

Analisis Sentimen pada Ulasan Google Maps Universitas Bina Darma Menggunakan Algoritma Random Forest

Erwin Satriansyah, Nurul Huda, Muhammad Izman Herdiansyah, dan Nurul Adha Oktarini Saputri

Universitas Bina Darma

Jl. Jenderal Ahmad Yani No.3, 9/10 Ulu, Kecamatan Seberang Ulu I, Kota Palembang, Sumatera Selatan

E-mail: erwin.satriansyah25@gmail.com, nurul_huda@binadarma.ac.id,

m.herdiansyah@binadarma.ac.id, nuruladhaos@binadarma.ac.id

Abstrak

Perkembangan teknologi informasi dan komunikasi telah membawa perubahan signifikan dalam interaksi dan pertukaran informasi, salah satunya melalui platform ulasan online seperti *Google Maps*. Ulasan ini menjadi sumber informasi penting bagi calon mahasiswa dan masyarakat dalam menilai kualitas institusi pendidikan. Penelitian ini bertujuan untuk menganalisis sentimen pada ulasan Universitas Bina Darma, dengan harapan dapat memberikan wawasan mengenai faktor-faktor yang mempengaruhi kepuasan mahasiswa, seperti kualitas pengajaran, fasilitas, dan layanan administrasi. Penelitian ini menggunakan model CRISP-DM (*Cross Industry Standard Process for Data Mining*) sebagai pedoman dalam analisis. Analisis dilakukan dengan menggunakan data hasil *web scraping* dari ulasan *Google Maps* pada Universitas Bina Darma. Dalam penelitian ini, algoritma *random forest* dengan teknik SMOTE digunakan untuk menyeimbangkan data sekaligus meningkatkan kinerja algoritma *random forest* dengan menggunakan ekstraksi fitur TF-IDF dan *k-fold cross validation*. Analisis terhadap 875 data ulasan hasil pembersihan menggunakan rasio 70:30 dalam membagi data latih dan data uji. Hasil evaluasi menunjukkan bahwa algoritma *random forest* dengan teknik SMOTE mencapai nilai akurasi tertinggi yaitu 92%. Performa yang dihasilkan algoritma *random forest* cukup baik dalam analisis sentimen ulasan *Google Maps*. Analisis ini diharapkan mampu menjadi masukan dan wawasan untuk mendukung meningkatkan reputasi dan daya saing Universitas Bina Darma di dunia pendidikan.

Kata kunci : Analisis Sentimen, *Random Forest*, *Term Frequency-Inverse Document Frequency*

Pendahuluan

Perkembangan teknologi informasi dan komunikasi telah mengubah cara orang berinteraksi dan berbagi informasi. Salah satu dampak signifikan dari perkembangan ini adalah munculnya platform ulasan *online*, seperti *Google Maps*, yang memungkinkan pengguna untuk memberikan penilaian dan komentar tentang berbagai tempat, termasuk institusi pendidikan. *Google Maps* adalah layanan pemetaan *web* atau peta *online* gratis dan merupakan salah satu produk yang dikembangkan oleh *Google*. Berbagai macam fitur yang dapat dilihat dalam *Google Maps* yaitu peta jalan, citra satelit, foto udara, pemandangan jalan dengan interaktif 360°, lalu lintas real time [1]. *Google Maps* adalah bagian dari produk *Google* yang memberikan informasi Lokasi dan petunjuk arah kepada penggunaanya [2]. Suatu Lembaga Pendidikan khususnya

pada universitas harus tahu bagaimana masyarakat melihat mereka untuk meningkatkan daya saing dan reputasi. Ulasan yang ada di *Google Maps* dapat memberikan gambaran tentang kelebihan dan kekurangan yang ada di universitas tersebut.

Informasi yang dikumpulkan dari *Google Maps* akan dianalisis untuk mendapatkan sebuah kesimpulan sentimen atau yang biasa disebut analisis sentimen. Analisis sentimen adalah terapan dari *text mining* yang hasilnya berupa positif, negatif, dan netral [3]. *Text Mining* merupakan proses mengeksplorasi dan menganalisis data teks tidak terstruktur dalam jumlah besar dengan menggunakan perangkat lunak yang dapat mengidentifikasi konsep, pola, topik, kata kunci, dan atribut lain dalam data. Cara kerja teknik ini adalah dengan mengekstraksi variasi dalam jumlah besar data yang tersedia dan mencoba menemukan pola yang sesuai dengan apa yang diharapkan dari kumpulan

data teks yang ada [4]. Analisis sentimen adalah proses otomatis yang melibatkan pemahaman dan transformasi data teks untuk mengungkap informasi yang terkandung dalam kalimat atau teks dalam bentuk opini. Tujuan utama analisis sentimen adalah untuk menentukan apakah sentimen yang diungkapkan dalam teks memiliki hubungan positif atau negatif dengan isu atau objek tertentu. Analisis sentimen juga dapat membantu menyampaikan pandangan atau opini orang tentang isu, peristiwa, kepuasan layanan, prediksi harga saham, dan analisis persaingan melalui data teks [5]. Dalam analisis sentimen, beberapa algoritma dapat digunakan, seperti *Naïve Bayes Classifier*, *K-NN*, *Support Vector Machine*, dan *Random Forest*.

Random Forest adalah serangkaian decision trees yang dibangun dari sampel secara acak, tetapi dengan membelah simpul yang berbeda. Model ini bekerja dengan menggunakan subset fungsi di setiap pohon dan kemudian mencoba menemukan ambang batas terbaik untuk memisahkan data [6]. Sebagian besar analisis sentimen ini berhubungan dengan orang di media social. Selain polaritas positif dan negatif, polaritas terkadang juga dianggap sebagai rentang di mana suatu dokumen dapat berisi pernyataan dengan polaritas campuran [7]. Beberapa penelitian menggunakan algoritma random forest yaitu pada penelitian Jennifer [8] melakukan analisis sentiment pada komentar pengguna TikTok menggunakan algoritma *Random Forest* dengan *dataset* 15.000 komentar dan menghasilkan akurasi sebesar 92%, dengan presisi dan *recall* diatas 80%. Hal ini menunjukkan performa algoritma yang baik dalam konteks sosial media. Pada penelitian selanjutnya [9] tentang analisis sentimen analisis dalam opini youtube menggunakan *Word2Vec* dan *Random Forest* dengan 31.947 data komentar yang diunduh dari youtube debat pemilihan umum presiden 2019 yang menghasilkan akurasi model antara 90,1% dan 91%. Selain itu penelitian Larasati [3] yang menganalisis sentiment ulasan pada aplikasi DANA menggunakan algoritma *Random Forest* mencatat akurasi sebesar 85%, dengan distribusi sentiment yang didominasi oleh kelas positif. Penelitian lain oleh Sriani [5] menggunakan algoritma *Naïve Bayes* dan C4.5 dalam menganalisis aplikasi JKN Mobile dan menemukan bahwa C4.5 lebih unggul dalam presisi, algoritma tersebut membutuhkan parameter tuning yang kompleks. Beberapa penelitian tersebut memperlihatkan relevansi model berbasis *decision tree* untuk klasifikasi teks. Meskipun sebagian penelitian menggunakan platform berbeda, hasil menunjukkan bahwa *Random Forest* memiliki kemampuan yang andal dalam klasifikasi teks. Namun, belum ditemukan penelitian yang secara spesifik menggunakan data ulasan institusi Pendidikan dari *Google Maps* seperti yang dilakukan dalam studi ini, menjadikan konteks penelitian ini lebih unik dan relevan bagi pengembangan reputasi institusi pendidikan.

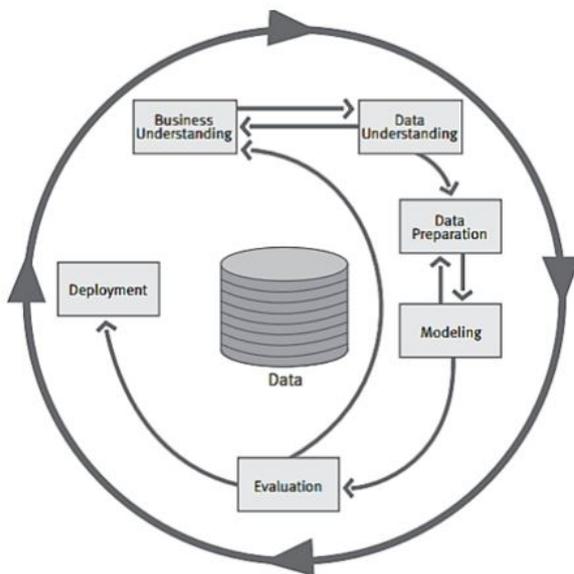
Data ulasan pada Universitas Bina Darma di

Google Maps menggunakan teknik web scraping, yaitu metode pengambilan informasi dari halaman *web* untuk membuat data yang dapat dianalisis dan digunakan untuk berbagai tujuan [10]. *Web scraping* merupakan metode memperoleh informasi secara otomatis dari situs web, dan data diperoleh melalui ekstraksi. Penelitian ini menggunakan aplikasi *web SerpApi* untuk scraping pada ulasan *Google Maps*. Hasil data yang didapatkan selanjutnya dilakukan *text preprocessing*. *Text preprocessing* adalah proses pembersihan teks sehingga memiliki kualitas yang baik dan memenuhi persyaratan untuk dieksekusi dalam suatu algoritma. Beberapa langkah pada pemrosesan teks adalah *case folding*, *cleansing*, *tokenization*, *normalisasi slang word*, *stemming*, dan *stopword*. Selanjutnya melakukan *labelling* menggunakan VADER untuk menentukan apakah teks tersebut termasuk dalam kategori positif, negatif, atau netral. Vader merupakan salah satu algoritma yang dirancang untuk analisis sentimen dengan cepat, akurat dan efektif dalam situasi yang memerlukan respon cepat tanpa memerlukan pelatihan yang rumit. Algoritma ini didasarkan pada kamus kata yang diberi skor sentiment dan aturan linguistik untuk menafsirkan teks. VADER juga cenderung mendeteksi sentimen negatif lebih baik dibandingkan algoritma serupa seperti RoBERTa dan Textblob [11].

Dalam penelitian ini, analisis sentimen diterapkan untuk menentukan jenis komentar masyarakat apakah termasuk ke dalam kategori positif, negatif, atau netral dan aspek yang dibahas dalam ulasan seperti kebersihan, fasilitas, dan layanan. Teks editor yang digunakan untuk pengolahan data teks menggunakan bahasa pemrograman *python* dengan memanfaatkan aplikasi *Google Colab*. *Google Colab* merupakan IDE (*Integrated Environment Development*) dengan jenis *Jupyter Notebook* dengan ekstensi file **.ipynb* berbasis *web*. *Google Colab* menyediakan berbagai macam fasilitas Sebagian besar *library* perangkat lunak maupun perangkat keras seperti media penyimpanan bawaan *Google Drive*, prosesor CPU, TPU, dan RAM, serta media lainnya, seperti *Matplotlib* untuk membuat grafik [12].

Berdasarkan uraian diatas, tujuan penelitian ini dilakukan untuk melakukan analisis sentimen pada ulasan Universitas Bina Darma dengan menggunakan algoritma random forest, penelitian ini diharapkan dapat memberikan manfaat bagi proses pengembangan rencana untuk meningkatkan kualitas pendidikan di Universitas. Selain itu, hasilnya dapat membantu Universitas Bina Darma meningkatkan reputasinya dan daya saingnya di dunia pendidikan melalui pemahaman yang lebih baik tentang persepsi publik.

Metode Penelitian



Gambar 1: Metode CRISP-DM

Penelitian ini menggunakan metode CRISP-DM (*Cross-Industry Standard Process for Data Mining*) yang bertujuan untuk menyediakan rencana langkah demi langkah untuk proses pengumpulan data. Langkah-langkah tersebut terbagi menjadi enam, yaitu *business understanding*, *data understanding*, *data preparation*, *modeling*, *evaluation*, dan *deployment* [13]. Metode ini dikenal sebagai metodologi data yang lengkap untuk kebutuhan proyek bisnis dan merupakan salah satu metode yang paling banyak digunakan dalam bidang ilmu data [14]. Dapat dilihat pada Gambar 1 terdapat 6 langkah CRISP-DM sebagai berikut:

1. *Business Understanding*

Pada tahap awal, dilakukan observasi pada ulasan *Google Maps*, kemudian membangun model analisis sentimen dengan menggunakan metode klasifikasi dan kemudian buat perbandingan dengan penelitian sebelumnya dengan mempelajari studi pustaka dan literatur serupa.

2. *Data Understanding*

Selanjutnya pada tahap kedua, dilakukan pengumpulan data awal dari ulasan pada Universitas Bina Darma di *Google Maps* dengan melakukan web scrapping data sebanyak 1.000 ulasan menggunakan aplikasi *SerpApi*. Data ulasan yang dikumpulkan yang telah diambil berupa nama akun, rating, dan ulasan pengguna.

3. *Data Preparation*

Pada tahap ketiga, dilakukan persiapan data meliputi pembersihan data, pemberian label, pembagian data latih dan data uji, dan

ekstraksi fitur. Menyiapkan *dataset* yang akan digunakan untuk pemodelan data mining dengan melakukan pembersihan data, penghapusan karakter (*case folding, cleansing, tokenization, slang word, stemming, stopwords*). Kemudian pelabelan data set hasil *text preprocessing* didasarkan pada pendekatan berbasis *lexicon* dengan menggunakan *library Python* yaitu *Vader (Valence Aware Dictionary and Sentiment Reasoner)*. Metode pelatihan vader didasarkan pada pendekatan *human-electric*, menggabungkan analisis kualitatif dan validasi empiris dengan kebijaksanaan dan penilaian manusia. Vader digunakan untuk mengklasifikasikan tiap data ke dalam kelas positif, negatif, atau netral berdasarkan penyatuan atribut *compound* [15]. Pada tahap *text preprocessing* dilakukan pembagian data latih (70%) dan data uji (30%).

4. *Modeling*

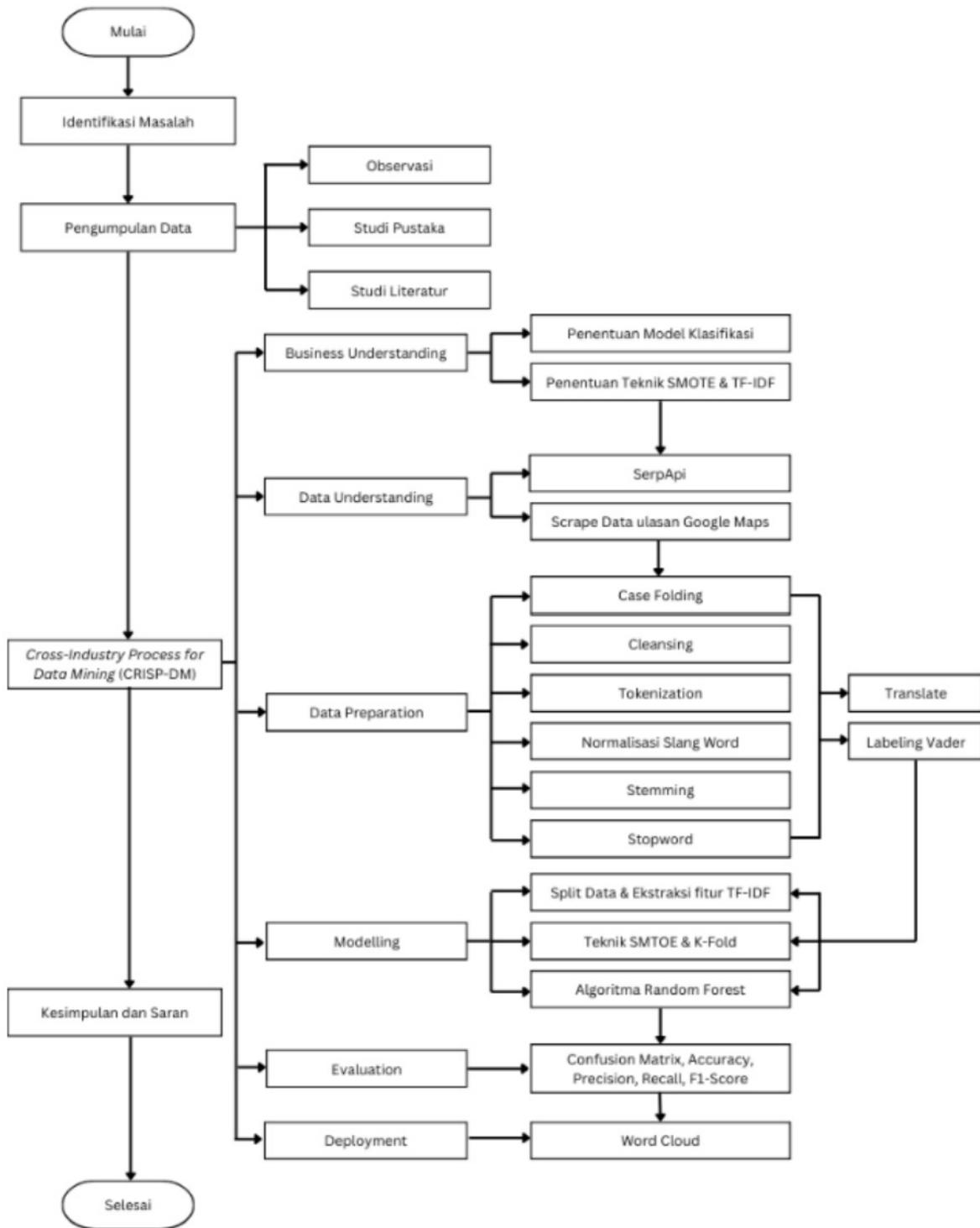
Pada tahap keempat, dilakukan pemodelan menggunakan algoritma *Random Forest* dan menggunakan Teknik SMOTE untuk menyeimbangkan data latih dan meningkatkan kinerja *Random Forest* lalu menggunakan *K-fold cross validation* untuk menilai lebih lanjut akurasi model. SMOTE adalah suatu algoritma yang berfungsi memverifikasi distribusi data yang seragam dalam suatu kelas kumpulan data dengan membuat sampel buatan dari kelas minoritas. Konsepnya adalah memilih sampel secara acak dari kelas minoritas, lalu menghitung nilai k terdekat sebagai sampel, lalu sampel buatan ditambahkan ke kumpulan data [16]. Dalam model ini, kumpulan data awal dipisahkan secara acak menjadi k kali lipat yang sama. Proses validasi *K-Fold Cross Validation* dimulai dengan membagi data menjadi kelompok dengan ukuran yang sama disebut lipatan (k).

5. *Evaluation*

Pada tahap kelima, mengukur dan membandingkan kinerja algoritma *Random Forest* sebelum dan sesudah menggunakan SMOTE berdasarkan akurasi, presisi, *recall*, dan *f-1 score* untuk menerima model dengan kinerja terbaik. Kemudian menilai efektivitas *fold* terbaik dari ke- k menggunakan *Confusion Matrix*.

6. *Deployment*

Pada tahap terakhir deployment, dilakukan visualisasi teks yang banyak muncul pada tiap kelas sentiment menggunakan word cloud berdasarkan hasil model klasifikasi terbaik. Adapun alur dari keseluruhan tahapan dapat dilihat pada Gambar 2.



Gambar 2: Alur penelitian

Ekstraksi Fitur

Ekstraksi fitur menggunakan TF-IDF (*Term Frequency Inverse Document Frequency*) merupakan Teknik pengambilan informasi yang mempertimbangkan term frequency dan dokumen inversnya (IDF). Algoritma ini digunakan untuk memberi bobot pada kata kunci dalam setiap dokumen dan menghitung berapa kali kata kunci tersebut muncul

dalam setiap dokumen [7]. Menurut [17] TF-IDF memiliki beberapa keterbatasan dalam konteks sistem pencarian informasi dalam dokumen teks. Metode ini tidak memperhitungkan konteks mana atau struktur Bahasa Ketika memilih dan mengklasifikasikan dokumen. Hal ini dapat menyebabkan ketidakakuratan dalam pencarian dokumen yang memenuhi kebutuhan pengguna jika terdapat istilah yang memiliki makna ganda atau jika kon-

teksnya perlu diperhatikan. Metode ini mempunyai kelemahan dalam pengolahan dokumen yang Panjang berbeda. Persamaan untuk menghitung TF-IDF ditunjukkan pada persamaan 1.

$$W_{ij} = tf_{ij} \times \log \left(\frac{N}{df} \right) \quad (1)$$

Ket:

tf_{ij} =Jumlah kata i muncul dalam dokumen j

df_i =Jumlah dokumen yang mengandung i

N =Total seluruh dokumen

Evaluasi *Confusion Matrix*

Untuk mengevaluasi dan mengukur kinerja suatu model pengetahuan yang dihasilkan oleh proses pelatihan secara umum dengan menggunakan *confusion matrix*. *Confusion matrix* adalah model nilai fungsi untuk menentukan apakah klasifikasi metode yang dievaluasi memiliki label positif, negatif, atau netral [3].

Tabel 1: *Confusion Matrix* 2 kelas

Kelas Aktual	Kelas Prediksi	
	Positive	Negative
Positive	(TP) True Positive	(FN) False Negative
Negative	(FP) False Positive	(TN) True Negative

Dalam beberapa penelitian, nilai accuracy masih digunakan sebagai variable utama untuk mengukur kinerja model. akurasi menunjukkan seberapa dekat hasil klasifikasi dengan nilai sebenarnya. Akurasi diperoleh dengan membandingkan antara data yang diklasifikasikan dengan benar dengan seluruh kumpulan data. Berikut rumus untuk menghitung akurasi pada *confusion matrix*.

$$accuracy = \frac{TP + TN}{TP + FP + FN + TN} \quad (2)$$

K-Fold Cross Validation

Dalam proses model *K-Fold cross validation* ini, kumpulan data awal dipisahkan secara acak menjadi k kali lipat yang sama. Proses validasi *K-Fold Cross Validation* dimulai dengan membagi data menjadi kelompok dengan ukuran yang sama disebut lipatan (k). Pada setiap putaran, lipatan yang berbeda digunakan untuk *data test*, sedangkan lipatan lainnya digabungkan untuk *data train* [18].

Hasil dan Pembahasan

Business Understanding

Pada ulasan *Google Maps* terdapat sebuah rating untuk menilai dan kolom ulasan, namun

ulasan dan rating terkadang tidak relevan atau ulasan masyarakat terlihat jelek tetapi rating yang diberikan termasuk dalam kategori baik. Dalam hal ini, analisis dilakukan untuk melihat apakah ulasan Masyarakat publik termasuk dalam sentiment positif, negatif, atau netral. Sebelum melakukan analisis menggunakan algoritma *Random Forest*, yaitu melakukan teknik TF-IDF untuk ekstraksi fitur kemudian teknik SMOTE untuk menyeimbangkan data. Hasil klasifikasi berdasarkan nilai terbaik dari *K-fold cross validation* dengan nilai *accuracy*, *recall*, *precision*, dan *f-1 score*.

Data Understanding

Pengumpulan data awal melalui *web scraping* pada *Google Maps* menggunakan api yang telah disediakan oleh SerpApi. Total ulasan Universitas Bina Darma pada *Google Maps* sebanyak 1.669 ulasan dengan rating 4.8. Namun, data set yang berhasil diambil menggunakan pustaka SerpApi *python* menghasilkan 1000 ulasan yaitu dari semua ulasan yang memiliki komentar sampai dengan yang kosong lalu disimpan dalam format CSV. Lihat Tabel 2.

Tabel 2: Hasil *Scraping*

Nama	Rating	Komentar
Thania f	5.00	Universitas Bina Darma adalah salah satu kampus terbaik di kota Palembang dan juga kampus ternyaman serta banyak di minati oleh seluruh pemuda dan pemudi.
miko bayu	1.00	Dari luar aja kelihatan jelek kampus nya, materi perkuliahan nya gak berkualitas, masih mending ikut les aja, materi nya berkualitas diajarin sampai bisa, beda dengan di kampus bina darma, ngajar sesuka hati, kalau gak bisa ngerjain tugas dikatain bodoh, kayak dia pinter aja, rugi saya kuliah di bina darma

Data Preparation

Fase persiapan data merupakan fase untuk memastikan data siap digunakan dan untuk dianalisis dengan cara manipulasi data mentah hasil *scraping* menghasilkan data set terstruktur yang siap digunakan pada tahap *modeling*.

Text Preprocessing

Text Prerocessing melibatkan beberapa teknik seperti *case folding*, *cleansing*, *tokenizing*, *slang word*, *stemming*, dan *stopword*. Semua langkah bekerja sama untuk memastikan model berfokus pada bagian tinjauan yang paling relevan untuk analisis sentiment yang akurat. Langkah pertama adalah *case folding*, yaitu proses mengubah seluruh teks atau kalimat pada *dataset* jadi huruf kecil atau *lower case*, lihat Tabel 3.

Tabel 3: Hasil *Case Folding*

Ulasan	Case Folding
Universitas Bina Darma adalah universitas swasta yang banyak memiliki program studi. Jadi tunggu apalagi! Ayo kita daftar di Universitas ini.	universitas bina darma adalah universitas swasta yang banyak memiliki program studi. jadi tunggu apalagi! ayo kita daftar di universitas ini.

Cleansing atau membersihkan Kumpulan data dari karakter atau elemen lain yang tidak relevan lainnya, seperti tanda baca, karakter khusus, maupun *emoticon*. *Cleansing* juga mencakup penghapusan spasi berlebih atau karakter ganda yang dianggap tidak informatif dan dapat mengganggu analisis, lihat Tabel 4.

Tabel 4: Hasil *Cleansing*

Case Folding	Cleansing
universitas bina darma adalah universitas swasta yang banyak memiliki program studi. jadi tunggu apalagi! ayo kita daftar di universitas ini.	universitas bina darma adalah universitas swasta yang banyak memiliki program studi jadi tunggu apalagi ayo kita daftar di universitas ini

Tahap ketiga melibatkan *tokenization*, yaitu membagi data teks menjadi karakter tunggal atau kata individual yang disebut dengan “token”. Proses tokenizing menggunakan pustaka NLTK, lihat Tabel 5.

Tabel 5: Hasil *Tokenization*

Cleansing	Tokenization
universitas bina darma menjadi pts terbaik swasta di sumatera selatan berdasarkan edu rank kampus dengan fasilitas yang lengkap bermutu baik dari pelayanan	['universitas', 'bina', 'darma', 'menjadi', 'pts', 'terbaik', 'swasta', 'di', 'sumatera', 'selatan', 'berdasarkan', 'edu', 'rank', 'kampus', 'dengan', 'fasilitas', 'yang', 'lengkap', 'bermutu', 'dari', 'pelayanan']

Slang word Normalization merupakan perbaikan bahasa tidak resmi menjadi kata baku berdasarkan data set ‘Kamus Alay’ yang dikembangkan oleh [19]. Mengubah teks token apabila ditemukan atau terdaftar pada data set kamus, lihat Tabel 6.

Tabel 6: Hasil *Slang Word*

Tokenization	Slang Word
['kalau', 'untuk', 'mahasiswa', 'yg', 'sambil', 'kerja', 'mending', 'jgn', 'di', 'sini', 'karna', 'kelas', 'karyawan', 'dgn', 'kelas', 'biasa', 'sama', 'aja', 'kalian', 'di', 'tuntut', 'seperti', 'mahasiswa', 'kelas', 'biasa', 'jd', 'tidak', 'ad', 'ke', 'mudahan', 'buat', 'mahasiswa', 'kelas', 'karyawan']	['kalau', 'untuk', 'mahasiswa', 'yang', 'sambil', 'kerja', 'mending', 'jangan', 'di', 'sini', 'karena', 'kelas', 'karyawan', 'dengan', 'kelas', 'biasa', 'sama', 'saja', 'kalian', 'di', 'tuntut', 'seperti', 'mahasiswa', 'kelas', 'biasa', 'jadi', 'tidak', 'ada', 'ke', 'mudahan', 'buat', 'mahasiswa', 'kelas', 'karyawan']

Tahap selanjutnya stemming, yaitu proses menghapus kata awalan atau akhiran kata sehingga menjadi kata dasarnya. Stemming membantu mengidentifikasi kata-kata dengan akar yang sama meskipun kata tersebut sedikit berbeda dalam tense atau konjugasi, lihat Tabel 7.

Tabel 7: Hasil *Stemming*

Slang Word	Stemming
['universitas', 'bina', 'darma', 'ubd', 'palembang', 'merupakan', 'perguruan', 'swasta', 'pts', 'terbaik', 'di', 'sumatra', 'selatan', 'pelayanan', 'kampus', 'ini', 'sangat', 'ramah', 'baik', 'itu', 'dari', 'dosen', 'pegawai', 'sangat', 'ramah', 'baik', 'itu', 'dari', 'dosen', 'pegawai', 'serta', 'kating', 'dengan', 'kampusnya', 'juga', 'bersih', 'dan', 'dilengkapi', 'dengan', 'fasilitas', 'yang', 'memadai', 'sangat', 'reccomended', 'bagi', 'yang', 'masih', 'bingung', 'cari', 'kampus', 'untuk', 'melanjutkan', 'pendidikan']	['universitas', 'bina', 'darma', 'ubd', 'palembang', 'rupa', 'guru', 'tinggi', 'swasta', 'pts', 'baik', 'di', 'sumatra', 'selatan', 'layan', 'kampus', 'ini', 'sangat', 'ramah', 'baik', 'itu', 'dari', 'dosen', 'pegawai', 'serta', 'kating', 'kampus', 'juga', 'bersih', 'dan', 'lengkap', 'dengan', 'fasilitas', 'yang', 'pada', 'sangat', 'reccomended', 'bagi', 'yang', 'masih', 'bingung', 'cari', 'kampus', 'untuk', 'lanjut', 'didik']

Tahap terakhir pada *text preprocessing* yaitu *stopword*. *Stopword* merupakan proses menghilangkan kata yang tidak memiliki arti penting dan tidak berkontribusi besar dalam memahami konten, lihat Tabel 8.

Tabel 8: Hasil *Stopword*

Stemming	Stopword
['universitas', 'bina', 'darma', 'salah', 'satu', 'kampus', 'baik', 'di', 'sumatra', 'selatan', 'dan', 'no', 'kampus', 'swasta', 'baik', 'di', 'sumatera', 'selatan', 'bina', 'darma', 'hebat', 'dan', 'mutu']	['universitas', 'bina', 'darma', 'salah', 'kampus', 'sumatra', 'selatan', 'kampus', 'swasta', 'sumatera', 'selatan', 'bina', 'darma', 'hebat', 'mutu']

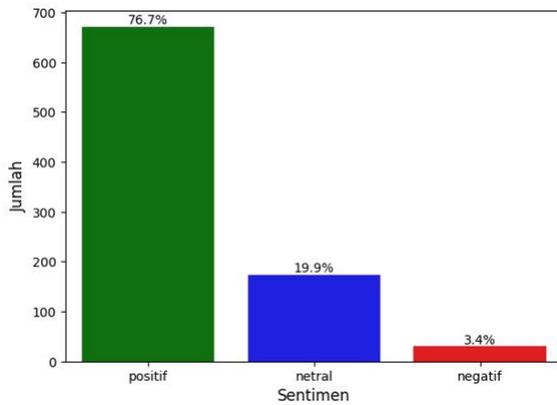
Labeling Data VADER

Labeling sentimen menggunakan VADER dengan *dataset* yang telah di translate dapat dilakukan menggunakan pustaka NLTK yaitu *SentimentIntensityAnalyzer*. Pembagian label berdasarkan VADER menghasilkan 671 data tergolong ke dalam kelas positif, 174 data kelas netral, dan 30 data kelas negatif, lihat Tabel 9.

Tabel 9: Hasil *Labeling*

Dokumen	Sentimen	Compound
['kuliah', 'bagus', 'cocok', 'buang', 'tenaga', 'uang', 'kompeten', 'dosen', 'ajar', 'layan', 'mahasiswa', 'payah']	negatif	-6.872
['universitas', 'bina', 'darma', 'palembang', 'pts', 'sumatra', 'selatan', 'rupa', 'universitas', 'sumatra', 'selatan', 'kualitas', 'unggul', 'mutu', 'layan', 'didik', 'ajar', 'bagus']	positif	8.622

Sebanyak 875 ulasan pada dataset telah mendapatkan label sentimennya berdasarkan pada diagram Gambar 4.10 terdapat 76.7% kelas positif, 19.9% kelas netral, dan 3.4% kelas negatif, lihat Gambar 3.



Gambar 3: Diagram Labeling Vader

Modeling

Pembagian Data

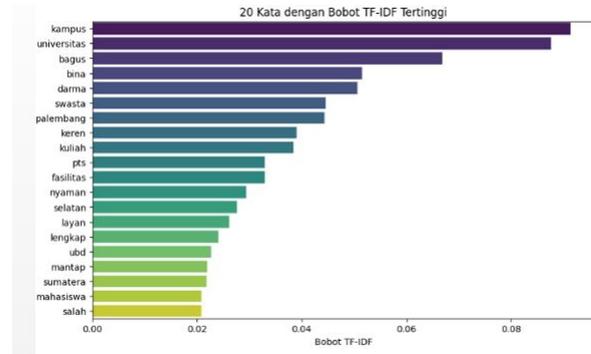
Pembagian data menggunakan perbandingan 70:30, yaitu 70% dari jumlah data menjadi data latih dan 30% sebagai data uji. Pembagian data dengan perbandingan 70:30 menghasilkan 612 data latih dan 263 data uji.

Ekstraksi Fitur

Ekstraksi fitur menggunakan Teknik TF-IDF untuk mengubah data teks menjadi angka dengan penetapan parameter *min_df* bernilai dua agar TF-IDF mengabaikan teks yang muncul kurang dari dua dokumen. Pembobotan kata menggunakan TF-IDF pada penelitian ini menggunakan Pustaka *TfidfVectorizer* dari Python. Hasil pembobotan dan bobot tertinggi lihat pada Tabel 10 dan Gambar 4.

Tabel 10: Hasil Pembobotan

Kata	Bobot TF-IDF
kampus	0.091362
universitas	0.087618
bagus	0.066911



Gambar 4: Bobot Tertinggi TF-IDF

Implementasi *Random Forest*

Sebelum klasifikasi menggunakan algoritma *Random Forest*, data latih yang telah dibobot dengan TF-IDF akan diseimbangkan dengan Teknik SMOTE untuk mengatasi ketidakseimbangan kelas. Metode *k-fold cross validation* dengan k-5 digunakan untuk mengidentifikasi akurasi optimal, di mana data dibagi menjadi k lipatan untuk pengujian dan pelatihan. Membandingkan akurasi pada nilai k yang berbeda membantu menentukan konfigurasi yang memberikan kinerja terbaik. Hasil rata-rata dari model klasifikasi *Random Forest* menggunakan *fold* k-5 yang sudah di *oversampling* dengan jumlah pohon 400 dan kedalaman 65 lihat Tabel 11.

Tabel 11: Hasil rata-rata K5

Accuracy	Recall	Precision	F1-Score
92.63%	92.69%	92.72%	92.64%

Selain mengevaluasi performa algoritma *Random Forest* terhadap dataset yang diolah, dilakukan juga perbandingan dengan hasil penelitian lain menggunakan algoritma yang berbeda, termasuk yang masuk dalam kategori *State of the Art* (SoTA). Berdasarkan [3] metode *Naïve Bayes* pada analisis sentimen aplikasi Dana mencapai akurasi sekitar 85%, sementara pada penelitian [11] membandingkan performa algoritma berbasis *deep learning* seperti BERT dan RoBERTa pada data *Twitter*, dan menunjukkan bahwa RoBERTa mampu mencapai akurasi hingga 91%. Namun hasil penelitian ini menunjukkan bahwa algoritma *Random*

Forest yang telah dioptimasi menggunakan teknik SMOTE dan TF-IDF berhasil memperoleh akurasi 92.63%, presisi 92.72%, *recall* 92.69% dan *F1-score* 92.64%. Hasil ini menunjukkan bahwa dalam konteks data ulasan *Google Maps* untuk institusi Pendidikan, *Random Forest* mampu melampaui performa metode lain dan menjadi pilihan yang sangat efektif dan efisien.

Evaluation

Evaluasi dilakukan dengan melihat perbandingan *dataset* sebelum dan sesudah *oversampling* menggunakan SMOTE dan menggunakan confusion matrix untuk melihat *fold* (lipatan) terbaik dari k-5 dalam proses klasifikasi algoritma *Random Forest* yang telah dilakukan sebelumnya.

Tabel 12: Perbandingan SMOTE

SMOTE	Accuracy	Precision	Recall	F1-Score
Sebelum	79.56%	68.61%	48.36%	76.49%
Sesudah	92.63%	92.72%	92.69%	92.64%

Tabel 12 menunjukkan perbandingan dataset sebelum dan sesudah *oversampling* SMOTE dengan fold k-5 kali. Tabel tersebut menggambarkan ketepatan hasil klasifikasi sentimen lebih besar daripada sebelum *oversampling* SMOTE.

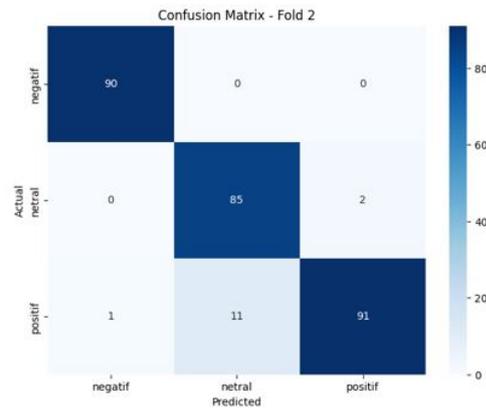
K-Fold Cross Validation

K-Fold Cross Validation dilakukan untuk memastikan distribusi kelas tetap seimbang di setiap *fold*. Dalam model ini, kumpulan data awal dipisahkan secara acak menjadi k kali lipat yang sama. Pada setiap *fold*, model dilatih menggunakan 4 *fold* sebagai data latih dan dievaluasi pada 1 *fold* yang tersisa sebagai data uji. *Fold* terbaik dari dataset yang telah di *oversampling* dipilih berdasarkan akurasi tertinggi yang dicapai pada data uji.

Tabel 13: Hasil K-Fold Cross Validation

Fold	Accuracy
K-1	0.93
K-2	0.95
K-3	0.90
K-4	0.91
K-5	0.93

Dari Tabel 13, peneliti melakukan 5 kali pengujian K-Fold Cross Validation. Hasil menunjukkan fold ke = 2 sebagai fold terbaik dengan akurasi sebesar 95%. Evaluasi pada fold tebaik menggunakan confusion matrix dengan rasio 70:30.



Gambar 5: Visualisasi *Confusion Matrix*

Gambar 5 menunjukkan visualisasi model *fold* ke dua berhasil memprediksi ulasan positif sebanyak 91 ulasan, 85 ulasan netral, dan 90 ulasan negatif. *Heatmap* menunjukkan hasil metrik 91 TP (*True Positive*), 12 FP (*False Positive*), 2 FN (*False Negative*), dan 175 TN (*True Negative*). Berikut hasil perhitungan *confusion matrix*:

1.

$$\begin{aligned}
 \text{Akurasi} &= \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \\
 &= \frac{91 + 175}{91 + 175 + 12 + 2} \times 100\% \\
 &= \frac{266}{280} \times 100\% = 0.95
 \end{aligned}$$

2.

$$\begin{aligned}
 \text{Presisi} &= \frac{TP}{TP + FP} \times 100\% \\
 &= \frac{91}{91 + 12} \times 100
 \end{aligned}$$

3.

$$\begin{aligned}
 \text{Recall} &= \frac{TP}{TP + FN} \times 100 \\
 &= \frac{91}{91 + 2} \times 100
 \end{aligned}$$

4.

$$\begin{aligned}
 \text{F1 - Score} &= 2x \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \\
 &= 2x \frac{0.8834 \times 0.9784}{0.8834 + 0.9784} \\
 &= \frac{0.8643}{1.8618} = 0.9284
 \end{aligned}$$

- [4] F. Fathonah dan A. Herliana, "Penerapan Text Mining Analisis Sentimen Mengenai Vaksin Covid-19 Menggunakan Metode Naïve Bayes," *Jurnal Sains dan Informatika*, vol. 7, no. 2, hlm. 155–164, <https://doi.org/10.34128/jsi.v7i2.331>, 2021.
- [5] S. Sriani dan S. Suhardi, "Analisis Sentimen Pengguna Aplikasi Mobile JKN Menggunakan Algoritma Naïve Byes Classifier dan C4.5," *Journal of Science Social Research*, vol. 7, no. 2, hlm. 555–563, <https://doi.org/10.24176/biner.v2i1.13399>, 2024.
- [6] D. P. Sinambela, H. Naparin, M. Zulfadhilah, dan N. Hidayah, "Implementasi Algoritma Decision Tree dan Random Forest dalam Prediksi Perdarahan Pascasalin," *Jurnal Informasi dan Teknologi*, hlm. 58–64, <https://doi.org/10.60083/jidt.v5i3.393>, 2023.
- [7] J. Supriyanto, D. Alita, dan A. R. Isnain, "Penerapan Algoritma K-Nearest Neighbor (K-NN) Untuk Analisis Sentimen Publik Terhadap Pembelajaran Daring," *Jurnal Informatika dan Rekayasa Perangkat Lunak*, vol. 4, no. 1, hlm. 74–80, <https://doi.org/10.33365/jatika.v4i1.2468>, 2023.
- [8] D. P. Jennifer, N. C. Ningrum, S. Pinasty, S. M. N. Edma, W. N. Andini, dan E. Widodo, "Analisis Sentimen Pada Pengguna Tiktok Menggunakan Metode Random Forest (Studi Kasus: Jessica-Mirna)," *Innovative: Journal Of Social Science Research*, vol. 4, no. 3, hlm. 14477–14489, 2024.
- [9] S. Khomsah, "Sentiment analysis on youtube comments using word2vec and random forest," *Telematika*, 18: 61, 2021.
- [10] M. Djufri, "Penerapan Teknik WEB Scraping untuk Penggalan Potensi Pajak (Studi Kasus pada Online Market Place Tokopedia, Shopee dan Bukalapak)," *Jurnal BPPK: Badan Pendidikan Dan Pelatihan Keuangan*, vol. 13, no. 2, hlm. 65–75, <https://doi.org/10.48108/jurnalbppk.v13i2.636>, 2020.
- [11] D. Nurmadewi, Z. F. Jailani, and N. K. S. Manik, "Comparison of the Performance of the VADER and RoBERTa Algorithms on Twitter," *Sistemasi: Jurnal Sistem Informasi*, vol. 13, no. 4, hlm. 1547–1557, <https://doi.org/10.32520/stmsi.v13i4.4198>, 2024.
- [12] L. Nursinggah, T. Mufizar, dan U. Perjuangan, "Analisis Sentimen Pengguna Aplikasi X Terhadap Program Makan Siang Gratis Dengan Metode Naïve Bayes Classifier," *J. Inform. dan Tek. Elektro Ter.*, vol. 12, no. 3, <https://doi.org/10.23960/jitet.v12i3.4336>, 2024.
- [13] S. Navisa, L. Hakim, dan A. Nabilah, "Komparasi Algoritma Klasifikasi Genre Musik pada Spotify Menggunakan CRISP-DM," *Jurnal Sistem Cerdas*, vol. 4, no. 2, hlm. 114–125, <https://doi.org/10.37396/jsc.v4i2.162>, 2021.
- [14] M. Yasir dan R. Suraji, "Perbandingan Metode Klasifikasi Naïve Bayes, Decision Tree, Random Forest Terhadap Analisis Sentimen Kenaikan Biaya Haji 2023 Pada Media Sosial Youtube," *Jurnal Cahaya Mandalika*, ISSN 2721-4796 (online), vol. 3, no. 2, hlm. 180–192, 2023.
- [15] Y. Asri, W. N. Suliyanti, D. Kuswardani, dan M. Fajri, "Pelabelan Otomatis Lexicon Vader dan Klasifikasi Naive Bayes dalam menganalisis sentimen data ulasan PLN Mobile," *Jurnal Petir*, 15 (2), 264–275, <https://doi.org/10.33322/petir.v15i2.1733>, 2022.
- [16] N. Sharfina dan N. G. Ramadhan, "Analisis SMOTE Pada Klasifikasi Hepatitis C Berbasis Random Forest dan Naïve Bayes," *JOINTECS (Journal of Information Technology and Computer Science)*, vol. 8, no. 1, hlm. 33–40, <https://doi.org/10.31328/jointecs.v8i1.4456>, 2023.
- [17] L. Annisa dan A. D. Kalifa, "Analisis Teknik TF-IDF Dalam Identifikasi Faktor-Faktor Penyebab Depresi Pada Individu," *Gudang Jurnal Multidisiplin Ilmu*, vol. 2, no. 1, hlm. 302–307, 2024.
- [18] L. Mardiana, D. Kusnandar, dan N. Satyahadewi, "Analisis Diskriminan Dengan K Fold Cross Validation Untuk Klasifikasi Kualitas Air Di Kota Pontianak," *Bimaster: Buletin Ilmiah Matematika, Statistika dan Terapannya*, vol. 11, no. 1, 2022.
- [19] N. A. Salsabila, Y. A. Winatmoko, A. A. Septiandri, and A. Jamal, "Colloquial indonesian lexicon," 2018 International Conference on Asian Language Processing (IALP), IEEE, hlm. 226–229, <https://doi.org/10.1109/IALP.2018.8629151>, 2018.