimen* model disusun juga dashboard hasil analisis sentimen *E-Parking* berbasis website sederhana menggunakan *framework* Flask Python sebagai server side, library pandas untuk membaca dataset, kemudian HTML, CSS sebagai pondasi tampilan web. Berikut dashboard untuk menampilkan hasil analisis sentimen dengan pendekatan BERT pada Gambar 3.13., Gambar 3.14., Gambar 3.15., Gambar 3.16. dan Gambar 3.17



Gambar 3. 13 Desain Halaman Dashboard

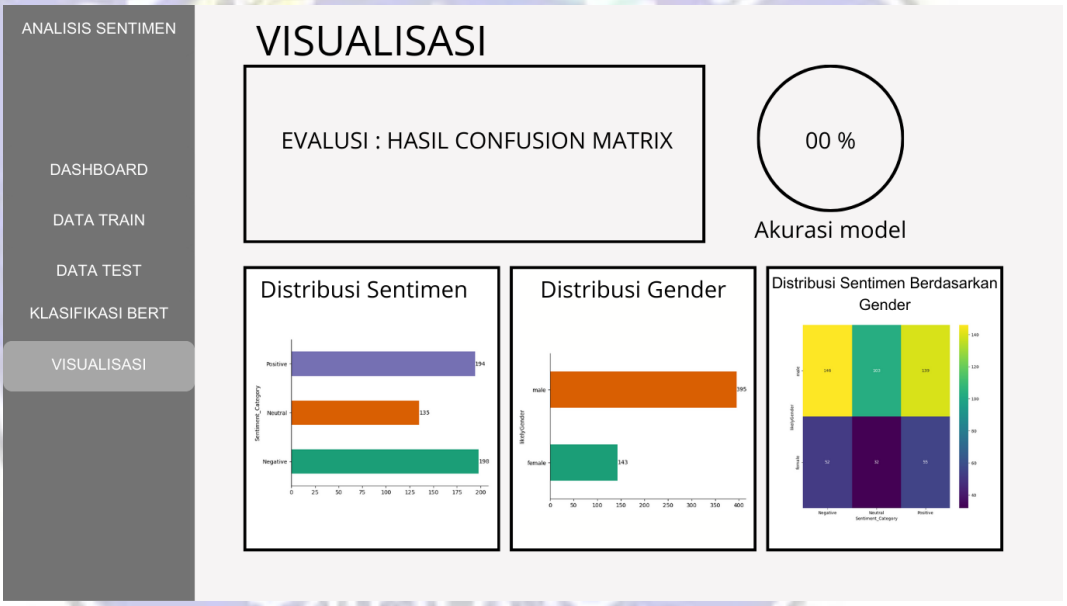


Gambar 3. 14 Desain Halaman Data Train



Gambar 3. 15 Desain Halaman Data Test

Gambar 3. 16 Desain Halaman Klasifikasi BERT



Gambar 3. 17 Halaman Visualisasi Data