# PENDAHULUAN

## Latar Belakang

## Tujuan

# LANDASAN TEORI

## Studi Pustaka

# METODE PENELITIAN

## Kerangka Pemikiran

Kerangka berpikir adalah gambaran yang menggambarkan bagaimana penelitian akan berlangsung secara logis dan menyeluruh. Tahapan-tahapan dalam kerangka berpikir penelitian meliputi antara lain:

1. Pengumpulan Data

Penelitian ini menggunakan data dari platform *e-commerce* Tokopedia. Data yang diambil adalah riwayat ulasan pelanggan yang pernah berbelanja di toko Hanafashion\_shop, yang diperoleh melalui teknik web scraping.

1. Pelabelan

Pelabelan adalah tahap di mana setiap ulasan diberi label yang akan digunakan dalam proses pelatihan pada tahap klasifikasi. Atribut ulasan berisi tentang pengalaman pelanggan terkait kepuasan mereka saat berbelanja di toko Hanafashion\_shop. Pelabelan dilakukan berdasarkan nilai rating produk, di mana rating 1-3 dikategorikan sebagai negatif, sementara rating 4-5 dikategorikan sebagai positif.

1. Preprocessing

Adapun tahapan yang dilakukan pada preprocessing yaitu sebagai berikut :

1. *Cleaning*, yaitu untuk menghapus data yang memiliki nilai yang sama *(duplicate)* dan data yang kosong (*nan)*.
2. *Normalize*, yaitu untuk mengoreksi kata-kata yang salah ketik atau singkatan agar kembali ke bentuk aslinya.
3. *Stopword*, yaitu untuk menghapus kata-kata umum yang tidak memiliki informasi penting untuk analisis dari teks.
4. *Tokenizing*, yaitu untuk melakukan pemenggalan pada tiap suku kata.
5. *Stemming*, yaitu untuk mengubah kata-kata menjadi bentuk dasarnya dengan menghapus akhiran atau imbuhan.
6. Klasifikasi

Proses klasifikasi dibedakan menjadi dua proses, diantaranya :

* 1. *Training* adalah proses melatih algoritma klasifikasi, yakni *Support Vector Machine* dan *IndoBert*, agar dapat berfungsi sesuai harapan. Pertama, data atribut ulasan diberi bobot menggunakan perhitungan TF-IDF, namun hanya memperhitungkan frekuensi istilah *(term frequency)*. Proses ini menghasilkan model klasifikasi yang kemudian digunakan dalam tahapan *testing*.
  2. *Testing* adalah tahap di mana dataset diklasifikasikan dengan menggunakan model klasifikasi yang dihasilkan selama proses *training* data. Pada tahap ini, ulasan dikategorikan ke dalam sentimen positif dan negatif. Langkah-langkah proses ini dapat dilihat pada gambar di bawah.

A screenshot of a cell phone

Description automatically generated

**Gambar 3. 1** Kerangka Berpikir

## Bahan/Data

### Prosedur Pengumpulan Data

Penelitian ini menggunakan metode scraping menggunakan Selenium untuk mengumpulkan data ulasan toko Hanafashion\_shop di situs web Tokopedia. Metode ini memungkinkan pengendalian browser untuk mengakses halaman toko Hanafashion\_shop, mengambil ulasan pelanggan, dan mengumpulkan data secara otomatis. Dengan demikian, penelitian ini dapat menganalisis ulasan pelanggan dengan efisien dan mendalam untuk memperoleh wawasan yang relevan terkait dengan pengalaman pelanggan di toko Hanafashion\_shop di Tokopedia. Tampilan flowchart pengumpulan data dan scraping data Tokopedia dapat dilihat pada gambar di bawah.

A screenshot of a computer

Description automatically generated

**Gambar 3. 2** Produk Hanafashion\_shop

Selanjutnya, Selenium akan berinteraksi dengan halaman toko Hanafashion\_shop di situs web Tokopedia untuk mencari ulasan yang relevan. Proses ini mungkin melibatkan mengklik pada halaman produk yang tepat, menavigasi melalui halaman ulasan pelanggan, dan mengumpulkan data ulasan yang ada. Selenium akan mengekstrak teks ulasan dari elemen HTML yang sesuai dan menyimpannya dalam format csv untuk preprocessing lebih lanjut. Langkah-langkah tambahan seperti pembersihan data, penghapusan karakter khusus, atau penggabungan ulasan yang terpisah juga dapat dilakukan untuk memastikan data siap untuk analisis sentimen.

### Data yang diperoleh

Penelitian ini menggunakan data ulasan toko Hanafashion\_shop yang diambil melalui platform Tokopedia sebagai sumber data primer. Data ulasan pelanggan yang dikumpulkan dari toko Hanafashion\_shop di Tokopedia akan digunakan sebagai dataset utama dalam penelitian ini. Dengan menggunakan data ulasan yang diperoleh secara langsung dari toko Hanafashion\_shop, penelitian ini dapat memberikan wawasan yang lebih spesifik dan relevan terkait dengan pengalaman pelanggan dalam berinteraksi dengan produk-produk yang ditawarkan oleh toko tersebut. Contoh flowchart dan hasil pengumpulan data dapat dilihat pada gambar di bawah.

A black background with white squares

Description automatically generated

**Gambar 3. 3** Flowchart Pengumpulan Data



**Gambar 3. 4** Data Ulasan Tokopedia

Data yang diambil setiap kali melakukan preprocessing berjumlah maksimal 200 baris data.  
A screenshot of a computer code

Description automatically generated

**Gambar 3. 5** Code Scraping Data

Pengguna perlu memasukkan URL sumber produk Hanafashion\_shop dari platform Tokopedia, serta jumlah data yang ingin di-scrape. Selain itu, pengguna perlu menentukan kisaran rating untuk ulasan produk yang akan di-scrape. Rentang rating ini akan digunakan untuk menentukan label sentimen ulasan, rating 1-3 dianggap negatif, sedangkan rating 4-5 dianggap positif. Setelah semua input dimasukkan, pengguna dapat mengklik tombol untuk memulai proses scraping data.

### Labeling

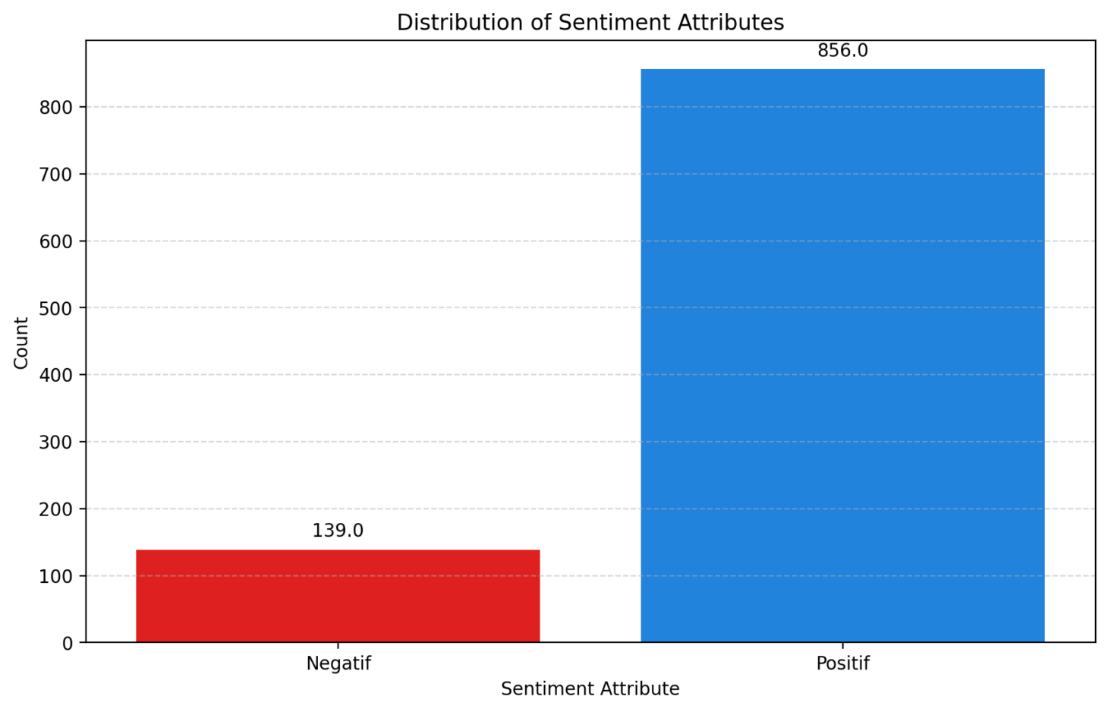
Pelabelan adalah tahap di mana ulasan produk diberi label yang nantinya akan digunakan untuk melatih model dalam proses klasifikasi. Ulasan tersedia untuk 10 produk dengan ulasan terbanyak di toko Hanafashion\_shop. Labeling didasarkan pada rating yang diberikan oleh pelanggan: rating 1-3 dianggap ulasan negatif, sedangkan rating 4-5 dianggap ulasan positif. Kelemahan pelabelan berdasarkan rating adalah adanya subjektivitas dalam hasilnya. Pada tahap pelabelan, ulasan diberi label sentimen positif dan negatif. Kode program untuk pelabelan dapat dilihat pada gambar di bawah ini.

A screenshot of a computer program

Description automatically generated

**Gambar 3. 6** Code Labeling Data

Dari 995 data yang telah diambil dari web scraping sebelumnya, dilakukan pelabelan berdasarkan rating dan menghasilkan sentiment positif dan negatif sebagai berikut :



**Gambar 3. 7** Visualisasi Sentimen

### Cleaning

Dalam proses *cleaning* data, tahap awal adalah mengidentifikasi dan menghapus nilai yang terduplikasi. Selain itu, langkah penting lainnya adalah menangani nilai yang hilang (NaN) agar data menjadi lebih bersih dan siap digunakan dalam pelatihan model SVM dan indoBert, sehingga kinerja model dapat dioptimalkan. Adapun *flowchart* dan kode program dari cleaning data dapat dilihat pada gambar di bawah.



**Gambar 3. 8** Flowchart Cleaning Data



**Gambar 3. 9** Code Cleaning Data

### Normalize

Pada tahapan *normalize* data, fokus utamanya adalah memperbaiki kata-kata yang mungkin disingkat atau tertulis dengan typo seperti "seller", "dll", "bgos", dan sejenisnya. Proses ini bertujuan untuk memastikan konsistensi dalam penggunaan kata-kata sehingga data menjadi lebih mudah dipahami dan diolah. Adapun contoh dari tahap *normalize* dapat dilihat pada table di bawah.

**Tabel 3. 1** Contoh Normalize

|  |  |
| --- | --- |
| **Text Input** | **Text Output** |
| seller responsif brg ny ok bangettt | penjual responsif, barang nya oke banget |

Normalisasi data adalah tahap penting dalam pra-pemrosesan data yang bertujuan untuk menghasilkan konsistensi dalam representasi data. Dengan melakukan normalisasi, kita dapat menghindari ambiguitas dan meningkatkan akurasi analisis data. Selain itu, normalisasi membantu dalam menyediakan input yang lebih baik untuk model pembelajaran mesin, seperti SVM dan indoBert, karena mengurangi variasi dalam data yang mungkin membingungkan model. Dengan demikian, normalisasi data membantu meningkatkan kualitas dan konsistensi data, yang pada gilirannya dapat meningkatkan kinerja model yang dibangun berdasarkan data tersebut.



**Gambar 3. 10** Code Normalize

### Stopword

Pada tahap ini, dilakukan penghapusan kata-kata yang kurang relevan atau sering muncul dalam teks, yang dikenal sebagai stopword. Stopword ini mencakup kata-kata penghubung dan kata-kata keterangan umum yang tidak memberikan kontribusi signifikan terhadap makna atau inti teks, seperti "sebuah", "oleh", "pada", dan lainnya. Proses ini bertujuan untuk menyederhanakan representasi teks, memfokuskan pada kata-kata kunci, dan mengurangi dimensi yang tidak perlu dalam analisis data, sehingga memungkinkan model untuk lebih fokus pada informasi yang penting. Adapun contoh dari tahap *stopword* dapat dilihat pada table di bawah.

**Tabel 3. 2** Contoh Stopword

|  |  |
| --- | --- |
| **Hasil *Normalize*** | **Hasil *Stopword*** |
| penjual responsif, barang nya oke banget | penjual responsif, barang oke banget |

A screen shot of a computer code

Description automatically generated

**Gambar 3. 11** Code Stopword

**Tabel 3. 3** List Stopword

|  |  |  |  |
| --- | --- | --- | --- |
| **Kata** | **Kata** | **Kata** | **Kata** |
| adalah  adanya  adapun  agak  agaknya  agar  akan  akankah  akhir  akhiri  akhirnya  aku  akulah  amat  amatlah  anda  andalah  antar  antara  antaranya  apa  apaan  apabila  apakah  apalagi  apatah  artinya  asal  asalkan  atas  atau  ataukah  ataupun  awal  awalnya  bagai  bagaikan  bagaimana  bagaimanakah  bagaimanapun  bagi  bagian  bahkan  bahwa  bahwasanya  baik  bakal  bakalan  balik | banyak  bapak  baru  bawah  beberapa  begini  beginian  beginikah  beginilah  begitu  begitukah  begitulah  begitupun  bekerja  belakang  belakangan  belum  belumlah  benar  benarkah  benarlah  berada  berakhir  berakhirlah  berakhirnya  berapa  berapakah  berapalah  berapapun  berarti  berawal  berbagai  berdatangan  beri  berikan  berikut  berikutnya  berjumlah  berkali-kali  berkata  berkehendak  berkeinginan  berkenaan  berlainan  berlalu  berlangsung  berlebihan  bermacam  bermacam-macam | bermula  bersama  bersama-sama  bersiap  bersiap-siap  bertanya  bertanya-tanya  berturut  berturut-turut  bertutur  berujar  berupa  besar  betul  betulkah  biasa  biasanya  bila  bilakah  bisa  bisakah  boleh  bolehkah  bolehlah  buat  bukan  bukankah  bukanlah  bukannya  bulan  bung  cara  caranya  cukup  cukupkah  cukuplah  cuma  dahulu  dalam  dan  dapat  dari  daripada  datang  dekat  demi  demikian  demikianlah  dengan | di  dia  diakhiri  diakhirinya  dialah  diantara  diantaranya  diberi  diberikan  diberikannya  dibuat  dibuatnya  didapat  didatangkan  digunakan  diibaratkan  diibaratkannya  diingat  diingatkan  diinginkan  dijawab  dijelaskan  dijelaskannya  dikarenakan  dikatakan  dikatakannya  dikerjakan  diketahui  diketahuinya  dikira  dilakukan  dilalui  dilihat  dimaksud  dimaksudkan  dimaksudkannya  dimaksudnya  diminta  dimintai  dimisalkan  dimulai  dimulailah  dimulainya  dimungkinkan  dini  dipastikan  diperbuat  diperbuatnya  dipergunakan |

### Tokenizing

Pada tahap *tokenizing*, dokumen diubah menjadi serangkaian term dengan menghapus semua karakter tanda baca yang ada pada token. Tujuan dari proses ini adalah untuk menghasilkan kumpulan kata-kata yang merupakan representasi dari teks atau dokumen tersebut. Dengan demikian, output yang dihasilkan adalah sekumpulan kata-kata yang membentuk inti dari teks, tanpa kehadiran tanda baca yang mungkin tidak relevan dalam analisis atau pemrosesan berikutnya. Adapun contoh dari tahap *tokenizing* dapat dilihat pada table di bawah.

**Tabel 3. 4** Contoh Tokenizing

|  |  |
| --- | --- |
| **Hasil *Stopword*** | **Hasil *Tokenizing*** |
| penjual responsif, barang oke banget | penjual | responsif | barang | oke | banget |

*Tokenizing* adalah proses memecah teks menjadi unit kata. Ini dilakukan dengan menggunakan karakter whitespace seperti enter, tabulasi, dan spasi sebagai pemisah kata. Namun, karakter tunggal seperti tanda kutip tunggal ('), titik (.), semikolon (;), titik dua (:), dan lainnya juga dapat berperan sebagai pemisah kata, tergantung pada konteksnya. Proses tokenisasi ini penting untuk mempersiapkan teks agar dapat diolah lebih lanjut, dengan menghasilkan kumpulan kata-kata yang mewakili teks tersebut dengan tepat.



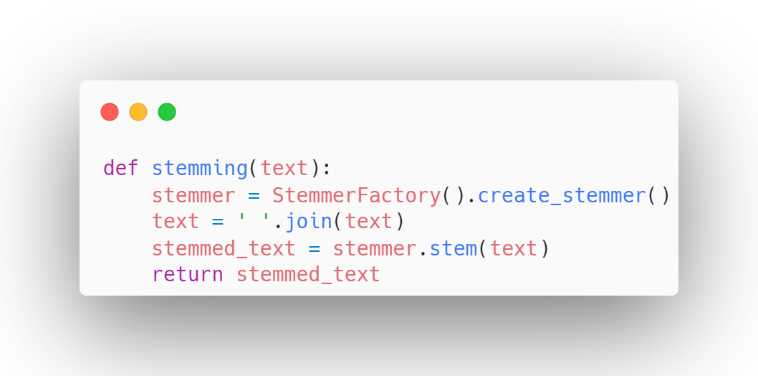
**Gambar 3. 12** Code Tokenizing

### Stemming

*Stemming* merupakan tahap dalam pra-pemrosesan teks yang bertujuan untuk menghapus imbuhan, awalan, dan akhiran dari kata-kata guna mengubahnya menjadi bentuk dasarnya. Proses ini membantu dalam menghasilkan representasi yang lebih konsisten dari kata-kata dalam teks, memungkinkan model untuk lebih mudah mengenali dan memahami makna kata-kata yang sebenarnya. Adapun contoh dari tahap *stemming* dapat dilihat pada table di bawah.

**Tabel 3. 5** Contoh Stemming

|  |  |
| --- | --- |
| **Hasil *Tokenizing*** | **Hasil *Stemming*** |
| penjual | responsif | barang | oke | banget | jual | responsif | barang | oke | banget |



**Gambar 3. 13** Code Stemming

### SMOTE

SMOTE, atau Teknik Oversampling Minoritas Sintetis, adalah pendekatan yang digunakan untuk menangani ketidakseimbangan data dengan cara yang berbeda dari metode oversampling konvensional [1]. SMOTE dapat mengatasi risiko overfitting, mempertahankan informasi, serta meningkatkan akurasi dan prediksi kelas. Proses SMOTE dimulai dengan identifikasi kelas minoritas dalam data, lalu menentukan jarak k tetangga terdekat dengan menghitung jarak Euclidean antara data minoritas. Selanjutnya, data sintetis dihasilkan dengan mengikuti garis yang menghubungkan antara k tetangga terdekat dan memasukkan titik acak pada garis tersebut. Formula untuk menghasilkan data sintetis dalam data yang tidak seimbang dengan SMOTE dapat ditemukan dalam persamaan (3. 1). Perhitungan jarak tetangga terdekat dalam data berskala numerik dapat dilakukan dengan menggunakan jarak Euclidean sebagaimana dijelaskan dalam Persamaan (3. 2).

|  |  |  |
| --- | --- | --- |
|  |  | (3. 1) |

|  |  |  |
| --- | --- | --- |
|  |  | (3. 2) |

### Pembobotan Kata

Dalam klasifikasi sentiment ulasan tokopedia, pembobotan kata digunakan untuk mendapatkan suatu kategori. Salah satu metode pembobotan adalah TF-IDF *(Term Frequency –Inverse Document Frequency)*.

*Term Weighting* TI-IDF adalah salah satu pembobotan yang sering digunakan dan merupakan gabungan dari *Term Frequency* *dan Inverse Document Frequency*. TF-IDF terdiri dari frekuensi term dan inverse dokumen yang didapatkan dari membagi seluruh jumlah dokumen terhadap jumlah dokumen yang memiliki term tersebut. Dalam TI-IDF bobot akan ditemukan dalam persamaan berikut:

|  |  |  |
| --- | --- | --- |
|  |  | (3. 3) |

Keterangan:

{tf}\_{i,j} = Bobot dari istilah i dalam dokumen j

{df}\_i = Frekuensi munculnya istilah i dalam dokumen j

N = Total istilah pada dokumen

### Hierarchical Dirichlet Process (HDP)

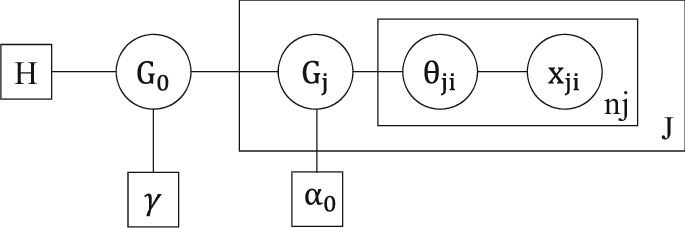
*Hierarchical Dirichlet Process* (HDP) adalah model statistik yang dapat diterapkan dalam proyek analisis sentimen untuk mengelompokkan data teks ke dalam berbagai topik atau aspek yang muncul secara alami. Dengan memungkinkan jumlah cluster yang tidak terbatas dan mengatasi masalah ketidakpastian tentang jumlah cluster yang tepat, HDP memungkinkan identifikasi yang lebih baik terhadap nuansa dan variasi dalam sentimen yang diekspresikan dalam data teks, yang berguna untuk memahami perasaan dan opini yang tersembunyi di balik bahasa manusia [2]. Setelah melakukan tahap preprocessing data, peneliti mengimplementasikan Hierarchical Dirichlet Process (HDP) dengan membuat kode untuk mengidentifikasi 10 topik utama dari data dan mengekstrak kata-kata yang mendefinisikan setiap topik. Secara manual, perhitungan HDP melibatkan proses yang lebih kompleks, termasuk pembentukan hierarki dari distribusi Dirichlet dan estimasi parameter secara iteratif untuk menemukan topik yang paling sesuai dengan data yang diamati. Sedangkan secara manual perhitungan atau rumus HDP adalah sebagi berikut:

|  |  |  |
| --- | --- | --- |
|  | 𝑃(𝜃) = 𝐷𝑃(𝛼, 𝐻) | (3. 4) |

di mana θ adalah distribusi topik, DP(α,H) mengindikasikan bahwa θ adalah sampel dari DP dengan parameter konsentrasi α dan distribusi dasar (base distribution) H. Metode HDP merupakan pengembangan dari metode LDA dimana metode HDP menghasilkan performa yang lebih bagus dibanding metode LDA dalam melakukan topic modeling pada teks, terlebih pada kasus dimana data teks akan bertambah secara terus-menerus. Berbeda dengan LDA, HDP tidak perlu menentukan jumlah topik sehingga data bisa terus bertambah tanpa perlu melatih ulang data. Ada beberapa probablitas, yaitu probabilitas global G0, serta probabilitas Gj untuk setiap kelompok j, dengan G0 didistribusikan sebagai Dirichlet Process (DP) dengan parameter konsentrasi γ dan distribusi dasar H, kami kemudian sampel Gj untuk setiap kelompok dari distribusi DP(α0, G0). Untuk setiap kelompok, kami sampel faktor-faktor θji dari distribusi Gj, dan kemudian sampel data individu xji dari distribusi F yang bergantung pada θji. Dengan demikian, kita dapat mengekspresikan persamaan kondisional sebagai berikut :

|  |  |  |
| --- | --- | --- |
|  | G0 | γ, H ∼ DP(γ, H) | (3. 5) |
|  | Gj | α0, G0 ∼ DP(α0, G0) | (3. 6) |
|  | θji | Gj ∼ Gj | (3. 7) |
|  | xji | θji ∼ F(θji) | (3. 8) |

Untuk memahami lebih baik, kita dapat membayangkan sebuah skenario pemodelan topik di mana kita memiliki k korpus dokumen, dan setiap dokumen memiliki nj jumlah kata. Kita diberikan distribusi awal H dan parameter konsentrasi γ, dan kita akan sampel pengukuran probabilitas global G0 dari DP(γ, H). Kita bisa menganggap G0 sebagai distribusi probabilitas, seperti yang ditunjukkan oleh konstruksi patah-stik pada bagian berikutnya. Kemudian, untuk j = 1, ..., k, kita sampel pengukuran probabilitas Gj untuk setiap dokumen dari DP(α0, G0). Untuk dokumen j, kita akan sampel nj kata, dengan setiap kata dinotasikan sebagai xji untuk i = 1, ..., nj, dengan cara pertama-tama melakukan sampel θji dari Gj dan kemudian sampel xji ∼ F(θji). Adapun ilustrasi dari *Hierarchical Dirichlet Process* (HDP) dapat dilihat pada gambar di bawah.



**Gambar 3. 14** Ilustrasi Hierarchical Dirichlet Process

### Support Vector Machine

*Support Vector Machine* (SVM) adalah teknik yang digunakan untuk melakukan prediksi, baik pada kasus klasifikasi maupun regresi. Prinsip dasar SVM adalah pemisah linier, yaitu kemampuan untuk mengklasifikasikan data yang secara linier dapat dipisahkan. Namun, SVM telah dikembangkan untuk bekerja pada masalah non-linier dengan mengadopsi konsep kernel, memungkinkan model bekerja dalam ruang berdimensi tinggi [3].

Pada Gambar 3.11, dijelaskan konsep dasar algoritma *Support Vector Machine* (SVM), yang berfokus pada mencari hyperplane terbaik sebagai pemisah antara dua kelas dalam sebuah data.

*A diagram of a graph

Description automatically generated with medium confidence*

**Gambar 3. 15** Klasifikasi Linear SVM

Pada input space terdapat dua kelas yang berbeda, +1 dan -1, beserta masing-masing pattern yang digambarkan dengan simbol kotak warna orange untuk pattern -1 dan simbol lingkaran hijau untuk pattern +1.

A black background with orange and green dots

Description automatically generated

**Gambar 3. 16** Detail Klasifikasi Linear SVM

­Pada gambar 3.7 dijelaskan dalam mengklasifikasi untuk mendapat hasil yang baik hyperplane digunakan untuk memisahkan menjadi dua kelas dengan mengukur margin *hyperplane* tersebut dan mencari titik maksimalnya. Margin adalah jarak antara hyperplane terdekat dengan pattern terdekat dari masing-masing kelas dan pattern yang paling dekat dengan *hyperplane* disebut *support vector*. Seperti gambar dibawah garis tidak putus-putus yangterletak tepat di tengah-tengah kedua keas. Sedangkan *support vector* tampak sebagai *pattern* yang berpotongan dengan garis putus-putus. Dari Gambar 3.6 bidang pemisah dapat dirumuskan :

m = jarak antara dua bidang

w = bidang normal

b = posisi relative terhadap origin

jarak garis dirumuskan wx+b=c ke origin adalah (c-b)/|w|

|  |  |  |
| --- | --- | --- |
|  |  | (3. 9) |

Margin m dimaksimalkan dengan memenuhi konstrain 2 bidang pembatas yang sejajar dan data yang ada pada bidang pembatas disebut *support vector.* Bidang pembatas kelas pertama membatasi kelas pertama sedangkan bidang pembatas kelas kedua membatasi kelas kedua. Sehingga diperoleh :

|  |  |  |
| --- | --- | --- |
|  | *i.w* + *b*  ≥ + 1 *for yi* = +1 | (3. 10) |
|  | *i.w* + *b*  ≥ + 1 *for yi* = -1 | (3. 11) |

Nilai maksimal margin harus memenuhi rumus di atas dan nilai b dan w dikalikan dengan sebuah konstanta yang akan menghasilkan nilai margin yang dikalikan dengan konstanta yang sama. Konstrain merupakan scaling constraint dengan dipenuhi rescaling b dan w. Karena maksimalkan dan minimalkan w dirumuskan dengan pertidaksamaan rumus di atas.

|  |  |  |
| --- | --- | --- |
|  | *yi* ( *i.w* + *b*) - 1 ≥ 0 | (3. 12) |

Dengan mengalikan b dan sebuah konstanta, maka menghasilkan nilai m kemudian dikalikan dengan konstanta yang sama. Konstrain merupakan scaling constraint yang dipenuhi dengan rescaling b dan w. Maksimalkan minimumkan .

Untuk mencari nilai margin terbesar untuk bidak pemisah terbaik dapat dirumuskan menjadi masalah optimasi konstrain, yaitu :

|  |  |  |
| --- | --- | --- |
|  |  | (3. 13) |

|  |  |  |
| --- | --- | --- |
|  | *s.t yi(* ≥ 0 | (3. 14) |

Dengan lebih mudah untuk menyelesaikan permasalah optimasi konstrain dalam formulasinya dirubah kedalam formula lagrangian yang menggunakan lagrange multiplier yang diubah menjadi :

|  |  |  |
| --- | --- | --- |
|  |  | (3. 15) |

Formula pencarian bidang pemisah terbaik ini adalah permasalahan *quadratic programming,* sehingga nilai maksimum global dari akan selalu dapat ditemukan setelah solusi permasalahan *quadratic programming* ditemukan (nilai ), maka kelas dari data pengujian x dapat ditentukan berdasarkan nilai dari fungsi keputusan :

|  |  |  |
| --- | --- | --- |
|  |  | (3. 16) |

= *support vector,*

Ns = jumlah *support vector*

*=* data yang akan diklasifikasikan

### IndoBert

Model word embedding yang telah dilatih sebelumnya, dikenal sebagai model pre-trained word embedding, dirancang untuk meningkatkan pemahaman makna dan sintaksis dari teks. Model-model ini dilatih menggunakan dataset besar yang beragam agar dapat mengenali pola bahasa secara luas. Pada tahun 2018, model canggih bernama Bidirectional Encoder Representations from Transformers (BERT) diperkenalkan dan berhasil mencapai hasil unggul dalam berbagai penelitian di bidang pemrosesan bahasa alami (NLP). BERT memanfaatkan arsitektur Transformer dengan mekanisme self-attention untuk memahami konteks hubungan antara kata-kata dalam teks. Di Indonesia, perkembangan signifikan terjadi pada tahun 2020 dengan hadirnya model pre-trained BERT yang dikenal sebagai IndoBERT. Model ini khusus disesuaikan untuk bahasa Indonesia, memungkinkan pemahaman yang lebih baik terhadap bahasa lokal dan meningkatkan performa model dalam tugas-tugas NLP berbahasa Indonesia [4].

Model ini bertujuan untuk meminimalkan gabungan fungsi kerugian dari Masked LM dan Next Sentence Prediction, sehingga menghasilkan model bahasa yang kuat dengan kemampuan yang ditingkatkan dalam memahami konteks dalam kalimat dan hubungan antar kalimat. Ada beberapa hal pada arsitektur BERT, diantaranya :

* BERT BASE memiliki 1 2 lapisan di tumpukan Encoder sedangkan BERT LARGE memiliki 24 lapisan di tumpukan Encoder . Ini lebih dari arsitektur Transformer yang dijelaskan dalam makalah asli ( 6 lapisan encoder ).
* Arsitektur BERT (BASE dan LARGE) juga memiliki jaringan feedforward yang lebih besar (masing-masing 768 dan 1024 unit tersembunyi), dan lebih banyak perhatian (masing-masing 12 dan 16) daripada arsitektur Transformer yang disarankan dalam makalah asli. Ini berisi 512 unit tersembunyi dan 8 kepala perhatian.
* BERT BASE berisi 110 juta parameter sedangkan BERT LARGE memiliki 340 juta parameter.

A close-up of a white rectangular object

Description automatically generated

**Gambar 3. 17** Arsitektur Bert Base & Bert Large

Model ini memproses masukan dimulai dengan token CLS, yang merupakan token klasifikasi, diikuti oleh serangkaian kata sebagai masukan. Token CLS ini berfungsi sebagai penanda awal untuk pemrosesan. Masukan kemudian diteruskan ke lapisan-lapisan di atasnya. Setiap lapisan menggunakan mekanisme perhatian mandiri dan hasilnya kemudian diteruskan ke jaringan feedforward sebelum akhirnya diserahkan ke pembangun enkode berikutnya. Model ini menghasilkan vektor dengan ukuran tersembunyi (768 untuk BERT BASE). Jika kita ingin menggunakan model ini untuk klasifikasi, kita dapat menggunakan keluaran yang terkait dengan token CLS.

A white rectangular object with a black background

Description automatically generated

**Gambar 3. 18** Bert Embeddings

## Perancangan Sistem

### Use Case Diagram

Berikut ini adalah use case diagram pada sistem analisis sentimen terhadap data ulasan Hanafashion\_shop di Tokopedia :



**Gambar 3. 19** Use Case Diagram

### Activity Diagram

*Activity diagram* merupakan alur aktivitas pengguna terhadap system. Dengan adanya *activity diagram* dapat mengetahui alur interaksi yang terjadi pada *use case diagram* :

1. *Activity Diagram Running System*

Diagram aktifitas ini menggambarkan aktifitas *system interface* menggunakan streamlit dijalankan. Adapun alur aktivitas proses running system dapat dilihat pada gambar di bawah.

A black background with white text

Description automatically generated

**Gambar 3. 20** Activity Diagram Running System

1. *Activity Diagram Preprocessing*

Diagram aktivitas ini menggambarkan *preprocessing* data dari ulasan Hanafashion\_shop yang ada di Tokopedia sebelum dilakukan proses training data menggunakan algoritma *Support Vector Machine* dan *indoBert*. Adapun alur aktivitas *preprocessing* dapat dilihat pada gambar di bawah.

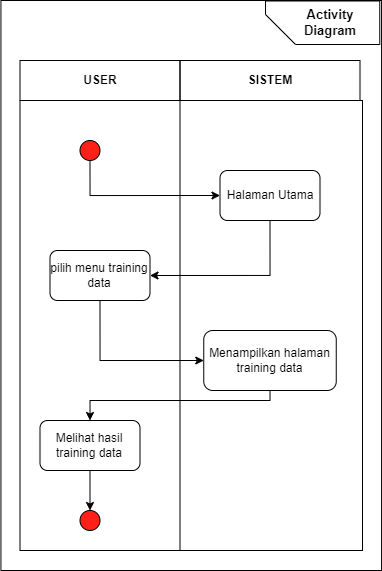
A screenshot of a cell phone

Description automatically generated

**Gambar 3. 21** Activity Diagram Preprocessing

1. *Activity Diagram Training Data*

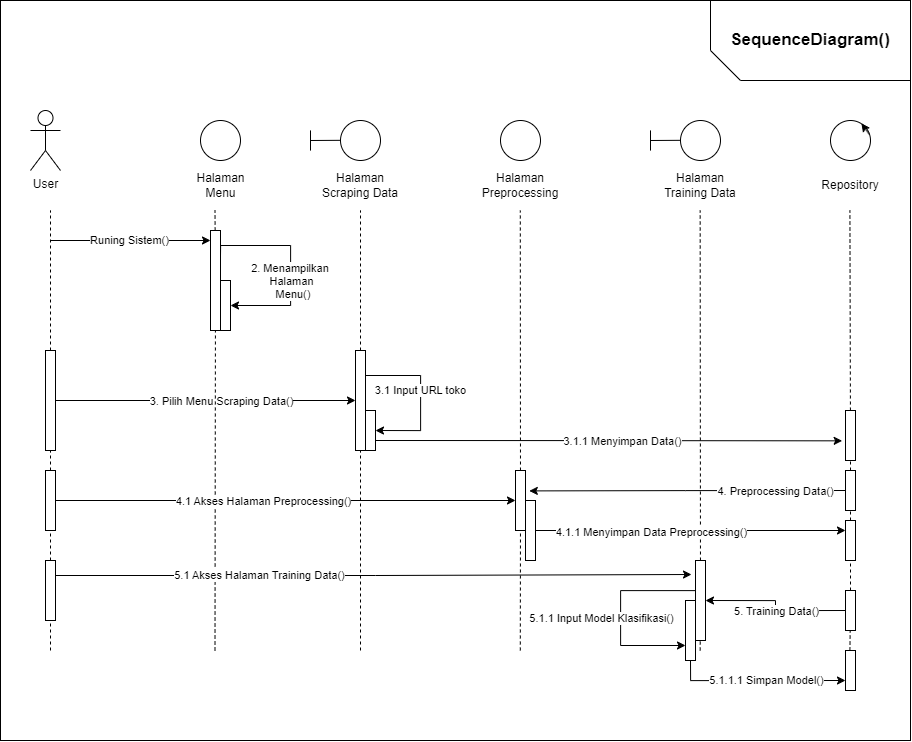
Diagram aktivitas ini menggambarkan tahapan training data menggunakan algoritma *Support Vector Machine* dan *indoBert*, user bisa memasukkan input parameter sebelum melakukan training data. Adapun alur aktivitas *training data* dapat dilihat pada gambar di bawah.



**Gambar 3. 22** Activity Diagram Training Data

### Sequence Diagram

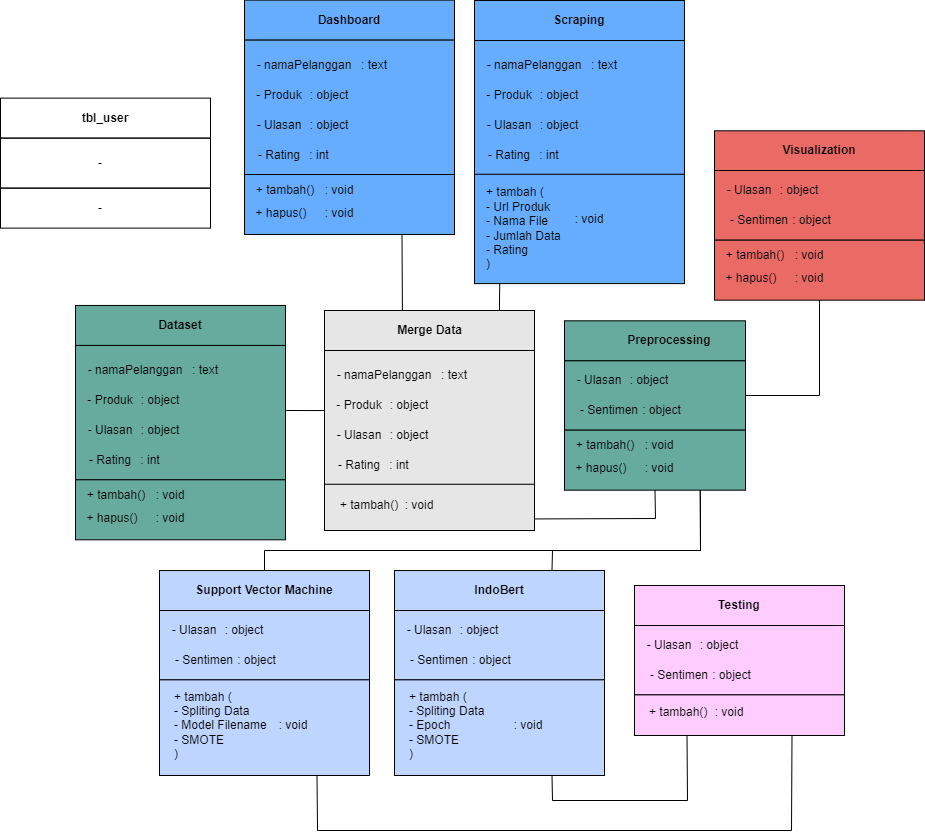
Diagram sequence menggambarkan detail alur proses berdasarkan urutan waktu. Dalam proses analisis sentimen, pengguna memulai dengan mengakses menu aplikasi yang tersedia. Sistem kemudian melakukan *scraping* data dari sumbernya, seperti media sosial, lalu melakukan *preprocessing* untuk membersihkan dan mempersiapkan data teks. Setelah itu, sistem melatih model analisis sentimen menggunakan data yang telah diproses dan menyimpannya ke dalam repository untuk penggunaan dan referensi di masa depan. Adapun tampilan proses *sequence diagram* dapat dilihat pada gambar di bawah.



**Gambar 3. 23** Sequence Diagram

### Class Diagram

Berikut ini adalah *class diagram* dari analisis sentiment pada toko Hanafashion\_shop dari platform Tokopedia menggunakan algoritma Support Vector Machine dan IndoBert.



**Gambar 3. 24** Class Diagram

# HASIL DAN PEMBAHASAN

## Environment Testing

Hasil dari langkah ini adalah persiapan perangkat keras dan perangkat lunak yang akan digunakan untuk mendesain sistem, mengembangkan sistem, dan mengujinya. Perangkat pengembangan yang dipakai termasuk:

1. Kebutuhan minimum perangkat keras

Perangkat minimum yang dibutuhkan untuk dapat menjalankan system perangkat computer atau laptop dengan detail spesifikasi sebagai berikut :

1. Intel Core i5-10351G
2. Memory RAM 8 GB
3. Memory SSD 1TB
4. Kebutuhan perangkat lunak
5. Sistem operasi *Microsoft Windows* 11
6. *Google Chrome*
7. *Visual Studio Code*
8. *Python & Library*

## Deskripsi Data

Dataset yang digunakan pada penelitian ini berisi atribut seperti nama pelanggan, produk yang dibeli, ulasan, dan rating pelanggan terhadap produk. Informasi dari atribut nama pelanggan memberikan identitas pelanggan dalam transaksi, produk menjelaskan jenis produk yang dibeli, ulasan mencatat feedback pelanggan, dan rating memberikan nilai kepuasan pelanggan terhadap produk. Data ini berasal dari Hanafashion\_shop di Tokopedia, fokus pada produk yang dijual di toko tersebut. Analisis dapat dilakukan untuk memahami preferensi pelanggan, kualitas produk, serta tren dalam ulasan dan rating produk di Hanafashion\_shop.

### Data Hasil Scraping

Pada tahapan ini, data ulasan produk E-commerce Tokopedia dikumpulkan melalui teknik scrapping data. Proses pengumpulan data dilakukan dengan mengakses situs Tokopedia dan mengekstrak ulasan produk menggunakan teknik web scrapping. Dengan menggunakan library Python seperti BeautifulSoup dan Requests, data ulasan produk yang terdapat pada halaman-halaman produk Tokopedia diunduh secara otomatis. Adapun sampel data hasil scraping dapat dilihat pada tabel di bawah.

**Tabel 4. 1** Sampel Data Hanafashion\_shop

|  |  |  |
| --- | --- | --- |
| **Nama Produk** | **Ulasan** | **Rating** |
| Miranda Basic Tank Top Simple Polos Wanita - TT029 | bahan biasa aja dan warna navy nya beda | 2 |
| Marina Tube Bra Crop Top Kemben Pendek Wanita - TT074 - Light Grey, L | yg dipesen kemben yg dikasih kaos | 2 |
| Danise Short Pants Celana Pendek Wanita - SP061 - yellow green, XL | Bahannya adem, bagus banget, sesuai foto | 5 |
| Sita Crop Tank Top Atasan Wanita - TT007 - Khaky, XL | bahan lumayan tebel, jahitan rapi, nyaman d pakai..makasih | 5 |
| Hanafe Crop Top Hana Fashion - Light Grey, M | Ukuran L nya lebih kecil. | 4 |

### Hierarchical Dirichlet Process (HDP)

Data yang telah terlebih dahulu dilakukan tahapan pre-processing, selanjutnya data tersebut dapat dilanjutkan pada analisis topic modeling. Analisis topik modeling yang pertama yaitu Hierarchical Dirichlet Process (HDP). Setelah itu dilakukan pembuatan topic yang terdiri dari kata apa saja dari masing-masing topic. Adapun topik HDP dapat dilihat pada tabel di bawah.

**Tabel 4. 2** Data Ulasan Hasil Ekstraksi Topik HDP Beserta Label Klasifikasi

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Ulasan** | **Topik1** | **Topik2** | **Topik3** | **Topik4** | **Topik5** | **Topik6** | **Sentimen** |
| Alhamdulillah guys, muat dan elastis bagus, anak suka banget | 0.00595 | 0.00417 | 0.00368 | 0.00394 | 0.00423 | 0.00344 | Positif |
| sayang banget kecil pdhl picture liat sesuai badan | 0.00477 | 0.00464 | 0.00612 | 0.00573 | 0.00308 | 0.00319 | Negatif |

Sesuai dengan langkah-langkah yang telah dijelaskan, terlebih dahulu akan dilakukan perhitungan skor coherence utnuk mengetahui jumlah topik yang optium. Jumlah topik yang optimum akan diindikasi dengan nilai skor coherence yang tertinggi. Sebagai langkah pertama dalam pembentukan model HDP adalah penentuan jumlah topik, telah dilakukan perhitungan skor coherence untuk mengetahui jumlah topik yang optimal dengan metode HDP, dimana hasil dari skor dapat ditabelkan sebagaimana pada Tabel 4.8 berikut.

**Tabel 4. 3** Skor Coherence Metode Hierarchical Dirichlet Process (HDP)

|  |  |
| --- | --- |
| **Jumlah Topik** | **Skor Coherence** |
| 2 | 0.777647 |
| **3** | **0.793783** |
| 4 | 0.784469 |
| 5 | 0.789677 |
| 6 | 0.731826 |
| 7 | 0.768110 |

Berdasarkan tabel di atas, dapat disimpulkan bahwa jumlah topik optimal adalah tiga, karena pada jumlah topik tersebut diperoleh skor coherence tertinggi. Skor coherence tertinggi yang dicapai dengan metode HDP adalah **0.793783**, yang memungkinkan pembentukan model HDP dengan jumlah topik optimal tersebut. Penulisan model didasarkan pada probabilitas kemunculan kata dalam setiap topik. Berdasarkan jumlah topik optimal ini, model akan dibentuk dengan dua topik. Dari hasil ekstraksi fitur, langkah selanjutnya adalah melatih data menggunakan algoritma SVM dan IndoBert. Penerapan topic modeling dengan *Hierarchical Dirichlet Proces*s *(HDP)* dapat meningkatkan performa pelatihan data.

### Data Training Model

Data training model digunakan untuk melakukan pelatihan model menggunakan data hasil preprocessing yang telah dipersiapkan sebelumnya. Dalam hal ini, user dapat memilih metode pelatihan model yang ingin digunakan, baik SVM maupun IndoBert. Dengan menyediakan akses langsung ke data yang telah dipersiapkan dan pilihan antara dua metode pelatihan yang berbeda, sub bab ini memfasilitasi pengguna untuk mengembangkan model yang sesuai dengan kebutuhan analisis data mereka, baik untuk kasus klasifikasi dengan SVM maupun untuk analisis teks yang lebih kompleks dengan IndoBert. Adapun data yang akan digunakan untuk training model dapat dilihat pada tabel di bawah.

**Tabel 4. 4** Sampel data preprocessing

|  |  |  |
| --- | --- | --- |
| **Ulasan** | **Preprocessing** | **Sentimen** |
| bahan biasa aja dan warna navy nya beda | Bahan biasa warna navy beda | Negatif |
| yg dipesen kemben yg dikasih kaos | pesen kemben kasih kaos | Negatif |
| Bahannya adem, bagus banget, sesuai foto | bahan adem bagus banget sesuai foto | Positif |
| bahan lumayan tebel, jahitan rapi, nyaman d pakai..makasih | bahan lumayan tebal jahitan rapi nyaman pakai terimakasih | Positif |
| Ukuran L nya lebih kecil. | ukuran lebih kecil. | Positif |

## Hasil Perbandingan Model

Untuk mengevaluasi kinerja dalam studi ini, peneliti menggunakan metode *confusion matrix* untuk menghitung recall, presisi, dan akurasi dari setiap kategori. Dari total 996 ulasan yang diambil, peneliti melakukan pembagian data menjadi dua bagian: 80% untuk data training dan 20% untuk data testing. Berikut adalah hasil evaluasi kinerja dari data pengujian yang kami peroleh menggunakan *confusion matrix*.

### Hasil Model *Support Vector Machine*

Dalam eksperimen menggunakan algoritma *Support Vector Machine* dengan splitting data 80/20, confusion matrix yang dihasilkan menunjukkan bahwa dari total 190 sampel, terdapat 159 prediksi positif yang benar (TP), 2 prediksi positif yang salah (FP), 1 prediksi negatif yang benar (TN), dan 28 prediksi negatif yang salah (FN). Sedangkan confusion matrix menggunakan SMOTE menunjukkan bahwa dari total 343 sampel, terdapat 156 prediksi positif yang benar (TP), 19 prediksi positif yang salah (FP), 165 prediksi negatif yang benar (TN), dan 3 prediksi negatif yang salah (FN). Evaluasi kinerja model dilakukan dengan menghitung metrik seperti akurasi, presisi, recall, dan F1-score, yang menyediakan pemahaman yang lebih holistik tentang kecocokan model terhadap data uji. Adapun tabel dan perhitungan evaluasi dapat dilihat sebagai berikut :

**Tabel 4. 5** Test Result Support Vector Machine

|  |  |  |
| --- | --- | --- |
| Actual Class | Positif | Negatif |
| Positif | 159 | 2 |
| Negatif | 28 | 1 |

**Positif :**

**Negatif :**

***Accuracy :***

**Tabel 4. 6** Test Result Support Vector Machine dengan SMOTE

|  |  |  |
| --- | --- | --- |
| Actual Class | Positif | Negatif |
| Positif | 156 | 19 |
| Negatif | 3 | 165 |

**Positif :**

**Negatif :**

***Accuracy :***

### Hasil Model *IndoBert*

Dalam eksperimen menggunakan model *IndoBERT* dengan splitting data 80/20, confusion matrix yang dihasilkan menunjukkan bahwa dari total 190 sampel, terdapat 154 prediksi positif yang benar (TP), 13 prediksi positif yang salah (FP), 11 prediksi negatif yang benar (TN), dan 12 prediksi negatif yang salah (FN). Sedangkan confusion matrix menggunakan SMOTE menunjukkan bahwa dari total 190 sampel, terdapat 150 prediksi positif yang benar (TP), 11 prediksi positif yang salah (FP), 15 prediksi negatif yang benar (TN), dan 14 prediksi negatif yang salah (FN). Evaluasi kinerja model dilakukan dengan menghitung metrik seperti akurasi, presisi, recall, dan F1-score, yang memberikan gambaran menyeluruh tentang kemampuan model IndoBERT dalam mengklasifikasikan data uji. Adapun tabel dan perhitungan evaluasi dapat dilihat sebagai berikut :

**Tabel 4. 7** Test Result IndoBert

|  |  |  |
| --- | --- | --- |
| Actual Class | Positif | Negatif |
| Positif | 154 | 13 |
| Negatif | 12 | 11 |

**Positif :**

**Negatif :**

***Accuracy :***

**Tabel 4. 8** Test Result IndoBert dengan SMOTE

|  |  |  |
| --- | --- | --- |
| Actual Class | Positif | Negatif |
| Positif | 150 | 11 |
| Negatif | 14 | 15 |

**Positif :**

**Negatif :**

***Accuracy :***

## Analisa Model Terbaik

Analisis model terbaik perlu dilakukan untuk memahami kinerja dari berbagai model yang telah dievaluasi dalam konteks penelitian atau tugas tertentu. Dalam tahap ini, hasil evaluasi dari setiap model akan dianalisis secara rinci untuk menentukan model mana yang paling cocok dengan dataset dan memenuhi kriteria kinerja yang diinginkan. Salah satu metode yang telah dievaluasi adalah *Support Vector Machine* (SVM), yang diterapkan dengan teknik SMOTE untuk menyeimbangkan data. Dengan akurasi sebesar 94%, presisi 98%, recall 89%, dan F1 score 93%, model SVM menonjol sebagai kandidat terbaik untuk tugas klasifikasi, menunjukkan kemampuannya dalam mengenali pola yang relevan dan menangani ketidakseimbangan kelas dalam dataset dengan efektif. Adapun perbandingan performa model dapat dilihat pada tabel di bawah.

**Tabel 4. 9** Perbandingan Performa Model

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Balancing** | **Metode** | **Akurasi** | **Precision** | **Recall** | **F1 Score** |
| Tanpa *SMOTE* | *Support Vector Machine* | 84% | 85% | 98% | 91% |
| *IndoBert* | 87% | 93% | 92% | 92% |
| *SMOTE* | ***Support Vector Machine*** | **94%** | **98%** | **89%** | **93%** |
| *IndoBert* | 87% | 91% | 93% | 92% |

## Prosedur Penggunaan Aplikasi

Prosedur Penggunaan Aplikasi menyajikan menu-menu yang tersedia, mulai dari dashboard yang memberikan gambaran umum tentang aplikasi, hingga langkah-langkah detail seperti scraping data untuk mengumpulkan informasi, merge data untuk menggabungkan beberapa dataset, dataset untuk melihat dataset yang tersedia, preprocessing untuk mempersiapkan data untuk analisis, visualization untuk visualisasi data, training data menggunakan algoritma *Support Vector Machine* dan *IndoBert* untuk proses pelatihan model menggunakan metode-metode tersebut, dan testing untuk menguji model yang telah dilatih. Dengan rangkaian menu ini, pengguna dapat dengan mudah mengakses berbagai fitur dan langkah-langkah dalam aplikasi untuk menganalisis data dan melakukan evaluasi model.

### Halaman Dashboard

Pada halaman dashboard, dilakukan proses untuk menampilkan data dan visualisasi ulasan dari Hanafashion\_shop yang kita dapatkan dari Tokopedia dengan menggunakan Python dan Selenium. Data yang kita tampilkan adalah untuk 10 produk dengan jumlah ulasan terbanyak dari toko Hanafashion\_shop. Sentimen dari ulasan ditampilkan berdasarkan rating pada masing-masing produk, sementara wordcloud digunakan untuk memvisualisasikan teks dari ulasan berdasarkan atribut tertentu pada data Hanafashion\_shop..Adapun script dan tampilan dashboard dapat dilihat pada gambar di bawah.



**Gambar 4. 1** Script Tampilan Dashboard

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A close up of a logo

Description automatically generated

**Gambar 4. 2** Tampilan Dashboard

Gambar di atas menunjukkan visualisasi untuk sentiment dari data dan wordcloud ulasan. Visualisasi sentiment menunjukkan ketidakseimbangan antara sentiment positif dan negatif, dikarenakan mayoritas rating produk Hanafashion\_shop cenderung tinggi. Dalam wordcloud untuk visualisasi teks positif, kata yang paling dominan adalah 'bagus', sedangkan untuk visualisasi teks negatif, kata yang muncul dominan adalah 'barang'.

### Halaman Scraping

Pada tahap implementasi, user bisa melakukan proses scraping data ulasan toko Hanafashion\_shop di Tokopedia menggunakan bahasa pemrograman Python dan library Selenium. Selenium digunakan untuk mengendalikan browser dan melakukan tugas-tugas seperti mengklik, memasukkan teks, dan menelusuri halaman web. Dengan menggunakan Selenium, sistem dapat mengakses halaman ulasan toko Pengrajin.com, menemukan elemen HTML yang berisi ulasan, dan mengekstrak teks ulasan tersebut. Selanjutnya, data ulasan yang berhasil diambil akan disimpan dalam format yang sesuai untuk proses analisis sentimen selanjutnya. Adapun script dan tampilan scraping dapat dilihat pada gambar di bawah.



**Gambar 4. 3** Script Tampilan Scraping



**Gambar 4. 4** Tampilan Scraping

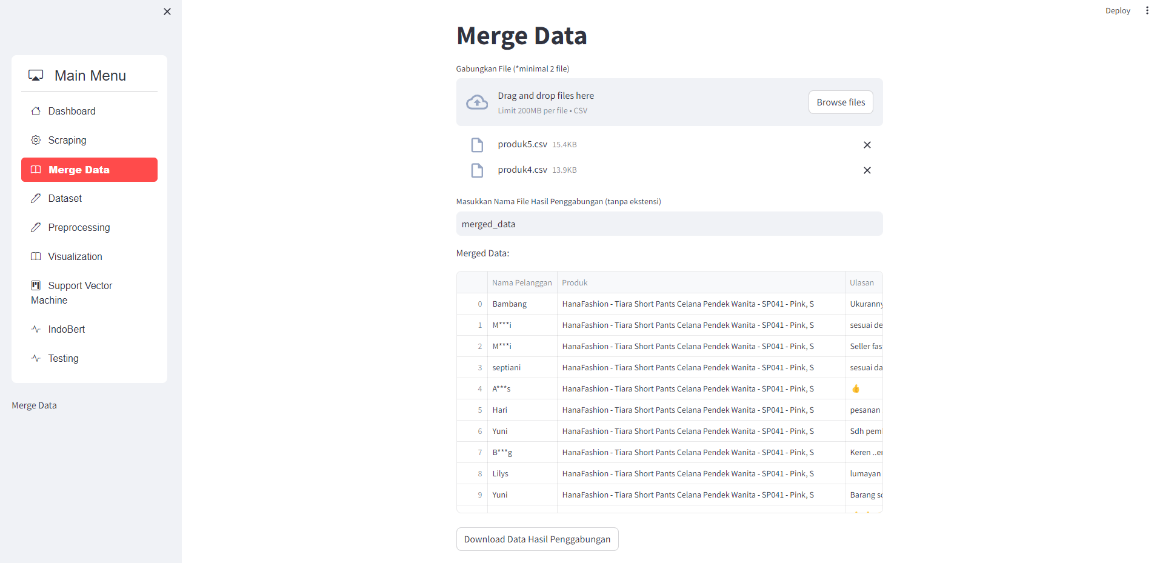
Implementasi halaman scraping di atas memanfaatkan fitur input URL, nama file, jumlah baris data, dan rentang rating produk yang ingin diambil. Pengguna diminta untuk memasukkan URL toko Hanafashion\_shop untuk menentukan produk yang akan diambil. Selanjutnya, mereka diminta untuk menentukan nama file tempat hasil scraping disimpan, jumlah baris data yang ingin diambil, dan rentang rating dari 1 hingga 5. Proses scraping bertujuan untuk mengumpulkan informasi penting seperti Nama Pelanggan, Nama Produk, Ulasan, dan Rating. Halaman scraping menampilkan kolom input URL dan gambaran visual dari elemen-elemen yang akan diambil dari toko Hanafashion\_shop.

### Halaman Merge Data

Pada tahap implementasi, user bisa menggabungkan data dari berbagai sumber menjadi satu set data yang lengkap dan terpadu. Dengan cara memetakan kolom-kolom dari masing-masing sumber dan menggunakan kriteria penggabungan yang ditentukan, seperti kunci primer atau kolom tertentu, halaman ini memungkinkan untuk menggabungkan baris-baris data yang sesuai. Tujuannya adalah untuk menciptakan satu set data yang komprehensif yang dapat digunakan untuk analisis lebih lanjut atau pelaporan, memungkinkan pengguna untuk mendapatkan pemahaman yang lebih baik tentang situasi atau tren yang ada dan membuat keputusan yang lebih terinformasi. Adapun script dan tampilan merge data dapat dilihat pada gambar di bawah.



**Gambar 4. 5** Script Merge Data



**Gambar 4. 6** Tampilan Merge Data

Implementasi halaman merge data di atas melibatkan penggabungan data untuk produk 9 dan produk 10 dari toko Hanafashion\_shop. Hasil penggabungan data ini dapat diunduh oleh user dan disimpan di komputer lokal mereka. Data yang dihasilkan dari penggabungan ini akan digunakan untuk proses lebih lanjut di halaman preprocessing, memungkinkan pengguna untuk melakukan persiapan data yang diperlukan sebelum melakukan analisis lebih lanjut.

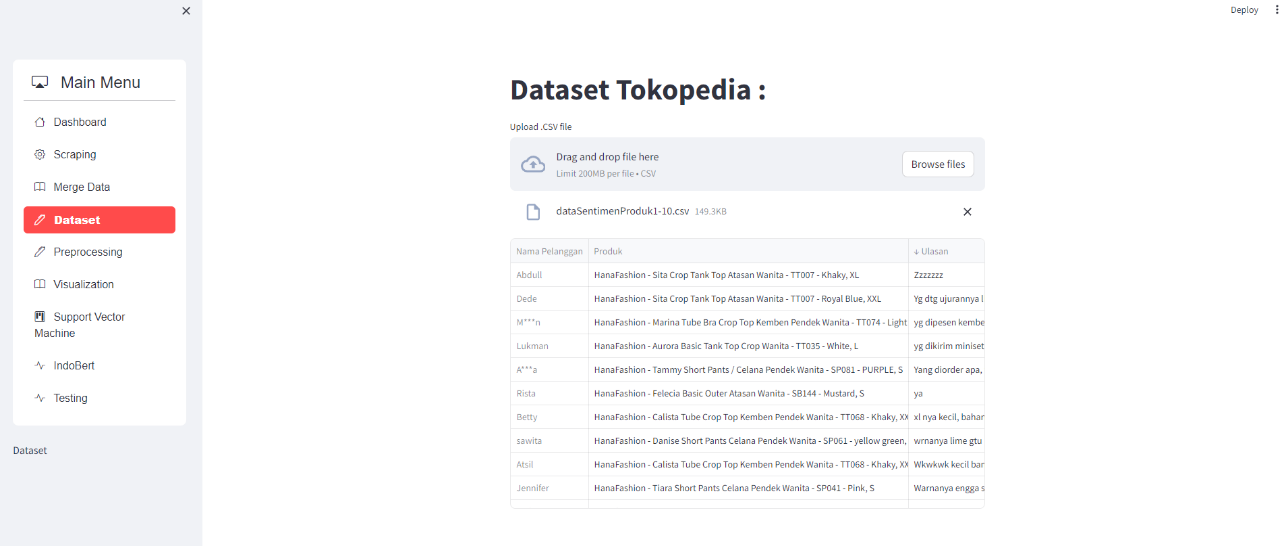
### Halaman Dataset

Pada tampilan dataset, user diberi akses untuk menampilkan dataset dari berbagai sumber dan proses dengan format .csv. Tujuannya adalah agar pengguna dapat memiliki gambaran menyeluruh tentang data yang akan mereka proses di tahap berikutnya. Dengan demikian, mereka dapat mengidentifikasi keberadaan data kosong atau duplikat, mempersiapkan langkah-langkah pembersihan yang diperlukan di tahap *preprocessing* data. Adapun script dan tampilan dataset dapat dilihat pada gambar di bawah.

A screenshot of a computer program

Description automatically generated

**Gambar 4. 7** Script Dataset



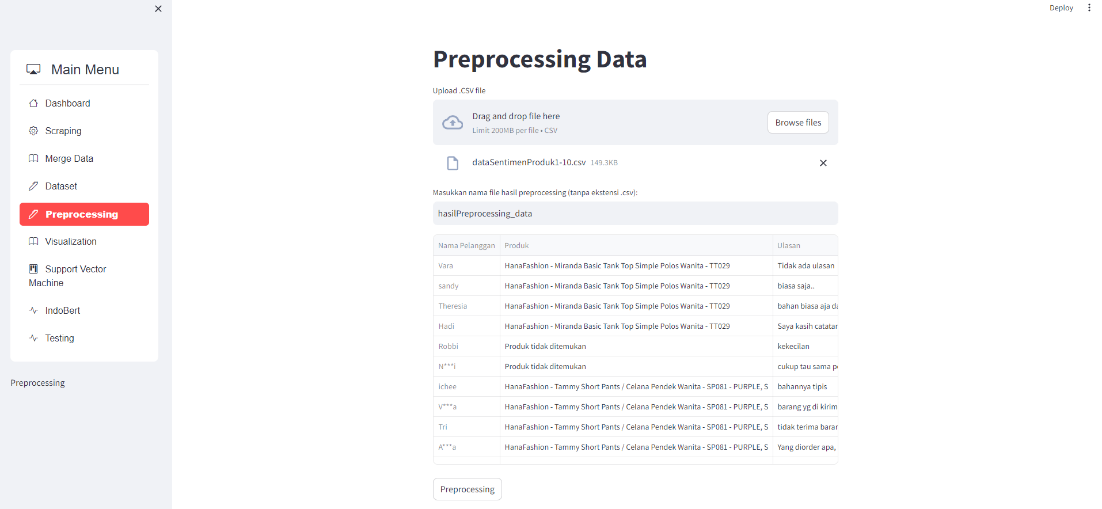
**Gambar 4. 8** Tampilan Dataset

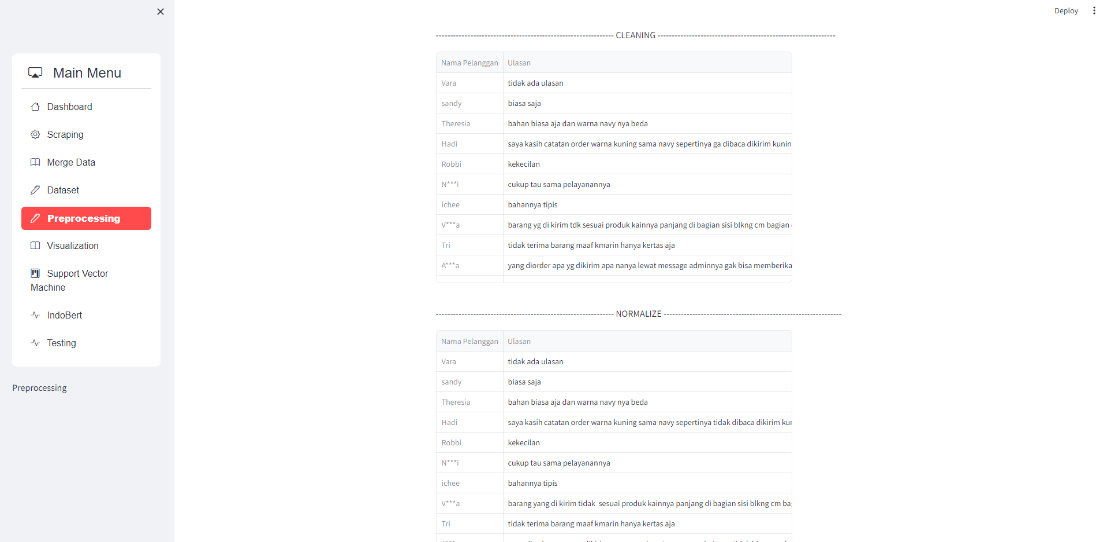
### Halaman Preprocessing

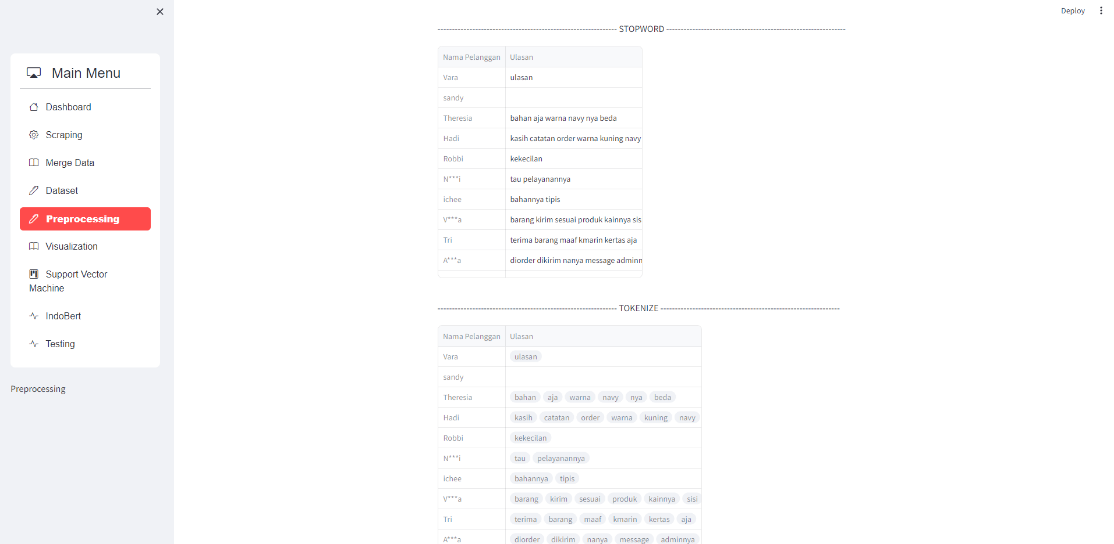
Pada tahap implementasi preprocessing, data ulasan toko Hanafashion\_shop yang telah berhasil diambil akan melalui serangkaian langkah untuk mempersiapkannya sebelum dilakukan analisis sentimen. Proses preprocessing meliputi langkah-langkah seperti menghapus karakter khusus, mengubah teks menjadi huruf kecil, menghilangkan stopwords (kata-kata umum yang tidak memberikan makna signifikan), serta melakukan tokenisasi untuk memisahkan kata-kata dalam ulasan. Selain itu, langkah-langkah tambahan seperti stemming atau lemmatisasi dapat dilakukan untuk mengubah kata-kata menjadi bentuk dasar mereka. Semua langkah ini bertujuan untuk membersihkan dan mempersiapkan data ulasan agar siap digunakan dalam proses analisis sentimen selanjutnya dengan metode *Support Vector Machine* dan *IndoBert*. Adapun tampilan dan script *preprocessing* dapat dilihat pada gambar di bawah.

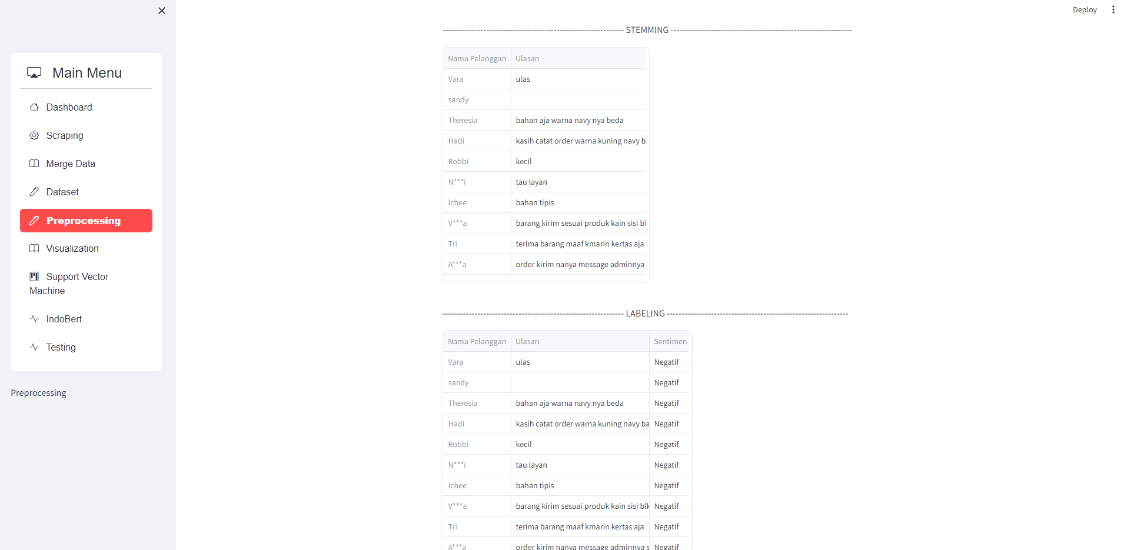


**Gambar 4. 9** Script Preprocessing









**Gambar 4. 10** Tampilan Preprocessing

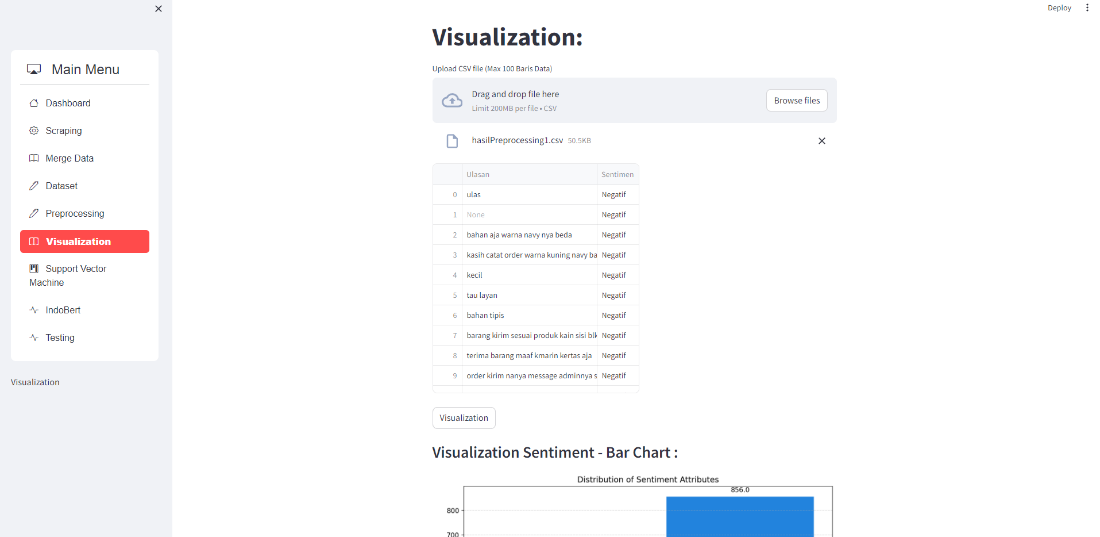
Pada tampilan diatas sistem melakukan preprocessing terhadap 1990 baris data. Tujuan dari preprocessing data adalah untuk mempersiapkan data agar memenuhi syarat sebelum dilakukan training ke model *Support Vector Machine* dan *IndoBert*. Proses dari tahapan preprocessing diantaranya yaitu cleaning, normalize, stopword, tokenize, stemming, dan labelling. Data tersebut beberapa atribut sebelum dilakukan preprocessing, lalu hasil akhir dari *preprocessing* hanya mengambil berapa dari kondisi atribut awal yaitu mengambil 996 dari 1113 Ulasan dan Rating, karena peneliti ingin melakukan pengolahan teks pada atribut ulasan sebelum dilakukan pengujian sentiment.

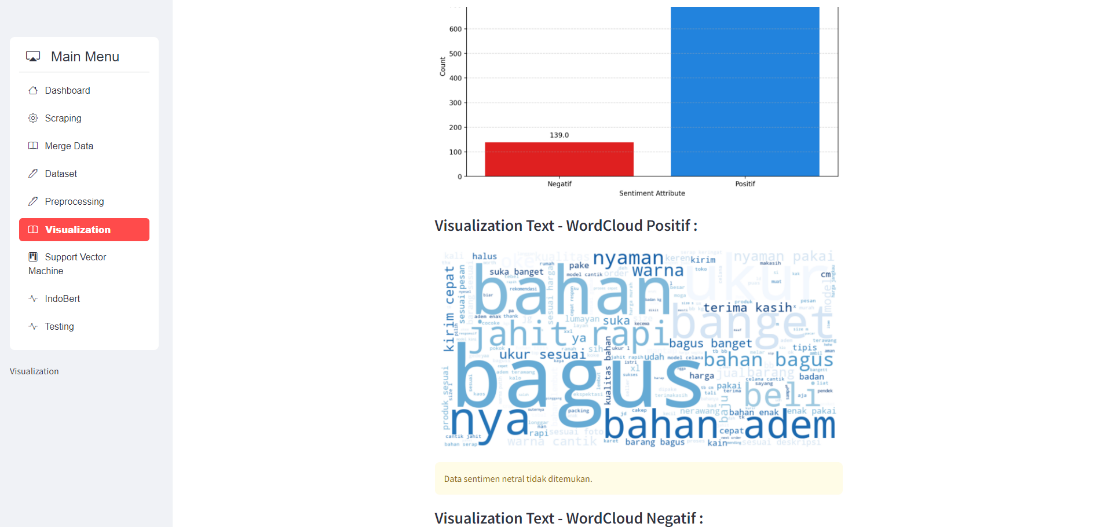
### Halaman Visualization

Pada tampilan visualization, fokusnya adalah pada visualisasi sentimen negatif dan positif serta visualisasi teks dari sentimen tersebut menggunakan wordcloud. Visualisasi sentimen negatif dan positif memungkinkan pengguna untuk dengan cepat memahami pola umum dari data sentimen, memungkinkan identifikasi tren dan pola yang mungkin tersembunyi. Sementara visualisasi teks menggunakan wordcloud memberikan representasi visual dari frekuensi kata-kata dalam teks, dengan ukuran kata yang lebih besar menunjukkan frekuensi yang lebih tinggi, memberikan wawasan langsung tentang tema atau topik yang dominan dalam sentimen negatif dan positif. Adapun script dan tampilan visualization dapat dilihat pada gambar di bawah.



**Gambar 4. 11** Script Visualization







**Gambar 4. 12** Tampilan Visualization

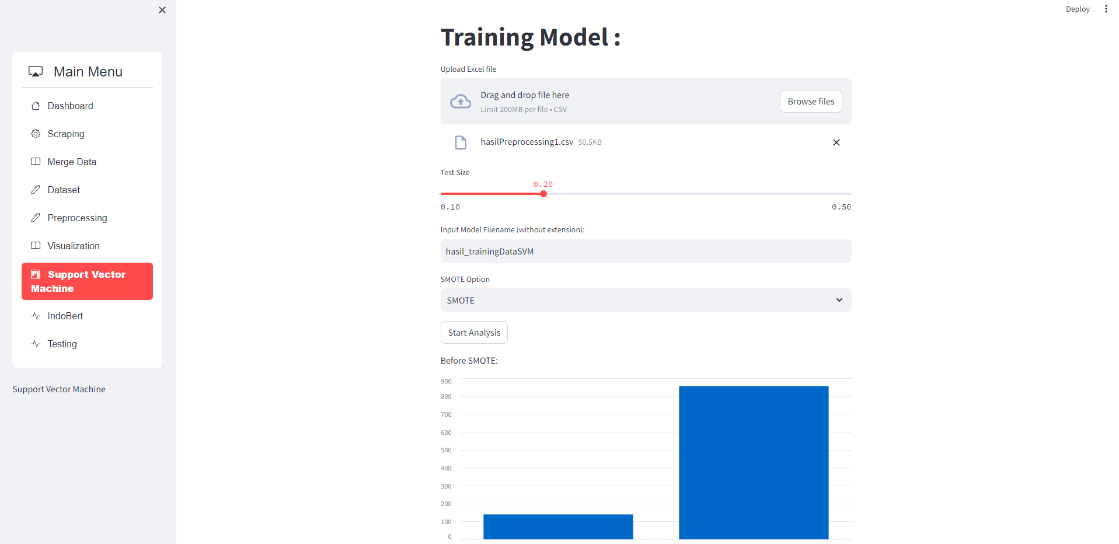
Pada hasil visualisasi diatas, user dapat dengan cepat mengidentifikasi sejumlah insight penting. Pertama, pola umum dari sentimen negatif dan positif dapat terlihat secara jelas, memberikan pemahaman mendalam tentang perasaan umum terhadap suatu topik atau produk yang ada di toko Hanafashion\_shop. Kedua, melalui wordcloud, dapat dilihat kata-kata kunci yang paling sering muncul dalam konteks sentimen negatif dan positif, memungkinkan pengguna untuk fokus pada aspek-aspek yang paling memengaruhi persepsi secara keseluruhan. Dengan demikian, halaman visualisasi memberikan pemahaman yang lebih dalam dan insight yang berguna bagi pengguna untuk mengambil tindakan yang sesuai.

### Halaman Support Vector Machine

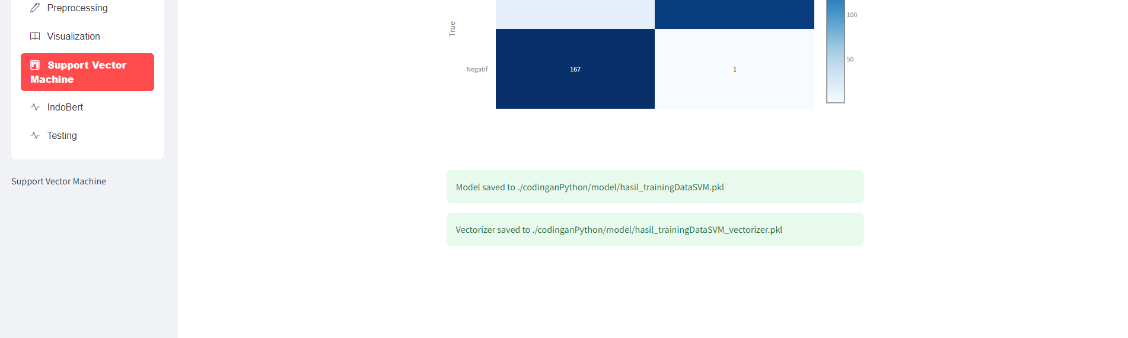
Pada tampilan *support vector machine*, terdapat beberapa komponen penting yang disajikan. Pertama, input data hasil preprocessing menjadi tahap awal yang mempersiapkan data untuk proses pelatihan model. Selanjutnya, terdapat input nama model dan vectorizer hasil dari proses pelatihan data, yang memungkinkan pengguna untuk melakukan proses testing dengan input ulasan dari user. Kemudian, proses splitting data train dan testing memberikan gambaran tentang bagaimana data dibagi menjadi dua subset untuk melatih model dan menguji performanya. Selain itu, terdapat opsi untuk menggunakan SMOTE (Synthetic Minority Over-sampling Technique) atau tanpa menggunakan SMOTE, yang memungkinkan pengguna untuk memilih apakah ingin menerapkan teknik oversampling pada data atau tidak, tergantung pada kebutuhan spesifik dan karakteristik dari dataset yang digunakan. Dengan demikian, halaman training data menyajikan informasi yang lengkap dan relevan untuk memahami proses pelatihan model SVM beserta opsi yang tersedia untuk meningkatkan kualitasnya.



**Gambar 4. 13** Script Support Vector Machine







**Gambar 4. 14** Tampilan Support Vector Machine

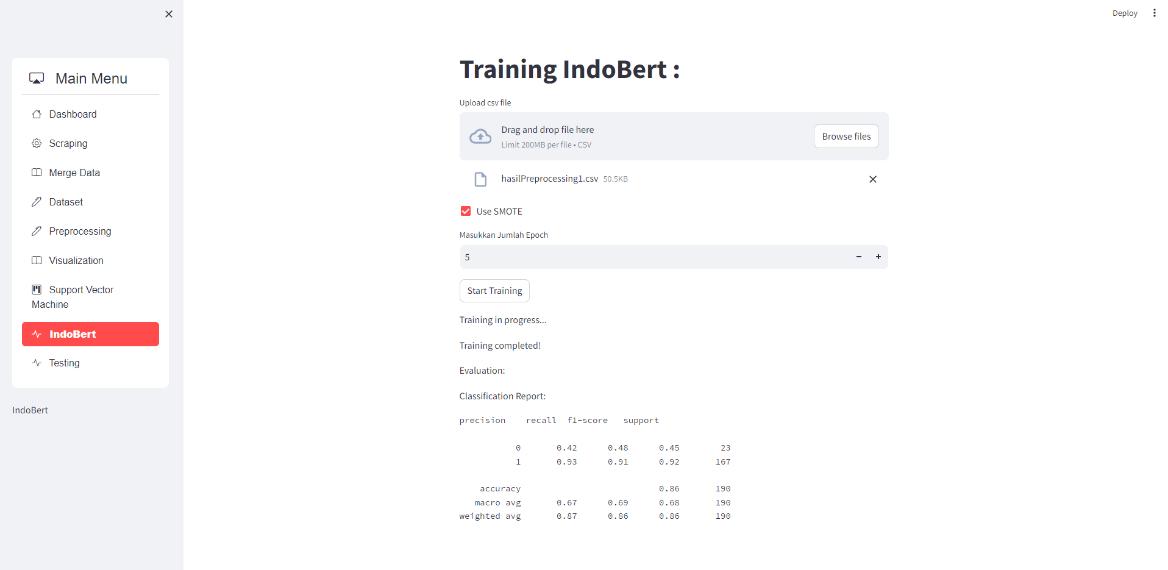
Pada gambar diatas, hasil *classification report* memberikan insight yang mendalam mengenai performa model *Support Vector Machine* yang telah dilatih. *Classification report* menyajikan metrik evaluasi seperti presisi, recall, dan f1-score untuk setiap kelas, serta akurasi secara keseluruhan. Dengan informasi ini, pengguna dapat memahami seberapa baik model SVM dalam mengklasifikasikan data ke dalam kelas yang berbeda dan mengidentifikasi area di mana model dapat ditingkatkan.

### Halaman IndoBert

Pada tampilan IndoBert, user diperlihatkan beberapa elemen yang esensial. Pertama, ada input data hasil preprocessing yang telah melalui tahap-tahap seperti tokenisasi, normalisasi, dan pembersihan data, mempersiapkan data untuk proses pelatihan model. Selanjutnya, terdapat opsi untuk menggunakan SMOTE (Synthetic Minority Over-sampling Technique) atau tanpa menggunakan SMOTE. Opsi ini memungkinkan user untuk menyesuaikan pendekatan penanganan ketidakseimbangan kelas dalam data mereka sesuai kebutuhan. Selain itu, user dapat memasukkan jumlah epoch yang diinginkan, mengontrol seberapa banyak iterasi melalui seluruh dataset dalam proses pelatihan model IndoBert. Dengan kombinasi opsi ini, pengguna memiliki fleksibilitas dalam menyesuaikan proses pelatihan model sesuai dengan kebutuhan dan karakteristik dataset mereka. Adapun script dan tampilan IndoBert dapat dilihat pada gambar di bawah.



**Gambar 4. 15** Script IndoBert



**Gambar 4. 16** Tampilan IndoBert

Pada gambar di atas, setelah proses pelatihan model menggunakan IndoBert dengan 5 epoch, hasilnya dapat dievaluasi menggunakan *classification report*. *Classification report* memberikan informasi rinci tentang performa model dalam mengklasifikasikan data ke dalam kelas yang berbeda. Metrik evaluasi seperti presisi, recall, dan f1-score disajikan untuk setiap kelas, bersama dengan akurasi secara keseluruhan. Dengan informasi ini, pengguna dapat memahami seberapa baik model IndoBert dalam mengenali setiap kelas, serta mengidentifikasi area di mana model mungkin perlu peningkatan. Misalnya, pengguna dapat melihat apakah ada kelas tertentu yang memiliki performa yang lebih rendah dan menyesuaikan strategi pelatihan atau parameter model sesuai kebutuhan.

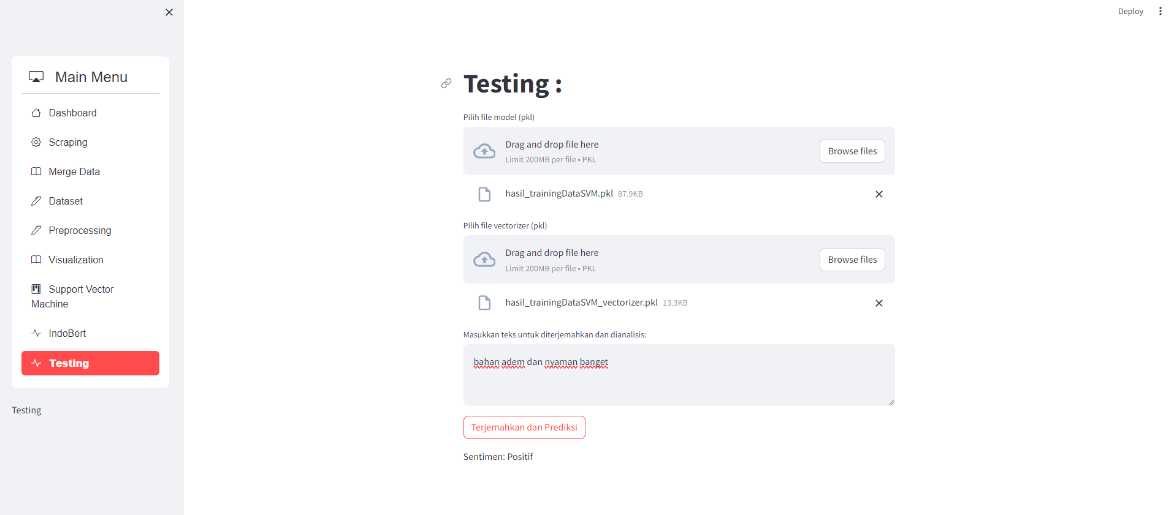
### Halaman Testing

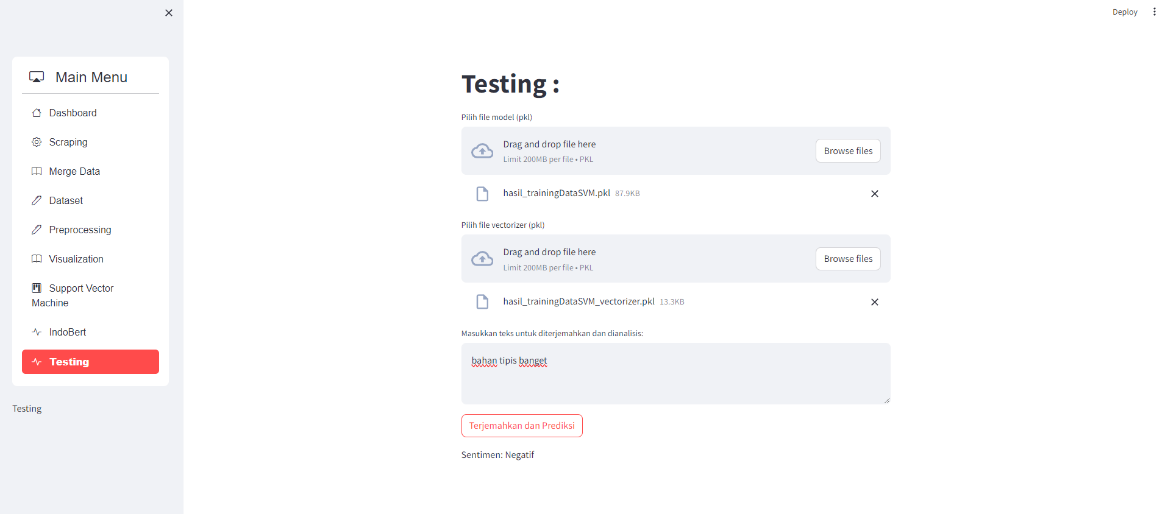
Pada tampilan testing, user diminta untuk memasukkan file model (model\_file) dan file *vectorizer* (vectorizer\_file) yang telah disimpan sebelumnya dari proses pelatihan. Selanjutnya, pengguna diminta untuk memasukkan teks yang akan dianalisis untuk sentimen. Setelah memasukkan teks, sistem akan menghasilkan output berupa sentimen negatif dan positif dari teks yang dimasukkan, yang membantu pengguna dalam memahami persepsi umum terhadap teks tersebut. Dengan demikian, pengguna dapat dengan cepat mengevaluasi sentimen dari teks yang dianalisis tanpa perlu melalui proses pelatihan ulang. Adapun script dan tampilan testing dapat dilihat pada gambar di bawah.

A screenshot of a computer program

Description automatically generated

**Gambar 4. 17** Script Testing





**Gambar 4. 18** Tampilan Testing

Pada hasil tampilan di atas dengan input "bahan adem dan nyaman banget" yang diklasifikasikan sebagai positif dan "bahan tipis banget" yang diklasifikasikan sebagai negatif menggunakan model hasil pelatihan data menggunakan SVM, pengguna dapat langsung melihat sentimen yang terkait dengan teks yang dimasukkan. Dengan demikian, pengguna bisa mendapatkan pemahaman yang cepat dan jelas tentang apakah teks tersebut dipandang positif atau negatif oleh model, memungkinkan pengambilan keputusan yang lebih tepat dan efisien terkait dengan analisis sentimen.