ANALISIS SENTIMEN OPINI MASYARAKAT TERHADAP PENGESAHAN RKUHP PADA TWITTER MENGGUNAKAN ALGORITMA *SUPPORT VECTOR MACHINE* (SVM) DAN *DECISION TREE* DENGAN *SYNTHETIC MINORITY OVERSAMPLING TECHNIQUE* (SMOTE)

Skripsi

Oleh

MUHAMMAD LANDY HAKIM

NIM : 11190910000093

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS SAINS DAN TEKNOLOGI

UNIVERSITAS ISLAM NEGERI SYARIF HIDAYATULLAH

JAKARTA

2023 M / 1444 H

UIN Syarif Hidayatullah Jakarta

ANALISIS SENTIMEN OPINI MASYARAKAT TERHADAP PENGESAHAN RKUHP PADA TWITTER MENGGUNAKAN ALGORITMA SUPPORT VECTOR MACHINE (SVM) DAN DECISION TREE DENGAN SYNTHETIC MINORITY OVERSAMPLING TECHNIQUE (SMOTE)

Skripsi

Sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer (S.Kom)

Oleh

MUHAMMAD LANDY HAKIM

NIM : 11190910000093

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS SAINS DAN TEKNOLOGI

UNIVERSITAS ISLAM NEGERI SYARIF HIDAYATULLAH JAKARTA

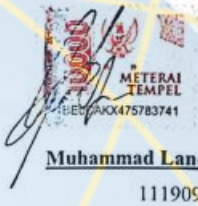
2023 M / 1444 H

PERNYATAAN ORISINALITAS

Dengan ini saya menyatakan bahwa:

1. Skripsi ini merupakan benar hasil karya asli saya yang diajukan untuk memenuhi persyaratan dalam memperoleh gelar Strata 1 di UIN Syarif Hidayatullah Jakarta.
2. Semua sumber yang saya gunakan dalam penulisan ini telah saya cantumkan sesuai dengan ketentuan yang berlaku di UIN Syarif Hidayatullah Jakarta.
3. Apabila dikemudian hari terbukti karya ini bukan hasil saya sendiri atau merupakan hasil jiplakan karya orang lain, maka dengan itu saya bersedia menerima sanksi yang berlaku di UIN Syarif Hidayatullah Jakarta.

Jakarta, 13 Juni 2023



Muhammad Landy Hakim

11190910000093

PERNYATAAN PERSETUJUAN PUBLIKASI SKRIPSI

Sebagai civitas akademika UIN Syarif Hidayatullah Jakarta, saya bertanda tangan dibawah ini:

Nama	: Muhammad Landy Hakim
NIM	: 11190910000093
Program Studi	: Teknik Informatika
Fakultas	: Sains dan Teknologi
Jenis Karya	: Skripsi

Demi pembuatan ilmu pengetahuan, saya menyetuji untuk memberikan kepada UIN Syarif Hidayatullah Jakarta Hak Bebas Royalti Non Ekslusif *(Non Exclusive Royalti Free Right)* atau karya ilmiah yang berjudul :

ANALISIS SENTIMEN OPINI MASYARAKAT TERHADAP PENGESAHAN RKUHP PADA TWITTER MENGGUNAKAN ALGORITMA *SUPPORT VECTOR MACHINE* (SVM) DAN *DECISION TREE* DENGAN *SYNTHETIC MINORITY OVERSAMPLING TECHNIQUE* (SMOTE)

Beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Non Ekslusif ini UIN Syarif Hidayatullah Jakarta berhak menyimpan, mengalihmediakan/formatkan, mengelola dalam bentuk pangkalan data, merawat dan mempublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik hak cipta. Demikian pernyataan ini saya buat sebagaimana mestinya.

Jakarta, 21 Juni 2023

(Muhammad Landy Hakim)

Nama : M Landy Hakim

Program Studi : Teknik Informatika

Judul : ANALISIS SENTIMEN OPINI MASYARAKAT TERHADAP PENGESAHAN RKUHP PADA TWITTER MENGGUNAKAN ALGORITMA *SUPPORT VECTOR MACHINE* (SVM) DAN *DECISION TREE* DENGAN *SYNTHETIC MINORITY OVERSAMPLING TECHNIQUE* (SMOTE)

ABSTRAK

Kebijakan pengesahan hukum di Indonesia sedang marak-maraknya diperbincangkan dikalangan publik terutama pada media sosial twitter. Twitter merupakan salah satu platform aplikasi yang digunakan masyarakat indonesia memberikan tanggapan melalui tweet yang dibuat. Tweet tersebut dapat memuat opini serta kritikan terhadap berita yang sedang ramai dibicarakan seperti masalah dibidang politik, sosial, ekonoomi dan lain-lain. Dalam tahun 2019 salah satu yang terjadi ialah perancangan pengesahan RKUHP, sehingga mahasiswa maupun rakyat melakukan aksi demo untuk menolak perancangan tersebut, pada akhirnya di tahun 2022, RKUHP tersebut disahkan oleh DPR RI. Tujuan dibuat penelitian ini yaitu untuk memberikan kemudahan bagi pengguna khususnya kepada masyarakat mendapatkan informasi respon dari masyarakat di Twitter terkait pengesahan RKUHP tersebut. Hasil penelitian ini bisa menjadi sebuah tolak ukur tingkat kepuasan dan penelitian masyarakat dengan data yang valid tanpa ada manipulasi data. Penelitian ini, melakukan analisis sentimen dengan menggunakan algoritma Support Vector Machine dan Decision Tree digabungkan dengan teknik SMOTE. Dataset berjumlah 2499 tweet yang didapatkan mengenai RKUHP dalam periode tertentu. Hasil dari penelitian ini menggunakan *confusion matrix* didapatkan hasil akurasi sebesar 81,2%, precision 85,33% recall 88,7%, f1-score 86,98% dengan AUC 0,759. Tingkat akurasi terbaik yaitu algoritma Support Vector Machine dengan nilai tanpa SMOTE dengan nilai 81,2 % pada skenario 90% data testing dan 10% data uji, sedangkan Decision Tree dengan SMOTE mendapatkan hasil terbaik tingkat akurasinya yaitu 77,6% pada skenario 75% data training dan 25% data testing.

Kata Kunci : Analisis Sentimen, SVM, Decision Tree, SMOTE

Jumlah Pustaka : 63 Jurnal, 3 Buku, 4 Website

Name : M Landy Hakim

Study Program : Informatics Engineering

Title : ANALISIS SENTIMEN OPINI MASYARAKAT TERHADAP PENGESAHAN RKUHP PADA TWITTER MENGGUNAKAN ALGORITMA *SUPPORT VECTOR MACHINE* (SVM) DAN *DECISION TREE* DENGAN *SYNTHETIC MINORITY OVERSAMPLING TECHNIQUE* (SMOTE)

ABSTRACT

The policy of legalization in Indonesia is being widely discussed among the public, especially on social media Twitter. Twitter is one of the application platforms used by the Indonesian people to provide responses through tweets made. These tweets can contain opinions and criticism of news that is being discussed, such as problems in the political, social, economic and other fields. In 2019, one of the things that happened was the design of the ratification of the RKUHP, so that students and people held demonstrations to reject the design, in the end in 2022, the RKUHP was passed by the DPR RI. The purpose of this research is to make it easier for users, especially the public, to get information on the response of the public on Twitter regarding the ratification of the RKUHP. The results of this study can be a benchmark for the level of satisfaction and research of the community with valid data without any data manipulation. This research conducts sentiment analysis using the Support Vector Machine and Decision Tree algorithms combined with the SMOTE technique. The dataset amounted to 2499 tweets obtained regarding the RKUHP in a certain period. The results of this study using confusion matrix obtained an accuracy of 81.2%, precision 85.33% recall 88.7%, f1-score 86.98% with AUC 0.759. The best accuracy rate is the Support Vector Machine algorithm with a value without SMOTE with a value of 81.2% in the scenario of 90% testing data and 10% test data, while Decision Tree with SMOTE gets the best accuracy rate of 77.6% in the scenario of 75% training data and 25% testing data.

Keywords ***: Sentiment Analysis, SVM, Decision Tree, SMOTE***

Bibilography ***: 63 Journals, 3 Books, 4 Websites***

KATA PENGANTAR

Bismillahirahmanirrahim

Segala puji serta syukur penulis panjatkan kehadirat Allah SWT tuhan semesta alam yang telah memberikan nikmat dan karunia-Nya sehingga peneliti bisa menyelesaikan skripsi ini. Shalawat serta salam semoga selalu tercurah kepada junjungan alam Nabi Muhammad SAW kepada keluarganya, sahabatnya, dan pengikutnya.

Skripsi yang berjudul "ANALISIS SENTIMEN OPINI MASYARAKAT TERHADAP PENGESAHAN RKUHP PADA TWITTER MENGGUNAKAN ALGORITMA SUPPORT VECTOR MACHINE (SVM) DAN DECISION TREE DENGAN SYNTHETIC MINORITY OVERSAMPLING TECHNIQUE (SMOTE)" yang telah disusun untuk memenuhi salah satu persyaratan sarjana komputer Program Studi Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Syarif Hidayatullah Jakarta.

Dalam proses pengerjaan skripsi ini tentu tidak terlepas dari bantuan, dukungan, motivasi, doa dari beberapa pihak. Maka dari itu peneliti ingin menyampaikan ucapan terima kasih kepada :

1. Allah SWT. Yang senantiasa mencurahkan segala nikmat, rahmat serta karunia-Nya kepada peneliti.
2. Bapak Husni Teja Sukmana, S.T,. M.Sc,. Ph.D selaku Dekan Fakultas Sains dan Teknologi.
3. Ibu Dewi Khairani, M.Sc selaku ketua Program Studi Teknik Informatika.
4. Ibu Dewi Khairani, M.Sc dan Ibu Nurul Faizah Rozy, M.T. sebagai dosen pembimbing yang senantiasa sabar dan ikhlas dalam memberikan bimbingan, arahan serta motivasi kepada peneliti.
5. Kedua orang tua peneliti yaitu Bapak Yadi Supriadi dan Ibu Siti Marfuah yang selalu mendukung penuh dan mendoakan agar penulisan skripsi ini dapat diselesaikan dengan sebaik-baiknya.
6. Ibu Arini, S.T,.M.T selaku dosen pembimbing akademik penulis yang telah mensupport dalam penyelesaian skripsi ini.
7. Kakak kandung peneliti yaitu Zulfa Nadia yang telah memfasilitasi dalam pengerjaan skripsi ini.

8. Seluruh teman-teman teknik informatika angkatan 2019 yang tidak disebutkan satu persatu, yang secara langsung dan tidak langsung terlibat dalam proses penyusunan skripsi ini.
9. Seluruh keluarga besar Himpunan Mahasiswa Teknik Informatika UIN Syarif Hidayatullah Jakarta, Senat Mahasiswa Universitas, yang telah memberikan kesempatan bagi peneliti untuk dapat merasakan terlibat dalam organisasi dikampus.
10. Seluruh keluarga besar Himpunan Mahasiswa Islam Komisariat Fakultas Sains dan Teknologi yang telah memberikan banyak kesempatan saya untuk berproses didalamnya.
11. Dan pihak lain yang tidak dapat penulis sebutkan satu-persatau namun tidak mengurasi rasa hormat serta terima kasih peneliti dalam penyelesaian skripsi ini.

Semoga pihak-pihak yang telah membantu peneliti dalam proses penyusunan skripsi mendapatkan keberkahan dari ALLAH SWT dan dengan skripsi ini dapat menjadi ilmu yang bermanfaat kedepannya. Peneliti menyadari masih banyak kekurangan pada penulisan skripsi ini, untuk itu penulis menerima kritik dan saran dari pembaca sekalian. Dan semoga kekurangan yang ditemukan dapat diperbaharui dan diperbaiki oleh peneliti lain dimasa yang akan datang.

Jakarta, 07 Mei 2023

M Landy Hakim

DAFTAR ISI

DAFTAR TABEL

DAFTAR GAMBAR

BAB I

PENDAHULUAN

1.1 Latar Belakang

Seiring perkembangan teknologi yang kian meningkat serta penggunanya yang sampai ini berkenbang dengan pesat dari tahun ke tahun, terutama dalam ranah media sosial. Dalam hal ini memungkinkan banyak pengguna yang memposting aktivitas sehari-harinya, salah satunya status berupa teks yang berisi suasana hati serta perilakunya (Putri & Kharisudin, 2022). Berhubungan dengan hal tersebut, analisis sentimen merupakan salah satu cara untuk mengetahui pendapat orang banyak terhadap perilaku sesuatu seperti layanan publik, isu, kinerja pemerintahan atau hal lainnya (Wati & Ernawati, 2021). Analisis sentimen ini dilakukan untuk melihat kecenderungan opini masyarakat terhadap isu isu yang berkembang terutama dimedia sosial. Analisis sentimen ini merupakan suatu pengklasifikasian dengan mengekstraksi pendapat, emosi, dan evaluasi seseorang yang tertulis dalam sebuah pembicaraan mengenai topik tertentu dengan memanfaatkan *Natural Language Processing*.(Aziz, 2022). Menurut Zulfa & Winarko, Analisis sentimen adalah sebuah riset komputasional dari opini, sentimen dan emosi yang diekspresikan secara tekstual (R. Sari, 2020). Oleh karena itu, analisis sentimen merupakan salah satu cabang dalam penelitian *text mining* yang melakukan klasifikasi dalam sebuah dokumen teks (Cahyaningtyas et al., 2021).

Indonesia merupakan negara hukum sebagaimana tertulis pada pasal 1 ayat (3) UUD 1945. Para ahli hukum mengklasifikasikan sistem hukum berdasarkan kriteria kriteria tertentu, contohnya dari aspek historis dan yuridis (Fitrah, 2021). Negara menetapkan aturan serta kebijakan untuk mengatur pola perilaku masyarakat (Informatika & Polinema, 2021). Penerapan hukum di indonesia saat ini sedang menjadi perbincangan publik dan menarik perhatian elemen masyarakat. Ditandai dengan mulainya pembahasan mengenai *Omnibus Law*, RUU Cipta Kerja hingga Pengesahan Kebijakan Rancangan Kitab Hukum Undang Undang Hukum Pidana (RKUHP). Dalam rangka menjawab persoalan dan kebutuhan hukum di indonesia, sejak juni 2015 sampai tahun 2019 Komisi III DPR RI dan Pemerintah RI telah melakukan pembahasan terkait Rancangan Undang-Undang tentang KUHP (Ruben & Sumigar, 2021). Pemerintah membuat rancangan ini setelah melalui proses yang sangat panjang dari generasi ke generasi, dan pada akhirnya tibalah saatnya RKHUP disahkan pada bulan desember tahun 2022. Dalam hal ini, banyak opini serta kritikan terhadap pemerintah dari masyarakat terkait pengesahan kebijakan kebijakan hukum yang ditetapkan oleh

pemerintah itu sendiri. (Imaduddin, 2022). hal ini lantaran membuat banyak menuai opini kontroversi terhadap elemen masyarakat baik para aktivis dari kalangan mahasiswa sampai media asing yang menentang kebijakan pengesahan KUHP tersebut (Isa, 2022).

Namun sebelum hal itu terjadi beberapa waktu, pembahasan dan pengesahan kebijakan hukum tersebut tersendat dengan adanya demo dari sekelompok Mahasiswa untuk menuntut RKUHP tersebut dibatalkan dan segera dicabut pengesahannya, karena mahasiswa berpendapat bahwasanya terdapat beberapa pasal yang menyudutkan rakyat bawah dan mematikan sistem demokrasi di Indonesia. Tepat setelah aksi demo tersebut pada tahun 2020, Presiden RI Jokowi Dodo akhirnya memutuskan untuk memberhentikan sementara pembahasan kitab RKUHP tersebut. Terlepas dari ulasan yang disampaikan oleh Komnas HAM dan Gerakan Masyarakat sipil tersebut, bahwasanya menemukan sejumlah problematika terbaru terhadap redaksi yang disediakan dalam pasal 598 dan pasal 599 RUU KUHP 2019, menurutnya jika resmi disahkan, rumusan tersebut akan bertentangan dengan standar yang berlaku dalam sejumlah instrumen hukum internasional dan hukum pidana internasional (Ruben & Sumigar, 2021).

Selain itu, menurut Menteri Hukum dan HAM, Yasona. H Laly mengatakan pengesahan ini merupakan suatu momen bersejarah dalam penyelenggaraan hukum di indonesia, dimana indonesia sebelumnya masih memakai hukum yang berasal dari produk Belanda serta sudah tidak relevan lagi dengan dengan kondisi dan kebutuhan hukum saat ini, dan masyarakat harus bangga dengan hukum yang telah dirancang oleh produk buatan Indonesia tersebut (V. Y. Susanto, 2022). KUHP yang digunakan hingga kini hanya terjemahan tidak resmi yang dikeluarkan pakar hukum pidana, sehingga anggota legislatif membuat kebijakan berupa RKUHP ini diakhir masa jabatannya pada tahun 2019 lalu (Solia et al., 2022).

Namun alih-alih dengan secara terbuka DPR RI akhirnya memutuskan mengesahkan rancangan RKUHP tersebut. Dalam hal ini masyarakat menilai pemerintah mengesahkan Kebijakan RKUHP merupakan sebuah hal yang sangat kontroversial terhadap keberlangsungan hidup masyarakat. Dari ratusan pasal yang tercantum dalam RKUHP, terdapat pasal yang memiliki keunikan jika ditinjau dari aspek perbandingan hukum (Fitrah, 2021). Masyarakat kerap memberikan sebuah opini dan pendapat nya menggunakan media sosial. Saat ini media sosial sebagai suatu kebutuhan masyarakat untuk mencari informasi, khususnya permasalahan permasalahan yang terjadi di Indonesia. salah satu media sosial yang sering digunakan oleh masyarakat seperti Twitter.

Twitter merupakan sarana masyarakat untuk memperoleh informasi seputar kehidupannya, seperti bisnis, hiburan, ekonomi, politik dan lain lainnya (Wati & Ernawati, 2021). Menurut laporan *We Are Social,* Pengguna aktif masyarakat yang menggunakan media sosial twitter mencapai 18,45 Juta pada tahun 2022. Angka tersebut membuat indonesia menempati posisi kelima negara yang pengguna twitter terbesar didunia (Rizaty, 2022). Dalam hal ini, dilandasi dengan banyaknya masyarakat yang menggunakan media sosial seperti twitter, membuat masyarakat selalu antusias untuk mengeluarkan pendapat serta berekspresi sepuasnya terhadap apa yang dirasanya tidak sesuai dengan apa yang diharapkan.

Penelitian ini tak lepas dari penelitian sebelumnya yang serupa sebagai bahan referensi pengoptimalan hasil yang terdapat beberapa penelitan, penelitian pertama dengan judul Penerapan Support Vector Machine dengan SMOTE untuk Klasifikasi Sentimen Pemberitaan *Omnibus Law* pada Situs CNNIndonesia.com penelitian ini memberikan hasil yang sedikit lebih baik dibandingkan dengan klasifikasi data tanpa SMOTE (Hutami et al., 2022). Selanjutnya dengan judul Analisis Sentimen pada rating aplikasi Shopee menggunakan Decision Tree berbasis SMOTE. Hasil penelitian ini menunjukan dengan SMOTE mendapatkan nilai yang terbaik. nilai *accuracy* (99,91%), AUC(0,999), *recall (99,98%)* dan nilai *precision* (99,98%). dari hasil kesimpulan SMOTE dapat berpengaruh terhadap nilai *accuracy* dan AUC, dan untuk nilai *recall* dan *precision* tidak berpengaruh (Cahyaningtyas et al., 2021). Penelitian selanjutnya dengan judul *Performance analysis of sentiments in Twitter dataset using SVM Model.* Dari hasil penelitian tersebut, performa dari pengklasifikasian SVM mencapai hasil yang memuaskan. Ada banyak ukuran kinerja model SVM yang tersedia. Model SVM Cross Validated dengan 10 fold dan looping through 3 kali dengan sampel rata rata 69, 67, dan 68 mendapatkan hasil yang baik dibandingkan dari pada model lainnya (Ramasamy et al., 2021). Selanjtunya dengan judul *Partner Sentiment Analysis for Telkom University on Twitter Social Media Using Decision Tree (CART) Algorithm,* dari hasil penelitian tersebut didapaktan hasil terbaik yang diperoleh model Decision Tree (CART) dengan *accuracy* 86,73%, *precision* 87,06%, *recall* 87,55% dan *f1-score* 86,52%. (Ryanto et al., 2022).

Menurut Wu & Kumar, *Support Vector Machine* merupakan salah satu algoritma dari sepuluh algoritma *supervised Learning* terbaik dalam hal *data mining*. (Rahma Yustihan & Pandu Adikara, 2021). SVM dikenal sebagai salah satu metode algoritma *supervised learning* yang mempunyai tingkat akurasi yang baik. Menurut Suyanto, bahwasannya *Decision Tree* merupakan suatu model yang menggunakan struktur pohon dalam suatu metode klasifikasi

yang sering digunakan secara praktis dan mempunyai keunggulan mengenai konsep yang jelas sehingga mudah dipahami dan di implementasikannya. (Asshiddiqi & Lhaksmana, 2020). Disamping itu, ada beberapa data yang nantinya terdapat ketidakseimbangan dalam memperoleh dataset. Oleh karena itu, Salah satu pendekatan dalam mengatasi banyak data yang tidak seimbang adalah SMOTE (*Synthetic Minority Oversampling Technique*). SMOTE merupakan suatu pendekatan yang men-generate sampel data secara sintetik, sehingga data menjadi seimbang antara data kelas mayoritas dan minoritas. Menurut Chawla, dkk, mengembangkan metode *sampling* SMOTE untuk mengatasi kelemahan yang terdapat pada sebuah metode *oversampling.* Metode SMOTE menambahkan beberapa data kelas minoritas dengan membangkitkan sebuah data buatan atau sintesis berdasar kelas k-tetangga terdekat antar kelas minoritas (Hairani et al., 2020). Dalam artian metode SMOTE ini digunakan untuk menyeimbangkan data pada beragam metode klasifikasi dan prediksi, seperti algoritma SVM, *Naïve Bayes*, *Decision Tree*, dan lainnya.

Dari hasil uraian diatas, untuk mengetahui bagaiamana perbandingan performa algoritma dengan SMOTE tersebut, maka penelitian ini melakukan penelitian yang berjudul "Analisis Sentimen Opini Masyarakat Terhadap Pengesahan RKUHP Pada Twitter Menggunakan Algoritma *Support Vector Machine* (*S*VM) Dan Algoritma Decision Tree Dengan *Synthetic Minority Oversampling Technique* (SMOTE).

1.2 Identifikasi Masalah

Berdasarkan latar belakang penelitian ini terdapat beberapa identfikasi masalahnya. sebagai berikut:

1. Banyaknya sentimen masyarakat yang melakukan kritik pro dan kontra terhadap pengesahan RKUHP di media sosial *Twitter* (Ihda Aulia Rahmah, 2022).
2. Dibutuhkan metode utnuk menganalisis sentimen opini masyarakat terkait pengesahan RKUHP dalam ranah keilmuan machine learning pada twitter.
3. Terdapat *imbalance data* dalam melakukan klasfikasi suatu data.

1.3 Rumusan Masalah

Pada penelitian ini berdasarkan dari latar belakang sebelumnya terdapat beberapa rumusan masalah yaitu

1. Bagaimana hasil performa Algoritma SVM dan Decision Tree dalam melakukan analisis sentimen terhadap pengesahan RKUHP?
2. Bagaimana pengaruh komposisi data latih dan data uji dengan SMOTE dan tanpa SMOTE terhadap performa model dalam melakukan klasifikasi terkait pengesahan RKUHP?

1.4 Batasan Masalah

Untuk mencapai penelitian yang lebih fokus dan terarah, maka penulis membatasi masalah sebagai berikut :

1.4.1 Proses

1. Data yang digunakan adalah data hasil komentar publik yang menjadi objek analisis sentimen dalam sebuah tweet pada media sosial twitter yang mengandung keyword berikut: #Pengesahan KUHP
2. Jumlah tweet yang diambil sebanyak 2499 Tweet yang berupa teks berbahasa indonesia.Proses pengambilan data tweet dilakukan pada rentang waktu 01 Desember 2022 – 01 Januari 2023.
3. Penggunaan emoji dihapus dan bahasa asing diabaikan,
4. Klasifikasi sentimen meliputi Positif dan Negatif.
5. Evaluasi performa algoritma SVM dan Decision Tree dengan SMOTE dan tanpa SMOTE, meliputi aspek *accuracy, precision, recall, f1-Score dan Area Under Curve.*

1.4.2 Metode

1. Labeling menggunakan *Transformers.*
2. Dalam penelitian ini menggunakan teknik klasifikasi algoritma SVM dan Decision Tree dengan SMOTE.
3. Proses *stemming* menggunakan algoritma Nazief & Adriani.
4. Ekstraksi fitur menggunakan TF-IDF.
5. Evaluasi performa model menggunakan metode *confusion Matrix.*

1.4.3 Tools

1. Pengolahan data analisis sentimen ini menggunakan *Jupyter Notebook* dan *Google Collaboratory* dengan bahasa pemograman *Python.*

2. *Crawling data tweet* menggunakan *library snscrape*.
3. Spesifikasi Perangkat Laptop yang digunakan dalam penelitian adalah Infinix INBook X1 Processor Intel Core i3-1005G1, RAM 4 GB, SSD 256 GB, dan Intel UHD Graphics.

1.5 Tujuan Penelitian

Tujuan penelitian yang ingin dicapai oleh penulis dalam penelitian ini sebagai berikut:

1. Mengetahui performa klasifikasi Algoritma SVM dan Decision Tree pada sentimen masyarakat terhadap pengesahan RKUHP meliputi lima aspek diantaranya *accuracy, precision, recall, f1-score dan Area Under Curve.*
2. Mengetahui hasil performa terbaik diantara tiga skenario pengujian pada algoritma SVM dan algoritma Decision Tree dengan SMOTE dan tanpa SMOTE pada komposisi data latih dan data uji yang berbeda.

1.6 Manfaat Penelitian

Pada penelitian ini berdasarkan latar belakang sebelumnya terdapat beberapa manfaat sebagai berikut:

1.6.1 Bagi Penulis

1. Sebagai salah satu syarat kelulusan strata satu (S1) Program Studi Teknik Informatika, Fakultas Sains dan Teknologi Universitas Islam Negeri Syarif Hidayatullah Jakarta.
2. Menerapkan dan pengembangan ilmu yang telah didapat selama masa perkuliahan,
3. Menambah pengetahuan pada bidang data dan *machine learning* terkait analisis sentimen sehingga mendapatkan hasil yang bermanfaat.

1.6.2. Bagi Universitas

1. Menjadi sebuah referensi untuk penelitian sejenis kedepannya.
2. Mengukur kemampuan mahasiswa dalam penguasaan materi dan menerapkan ilmu yang didapat selama dibangku perkuliahan.

1.6.3. Manfaat bagi pembaca

1. Sebagai referensi dan tolak ukur terkait permasalahan mengenai terkait suatu Kebijakan RKUHP yang menjadi perbincangan hangat di media Twitter terhadap Hukum diindonesia.

1.7 Metode Penelitian

1.7.1. Metode penelitian

Pada penelitian ini penulis menggunakan metode pengumpulan data yang dilakukan dengan studi literatur dan observasi, yaitu dengan mempelajari berbagai buku-buku terkait, membaca jurnal, artikel sebagai referensi yang terkait dengan penelitian, serta melakukan *crawling data twitter* dengan Python 3.10.

1.7.2. Metode Klasifikasi Data

Pada penelitian ini penulis melakukan klasifikasi data dengan menggunakan *supervised learning* yaitu menggunakan kategori klasifikasi. Algoritma Klasifikasi yang digunakan *Support Vector Machine* dan *Decision Tree*. Dalam menganalisis sentimen, beberapa tahapan dilakukan dengan mengacu pada (Artikel et al., 2021) :

1. Identifikasi Masalah
2. Prapemrosesan data
3. Pemrosesan data
4. Evaluasi
5. Implementasi

1.8 Sistematika Pembahasan Penelitian

Sistematika pembahasan merupakan tahapan penyususan penelitianm sebagai berikut :

BAB 1 PENDAHULUAN

Bab ini berisi tentang latar belakang, rumusan masalahn batasan masalah, tujuan penelitian, manfaat penelitian kajian yang relevan dan sistematika penelitian.

BAB 2 LANDASAN TEORI

Bab ini merupakan penjelasan seluruh teori secara lengkap yang teraput akan penelitian.

BAB 3 METODOLOGI PENELITIAN

Bab ini memuat secara rinci metode peneltian yang digunakan beserta alasannya, seperti metode yang digunakan, jenis penelitian, pengumpulan data dan tahapan proses pengolahan penelitian.

BAB 4 IMPLEMENTASI DAN EKSPERIMEN

Bab ini berisi membahas tentang proses implementasi dari metode yang digunakan untuk menyelesaikan masalah penelitian.

BAB 5 HASIL DAN PEMBAHASAN

Bab ini berisi, (1) Hasil penelitian, klasifikasi bahasan disesuaikan dengan pendekatan, sifat penelitian, rumusan masalah dan fokus penelitian, (2) Pembahasan hasil penelitian.

BAB 6 PENUTUP

Bab ini berisi kesimpulan, saran atau rekomendasi penelitian selanjutnya, berdasarkan hasil analisis dan interpretasi data atau hasil yang telah diuraikan sebelumnya.

BAB II

LANDASAN TEORI

Bab ini menjelaskan definisi dan membahas teori-teori yang diperlukan guna untuk menunjang serta menjadi acuan dalam penelitian ini. Dalam bab ini juga membahas penelitian sebelumnya sehingga terjadi acuan untuk menerapkan teori teori yang sesuai diharapkan dapat mengarah pada tujuan yang ingin dicapai. Dalam bab ini akan membahas teori-teori secara berurutan pada bab ini.

2.1 *Sentiment Analysis*

Analisis sentimen merupakan suatu metode yang digunakan untuk mengekstrak opinion data, memahami serta mengolah tekstual data secara otomatis untuk melihat sentimen positif dan negatif didalamnya (F. V. Sari & Wibowo, 2019). Analisis juga disebut sebagai bidang studi yang yang menganalisis sebuah sikap, opini, dan emosi terhadap entitas dan atributnya yang dinyatakan dalam teks tertulis. Sejak penelitian yang ada aplikasi sentimen ini berfokus pada teks tertulis yang menggunakan *Natural Language Process* (NLP) dalam pemrosesan data. Analisis sentimen berfokus pada sebuah pendapat yang mengemukakan atau menyiratkan sentimen positif dan negatif, atau disebut juga pendapat positif dan pendapat negatif dalam bahasa sehari-hari. Analisis sentiment merupakan salah satu topik yang penelitian yang dapat mendeteksi suatu permasalahan yang sedang beredar di media sosial. Salah satunya seperti, isu sosial dan politik. (Azizah et al., 2022)

Analisis sentimen sangat penting untuk bisnis dan organisasi karena ingin mencari opini konsumen atau publik tentang produk layanan mereka. Tidak hanya itu, analisis sentimen dapat juga dipakai pemerintahan yang bertujuan untuk mengetahui opini publik tentang kebijakan. Menurut D.T Hermanto, Analisis sentimen atau yang disebut *opinion mining* digunakan untuk menganalisis atau mengklasifikasikan pengguna dari kata, kalimat ataupun dokumen (Iskandar & Nataliani, 2021). Dalam hal ini, bahwasannya analisis sentimen dapat dianggap sebagai sub area penting dari analisis semantik karena bertujuan untuk mengenali topik yang dibicarakan dan sentimen mereka terhadap topik tersebut (Liu, 2019).

2.1.1 Level Analisis Sentimen

1. Tingkat Dokumen

   Level ini merupakan mengklasifikasikan seluruh dokumen opini mengungkapkan sentimen positif dan negatif. Tingkat analisis ini mengasumsikan

bahwa setiap dokumen mengungkapkan pendapat tentang satu entitas. Jadi tidak berlaku untuk dokumen yang mengevaluasi atau menmbandingkan banyak entitas.

2. Tingkat Kalimat

Pada tingkat ini merupakan pengklasifikasian kalimat dan menentukan masing-masing dari kalimat yang menyatakan pendapat positif, negatif atau netral. Tingkat analisis ini mengasumsikan bahwa setiap dokumen mengungkapkan pendapat tentang satu entitas. Analisis ini berkaitan dengan klasifikasi subjektivitas, yang membedakan kalimat objektif ialah kalimat yang mengungkapkan informasi faktual dari suatu kalimat yang disebut subyektif.

3. Tingkat Entitas/Aspek

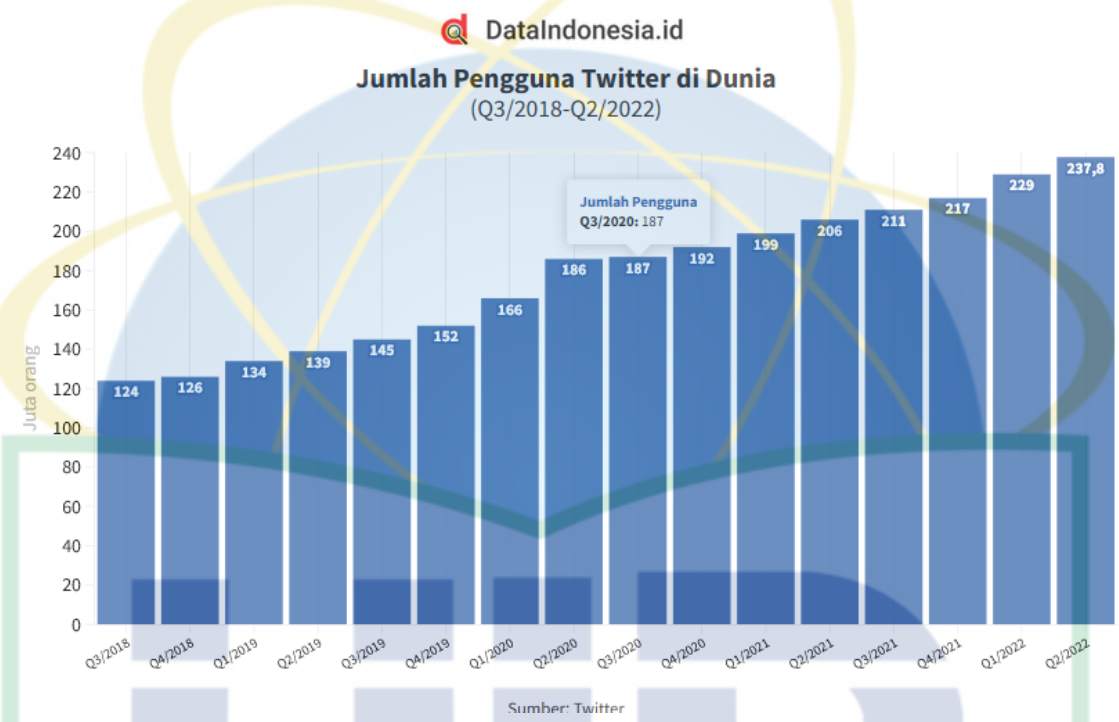
Tingkat analisis tingkat dokumen dan tingkat kalimat tidak menemukan apa yang sebenernya disukai atau tidak disukai orang. Alih-alih melihat konstruksi bahasa (dokumen, paragrad, kalimat, klausa, atau frase), tingkat aspek ini langsung melihat opini itu sendiri. Tingkat ini didasarkan pada gagasan bahwa opini terdiri dari sentimen positif dan negatif dan target. Menyadari pentingnya target opini juga membantu untuk memahami masalah analisis sentimen.

2.2 Rancangan Kitab Undang-Undang Hukum Pidana

Rancangan Kitab Undang Undang Hukum Pidana merupakan kitab yang disusun oleh DPR yang didalamnya memiliki peran sentral dan sebagai tulang punggung dalam sistem hukum pidana di Indonesia. RUU KUHP merupakan pembaruan hukum yang telah muncul sejak era kemerdekaan yang dilandasi dengan diterbitkan pasal II Aturan peralihan UUD 1945 menjadi dasar hukum untuk menghindari kekosongan hukum. RUU KUHP ini memuat asas-asas umum (fundamental) antara lain asas legalitas yang disusun secara progresif dan sesuai dengan kepribadian indonesia dan sebuah perkembangan revolusi. (F. Faisal & Rustamaji, 2021). Perkembangan KUHP sekarang diberlakukan KUHP yang bersumber dari hukum pidana yang berasal dari hukum kolonial belanda. Tujuan utama RUU KUHP diperbaharui ialah penanggulangan kejahatan. Yang bertitik tolak pada KUHP (*WvS*) yang dipandang sebagai induk sebagai wujud dari kodifikasi dan unfikasi. RUU KUHP ini memuat sebanyak 627 pasal. RUU KUHP terbaru disahkan oleh Dewan Perwakilan Rakyat (DPR) dalam rapat paripurna pada Selasa 6 Desember 2022.

2.3 *Twitter*

Twitter merupakan salah satu media sosial yang menyediakan fasilitas untuk para pengguna mengemukakan pendapat serta termasuk media sosial yang bekerja secara *real time* dan menyajikan berita terbaru. Twitter dibuat pada bulan maret 2006 dan diluncurkan pada bulan juli 2006. Data pada kuarter ke empat pada tahun 2019 membuktikan bahwa jumlah pengguna harian aktif pada *twitter* sebanyak 159 juta pengguna. (Hafidz & Yanti Liliana, 2021). Adapun *twitter* dalam laporan resminya mencatat pada tahun 2022 secara global pengguna *twitter* mencapai 237,8 juta pengguna pada kuartal II/2022. (dataindonesia.id, 2022).



Gambar 2. 1 Data Twitter

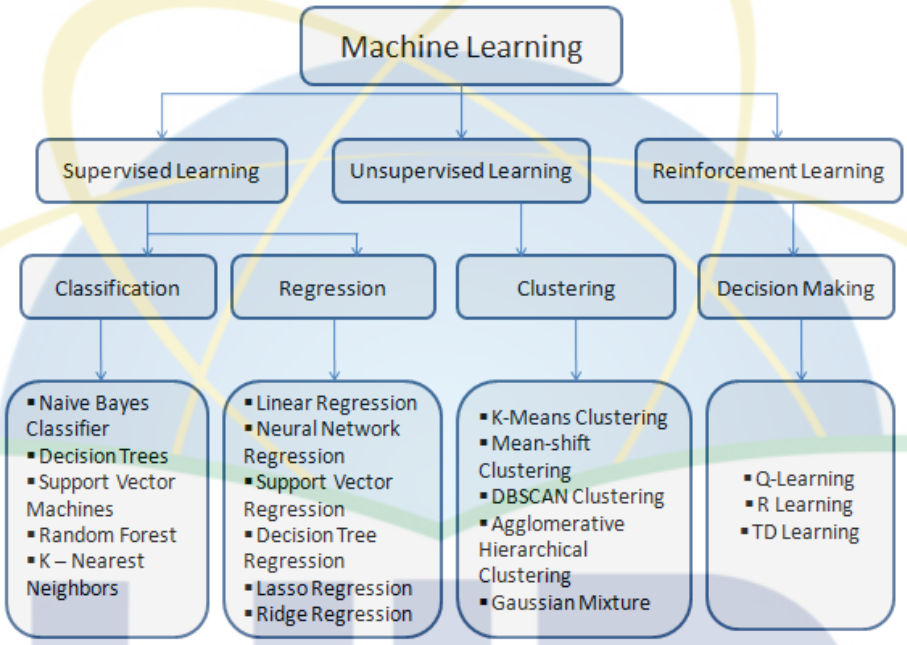
Kebiasaan masyarakat megutarakan opini nya melalui media twitter dalam menanggapi kejadian dan isu-isu yang beredar disekitar lingkungannya dapat menjadi salah satu acuan twitter menjadi media sosial yang sangat populer dikalangan masyarakat. (Informatika & Polinema, 2020). Selain itu, Twitter merupakan salah satu media sosial yang paling populer untuk mendapatkan sumber-sumber data pada analisis teks serta emosi. (Wati & Ernawati, 2021).

2.4 *Machine Learning*

Machine learning atau yang dikenal sebagai dengan pembelajaran mesin merupakan sebuah ilmu komputer yang bisa bekerja tanpa diprogram secara eksplisit. Pembelajaran mesin merupakan kecerdasan buatan yang didalamnya mempelajari bagaimana membuat sebuah data.

Machine learning merupakan teknik yang cepat dan kuat dalam menemukan sebuah masalah baru dalam penelitian (Liu, 2019).

Secara definisi, *Machine Learning* merupakan suatu ilmu atau studi yang mempelajari algoritma dan model statistik yang digunakan oleh sebuah komputer dalam melakukan tugas tertentu tanpa sebuah instruksi eksplisit. Istilah lebih umumnya yaitu bagaimana sebuah komputer yang dapat belajar dari lingkungan sekitar sehingga memilki "*knowledge"* yang berkembang.



Gambar 2. 2 Teknik Machine Learning

Ada beberapa teknik yang dimiliki oleh *machine learning* , namun yang paling populer yaitu *Supervised Learning* dan Unsupervised Learning dan Reinforcement Learning.

1. *Supervised Learning*

   Sebagaian besar praktik machine learning mengandalkan algortima *supervised learning.* Teknik *Supervised Learning* ini merupakan teknik yang menggunakan data terlabel, contohnya input dan outpun nya sudah diketahui. Dalam pemrosesan data memerlukan bantuan data yang dikumpulkan dari periode sebelumnya untuk melatih dan menentukan sebuah model dari algoritma yang dipilih.

   Permasalahan-permasalahan yang terkait dengan supervised learning dapat dikategorikan menjadi dua jenis yaitu : *Clasification* dan *Regression.* Algortima yang termasuk dalam supervised learning meliputi *Naïve Bayes, Decision Tree, Support Vector Machine, Random Forest, K-Nearest Neighbors.*(Bhatia, 2019)

2. *Unspervised Learning*

Pada teknik ini dalam pemrosesan tidak memerlukan data yang sebelumnya sebagai input. Dalam teknik ini memungkinkan pemodelan nya untuk belajar sendiri dengan menggunakan data yang telah diberikan. Dengan artian bahwa teknik ini berbeda dengan teknik *supervised learning* yang dimana teknik ini tidak memerlukan sebuah label. Oleh karena itu fokus pada teknik ini ialah masalah *clustering* dan asosiasi (Zenvita et al., 2021). Algoritma yang termasuk kedalam teknik *unsupervised learning* yaitu *Linier Regression, Neural Network Regression, Support vector Regression, Decision Tree Regression, Lasso Regression dan Ridge Regression.*(Bhatia, 2019)

3. *Reinforcement Learning*

Teknik ini merupakan pembelajarn model algoritma untuk membuat suatu keputusan. Teknik ini banyak digunakan dengan variasi dari teknik learning yang lain. Oleh karena itu, penggunaan reinforcement learning tidak bisa memberikan sebuah hasil yang akurat. Algoritma yang termasuk dalam teknik ini ialah *Q-Learning, R-Learning dan TD-Learning.*(Bhatia, 2019)

2.5 *Natural Language Process*

Natural Language Process atau yang disebut dengan Pengolahan Bahasa Alami merupaka suatu disiplin ilmu yang menerapkan model komputasi dari bahasa sehingga memungkinkan terjadinya sebuah interaksi antara manusia dengan komputer dengan sebuah perantaraan bahasa alami yang disering dipakai oleh sebagaian manusia. *Natural Language Process* (NLP) memodelkan pengetahuan seputar bahasa, baik dari segi kata, serta bagaimana kata-kata tersebut tergabung dalam suatu kalimat dan konteks kata dalam suatu kalimat. Model komputasi dari *Natural Language Process* ini dapat berguna untuk keperluan dalam suatu penelitian misalnya seperti meneliti sifat-sifat dari suatu bentuk bahasa alami maupun untuk memudahkan komunikasi antar manusia dengan komputer (Kulkarni & Shivananda, 2019).

Natural Language Process (NLP) merupakan bidang penelitian dan suatu aplikasi yang mengekplorasi bagaimana suatu komputer dapat digunakan untuk memahami dan memroses suatu teks atau ucapan bahasa alami untuk hal-hal yang bermanfaat. (E. B. Susanto et al., 2022). Dalam *Natural Language Process* ini terdapat beberapa disiplin ilmu dalam beberapa bidang pengetahuan diantaranya (Abdulrohim & Kom, n.d.);

1. Fonetik dan Fonologi : Bidang ini berhubungan dengan suara yang menghasilkan kata-kata yang dapat dikenali. Bidang ini menjadi apabila dipakai dalam suatu aplikas yang menerapakan metode *speech based system.*
2. Morfologi : Bidang ini merupakan sebuah pengetahuan tentang kata dan bentuknya yang dapat dibedakan antara satu kata dengan yang lainnya.
3. Sintaksis : Bidang ini ialah pengetahuan yang didalamnya membahas tentang urutan sebuah kata dalam pembentukan suatu kalimat.
4. Semantik : Sebuah pengetahuan yang mempelajari arti arti suatu kata dan bagaimana memproses arti dari kata-kata tersebut menjadi suatu arti kata dari kalimat yang utuh.
5. Pragmatik : Bidang pengetahuan ini berfokus pada tentang pengetahuan yang mempunyai konteks kata/kalimat yang berkaitan dengan keadaan atau situasi pada suatu kalimat.
6. *Discourse Knowledge* : Bidang ini merupakan berhubungan dengan antar kalimat yaitu yang didalamnya membahas pengenalan apakah suatu kalimat yang telah dikenali dapat mempengaruhi kalimat selanjutnya atau tidak.
7. *World Knowledge* : mencakup seluruh pengetahuan arti kata secara umum dan mencari apakah terdapat arti khusus bagi suatu kata dalam suatu percakapan dengan konteks tertentu.

2.6 *Text Mining*

Text Mining merupakan sebuah proses penambangan yang dilakukan oleh komputer untuk mendapatkan sesuatu yang baru dan tidak diketahui, atau menemukan suatu informasi yang tersirat secara implisit. (F. V. Sari & Wibowo, 2019). *Text mining* selalu berurusan dengan kata-kata, jutaan kata-kata yang disimpan melalui berbentuk file elektronik. Untuk proses penambangan teks-teks yang ada didalam dokumen membutuhkan suatu mekanisme untuk mendapatkan informasi-informasi yang lebih bernilai dan terstruktur. Diantar tahapan atau mekanisme yang dilakukan ialah *tokenizing, filtering, stemming, tagging.* Tantangan umum yang dihadapi dalam *text mining* yaitu bahasa. Karena dalam pengimplemntasian bahasa dapat memengaruhi metode ataupun teknik yang digunakan.(Kulkarni & Shivananda, 2019).

Tujuan dari text mining tidak jauh beda dengan data mining yaitu menemukan suatu pola pada suatu data agar dapat dimanfaatkan manusia untuk memudahkan dalam melakukan

pekerjaannya. Ada beberapa implementasi pada *text mining* pada kasus nyata, diantaranya (M. R. Faisal & Kartini, 2020);

1. Klasifikasi Email untuk menentukan email spam atau tidak spam.
2. Mengetahui sentimen terhadap tokoh publik dengan pengklasifikasian pada komentar masyarakat di media sosial.
3. Sebagai alat pemantauan keadaan fasilitas publik dengan mengklasifikasikan komentar-komentar.
4. Sebagai aplikasi untuk melihat kinerja pelayanan publik dalam suatu lembaga pemerintahan.
5. Untuk mengetahui sentimen terhadap produk atau jasa di media sosial maupun *E-Commerce.*
6. Sebagai aplikasi untuk pemantauan bencana Alam yang ditulis langsung oleh korban, netizen dan pesan yang tidak berhubungan dengan bencana tersebut.

2.7 *Preprocessing*

Pre-processing merupakan sebuah proses awal *text mining* untuk mempersiapkan suatu text menjadi data numerik yang bisa diolah lebih lanjut.(Ginantra et al., 2022). Pada tahapan preprocessing dilakukan dengan teknik *Natural Language Process (NLP).* (Arsi et al., 2022). *Preprocessing* data biasanya dilakukan dengan cara mengeliminasi data yang tidak sesuai oleh kerja sistem. Melalui *preprocessing*, memungkinkan sebuah proses mining akan berjalan dengan baik, efektif dan lebih efisien karena data sudah melalui tahap pembersihan data agar dapat diolah oleh sistem (Kulkarni & Shivananda, 2019).

Preprocessing sangat penting pada data *training* untuk mendukung proses melatih algoritma agar dapat mengoreksi data-data yang tidak teroganisir menjadi terorganisir oleh sistem. *Preprocessing* dilakukan dalam enam tahap, diantaranya (Kevin et al., 2020).

1. *Cleansing :* Tahap ini merupakan tahap pembersihan atau eliminasi aksara nonalfabetis untuk menurunkan *noise.* Aksara yang dihapus seperti tanda baca, simbol-simbol hashtag, dan lainnya.
2. *Case Folding* : tahap ini berfungsi untuk mengkonversi karakter alfabet a sampai z yang sudah melalui tahap cleansing.
3. *Tokenizing* : Tahap ini merupakan proses pemecahan kalimat berdasarkan setiap kata dalam proses penyusunan suatu kalimat.

4. Normalisasi atau *konversi slangword* : proses ini merupakan proses pengubahan suatu kata dari kata yang tidak baku menjadi kata baku.
5. *Filtering/Stopword Removal* : proses ini merupakan proses memilah agar kata kata yang tidak penting atau tidak terdapat makna didalamnya dihapus untuk sebuah analisis sentimen.
6. *Stemming* : Stemming merupakan proses mentranformasi kata kata yang terdapat dalam sebuah dokumen ke kata-kata akarnya dengan menerapkan aturan-aturan tertentu. Terdapat dua aturan dalam *stemming* yaitu pendekatan kamus dan pendekatan aturan.

2.8 Algoritma stemming Nazief & Adriani

Algoritma Nazief & Adriani merupakan salah satu algoritma stemming yang dikembankan pertama kali oleh Bobby Nazief & Mirna Adriani. (Choesni Herlingga et al., 2020). Terdapat tiga komponen pada algoritma Nazief & Adriani, yaitu urutan perubahan kata (*rule*), pengelompokan imbuhan, kamus (dictionary) (Rezalina, 2020). Algortima ini didasarkan dalam morfologi bahasa indonesia yang begitu luas, yang disatukan menjadi satu kelompok dan enkapsulasi menjadi suatu imbuhan yang diizinkan dan tidak diizinkan (Choesni Herlingga et al., 2020). Dalam algoritma Nazief & Adriani mempunyai kamus kata dasar dalam pencocokan kata setelah dilakukan *stemming* kata.

Algoritma Nazief & Adriani merupakan salah satu algoritma yang sering kali digunakan karena kefektivitas nya yang begitu uggul oleh algoritma *stemming* yang lain. Algoritma Nazief & Adriani ini memliki tahap-tahap sebagai berikut : (Putu et al., 2020).

1. Langkah awal yang pertama adalah cari kata yang ingin di stemming dalam kamus kata dasar. Jika ditemukan maka diasumsikan kata dasasr. Maka algoritma nya akan berhenti.
2. Buang kata yang mengandung *inflection Suffixes* (“-lah”, “-kah”, “-ku”, “-mu”, atau “-nya”) atau jika terdapat partikel (“-lah”, “-kah”, “-tah”, “-pun”) maka proses langkah ini diulangi untuk menghilangkan *posesive pronouns* (“-ku”, “-mu”, atau “-nya”), jika ada.
3. Hapus *Derivation Suffixes* (“-i”, “-an”, atau “-kan”), jika ditemukan kata tersebut dalam kamus, maka algortima berhenti. Jika tidak, maka ke langkah 3a.

a) Jika “-an” telah dihapus lalu huruf terakhir itu dari kata tersebut adalah “-k”, maka huruf “-k”, akan dihapus. Jika kata tersebut dalam kamus ditemukan maka algortitma akan berhenti. Sebaliknya jika kata tidak ditemukan maka lanjutkan ke langkah 3b.

b) Akhiran yang dihapus (“-i”, “-an”, atau “-kan”) dikembalikan, lanjut ke langkah 4.

4. Hilangkan atau hapus Derivation prefix. Jika langkah ke 3 ada sufiks yang dihapus, maka lanjut ke langkah 4a, jika tidak maka ke langkah 4b.

   a) Periksa tabel kombinasi awalan akhiram yang tidak diijinkan, jika dapat ditemukan maka algortima berhenti, jika tidak maka selanjutnya lankgah ke 4b.

   b) Psada langkah ini melakukan sebuah *looping* sebanyak 3 kali. Jika kata dasar belum dapat ditemukan , maka lanjut ke langkah 5, jika sudah algoritma akan berhenti. Catatan jika awalan kedua tesebut sama dengan awalan pertama , maka algoritma akan berhenti.
5. Melakukan *recording.*
6. Jika semua langkah selesai, namun tidak berhasil maka kata awal diasumsikan sebagai kata *root word* (kata dasar). Proses selesai.

2.9 *Synthetic Minority Oversampling Techinique* (SMOTE)

SMOTE merupakan kepanjangan dari kata *Synthetic Minority Oversampling Technique* yang berfungsi sebagai menyeimbangkan jumlah distribusi data sampel pada kelas minoritas dengan menyeleksi sebuah data sampel tersebut menjadi data sampel yang seimbang dengan jumlah sampel mayoritas (Keputusan Dirjen Penguatan Riset dan Pengembangan Ristek Dikti et al., 2017). Penjelasan singkatnya ialah teknik yang dimanfaatkan untuk masalah ketidaseimbangan suatu data (*imbalanced data).* SMOTE bekerja dengan memodifikasi dataset yang tidak seimbang dengan cara membuat data sintetik yang baru pada kelas minoritas dengan tujuan untuk meningkatkan kinerja dalam metode klasifikasi tersebut. (Cahyaningtyas et al., 2021). Jika dibandingkan dengan *oversampling* yang dilakukan dengan duplikasi (*random over sampling)* kelebihan SMOTE dapat dilihat dengan mempertimbangkan efek pada wilayah keputusan dalam ruang fitur (WIJAYANTI et al., 2021).

Tahapan dalam melakukan SMOTE dimulai dari menghitung suatu jarak antar data pada data minoritas. Selanjutnya menentukan nilai presentase SMOTE kemudian menentukan

jumlah k terdekat dan terakhir untuk menciptakan suatu data buatan dengan suatu persamaan berikut. (Keputusan Dirjen Penguatan Riset dan Pengembangan Ristek Dikti et al., 2017);

$$x_{syn} = x_{i+} (x_{knn} - x_i) \times \delta \quad (2)$$

Dengan x_{syn} merupakan sebuah data sintesis yang diciptakan x_i data yang akan direplikasi, x_{knn} data yang memiliki jarak yang paling dekat dengan data yang akan direplikasi dan δ nilai random antara 0 dan 1. (Keputusan Dirjen Penguatan Riset dan Pengembangan Ristek Dikti et al., 2017).

Tabel 2. 1 Perbandingan SMOTE, ADASYN, RUS, dan ROS

SMOTE	ADASYN	*Random Under Sampling*	*Random Over Sampling*
Teknik Imbalance dengan melakukan *oversampling* pada kelas minoritas dengan membuat sampel sintesis (Ramadhanti et al., 2023)	Mengatasi masalah imbalance data dengan *oversampling* pada kelas minoritas dengan menggunakan bobot distribusi untuk data pada kelas minoritas (Ramadhanti et al., 2023).	membuat banyak instance penting terhapus sehingga dapat mempengaruhi kinerja classifier (Sir & Soepranoto, 2022).	Meningkatkan ukuran kelas minoritas dengan mensitesis sampel baru atau langsung mereplikasi secara acak dataset *training* (Sir & Soepranoto, 2022).
Pada penelitian (Maula Chamzah et al., 2022) Dapat meningkatkan akurasi pengklasifikasian data yang memiliki distribusi label yang tidak merata terhadap kelas minoritas.	mengurangi bias yang diakibatkan oleh ketidakseimbangan kelas (Pamuji et al., 2021).	Pendekatan ini memiliki kelemahan karena membuat banyak instance penting terhapus sehingga dapat mempengaruhi kinerja classifier (Sir & Soepranoto, 2022).	Cenderung memiliki akurasi yang rendah bila terdapat jumlah data yang signifikan dari suatu data (Arsi et al., 2022)

Pada penelitian (Ramadhanti et al., 2023) SMOTE Mempunyai tingkat optimalisasi akurasi yang baik dibandingkan dengan ADASYN.	Memiliki parameter yang digunakan untuk menentukan tingkat keseimbangan yang diharapkan (β) (Ramadhanti et al., 2023).	RUS dapat lebih efektif dan cepat dalam proses pelatihan prediksi imbalance class sebuah cacat *software* (Saputro & Rosiyadi, 2022).	*Instance* yang dihasilkan hanya meningkatkan besarnya jumlah kelas minoritas dengan hanya mereplikasi informasi yang sama (Saputro & Rosiyadi, 2022).

Metode penelitian sekarang	**Jenis**	***Sampling Srategy***
Synthetic Minority Oversampling Techinique (SMOTE)	SMOTE	*Minority*

2.10 *Support Vector Machine* (SVM)

Menurut Wu & Kumar, *Support Vector Machine* merupakan salah satu algoritma dari sepuluh algoritma *supervised Learning* terbaik dalam hal *data mining*. (Rahma Yustihan & Pandu Adikara, 2021). *Support Vector Machine* (SVM) dikembangkan oleh Boser, Guyon, Vapnik, dan pertama kali dipresentasikan pada tahun 1992 di Annual *Workshop on Computational Learning Theory* dan berkembang pesat sampai saat ini. Banyak penelitian yang menerapkan konsep SVM dikarenakan Tingkat Akurasi pada model yang dihasilkan oleh proses peralihan SVM sangat berhubungan dengan fungsi kernel dan parameter yang digunakan. (Rahman Isnain et al., 2021). Tujuan dari pembelajaran SVM ialah menemukan nilai *margin* maksimum untuk memisahkan dua kelas data. (Rahma Yustihan & Pandu Adikara, 2021).

Didalam dunia nyata, pada umumnya permasalahan tersebut bersifat *non-linier,* sehingga garis yang disebut *hyperplane* tidak dapat membagi kedua kelas secara penuh dan untuk mengatasi permasalahan tersebut dengan menggunakan fungsi *kernel.* SVM sangat cepat dan begitu efektif untuk proses pengolahan klasifikasi teks, dalam isitilah geometris disebut sebuah klasifikasi biner, dimana terdapat *hyperplane* yang menentukan ruang fitur antar titik-titik yang mewakili situasi poisitf maupun negatif (Rahman Isnain et al., 2021). Algoritma SVM memiliki persamaan sebagai berikut (Liang, 2021) :

$f(x) = w.x + b$

$f(x) = \sum_{t=1}^{m} a_i\, y_i K(x, x_{i)} + \ b$

Keterangan :

w : Parameter *hyperplane* yang dicari (gerak yang tegak lurus antara garis hyperplane & titik *support vector machine*)

x : titik data masukan

a_i : nilai bobot pada setiap titik data

$K(x, x_{i)}$: Fungsi Kernel

b : Parameter *hyperplane* (nilai bias)

Tabel 2. 2 Perbandingan SVM, KNN dan *Naive Bayes*

Support Vector Machine	KNN	*Naive Bayes*
Kemampuannya dalam komputasi data berdimensi tinggi (Arsi et al., 2021).	Tangguh terhadap *training data* yang noisy dan efektif apabila data latihnnya besar dan performa cukup baik (Keputusan Dirjen Penguatan Riset dan Pengembangan Ristek Dikti et al., 2017).	memiliki waktu klasifikasi yang singkat (Gunawan et al., 2018).
Tingkat akurasi terbaik dan tidak dipengaruhi besar kecilnya data uji (Ramadhon, 2020).	waktu yang digunakan untuk komputasi sangatlah lama jika data latihnya besar dan sangat sensitive dengan ciri yang redundan atau relevan (Keputusan Dirjen Penguatan Riset dan Pengembangan Ristek Dikti et al., 2017).	Metode ini untuk banyak dataset dengan performa yang cepat dalam mengklasifikasi data dan memiliki akurasi tinggi (Utami & Dwi Hartanto, n.d.).
Mampu menjadi pengklasifikasi text yang baik	Bergantung pada label kategori yang melekat pada	membutuhkan jumlah data pelatihan (*training data*)

untuk pengujian sentimen (Wati & Ernawati, 2021).	dokumen pelatihan mirip dengan dokumen tes (Adhi Putra, 2021).	yang kecil untuk menentukan estimasi parameter yang diperlukan dalam proses pengklasifikasian (Imron, 2019).
Cenderung relatif baik ketika terdapat margin pemisahan data yang jelas (Trivusi, 2022a).	Membutuhkan nilai k dan metrik jarak, yang relatif lebih sedikit jika dibandingkan dengan algoritma machine learning lainnya (Trivusi, 2022b).	Hanya mencari probabilitas pada setiap kata pada data latih, kemudian mencari kata pada data uji yang sesuai (Ramadhon, 2020).
Metode penelitian sekarang	**Jenis Kernel**	***Cost***
Support Vector Machine	*Linear*	0.25

2.11 *Decision Tree*

Decision Tree adalah salah satu algoritma machine learning yang digunakan untuk membuat suatu keputusan seperti struktur pohon yang memodelkan kemungkinan hasil, biaya serta utilitas dan kemungkinan suatu konsekuensi. Decision Tree merupakan suatu model klasifikasi seperti pohon, dimana dalam setiap cabang pohon mempresentasikan suatu pilihan, dan daun pohon mempresentasikan sebagai hasil keputusan. (Hidayatullah & Warih Maharani, 2022) . *Decision Tree* terbuat dari 3 simpul jenis yaitu *leaf, root,* dan yang terakhir yaitu simpul perantara yang berhubungan dengan pengujian. (Puspita & Widodo, 2021). Kelebihan dari *decision tree* ini ialah dapat dilihat dari area pengambilan keputusan yang dapat mengubah suatu hasil keputusan menjadi lebih sederhana dan lebih spesifik dari suatu keputusan yang kompleks. (Hidayatullah & Warih Maharani, 2022).

Dalam membuat keputusan *Decision Tree* melakukan dengan cara partisi data rekursif. Dalam tahapannya, penentuan aturan terbaik dalam *splitting,* dan data dari *node* akan di partisi menjadi *child nodes* dengan kriteria tertentu. (Akmal Iftikar, 2022).

Tabel 2. 3 Perbandingan *Decision Tree, Random Forest dan Logistic Regression*

Decision Tree	*Random Forest*	*Logistic Regression*

konsep yang jelas sehingga dapat mudah untuk dipahami dan mengimplementasikannya mudah dengan menggunakan algoritma rekursif (Asshiddiqi & Lhaksmana, 2020)	Dapat mengatasi noise dan missing value (Maziida, 2018).	berguna dalam masalah klasifikasi biner (Hendriyana et al., 2022).
bekerja dengan cara melakukan pencarian secara rakus (greedy) sehingga dapat dimungkinkan mendapat hasil tidak optimal tetapi memberikan solusi yang mendekati nilai optimum dalam waktu yang cukup cepat (Asshiddiqi & Lhaksmana, 2020).	Metode ensemble paling kuat dengan kinerja tinggi dalam hal data dimensi tinggi (Andreyestha & Azizah, 2022)	Rentan terhadap underfitting dataset yang kelasnya tidak seimbang (Hendriyana et al., 2022).
Sangat kuat, populer, berbasis logika, dan mudah dipahami (Permana et al., 2021).	Membutuhkan tuning model yang tepat untuk data (Maziida, 2018).	efisien dan ampuh untuk menganalisis efek dari sekelompok variabel independen dengan hasil biner dengan mengukur kontribusi unik setiap variabel independen untuk memprediksi output dari variabel dependen kategoris (Fazrin et al., 2022).

Metode penelitian sekarang	Jenis	Parameter
Decision Tree	CART	*Max Depth*

2.12 Algoritma TF-IDF

Term Frequency (TF) merupakan salah satu proses pembobotan yang digunakan untuk menentukan jumlah kata yang memiliki tingkat munculnya pada suatu dokumen. *Inverse Document Frequency* (TF-IDF) adalah proses dalam pemberian bobot yang digunakan untuk mengukur intensitas kemunculan dalam suatu kata. (E. B. Susanto et al., 2022). TF-IDF merupakan suatu proses pembobotan untuk dalam menentukan jumlah kata dan mengukur intensitas kemunculan dalam suatu kata didalam dokumen.

TF-IDF biasa digunakan untu mengubah data dokumen menjadi data numerik. Dalam perhitungan TF-IDF, pertama yang harus dihitung adalah *Term Frequency* per kata dengan bobot masing masing adalah 1 (Kevin et al., 2020).

Rumus pembobotan suatu kata menggunakan Algoritma TF-IDF mempunyai rumus. Rumus TF pada persamaan ke 1, IDF terdapat pada persamaan ke 2, sedangkan TF-IDF pada persamaan 3:

$$W_{(t,d)} = TF_{(t,d)} \quad (1)$$

$$IDF(t) = (\log(\frac{N}{df_{(t)}}) \quad (2)$$

$$TF - IDF = TF(t,d) \times IDF_{(t)} \quad (3)$$

Keterangan :

TF(t,d) : frekuensi pada term t dan dokumen d

N : Jumlah total dokumen

Df(t) : Jumlah dokumen term t

2.13 *Confusion Matrix*

Confusion matrix merupakan suatu model tabel yang berisikan dari banyaknya baris data uji yang dipresiksi benar atau tidak benarnya suatu data dalam sebuah model klasifikasi. (Asshiddiqi & Lhaksmana, n.d, 2020). Dengan kata lain *confusion matrix* digunakan untuk mengetahui informasi hasil klasifikasi aktual yang diprediksi oleh sebuah sistem. Menurut (Rizkia et al., 2019), *Confusion matrix* merupakan sebuah metode untuk mengukur performansi pada *supervised learning.* Pada tabel dibawah merupakan tabel *confusion matrix* yang diperlukan untuk menentukan kinerja paa suatu model klasifikasi (Asshiddiqi & Lhaksmana, n.d, 2020):

Tabel 2. 4 *Confusion Matrix*

Confusion Matrix	*Prediction True*	*Prediction False*
Actual Positive	*True Positive* (TP)	*False Negative* (FN)
Actual Negative	*False Positive* (FP)	*True Negative* (TN)

Penjelasan tabel *confusion matrix* sebagai berikut: (Asshiddiqi & Lhaksmana, n.d.)

1. *True Positive* (TP) : Jumlah prediksi yang bernilai benar, dengan nilai kelas aktual positif dan prediksi sebagai positif..

2. *False Negative* (FN) : Jumlah prediksi yang bernilai salah, dengan nilai kelas aktual positif dan prediksi sebagai negatif..

3. *False Positive* (TP) : Jumlah prediksi yang bernilai benar, dengan nilai kelas aktual negatif dan prediksi sebagai positif..

4. *True Negative* (TN) : Jumlah prediksi yang bernilai salah, dengan nilai kelas aktual negatif dan prediksi sebagai negatif.

Dalam tahapan *confusion matrix* terdapat beberapa komponen yang dapat menentukan performace terhadap evaluasi dalam suatu model, diantaranya:

a. *Accuracy*

Accuracy adalah suatu parameter perbandingan yang memprediksi benar dalam suatu data penuh.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

b. *Precision*

Precision adalah suatu parameter perbandingan dalam memprediksi benar positif dengan hasil yang diprediksi positif secara keseluruhan.

$$Precision = \frac{(TP)}{(TP + FP)}$$

c. *Recall*

Recall adalah parameter perbandingan dalam memprediksi benar positif dengan data benar positif secara keseluruhan data.

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

d. *F1-Score*

FI-Score merupakan rasio perbandingan dalam menentukan rata-rata dalam *precision* dan *recall* yang telah dibobotkan.

$$FI - score = \frac{(2 \times recall \times precision)}{recall + precision}$$

e. AUC (*Area Under The Curve*)

AUC digunakan untuk memberi suatu kemudahan dalam membandingkan dalam suatu model dengan model lainnya, dengan artian AUC ialah perhitungan luas area yang berada dibawah curve ROC (Receiving Operating Charateristic) atau integral ROC. (Bhatia, 2019). Selain itu, AUC (*Area Under Curve*) adalah kriteria evaluasi yang menggunakan sensitivitas dan spesifitas sebagai dasar pengukuran. Berikut ini tabel keterangan untuk masing-masing interval nilai AUC (Putri & Kharisudin, 2022).

Tabel 2. 5 Kategori nilai AUC

AUC Score	Model Quality
0.9 – 1.0	*Excellent*
0.8 – 0.9	*Good*
0.7 – 0.8	*Fair*
0.6 – 0.7	*Poor*
0.5 – 0.6	*Failure*

2.14 Python

Python adalah suatu bahasa pemograman yang dinamis yang biasa digunakan dalam data science karena memiliki banyak standar library dan mudah untuk di pelajari. Pemograman

python sendiri dikembangkan oleh Guido van Rossum tahun 1990 di Amsterdam dan dikembangkan lanjut oleh *Python Software Foundation.* Python secara umum berbentuk pemrograman berorientasi pada objek, pemrrograman imperatif, dan pemrograman fungsional. Hal yang membedakan pemograman *python* dengan bahasa yang lain adalah dalah penulis kode program yang terbilang cukup mudah dalam implementasinya dan merupakan salah satu produk open source pada multiplatform. Bahkan *python* dapat mendukung untuk suatu sistem operasi (Morgan Peters, 2020).

Python dapat digunakan untuk berbagai macam tugas, seperti untuk aplikasi dekstop, database, jaringan pemograman, pembuatan aplikasi game bahkan pengembangan seluler. Tidak hanya itu, *Python* seringkali dipergunakan untuk suatu keperluan penelitian untuk menganalisis data yang berukuran besar atau yang dikenal *big data*, dan cabang data science lainnya. (Zenvita et al., 2021). Selain itu, *pyhton* memiliki beberapa fitur diantaranya :

1. Memiliki koleksi kepustakaan yang sangat luas artinya, telah tersedia modul-modul siap pakai untuk keperluan.
2. Tata bahasa yang mudah dipahami, jelas dan sederhana.
3. Berorientasi prosedural dan objek sekaligus.
4. memiliki aturan tampilan kode sumber yang sangat mudah untuk melakukan pengecekan, pembacaan kembali, dan penulisan.
5. sistem yang sangat mudah dalam pengelolaan memori otomatis.

Karena memiliki banyak fitur yang memudahkan para programmer dalam menyelesaikan tugasnya, *python* sendiri menjadi salah satu bahasa yang populer saat ini dan yang paling banyak dipakai untuk analisis suatu data yang jumlahnya besar pada saat ini.

2.15 Studi Literatur

Tabel 2. 6 Studi Literatur

No.	Judul	Penulis	Tahun	Kelebihan	Kekurangan	Penelitian Penulis Sekarang
1	*Partner Sentiment Analysis for Telkom University on Twitter Social Media Using Decision Tree (CART) Algorithm*	Sean Akbar Ryanto, Donni Richasdy, Widi Astuti	2022	- Klasifikasi Sentimen menjadi 3 kategori : Positif, Negatif dan Netral - Menggunakan N-Gram dalam proses ekstraksi fitur	- Tidak menggunakan *Split Validation*	• Menggunakan Split Validation dalam 3 skenario
2	Perbandingan Metode K-NN, Naïve Bayes, Decision Tree untuk Analisis Sentimen Tweet Twitter Terkait Opini Terhadap PT PAL Indonesia	Franly Salmon Pattiiha, Hendry	2022	- Menggunakan 3 Algortima Supervised Learning	- pelabelan bersifat manual.	• pelabelan sentimen dengan otomatis menggunakan library google.trans

3	*Twitter sentiment analysis of bangkok tourism during covid-19 pandemic using support vector machine algorithm*	Sontayasara, Thanapat, Jariyapongpaiboon, Sirawit, Arnon Promjun, Napat Seelpipat,Kumpol Saengtabtim,JingTang, and Natt Leelawat	2021	- Pengujian Menggunakan 4 Kernel SVM - Labeling sentimen Positif, Negatif, Netral	- Data yang digunakan sedikit hanya 1,101 tweets - Pada tahap preprocessing tidak ada *case folding, normalization,*	• Membandingkan 2 Algoritma Decision Tree dan SVM • Menggunakan teknik crawling data dengan library snscrape
4	*Sentiment analysis of twitter data related to Rinca Island development using Doc2Vec and SVM and logistic regression as classifier*	Tirta Hema Jaya Hidayat, Yova Ruldeviyani, Achmad Rizki Aditama, Gusti Raditia Madya, Ade Wija Nugraha, Muhammad Wijaya Adisaputra	2021	- Menggunakan Doc2Vew. (PV-DM & PV-DBOW)	- Tidak mempunyai fitur algoritma stemming	• Menggunakan algortima Nazief & Adriani.
5	Perbandingan Metode KNN, Decision Tree, dan Naïve Bayes Terhadap Analisis	Rani Puspita, Agus Widodo	2021	- Menggunakan 3 Algortima - Klasfikasi sentimen	- Pengujian performa metrik hanya *Accuracy dan Precision.*	• Pengujian performa *confusion matrix* dengan *Accuracy, Recall, Precision, dan F1-Score.*

	Sentimen Pengguna Layanan BPJS			Positif, Negatif, Netral.		
6	*Comparison of Kernel Support Vector Machine Multi-Class in PPKM Sentiment Analysis on Twitter*	Andi Nurkholis, Debby Alita, Aris Munandar.	2022	- Menggunakan kernel multi-class SVM	- Jumlah data yang diperoleh imbalance - Tidak ada normalisasi pada text pre-processing	• Menggunakan Optimalisasi *Synthetic Minority Oversampling Technique* (SMOTE)
7	*Sentiment Analysis of Nepali COVID19 Tweets Using NB, SVM AND LSTM*	Milan Tripathi	2021	- Menggunakan 3 Algoritma SVM, NB dan LSTM.	- Menggunakan python sebagai bahasa pemograman	• Menggunakan data teks Berbahasa Indonesia
8	*Performance analysis of sentiments in Twitter dataset using SVM models*	Lakshmana Kumar Ramasamy, Seifedine Kadry, Yunyoung Nam, Maytham N. Meqdad.	2021	- Pengujian Kernel Kernel SVM	- Tidak mengukur AUC	• Menggunakan SVM Kernel Linear • Performa Model Quality AUC

9	Analisis Sentimen Pengguna Aplikasi Dana Berdasarkan Ulasan Pada Google Play Menggunakan Metode *Support Vector Machine*	Abitdavy Athallah Muhammad , Ermatita , Desta Sandya Prasvita	2022	- Pengujian menggunakan 3 Kernel SVM dengan Gridsearch CV.	- Tidak menggunakan nilai CV pada Gridsearch CV.	• Menggunakan kernel Linear dengan Cost = 0.25
10	Analisis Sentimen Twitter Bahasa Indonesia Menggunakan Pendekatan Machine Learning	Aloysius Kurniawan Santoso	2022	- Pengujian dengan 4 Metode - Menggunakan Prepruning Max Depth pada Decision Tree	- Tidak menggunakan normalisasi	• Penelitian sekarang memakai Max Depth dengan Hyperparameter Grid Search CV

Tabel 2. 7 Keunikan peneliti dengan penelitian sebelumnya

Nama Penulis	Hutami, Wijayanto & Dina, 2022	Abitdavy, Ermatita, Prasvita, 2022	Ryanto, Richasdy & Astuti, 2022	Akmal Iftikar, 2022	Rizkia, 2019	Peneliti Sekarang
Seleksi Fitur	Pembobotan TF-IDF	Pembobotan dengan TF-IDF	Pembobotan dengan TF-IDF	Pembobotan TF-IDF	Pembobotan dengan TF-IDF	Menggunakan Pembobotan TF-IDF
Labeling	Labeling manual terdiri dari 1940 data, dengan 135 data sentimen positif, 1557 data sentimen netral, dan 248 data sentimen negatif.	Labeling manual terdiri dari 1366 dengan kategori positif dan negatif.	Labeling manual terdiri dari 1855 data tweet dengan 3 kategori sentimen positif, negatif dan netral.	Labeling manual dan vader terdiri dari 3000 data.	Labeling manual terdiri dari 1934 data dengan 3 kategori yaitu positif, negatif, dan netral.	Pelabelan otomatis dengan library google.trans terdiri dari 2499 data, dengan 1653 sentimen positif, dan 843 sentimen negatif.
Splitting Data	Pengujian dengan dua skenario yaitu 80%:20% dan 90%:10%	Pengujian dengan satu skenario 80:20%	Pengujian dengan 3 skenario 80:20, 75:25, dan 70:30	Pengujian dengan skenario data latih data uji 80:20.	Pengujian dengan data latih dan data uji 70:30.	Dengan tiga skenario data latih dan data uji yaitu : 90%:10%,

						80%:20% dan 75%:25%.
Tahapan data pengujian	Dilakukan dua kali pengujian dengan menggunakan algoritma SVM tanpa SMOTE dan SVM dengan SMOTE. Didapatkan penerapan dengan SMOTE memberikan hasil sedikit lebih baik.	Dilakukan pengujian dengan dua kali tahapan yaitu SVM dengan 3 kernel dan SVM dengan seleksi fitur Chi-square. Didapatkan penerapan dengan seleksi fitur tersebut memberikan hasil akurasi tertinggi.	Dilakukan pengujian sebanyak tiga kali tahapan dengan Decision Tree CART yaitu *oversampling, preprocessing* dan TF-IDF.	Pengujian dilakukan 2 skenario dengan 4 algoritma klasifikasi yaitu *naive bayes, decision tree, support vector machine,* dan *Stacking Ensemble.* Didapatkan metode *stacking ensemble* akurasi tertinggi	Pengujian dilakukan dengan decision tree dengan 3 skenario fitur utama TF-IDF, didapatkan skenario bigram memiliki akurasi tertinggi.	Pengujian dilakukan empat kali dengan algoritma SVM dan Decision Tree dengan *hyperparameter gridsearch CV* tanpa SMOTE dan dengan SMOTE. Didapatkan tahapan tanpa SMOTE pada data latih 90%:10% mendapatkan hasil akurasi 81,2%.

Untuk penelitian ini memiliki kelebihan antara lain :

1. Labeling

   Peneliti menggunakan labeling yaitu *Transformers*. Dengan labeling menggunakan *Transformers* ini, peneliti dapat mengidentifikasi sentimen positif dan negatif terkait data twitter pada pengesahan RKUHP secara otomatis, sehingga memudahkan peneliti untuk mengetahui tingkat opini masyarakat positif dan negatif terkait Pengesahan RKUHP tersebut.
2. Penggunaan tiga Algoritma yakni : SVM dan Decision Tree, serta penggunaan SMOTE.

   Merujuk studi literatur sebagaian besar penelitian terkait analisis sentimen hanya menggunakan dua metode. Oleh karena itu sebagai nilai lebih dan pembeda peneliti menambahkan algoritma SMOTE untuk membandingkan tingkat performa model tersebut. Dengan tujuan untuk meningkatkan akurasi pada klasifikasi.
3. *Area Under Curve*

   Penelit menambahkan parameter model quality untuk menunjukkan kualitas algoritma dalam performa model SVM dan Decision Tree dengan SMOTE dan tanpa SMOTE dalam melakukan Analisis Sentimen terhadap Pengesahan RKUHP, yang belum dilakukan penelitian sebelumnya

BAB III
METODOLOGI PENELITIAN

3.1 Metode Pengumpulan data

Pada tahap ini merupakan tahap mengumpulkan data dan informasi yang dapat menunjang proses penelitian, penulis mencari dataset untuk mengklasifikasikan model penelitian yang akan digunakan , tidak hanya itu, jurnal, artikel, buku, skripsi serta website untuk dijadikan bahan pengetahuan untuk melakukan penelitian ini.

3.1.1 Studi Literatur

Penulis melakukan studi literature terkait dengan pengumpulan tinjauan teori-teori yang berkaitan dengan penelitian ini. Sumber sumber dari penelitian ini berasal dari jurnal, buku referensi, skripsi serta situs website terkait dalam *Natural Language Process* (NLP), *Data mining, Machine Learning, synthetic Minority Oversampling Techinque,*serta Algoritma *Support Vector Machine* (SVM) dan *Decision Tree.*

3.1.2 Observasi

pada tahap ini, penulis melakukan observasi data dengan mengumpulkan data dengan memanfaatkan *library* dari python yaitu *library snscrape*. Dataset yang akan digunakan dalam penelitian ini, ialah menggunakan teknik crawling pada Twitter yang diakses menggunakan *library python snscrape*, dan hasil crawling berupa teks berbahasa indonesia. Setelah itu, didapatkan data dari platform tersebut tentang tweet atau komentar masyarakat terhadap kebijakan pengesahan RKUHP yang menjadi sebuah fenomena dan perbincangan hangat di Indonesia. Untuk pengambilan data dilakukan pada tanggal 01 Desember 2022 – 01 Januari 2023. Pada periode pengumpulan data tersebut, tanggal tersebut dipilih karena periode tersebut merupakan detik-detik pengesahan RUU KUHP diresmikan oleh DPR RI dan setelah pengesahan RKUHP.

3.2 *Preprocessing*

Pada tahap ini, penulis menggunakan 6 tahapan pada pra-pemrosesan data dan sebelum melakukan tahap selanjutnya agar mudah untuk dianalisis. Terdapat 6 tahapan yaitu :

1. *Case Folding*

2. Cleansing

3. Tokenization

4. Stopword Removal

5. Normalization

6. Stemming

3.3 *Synthetic Minority Oversampling Technique* (SMOTE)

Proses SMOTE ini digunakan untuk dokumen yang mempunyai ketidakseimbangan suatu data yang telah di proses pada tahap *preprocessing*. Dengan memakai SMOTE ini diharapkan ada hasil dari model SVM dan *Decision Tree* mendatkan hasil terbaik dalam pengklasifikasian suatu data.

3.4 Ekstraksi Fitur

Setelah tahapan proses preprocessing dilakukan, selanjutnya pada tahap ini menggunakan ekstraksi fitur *Term Frequency-Inverse Document Frequency* (TF-IDF) dalam pembobotan kata yang berfungsi mengubah teks menjadi sebuah vektor.

3.5 Klasifikasi SVM dan *Decision Tree*

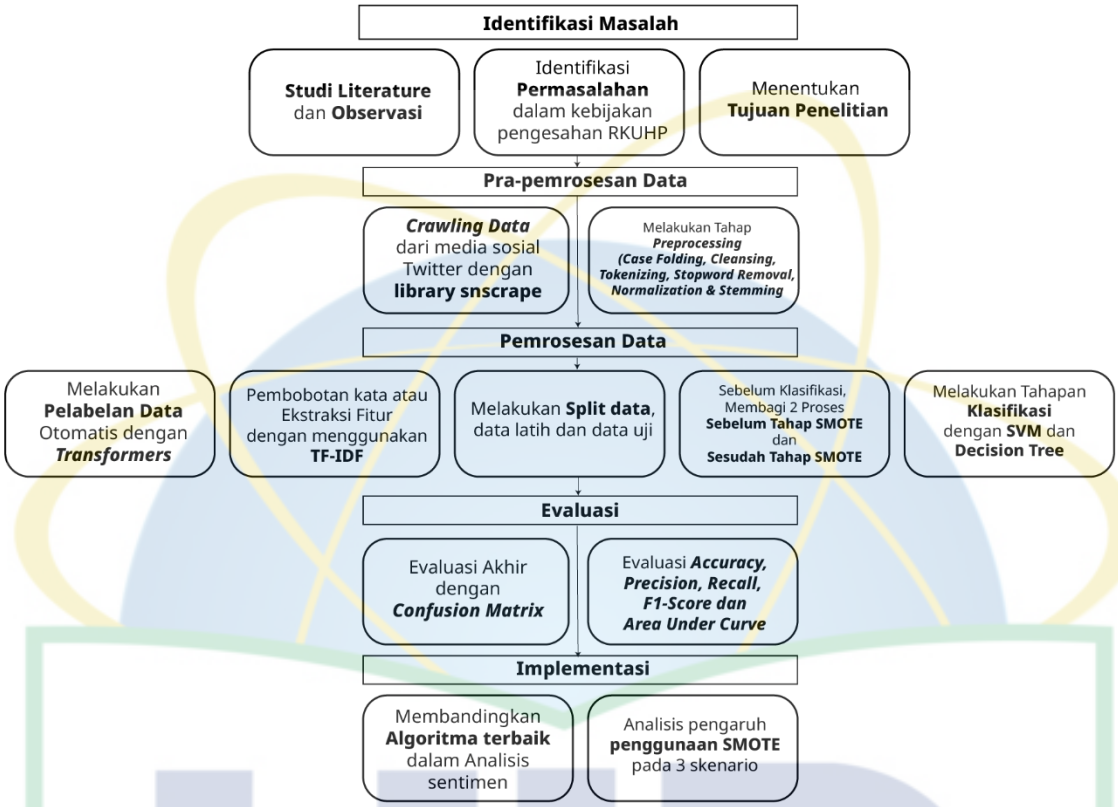
Setelah data *tweet* telah melakukan tahap ekstraksi fitur maka selanjutnya dataset dibagi menjadi data latih dan data uji menggunakan train-test-split pada *library python*. Model dilatih menggunakan lima skenario jumlah pembagian komposisi data yang berbeda.

3.6 Evaluasi Model SVM dan *Descision Tree*

Hasil dari pemodelan klasifikasi SVM dan *Decision Tree* sebelumnya, perlu adanya evaluasi model menggunakan *confusion matrix* untuk mengukur performa dalam memprediksi sautu data uji. Terdapat parameter dalam menguji evaluasi model tersebut yaitu *accuracy, precision, recall* dan *f1-score.*

3.7 Alur Penelitian

Alur tahapan ini menujukkan mekanisme rangkaian penelitian dari awal hingga akhir yang dibuat berdasarkan pemikiran penulis yang diimplementasikan kedalam bentuk *flowchart*. Berikut alur penelitian penulis sebagai berikut :



Gambar 3. 1 Alur Penelitian

BAB IV

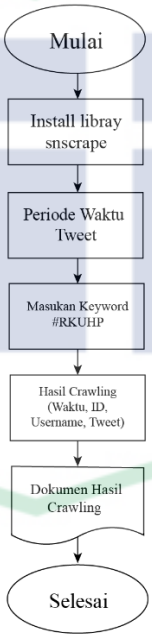
IMPLEMENTASI DAN EKSPERIMEN

4.1 Pengumpulan Data

Tahap awal dalam pengumpulan data yaitu *crawling data* pada tweet dengan menggunakan library snscrape. *Library snscrape* merupakan library yang terdapat dalam bahasa pemograman pyhton yang berfungsi unutk melakukan *crawling data* pada twitter. Dalam *crawling data* dengan library sncsrape ini memiliki kelebihan dalam pengambilan data twitter tidak terbatas waktu *crawling data*, dimana pada *Twiter* API hanya bisa 1 minggu ke belakang untuk *crawling data.* Oleh karena itu dalam penggunaan *snscrape* ini memudahkan untuk pengambilan data twitter dalam kurun waktu yang telah lama. Berikut langkah-langkah dalam *crawling data* :

1. *Install library snscrape* pada *jupyter notebook* dengan kode “pip install snscrape”.
2. Setelah *install library* tersebut, kemudian masukan tanggal mulai untuk melakukan crawling data dan sampai kapan crawling data yang dibutuhkan.
3. Selanjutnya, data tweet yang akan diambil hanya tweet yang mengandung kata kunci ‘pengesahan KUHP’ atau dengan hashtag #pengesahan KUHP, serta tidak memasukan *retweet.*
4. Menyimpan data ke dalam bentuk file csv.

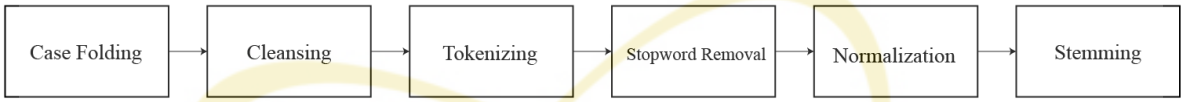
Berikut adalah gambaran dari proses *crawling data* :



Gambar 4. 1 *Flow Chart Crawling*

4.2 *Preprocessing*

Preprocessing merupakan salah satu tahap *text mining*, hal tersebut perlu dilakukan karena untuk memudahkan dalam menganalisis atau memproses data. Dalam melakukan tahapan preprocessing ini penulis menggunakan bahasa pemograman python. Adapun tahap tahap preprocessing sebagai berikut :



Gambar 4. 2 Alur Preprocessing

Berikut contoh dari setiap proses yang dilakukan untuk melakukan tahapan preprocessing :

1. *Case Folding*

   Case Folding merupakan salah satu tahap *preprocessing* yang berfungsi untuk menyamaratakan penggunaan huruf kapital pada suatu text menjadi huruf kecil. Dalam pemrosesan case folding dilakukan dengan menggunakan fungsi *pandas* yaitu *Series.str.lower().* Setelah itu akan didapatkan teks yang yang telah diubah ke *lowercase* (huruf kecil) yang sebelumnya terdapat beberapa teks yang mempunyai huruf kapital.

   Berikut adalah script untuk melakukan case folding:

```
#-------------- Case folding -------------
# fungsi str.lower () pada pandas
tweet['Tweet'] = tweet['Tweet'].str.lower()

print('Hasil Case Folding: \n')
print(tweet['Tweet'].head(30))
print('\n\n\n')
```

Dalam proses tersebut didapatkan hasil *case folding* pada setiap tweet melalui tabel dibawah ini, dalam tabel ini setiap karakter yang mempunyai huruf kapital akan berubah menjadi huruf kecil :

Tabel 4. 1 Contoh Case Folding

Sebelum Case Folding	Setelah Case Folding
Menutup 2022, bisa kah ini masuk ke pasal 265 KUHP mengganggu ketrentaman tetangga ðŸ˜’ https://t.co/sLz0D6OINR"	menutup 2022, bisa kah ini masuk ke pasal 265 kuhp mengganggu ketrentaman tetangga ðŸ˜’ https://t.co/slz0d6oinr"

2. *Cleansing*

Tahapan *cleansing* ini merupakan tahap pembersihan atau eliminasi aksara nonalfabetis untuk menurunkan *noise*. Proses cleansing ini dilakukan setelah tahap case folding, dimana data yang dipakai merupakan data dari hasil case folding. Dalam cleansing ini dilakukan penghapusan URL, *@mention, link, double whitespaces, emoticon,* duplikasi teks, serta angka. Sedangkan untuk aksara yang dihapus seperti tanda baca, simbol-simbol hashtag, dan lainnya. Dalam proses ini menggunakan fungsi library NLTK (*Natural Language Toolkit*) pada python.

Berikut adalah *script* untuk proses *cleansing*:

```
import string
import re

from nltk.tokenize import word_tokenize
from nltk.probability import FreqDist
nltk.download('punkt')

# --------------- Tahap Cleansing -------------

def remove_tweet_special(text):
  # Menghilangkan tab, new line, ans back slice
  text = text.replace('\\t', " ").replace('\\n', "
").replace('\\u', " ").replace('\\',"")
  # menghapus non ASCII : Emoticon, chineseword. etc
  text = text.encode('ascii', 'replace').decode('ascii')
  # menghapus mention, link, hastag
```

```
  text = ' '.join(re.sub("([@#][A-Za-z0-9]+)|(\w+:\/\/\S+)"," ",
text).split())
  #remove URL
  return text.replace("http://", " ").replace("https://", " ").
replace("xexxa"," ")

tweet['Tweet'] = tweet['Tweet'].apply(remove_tweet_special)

#remove angka
def remove_number(text):
    return  re.sub(r"\d+", "", text)

tweet['Tweet'] = tweet['Tweet'].apply(remove_number)

#remove punctuation
def remove_punctuation(text):
    return text.translate(str.maketrans("","",string.punctuation))

tweet['Tweet'] = tweet['Tweet'].apply(remove_punctuation)

#remove whitespace leading & trailing
def remove_whitespace_LT(text):
    return text.strip()

tweet['Tweet'] = tweet['Tweet'].apply(remove_whitespace_LT)

#remove multiple whitespace into single whitespace
def remove_whitespace_multiple(text):
    return re.sub('\s+',' ',text)

tweet['Tweet'] = tweet['Tweet'].apply(remove_whitespace_multiple)

# remove single char
def remove_single_char(text):
    return re.sub(r"\b[a-zA-Z]\b", "", text)

tweet['Tweet'] = tweet['Tweet'].apply(remove_single_char)
```

```
#reset Data duplicate
tweet.drop_duplicates(subset="Tweet", keep='first', inplace = True)
```

Dalam proses *cleansing* didapatkan hasil sebagai berikut :

Tabel 4. 2 Contoh *Cleansing*

Sebelum Cleansing	Setelah Cleansing
menutup 2022, bisa kah ini masuk ke pasal 265 kuhp mengganggu ketrentaman tetangga ðŸ˜’ https://t.co/slz0d6oinr"	menutup bisa kah ini masuk ke pasal kuhp mengganggu ketrentaman tetangga

3. *Tokenizing*

Pada tahap ini merupakan proses pemecahan kalimat berdasarkan setiap kata dalam proses penyusunan suatu kalimat. Dimana dalam proses ini menggunakan library NLTK. Dan tahapan ini dilakukan setelah hasil proses dari cleansing.

Berikut adalah script dalam proses tokenizing :

```
# --- NLTK word tokenize ------
def word_tokenize_wrapper(text):
    return word_tokenize(text)

tweet['tweet_tokens'] = tweet['Tweet'].apply(word_tokenize_wrapper)

print('Hasil Tokenizing : \n')
print(tweet['tweet_tokens'].head(30))
print('\n\n\n')

tweet = pd.DataFrame(tweet)
tweet.to_csv("Tokenizing-KUHP.csv", index = False)
```

Dalam proses tersebut penggunaan kata dalam setiap kalimat akam dipecah dalam satu kesatuan kata. Berikut contoh hasil dari *tokenizing* :

Tabel 4. 3 Contoh *Tokenizing*

Sebelum Tokenizing	Sesudah Tokenizing
menutup bisa kah ini masuk ke pasal kuhp mengganggu ketrentaman tetangga	['menutup', 'bisa', 'kah', 'ini', 'masuk', 'ke', 'pasal', 'kuhp', 'mengganggu', 'ketrentaman', 'tetangga']

4. *Stopword Removal*

Pada tahapain ini merupakan proses penghapusan kata-kata yang tidak baku dan dianggap tidak penting dalam dokumen, seperti yang, kalo, bikin, masih, dalam, dan, ke, atau, dan seterusnya. Tahapan ini dilakukan setelah tahapan tokenizing.

Berikut adalah script dalam proses stopword removal

```
#import stopword
from nltk.corpus import stopwords

# ----------------------  stopword from NLTK stopword ------------
-------------------
# stopword indonesia
list_stopwords = stopwords.words('indonesian')


# ---------------------------- tambah stopword manual  ------------
------------------------
# append stopword tambahan
list_stopwords.extend(['yg', 'dg', 'dgn', "brt", "ppp", "ny",
'klo', 'kalo', 'amp', 'biar', 'bikin',
    'bilang','krn', 'nya', 'nih', 'sih', 'wkwk',
'xfxfxxfxfxfxfxbb',
    'si', 'tuh', 'utk', 'ya','jd', 'sdh', 'aja', 'n',
'xexxa','nyg', 'hehe', 'pen',
    'u', 'nan', 'loh','&amp', 'yah', 'yang', 'gini', 'yha',
'sjw','sm', 'scr',
    'also', 'ku', 'mu', 'dri','gue', 'apa','ape','gua',
'btw','lg','gw','deh','eh','kl',
```

```
'ttg','kzl','lah','krn','loh','kmrn','spet','masi','ni','kek','dr',
'dlm','ndan'])

# ---------------------- tambah stopword dari txt file ----------
-------------------------
# read stopword dari txt file dengan pandas
txt_stopword = pd.read_csv("stopwords.txt", names= ["stopwords"],
header = None)

# convert stopword string ke list & append stopword tambahan
list_stopwords.extend(txt_stopword["stopwords"][0].split(' '))

# ---------------------------------------------------------------
----------------------

# convert list to dictionary
list_stopwords = set(list_stopwords)


#remove stopword pada list token
def stopwords_removal(words):
    return [word for word in words if word not in list_stopwords]

tweet['tweet_tokens_SR'] =
tweet['tweet_tokens'].apply(stopwords_removal)
```

Dalam proses tahapan ini kata-kata yang mengandung unsur makna penting didalamnya tetap dipertahankan dan kata kata penghubung akan dihilangkan guna mempermudah dalam pemrosesan suatu data. Berikut hasil dari *stopword Removal* :

Tabel 4. 4 Contoh Stopword Removal

Sebelum Stopword Removal	Setelah Stopword Removal
['menutup', 'bisa', 'kah', 'ini', 'masuk', 'ke', 'pasal', 'kuhp', 'mengganggu', 'ketrentaman', 'tetangga']	['menutup', 'pasal', 'kuhp', 'mengganggu', 'ketrentaman', 'tetangga']

5. *Normalization*

Pada tahapan normalisasi ini kata kata yang sebelumnya mempunyai makna yang ambigu atau tidak sesuai dengan ejaan akan di ubah sesuai dengan teks normalisasi yang sudah dibuat berdasarkan hasil dari pengamatan dalam suatu dokumen teks yang akan diproses.

Berikut adalah *script* dari *Normalization*

```
# -- Normalisasi ----
normalizad_word = pd.read_excel("Normalisasi_teks.xlsx")

normalizad_word_dict = {}

for index, row in normalizad_word.iterrows():
    if row[0] not in normalizad_word_dict:
        normalizad_word_dict[row[0]] = row[1]

def normalized_term(document):
    return [normalizad_word_dict[term] if term in
normalizad_word_dict else term for term in document]

tweet['tweet_normalized'] =
tweet['tweet_tokens_SR'].apply(normalized_term)
```

Dalam proses tersebut mesin akan membaca dokumen normalisasi_teks yang didalamnya terdapat kata yang ambigu yang tidak sesuai makna pada suatu kata akan disesuaikan.

Berikut hasil dari *Normalization* :

Tabel 4. 5 Contoh *Normalization*

Sebelum Normalization	Sesudah Normalization

[‘menutup’, ‘pasal’, ‘kuhp’, ‘mengganggu’, ‘ketrentaman’, ‘tetangga’]	“[‘menutup’, ‘pasal’, ‘kuhp’, ‘mengganggu’, ‘ketenteraman’, ‘tetangga’]

6. *Stemming*

Pada tahapan ini merupakan suatu proses suatu kata yang mempunyai imbuhan akan di pecah menjadi kata dasar. Dalam proses *stemming* ini menggunakan Algoritma Nazief & Adriani. Dalam proses *stemming* ini menggunakan *library sastrawi* dan proses ini dilakukan setelah normalisasi telah selesai.

Berikut adalah *scripts* dari proses *stemming* :

```
#-------- Stemming -------
# import Sastrawi package
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory
import swifter

# create stemmer
factory = StemmerFactory()
stemmer = factory.create_stemmer()

# hasil stemmed
def stemmed_wrapper(term):
    return stemmer.stem(term)

term_dict = {}

for document in tweet['tweet_normalized']:
    for term in document:
        if term not in term_dict:
            term_dict[term] = ' '

print(len(term_dict))
print("------------------------")
```

```
for term in term_dict:
    term_dict[term] = stemmed_wrapper(term)
    print(term,":" ,term_dict[term])

print(term_dict)
print("------------------------")


# apply stemmed term to dataframe
def get_stemmed_term(document):
    return [term_dict[term] for term in document]

tweet['tweet_tokens_stemmed'] =
tweet['tweet_normalized'].swifter.apply(get_stemmed_term)
print(tweet['tweet_tokens_stemmed'])
```

Kata kata yang mempunyai kata imbuhan akan disesuaikan menjadi kata dasar dan kata dasar yang tidak mempunyai imbuhan ke-an, me-an tidak akan diproses.

Berikut adalah hasil dari proses *stemming* :

Tabel 4. 6 Contoh *Stemming*

Sebelum Stemming	Sesudah Stemming
['menutup', 'pasal', 'kuhp', 'mengganggu', 'ketenteraman', 'tetangga']	['tutup', 'pasal', 'kuhp', 'ganggu', 'tenteram', 'tetangga']

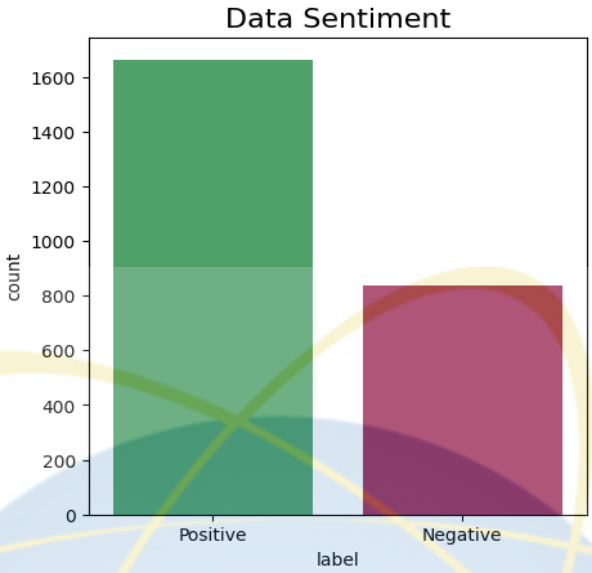
4.3 *Wordcloud*

Gambar 4. 3 Wordcloud

Dalam visualisasi data dengan memakai *wordcloud*, didapatkan kata dengan frekuensi tingkat kata terbanyak yaitu KUHP, Hukum, Dukung, Bangsa, Indonesia.

4.4 Pelabelan Otomatis dengan *Transformers*

Tahapan ini merupakan pemberian label pada dataset. Dalam pemberian label data tersebut menggunakan library *google.transs*. *Transormers* merupakan suatu *library* pada python yang berfungsi sebagai pemberian label positif dan negatif dalam suatu data secara otomatis. Pada tahapan pelabelan ini dilakukan setelah semua proses tahapan *preprocessing* selesai. selanjutnya dalam pelabelan transformers ini hanya bisa melakukan klasifikasi positif dan negatif dalam mendeteksi klasifikas sentimen tersebut. Dalam proses pelabelan ini transformers akan lebih efektif bila teks dalam tweet di terjemahkan terlebih dahulu ke dalam bahasa inggris, setelah itu *transformers* akan memroses kalimat dalam bentuk klasifikasi positif dan negatif. Pelabelan ini berisi 2499 data dan diperoleh hasil positif sebanyak 1653 dan negatif sebanyak 846.



Gambar 4. 4 Data Sentimen Labeling *Transformers*

Berikut adalah contoh data yang sudah dilabeli dengan *Transformers* dan akan dianalisis :

Tabel 4. 7 Hasil Labeling *Transformers*

No	Tweet	Sentimen
1	revisi kuhp dilakukan tidak mudah untuk membuat semua pihak sepakat dalam merumuskan rkuhp.	NEGATIVE
2	menurut kuhp yg baru hanya org tua perempuan yg bisa melaporkan krn hubungan tsb dilakukan atas suka sama suka tdk ada jaminan akan nikahi.	NEGATIVE
3	pengesahan kuhp merupakan wujud manifestasi reformasi hukum.	POSITIVE
4	jangan bunyikan petasan dan kembang api di malam tahun baru karena bisa terkena pasal uu	NEGATIVE

	kuhp tentang kebisingan yang baru disahkan dpr kemaren.	
5	kegentingannya apa yh kok smpe nekat bikin perppu skg waktunya anak htn rebutan isu hukum nih setelah anak pidana dgn kuhp barunya.	NEGATIVE
6	kuhp baru bawa indonesia ke paradigma modern hukum pidana dukung kuhp.	POSITIVE
7	tokoh masyarakat ajak rakyat dukung uu kuhp	POSITIVE
8	tuduhan uu kuhp bahayakan demokrasi sama sekali tidak tepat.	NEGATIVE
9	kuhp buat kemajuan indonesia.	POSITIVE
10	kuhp baru dibuat oleh orang orang hebat.	POSITIVE

4.5 Ekstraksi Fitur

Dalam sebuah tahapan ini merupakan suatu langkah untuk memberikan sebuah *term* atau kata pada proses hasil tweet. Dalam proses ini menggunakan library *features_extraction.text* dalam sklearn. Pemberian bobot kata ini menggunakan metode *Term Frequency-Inverse Document Frequency* (TF-IDF).

1. Tweet yang sudah dilakukan tahap preprocessing serta sudah diberi label dengan transformers setelah itu diberi nilai polaritas dengan Positif bernilai 1 dan Negatif bernilai 0. Sehingga didapatkan hasil dibawah ini:

Tabel 4. 8 Hasil Polaritas

No	*Tweet*	Label	Polaritas
1.	['kuhp', 'maju', 'indonesia']	Positif	1
2.	['kuhp', 'penting', 'rakyat']	Positif	1

3.	['bunyi', 'petas', 'kembang', 'api', 'malam', 'kena', 'pasal', 'uu', 'kuhp', 'bising', 'sah', 'dpr', 'kemarin']	Negatif	0
4.	['pasal', 'lalai', 'timbul', 'korban', 'jiwa', 'pasal', 'kuhp', 'tinggal', 'selidik', 'lalai']	Negatif	0
5.	['dukung', 'kuhp', 'bawa', 'indonesia', 'paradigma', 'modern', 'hukum', 'pidana']	Positif	1

2. Dalam pemrosesan fitur ini menggunakan library python yaitu *scikit-learn* . library ini sudah banyak digunakan untuk metode dalam pemodeln data serta machine learning. Dalam library scikit-learn peneliti menggunakan *TfidfVectorize*r. Berikut dibawah ini merupakan *syntax* dalam proses ekstraksi fitur.

```
from sklearn.feature_extraction.text import TfidfVectorizer
max_features = 2500

print ("-TF-IDF on Tweet Data-")

tfidf = TfidfVectorizer(max_features = max_features, binary=True)
tfidf_mat = tfidf.fit_transform(Tweet).toarray()

print("TF-IDF ", type(tfidf_mat), tfidf_mat.shape)

terms = tfidf.get_feature_names_out()

sums = tfidf_mat.sum(axis=0)

# menghubungkan term ke sums frequency
data = []
for col, term in enumerate(terms):
    data.append((term, sums[col] ))

ranking = pd.DataFrame(data, columns=['term','rank'])
ranking.sort_values('rank', ascending=False)
```

3. Hasil *Term Frequency-Inverse Document Frequency* (TF-IDF)

Dalam proses sebelumnya dengan menggunakan fungsi *TfidfVectorizer* peneliti men-set *max_features* = 2500 untuk memperoleh 2500 term dengan term terbesar.

Berikut hasil dari perhitungan TF-IDF:

Tabel 4. 9 Hasil TF-IDF

No	*Term*	*TF-IDF*
1.	kuhp	293.6159976199091
2.	dukung	188.37304136172648
3.	indonesia	172.7143811211307
4.	hukum	135.37367144662738
5.	bangsa	124.12142120052982
6.	maju	113.0752430308218
7.	uu	102.43175077671972
8.	kesah	101.20325567207273
9.	anak	78.41454366237159
10.	ruu	72.69310769626262
11.	buat	68.98066933376164
12.	Pasal	55.376395683567246
13.	masyarakat	49.89256234028645
14.	mantap	47.93596939775104
15.	rkuhp	46.71394031446183
16.	pidana	43.28568769297896
17.	adil	42.387578381758814
18.	produk	40.52646827402381
19.	demokrasi	39.3857570592137
20.	jamin	38.00905468291425

4.6 Klasifikasi

1. *Split Validation*

Pada tahap ini setelah ekstraksi fitur dilakukan, peneliti akan melakukan eksperimen pada data tersebut. Peneliti akan men-split data (membagi data) dalam kedua kelompok data latih dan data uji dengan jumlah yang berbeda pengujian ini akan menggunakan 3 seknario yang berbeda.

Tabel 4. 10 Klasifikasi Skenario

Skenario	Data Training		Data Testing	
	Jumlah	Persentase (%)	Jumlah	Persentase(%)
1.	1874	75%	625	25%
2.	1999	80%	500	20%
3.	2249	90%	250	10%

Pembagian jumlah data latih dan data uji menggunakan library pada python *scikit-learn* yaitu model train_test_split. Parameter yang dipakai ialah test_size . dalam pembagian data latih dan data uji dilakukan tahap pembagian data latih sebesar 90% dan data uji sebesar 10%. Untuk skenario selanjutnya dalam parameter test_size akan dimodifikasi dalam pembagian data sebesar 0.20 dan 0.25 untuk data uji nya.

```
#split data
from sklearn.model_selection import train_test_split
x = tweet['clean_tweets']
y = tweet['Polaritas']

x_train, x_test, y_train, y_test = train_test_split(x, y, test_size =
0.10, random_state=42)
```

4.7 Klasifikasi menggunakan SMOTE

Dalam tahapan ini peneliti menggunakan *Synthetic Minority Oversampling Technique* (SMOTE). Dalam SMOTE menggunakan library *imblearn.over_sampling* yaitu model

SMOTE. Dalam kasus terdapat ketidakseimbangan pada sebuah dataset dimana hasil dari positif dan negatif tidak seimbang, oleh karena itu penggunaan SMOTE ini diperlukan agar memaksimalkan klasifikasi dalam sebuah dataset.

```
from imblearn.over_sampling import SMOTE
sm = SMOTE(sampling_strategy='minority', random_state=42)
x_train_res, y_train_res = sm.fit_resample(x_train, y_train)
```

4.8 Klasifikasi menggunakan SVM

Setelah tahap pengujian pada data training dan data uji, selanjutnya masuk ke tahap klasifikasi menggunakan algoritma SVM. Dalam tahap ini peneliti melakukan uji klasifikasi menggunakan library pada python yaitu *scikit-learn* dengan modul svm.

Peneliti menggunakan parameter dalam pemodelan SVM ini dengan kernel linear dengan nilai *cost = 0.25* yang merupakan nilai *default.* Selanjutnya penulis memakai *hyperparameter tuning GridSearchCV*. Hyperparameter ini digunakan pada performa model yang dihasilkan sehingga untuk membantu menemukan parameter terbaik dalam suatu model klasifikasi.

Tabel 4. 11 Parameter SVM

Parameter	Value
Kernel	Linear
Cost	0,25
Hyperparameter GridSearchCV	5

Dari pemodelan diatas dapat dilakukan penulisan dalam *script* program sebagai berikut:

```
from sklearn.svm import LinearSVC
svm = LinearSVC(C=0.25)
param_grid = {'C': [0.1, 1, 10]}
svm_cv = GridSearchCV(svm, param_grid, cv=5)
```

```
svm_cv.fit(x_train, y_train)
```

Dalam proses tersebut akan dihasilkan sebuah model *machine learning* untuk mengklasifikasikan dalam memprediksi label atau sentimen pada data uji tersebut.

4.9 Klasifikasi Menggunakan *Decision Tree*

Setelah tahap pengujian pada data training dan data uji, selanjutnya masuk ke tahap klasifikasi menggunakan algoritma Decision Tree. Dalam tahap ini peneliti melakukan uji klasifikasi menggunakan library pada python yaitu *scikit-learn* dengan modul *tree*.

Selanjutnya peneliti memakai *hyperparameter tuning GridSearchCV* untuk membantu pemodelan dalam parameter dalam decision tree. *Hyperparameter* ini digunakan pada performa model yang dihasilkan sehingga untuk membantu menemukan parameter teroptimal dalam suatu model klasifikasi pada Decision Tree.

Tabel 4. 12 Parameter *Decision Tree*

Parameter	Value
Hyperparameter GridSearchCV	5
Max-Depth	10

```
from sklearn.tree import DecisionTreeClassifier
dt = DecisionTreeClassifier(max_depth=10)
param_grid = {'max_depth': [5, 10, 15]}
dt_cv = GridSearchCV(dt, param_grid, cv=5)
dt_cv.fit(x_test, y_test)
```

Dalam proses tersebut akan dihasilkan sebuah model *machine learning* untuk mengklasifikasikan dalam memprediksi label atau sentimen pada data uji tersebut.

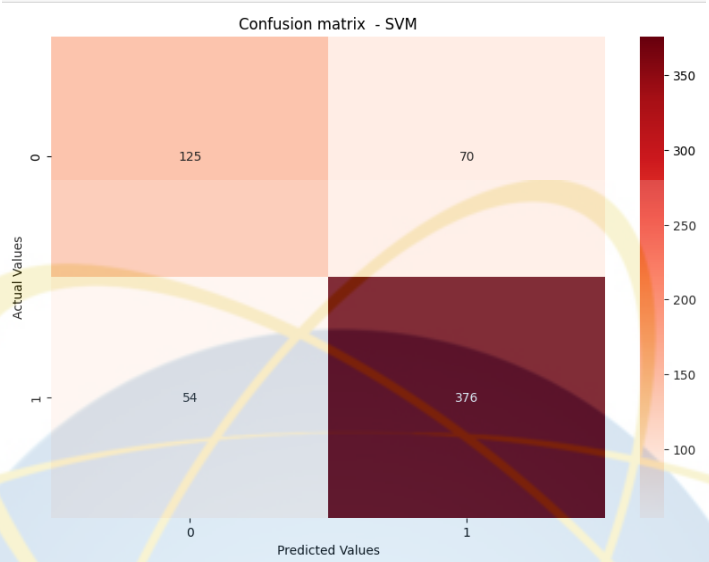
BAB V
HASIL DAN PEMBAHASAN

5.1 Evaluasi Model SVM dan *Decision Tree*

Pada sub bab sebelumnya peneliti sudah menjabarkan dalam menentukan model training dalam sebuah data latih dan data uji dengan splitting data ke dalam 3 skenario, yaitu untuk skenario ke-1, data latih sebesar 90% dan 10% untuk data uji. skenario ke-2 dengan rasio data latih sebesar 80% dan 20% data uji. skenario ke-3 dengan rasio data latih sebesar 75% dan 25% data uji.

5.1.1 *Confusion Matrix* SVM

Berikut adalah hasil dari *confusion matrix* dengan 25% data uji atau sama dengan 625 data uji.



Gambar 5. 1 *Heatmap Confusion Matrix* SVM skenario 1

Dari *confusion matrix* tersebut menunjukan bahwa mayoritas data berhasil di prediksi dengan benar atau sesuai dengan data aktualnya. Untuk keterangan lebih lanjut dapat dilihat pada tabel dibawah ini:

Tabel 5. 1 Hasil *Confusion Matrix* SVM skenario 1

Predicted / Actual	Negative (0)	Positive (1)
Negative (0)	125 (TN)	70 (FP)
Positive (1)	54 (FN)	376 (TP)

Dari tabel diatas, dapat dijelaskan bahwa:

- Sebanyak 125 data label negatif diprediksi benar sebagai label negatif (TN)
- Sebanyak 376 data label positif diprediksi benar sebagai label positif (TP)
- Sebanyak 54 data label positif diprediksi salah sebagai label negatif (FN)

- Sebanyak 70 data label negatif diprediksi salah sebagai label positif (FP)

Berdasarkan dari tabel *confusion matrix* digunakan dasar perhitungan metrik dalam menentukan performa model diantaranya *accuracy, precision, recall , f1-score* dan *Area Under Curve.* berikut proses perhitungan peneliti jabarkan sebagai berikut:

a. *accuracy*

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)}$$

$$Accuracy = \frac{376 + 125}{376 + 125 + 54 + 70}$$

$$Accuracy = \frac{501}{625} \times 100\% = 80.16\%$$

b. *precision*

$$Precision = \frac{(TP)}{(TP + FP)}$$

$$Precision = \frac{376}{376 + 70} \times 100$$

$$Precision = 84.3$$

c. *recall*

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

$$Recall = \frac{376}{376 + 54} \times 100$$

$$Recall = 87.44$$

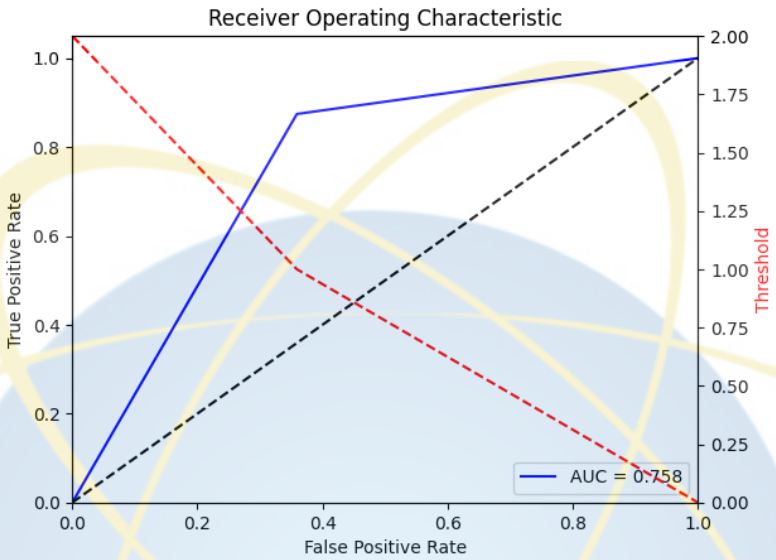
d. *f1-score*

$$FI - score = \frac{(2 \times recall \times precision)}{recall + precision}$$

$$FI - score = \frac{(2 \times 87.44 \times 84.3)}{87.44 + 84.3} \times 100$$

$$FI - score = 85.84$$

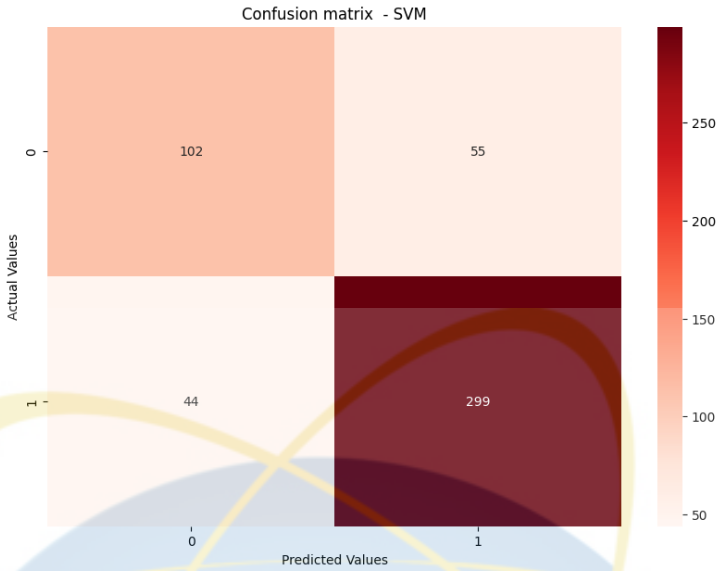
e. *Area Under Curve*



Gambar 5. 2 *Area Under Curve* SVM skenario 1

5.1.2 *Confusion Matrix* SVM

Berikut hasil dari *connfusion matrix* algoritma SVM dari 20% data uji atau sebesar 500 data uji.



Gambar 5. 3 *Heatmap Confusion Matrix* SVM skenario 2

Dari *confusion matrix* tersebut menunjukan bahwa mayoritas data berhasil di prediksi dengan benar atau sesuai dengan data aktualnya. Untuk keterangan lebih lanjut dapat dilihat pada tabel dibawah ini:

Tabel 5. 2 Hasil *Confusion Matrix* SVM skenario 2

Predicted Actual	Negative (0)	Positive (1)
Negative (0)	102 (TN)	55 (FP)
Positive (1)	44 (FN)	299 (TP)

Dari tabel diatas, dapat dijelaskan bahwa:

- Sebanyak 102 data label negatif diprediksi benar sebagai label negatif (TN)
- Sebanyak 299 data label positif diprediksi benar sebagai label positif (TP)
- Sebanyak 44 data label positif diprediksi salah sebagai label negatif (FN)
- Sebanyak 55 data label negatif diprediksi salah sebagai label positif (FP)

Berdasarkan dari tabel *confusion matrix* digunakan dasar perhitungan metrik dalam menentukan performa model diantaranya *accuracy*, *precision, recall , f1-score* dan *Area Under Curve.* berikut proses perhitungan peneliti jabarkan sebagai berikut:

a. *accuracy*

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)}$$

$$Accuracy = \frac{299 + 102}{299 + 102 + 55 + 44}$$

$$Accuracy = \frac{401}{500} \times 100\% = 80.2\%$$

b. *precision*

$$Precision = \frac{(TP)}{(TP + FP)}$$

$$Precision = \frac{299}{299 + 55} \times 100$$

$$Precision = 84.46$$

c. *recall*

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

$$Recall = \frac{299}{299 + 44} \times 100$$

$$Recall = 87.17$$

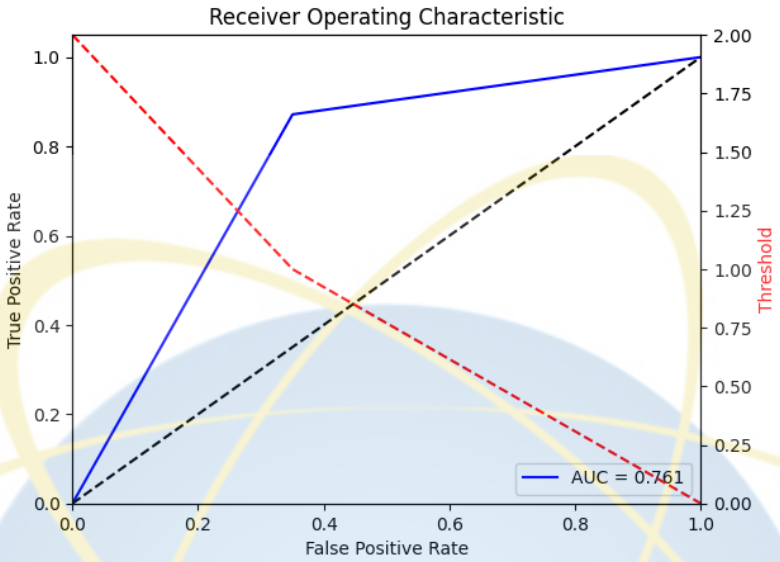
d. *f1-score*

$$FI - score = \frac{(2 \times recall \times precision)}{recall + precision}$$

$$FI - score = \frac{(2 \times 87.17 \times 84.46)}{87.17 + 84.46} \times 100$$

$$FI - score = 85.8$$

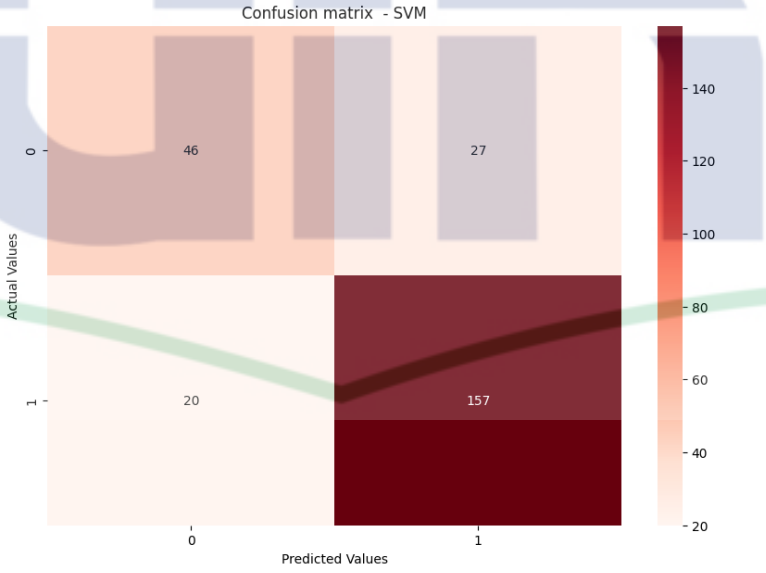
e. *Area Under Curve*



Gambar 5. 4 *Area Under Curve* SVM Skenario 2

5.1.3 *Confusion Matrix* SVM

Berikut hasil dari *connfusion matrix* algoritma SVM dari 10% data uji atau sebesar 250 data uji.



Gambar 5. 5 *Heatmap Confusion Matrix* SVM skenario 3

Dari *confusion matrix* tersebut menunjukan bahwa mayoritas data berhasil di prediksi dengan benar atau sesuai dengan data aktualnya. Untuk keterangan lebih lanjut dapat dilihat pada tabel dibawah ini:

Tabel 1. 3 Hasil Confusion Matrix SVM Skenario 3

Predicted / Actual	Negative (0)	Positive (1)
Negative (0)	46 (TN)	27 (FP)
Positive (1)	20 (FN)	157 (TP)

Dari tabel diatas, dapat dijelaskan bahwa:

- Sebanyak 46 data label negatif diprediksi benar sebagai label negatif (TN)
- Sebanyak 157 data label positif diprediksi benar sebagai label positif (TP)
- Sebanyak 20 data label positif diprediksi salah sebagai label negatif (FN)
- Sebanyak 27 data label negatif diprediksi salah sebagai label positif (FP)

Berdasarkan dari tabel *confusion matrix* digunakan dasar perhitungan metrik dalam menentukan performa model diantaranya *accuracy, precision, recall, f1-score* dan *Area Under Curve.* berikut proses perhitungan peneliti jabarkan sebagai berikut:

a. *accuracy*

$$Accuracy = \frac{(TP + TN)}{TP + TN + FP + FN}$$

$$Accuracy = \frac{157 + 46}{157 + 46 + 27 + 20}$$

$$Accuracy = \frac{203}{250} \times 100\% = 81.2\%$$

b. *precision*

$$Precision = \frac{(TP)}{(TP + FP)}$$

$$Precision = \frac{157}{157 + 27} \times 100$$

$$Precision = 85.33$$

c. *recall*

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

$$Recall = \frac{157}{157 + 20}$$

$$Recall = 88.7$$

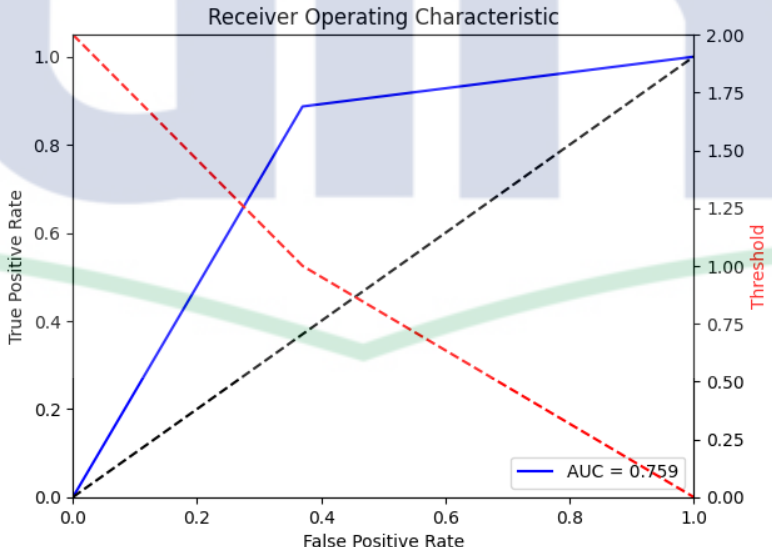
d. *f1-score*

$$FI - score = \frac{(2 \times recall \times precision)}{recall + precision}$$

$$FI - score = \frac{(2 \times 88.7 \times 85.33)}{88.7 + 85.33} \times 100$$

$$FI - score = 86.98$$

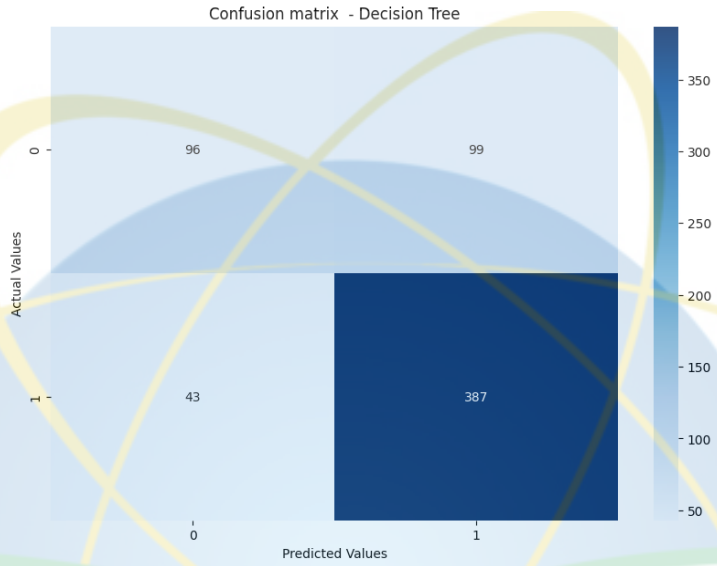
e. *Area Under Curve*



Gambar 5. 6 *Area Under Curve* SVM Skenario 3

5.1.4 *Confusion Matrix Decision Tree*

Berikut adalah hasil dari *confusion matrix* dengan 25 % data uji atau sama dengan 625 data uji.



Gambar 5. 7 *Heatmap Confusion Matrix Decision Tree* Skenario 1

Dari *confusion matrix* tersebut menunjukan bahwa mayoritas data berhasil di prediksi dengan benar atau sesuai dengan data aktualnya. Untuk keterangan lebih lanjut dapat dilihat pada tabel dibawah ini:

Tabel 5. 4 Hasil *Confusion Matrix* Decision Tree Skenario 1

Predicted / Actual	Negative (0)	Positive (1)
Negative (0)	96 (TN)	99 (FP)
Positive (1)	43 (FN)	387 (TP)

Dari tabel diatas, dapat dijelaskan bahwa:

- Sebanyak 96 data label negatif diprediksi benar sebagai label negatif (TN)

- Sebanyak 387 data label positif diprediksi benar sebagai label positif (TP)
- Sebanyak 43 data label positif diprediksi salah sebagai label negatif (FN)
- Sebanyak 99 data label negatif diprediksi salah sebagai label positif (FP)

Berdasarkan dari tabel *confusion matrix* digunakan dasar perhitungan metrik dalam menentukan performa model diantaranya *accuracy, precision, recall, f1-score dan Area Under Curve.* berikut proses perhitungan peneliti jabarkan sebagai berikut:

a. *accuracy*

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)}$$

$$Accuracy = \frac{387 + 96}{387 + 96 + 99 + 43}$$

$$Accuracy = \frac{483}{625} \times 100\% = 77.28\%$$

b. *precision*

$$Precision = \frac{(TP)}{(TP + FP)}$$

$$Precision = \frac{387}{387 + 99} \times 100$$

$$Precision = 79.63$$

c. *recall*

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

$$Recall = \frac{387}{387 + 43} \times 100$$

$$Recall = 90.0$$

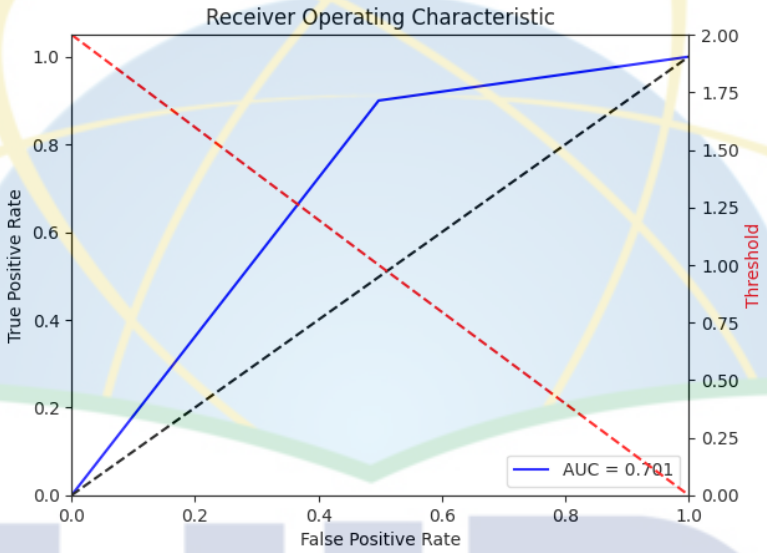
d. *f1-score*

$$FI - score = \frac{(2 \times recall \times precision)}{recall + precision}$$

$$FI - score = \frac{(2 \times 90.0 \times 79.63)}{90.0 + 79.63} \times 100$$

$$FI - score = 84.5$$

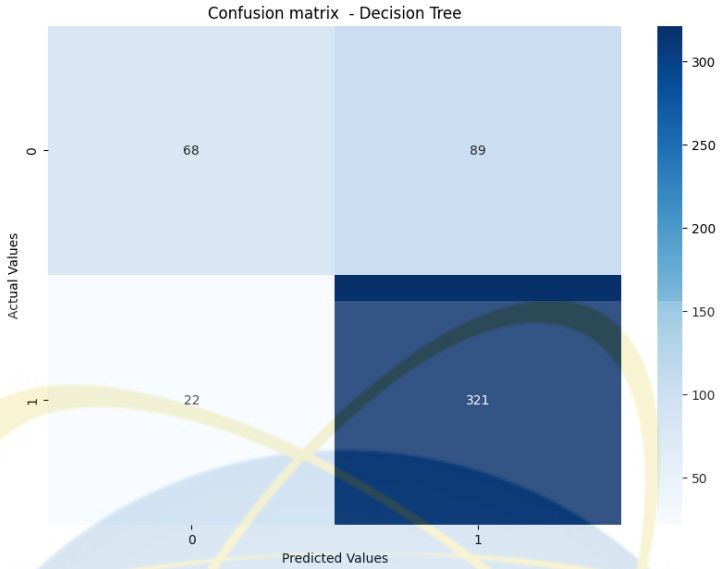
e. *Area Under Curve*



Gambar 5. 8 *Area Under Curve Decision Tree* skenario 1

5.1.5 *Confusion Matrix Decision Tree*

Berikut hasil dari *connfusion matrix* algoritma Decision Tree dari 20% data uji atau sebesar 500 data uji.



Gambar 5. 9 *Heatmap Confusion Matrix Decision Tree* skenario 2

Dari *confusion matrix* tersebut menunjukan bahwa mayoritas data berhasil di prediksi dengan benar atau sesuai dengan data aktualnya. Untuk keterangan lebih lanjut dapat dilihat pada tabel dibawah ini:

Tabel 5. 5 Hasil *Confusion Matrix Decision Tree* skenario 2

Predicted / Actual	Negative (0)	Positive (1)
Negative (0)	68 (TN)	89 (FP)
Positive (1)	22 (FN)	321 (TP)

Dari tabel diatas, dapat dijelaskan bahwa:

- Sebanyak 68 data label negatif diprediksi benar sebagai label negatif (TN)
- Sebanyak 321 data label positif diprediksi benar sebagai label positif (TP)
- Sebanyak 22 data label positif diprediksi salah sebagai label negatif (FN)
- Sebanyak 89 data label negatif diprediksi salah sebagai label positif (FP)

Berdasarkan dari tabel *confusion matrix* digunakan dasar perhitungan metrik dalam menentukan performa model diantaranya *accuracy*, *precision, recall , f1-score* dan *Area Under Curve.* berikut proses perhitungan peneliti jabarkan sebagai berikut:

a. *accuracy*

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)}$$

$$Accuracy = \frac{321 + 68}{321 + 68 + 89 + 22}$$

$$Accuracy = \frac{389}{500} \times 100\% = 77.8\%$$

b. *precision*

$$Precision = \frac{(TP)}{(TP + FP)}$$

$$Precision = \frac{321}{321 + 89} \times 100$$

$$Precision = 78.29$$

c. *recall*

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

$$Recall = \frac{321}{321 + 22} \times 100$$

$$Recall = 93.59$$

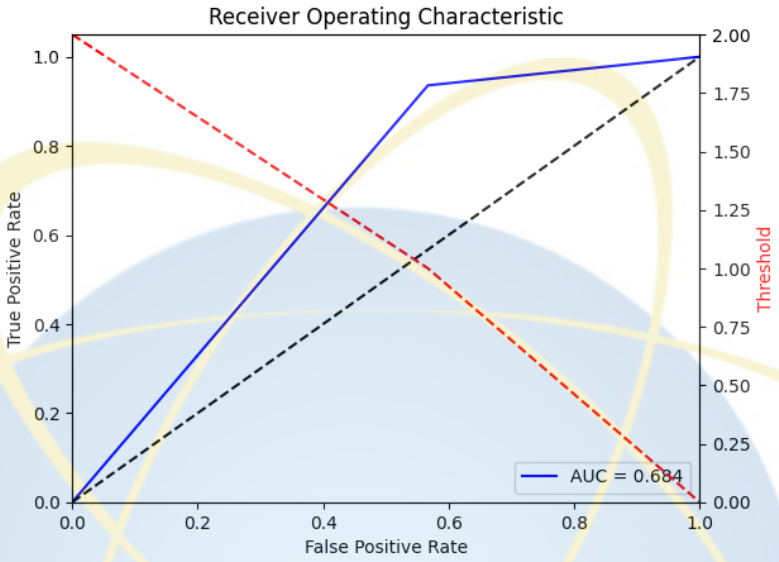
d. *f1-score*

$$FI - score = \frac{(2 \times recall \times precision)}{recall + precision}$$

$$FI - score = \frac{(2 \times 93.59 \times 78.29)}{93.59 + 78.29} \times 100$$

$$FI - score = 85.26$$

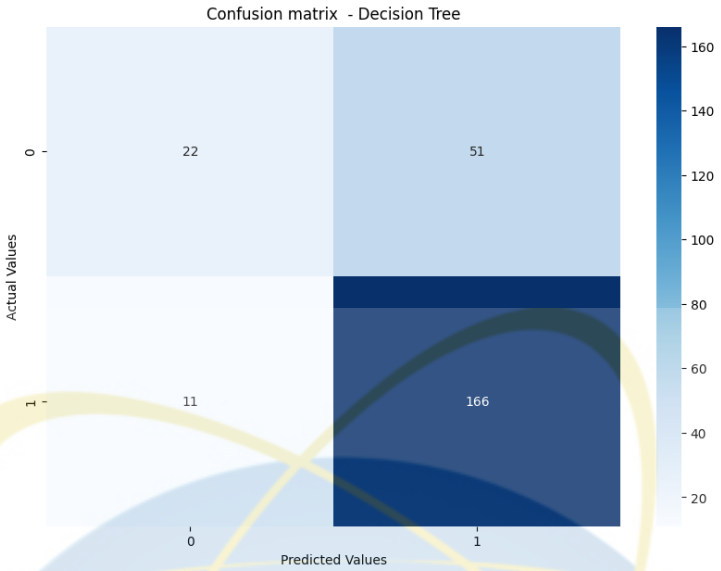
e. *Area Under Curve*



Gambar 5. 10 *Area Under Curve Decision Tree* skenario 2

5.1.6 *Confusion Matrix Decision Tree*

Berikut hasil dari *connfusion matrix* algoritma Decision Tree dari 10% data uji atau sebesar 250 data uji.



Gambar 5. 11 *Heatmap Confusion Matrix Decision Tree skenario 3*

Dari *confusion matrix* tersebut menunjukan bahwa mayoritas data berhasil di prediksi dengan benar atau sesuai dengan data aktualnya. Untuk keterangan lebih lanjut dapat dilihat pada tabel dibawah ini:

Tabel 5. 6 Hasil *Confusion Matrix Decision Tree* skenario 3

Predicted / Actual	Negative (0)	Positive (1)
Negative (0)	22 (TN)	51 (FP)
Positive (1)	11 (FN)	166 (TP)

Dari tabel diatas, dapat dijelaskan bahwa:

- Sebanyak 22 data label negatif diprediksi benar sebagai label negatif (TN)
- Sebanyak 166 data label positif diprediksi benar sebagai label positif (TP)
- Sebanyak 11 data label positif diprediksi salah sebagai label negatif (FN)
- Sebanyak 51 data label negatif diprediksi salah sebagai label positif (FP)

Berdasarkan dari tabel *confusion matrix* digunakan dasar perhitungan metrik dalam menentukan performa model diantaranya *accuracy*, *precision, recall , f1-score* dan *Area Under Curve.* berikut proses perhitungan peneliti jabarkan sebagai berikut:

a. *accuracy*

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Accuracy = \frac{166 + 22}{166 + 22 + 51 + 11}$$

$$Accuracy = \frac{188}{250} \times 100\% = 75.2\%$$

b. *precision*

$$Precision = \frac{(TP)}{(TP + FP)}$$

$$Precision = \frac{166}{166 + 51} \times 100$$

$$Precision = 76.5$$

c. *recall*

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

$$Recall = \frac{166}{166 + 11} \times 100$$

$$Recall = 93.79$$

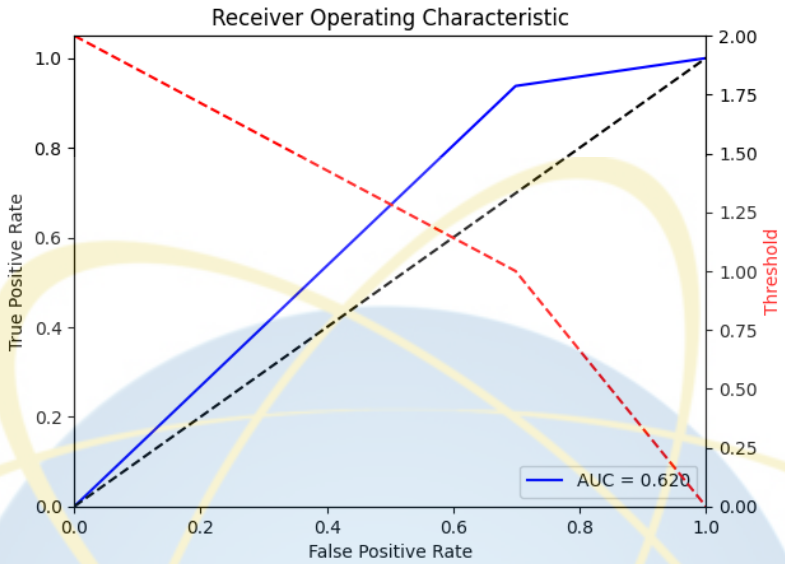
d. *f1-score*

$$FI - score = \frac{(2 \times recall \times precision)}{recall + precision}$$

$$FI - score = \frac{(2 \times 93.79 \times 76.5)}{93.79 + 76.5} \times 100$$

$$FI - score = 84.26$$

e. *Area Under Curve*



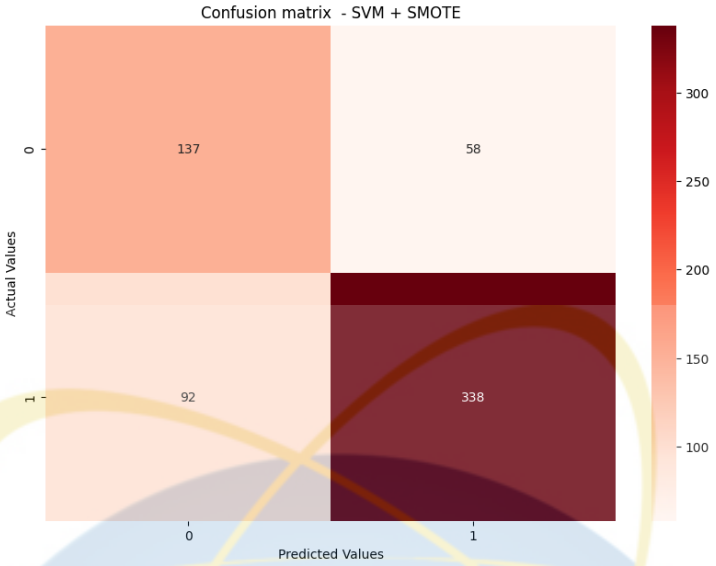
Gambar 5. 12 *Area Under Curve Decision Tree* skenario 3

5.2 Evaluasi Model SVM dan Decision Tree dengan SMOTE

Pada sub bab sebelumnya peneliti sudah menjabarkan dalam menentukan model training dalam sebuah data latih dan data uji dengan *splitting data* ke dalam 3 skenario, yaitu untuk skenario ke 1 untuk data latih sebesar 90% dan 10% untuk data uji. skenario ke 2 dengan rasio data latih sebesar 80% dan 20% data uji. skenario ke-3 dengan rasio data latih sebesar 75% dan 25% data uji. Selanjutnya peneliti dalam tahapan ini menggunakan SMOTE pada proses klasifikasi dengan SVM dan Decision Tree. Berikut hasil dari klasifikasi dengan SMOTE sebagai berikut :

5.2.1 *Confusion Matrix* SVM dengan SMOTE

Berikut adalah hasil dari *confusion matrix* dengan 25% data uji atau sama dengan 625 data uji.



Gambar 1. 13 Heatmap *Confusion Matrix* SVM dengan SMOTE skenario 1

Dari *confusion matrix* tersebut menunjukan bahwa mayoritas data berhasil di prediksi dengan benar atau sesuai dengan data aktualnya. Untuk keterangan lebih lanjut dapat dilihat pada tabel dibawah ini:

Tabel 5. 7 Hasil *Confusion Matrix* SVM dengan SMOTE skenario 1

Predicted / Actual	Negative (0)	Positive (1)
Negative (0)	137 (TN)	58 (FP)
Positive (1)	92 (FN)	338 (TP)

Dari tabel diatas, dapat dijelaskan bahwa:

- Sebanyak 137 data label negatif diprediksi benar sebagai label negatif (TN)
- Sebanyak 338 data label positif diprediksi benar sebagai label positif (TP)
- Sebanyak 92 data label positif diprediksi salah sebagai label negatif (FN)
- Sebanyak 58 data label negatif diprediksi salah sebagai label positif (FP)

Berdasarkan dari tabel *confusion matrix* digunakan dasar perhitungan metrik dalam menentukan performa model diantaranya *accuracy*, *precision, recall , f1-score* dan *Area Under Curve.* berikut proses perhitungan peneliti jabarkan sebagai berikut:

a. *accuracy*

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)}$$

$$Accuracy = \frac{338 + 137}{338 + 137 + 58 + 92}$$

$$Accuracy = \frac{475}{625} \times 100\% = 76.0\%$$

b. *precision*

$$Precision = \frac{(TP)}{(TP + FP)}$$

$$Precision = \frac{338}{338 + 58} \times 100$$

$$Precision = 85.35$$

c. *recall*

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

$$Recall = \frac{338}{338 + 92} \times 100$$

$$Recall = 78.86$$

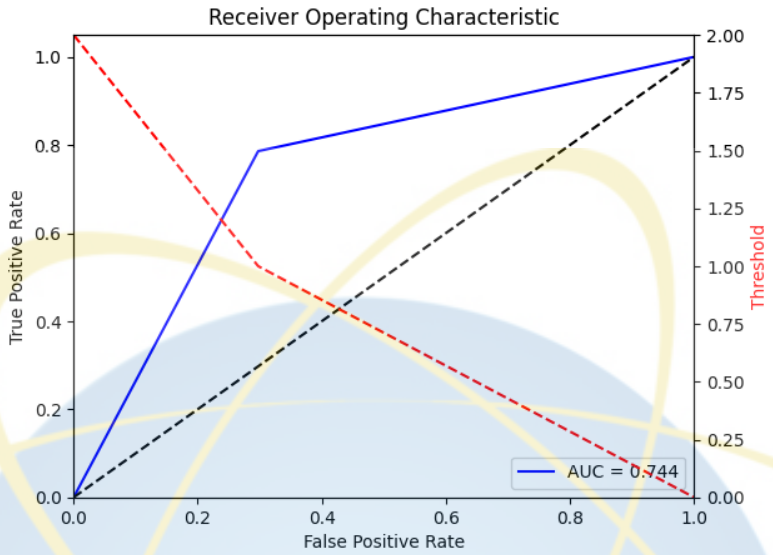
d. *f1-score*

$$FI - score = \frac{(2 \times recall \times precision)}{recall + precision}$$

$$FI - score = \frac{(2 \times 78.86 \times 85.35)}{78.86 + 85.35} \times 100$$

$$FI - score = 81.84$$

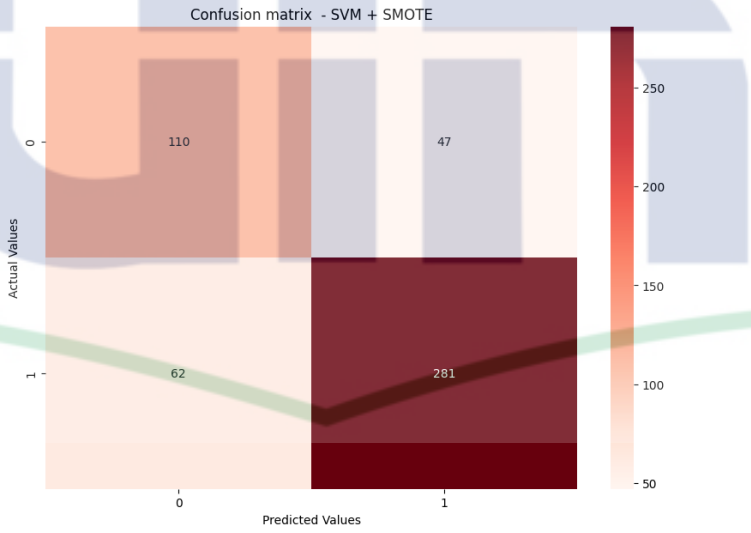
e. *Area Under Curve*



Gambar 5. 14 *Area Under Curve* SVM dengan SMOTE skenario 1

5.2.2 *Confusion Matrix* SVM dengan SMOTE

Berikut hasil dari *connfusion matrix* algoritma SVM dengan SMOTE dari 20% data uji atau sebesar 500 data uji dengan SMOTE



Gambar 5. 15 *Heatmap Confusion Matrix* SVM dengan SMOTE skenario 2

Dari *confusion matrix* tersebut menunjukan bahwa mayoritas data berhasil di prediksi dengan benar atau sesuai dengan data aktualnya. Untuk keterangan lebih lanjut dapat dilihat pada tabel dibawah ini:

Tabel 5. 8 Hasil *Confusion Matrix* SVM dengan SMOTE skenario 2

Predicted / Actual	Negative (0)	Positive (1)
Negative (0)	110 (TN)	47 (FP)
Positive (1)	62 (FN)	281 (TP)

Dari tabel diatas, dapat dijelaskan bahwa:

- Sebanyak 110 data label negatif diprediksi benar sebagai label negatif (TN)
- Sebanyak 281 data label positif diprediksi benar sebagai label positif (TP)
- Sebanyak 62 data label positif diprediksi salah sebagai label negatif (FN)
- Sebanyak 47 data label negatif diprediksi salah sebagai label positif (FP)

Berdasarkan dari tabel *confusion matrix* digunakan dasar perhitungan metrik dalam menentukan performa model diantaranya *accuracy, precision, recall , f1-score* dan *Area Under Curve.* berikut proses perhitungan peneliti jabarkan sebagai berikut:

a. *accuracy*

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)}$$

$$Accuracy = \frac{281 + 110}{281 + 110 + 47 + 62}$$

$$Accuracy = \frac{391}{500} \times 100\% = 78.2\%$$

b. *precision*

$$Precision = \frac{(TP)}{(TP + FP)}$$

$$Precision = \frac{281}{281 + 47} \times 100$$

$$Precision = 85.67$$

c. *recall*

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

$$Recall = \frac{281}{281 + 62} \times 100$$

$$Recall = 81.92$$

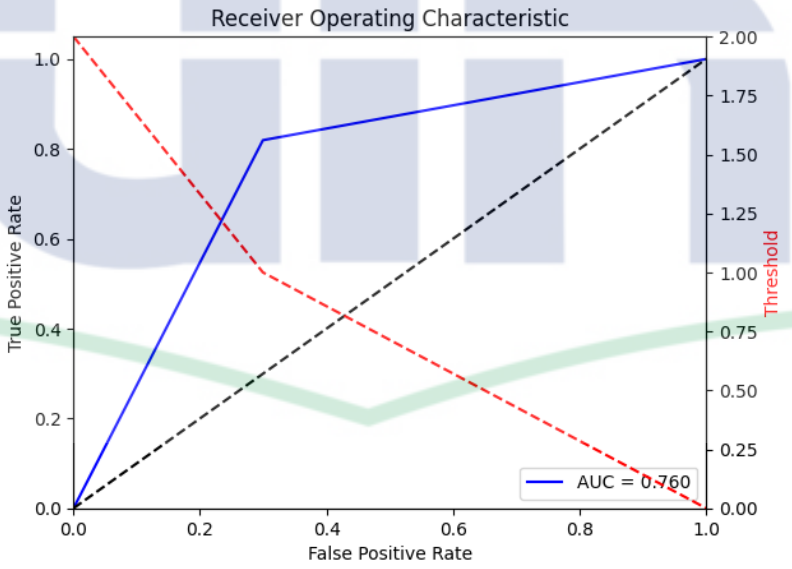
d. *f1-score*

$$FI - score = \frac{(2 \times recall \times precision)}{recall + precision}$$

$$FI - score = \frac{(2 \times 81.92 \times 85.67)}{81.92 + 85.67} \times 100$$

$$FI - score = 83.76$$

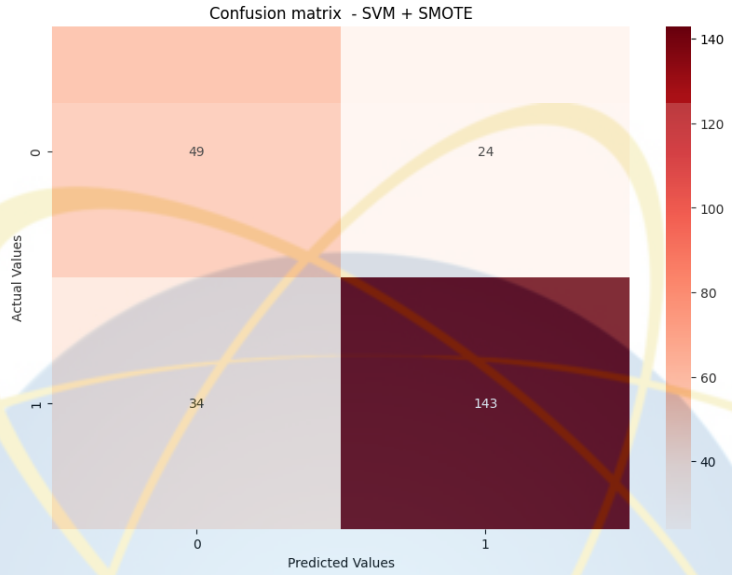
e. *Area Under Curve*



Gambar 5. 16 *Area Under Curve* SVM dengan SMOTE skenario 2

5.2.3 *Confusion Matrix* SVM dengan SMOTE

Berikut hasil dari *connfusion matrix* algoritma SVM dari 10% data uji atau sebesar 250 data uji dengan SMOTE.



Gambar 5. 17 *Heatmap Confusion Matrix* SVM dengan SMOTE skenario 3

Dari *confusion matrix* tersebut menunjukan bahwa mayoritas data berhasil di prediksi dengan benar atau sesuai dengan data aktualnya. Untuk keterangan lebih lanjut dapat dilihat pada tabel dibawah ini:

Tabel 5. 9 Hasil *Confusion Matrix* SVM dengan SMOTE skenario 3

Predicted / Actual	Negative (0)	Positive (1)
Negative (0)	49 (TN)	24 (FP)
Positive (1)	34 (FN)	143 (TP)

Dari tabel diatas, dapat dijelaskan bahwa:

- Sebanyak 49 data label negatif diprediksi benar sebagai label negatif (TN)
- Sebanyak 143 data label positif diprediksi benar sebagai label positif (TP)

- Sebanyak 34 data label positif diprediksi salah sebagai label negatif (FN)
- Sebanyak 24 data label negatif diprediksi salah sebagai label positif (FP)

Berdasarkan dari tabel *confusion matrix* digunakan dasar perhitungan metrik dalam menentukan performa model diantaranya *accuracy, precision, recall , f1-score* dan *Area Under Curve.* berikut proses perhitungan peneliti jabarkan sebagai berikut:

a. *accuracy*

$$Accuracy = \frac{(TP + TN)}{TP + TN + FP + FN}$$

$$Accuracy = \frac{143 + 49}{143 + 49 + 24 + 34}$$

$$Accuracy = \frac{192}{250} \times 100\% = 76.8\%$$

b. *precision*

$$Precision = \frac{(TP)}{(TP + FP)}$$

$$Precision = \frac{143}{143 + 24} \times 100$$

$$Precision = 85.63$$

c. *recall*

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

$$Recall = \frac{143}{143 + 34}$$

$$Recall = 80.79$$

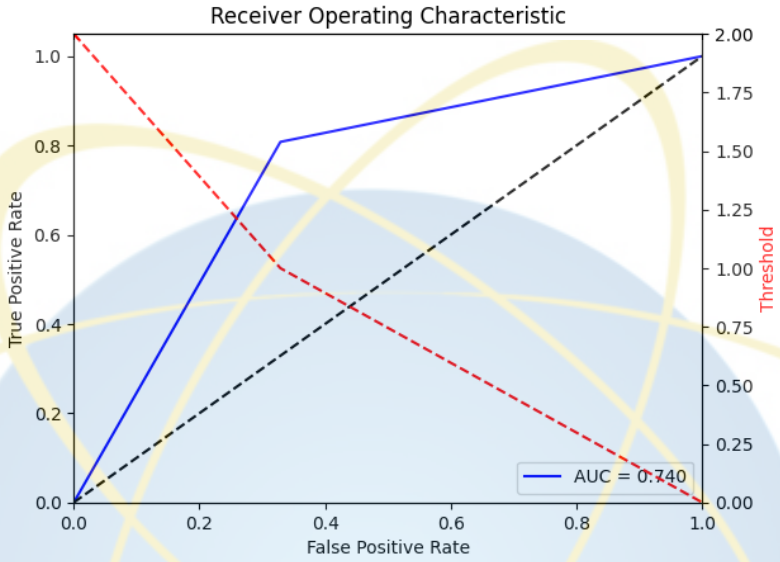
d. *f1-score*

$$FI - score = \frac{(2 \times recall \times precision)}{recall + precision}$$

$$FI - score = \frac{(2 \times 80.79 \times 85.63)}{80.79 + 85.63} \times 100$$

$$FI - score = 83.14$$

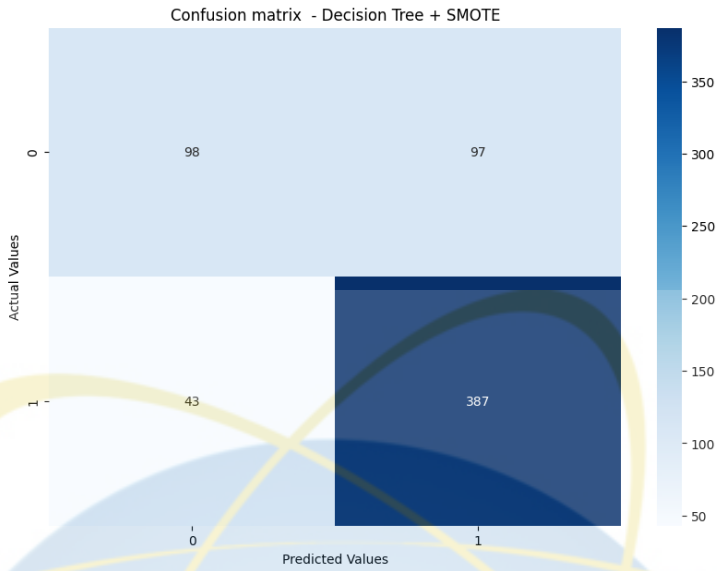
e. *Area Under Curve*



Gambar 5. 18 *Area Under Curve* SVM dengan SMOTE skenario 3

5.2.4 Confusion Matrix Decision Tree dengan SMOTE

Berikut adalah hasil dari *confusion matrix* dengan 25% data uji atau sama dengan 625 data uji dengan SMOTE.



Gambar 2. 19 *Heatmap Confusion Matrix Decision Tree* dengan SMOTE skenario 1

Dari *confusion matrix* tersebut menunjukan bahwa mayoritas data berhasil di prediksi dengan benar atau sesuai dengan data aktualnya. Untuk keterangan lebih lanjut dapat dilihat pada tabel dibawah ini:

Tabel 2. 10 Hasil *Confusion Matrix Decision Tree* dengan SMOTE skenario 1

Predicted / Actual	Negative (0)	Positive (1)
Negative (0)	98 (TN)	97 (FP)
Positive (1)	43 (FN)	387 (TP)

Dari tabel diatas, dapat dijelaskan bahwa:

- Sebanyak 98 data label negatif diprediksi benar sebagai label negatif (TN)
- Sebanyak 387 data label positif diprediksi benar sebagai label positif (TP)
- Sebanyak 43 data label positif diprediksi salah sebagai label negatif (FN)
- Sebanyak 97 data label negatif diprediksi salah sebagai label positif (FP)

Berdasarkan dari tabel *confusion matrix* digunakan dasar perhitungan metrik dalam menentukan performa model diantaranya *accuracy*, *precision, recall , f1-score* dan *Area Under Curve.* berikut proses perhitungan peneliti jabarkan sebagai berikut:

a. *accuracy*

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Accuracy = \frac{387 + 98}{387 + 98 + 97 + 43}$$

$$Accuracy = \frac{485}{625} \times 100\% = 77.6\%$$

b. *precision*

$$Precision = \frac{(TP)}{(TP + FP)}$$

$$Precision = \frac{387}{387 + 97} \times 100$$

$$Precision = 79.96$$

c. *recall*

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

$$Recall = \frac{387}{387 + 43} \times 100$$

$$Recall = 90.0$$

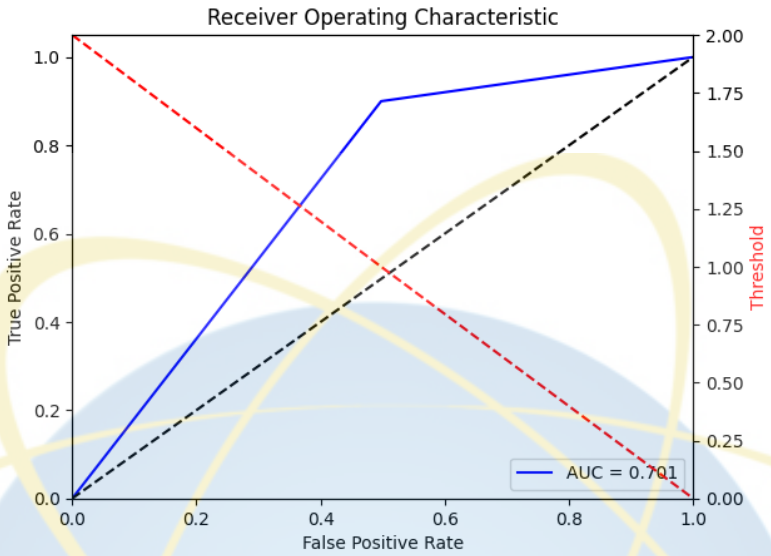
d. *f1-score*

$$FI - score = \frac{(2 \times recall \times precision)}{recall + precision}$$

$$FI - score = \frac{(2 \times 90.0 \times 79.96)}{90.0 + 79.96}$$

$$FI - score = 84.68$$

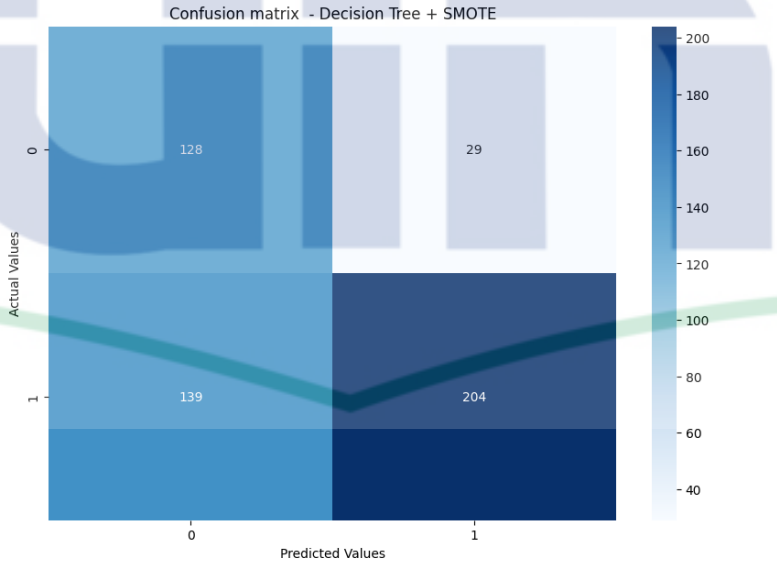
e. *Area Under Curve*



Gambar 3. 20 *Area Under Curve Decision Tree* dengan SMOTE skenario 1

5.2.5 Confusion Matrix Decision Tree dengan SMOTE

Berikut hasil dari *connfusion matrix* algoritma Decision Tree dari 20% data uji atau sebesar 500 data uji dengan SMOTE



Gambar 4. 21 *Heatmap Confusion Matrix Decision Tree* dengan SMOTE skenario 2

Dari *confusion matrix* tersebut menunjukan bahwa mayoritas data berhasil di prediksi dengan benar atau sesuai dengan data aktualnya. Untuk keterangan lebih lanjut dapat dilihat pada tabel dibawah ini:

Tabel 3. 11 Hasil *Confusion Matrix* Decision Tree dengan SMOTE skenario 2

Predicted / Actual	Negative (0)	Positive (1)
Negative (0)	128 (TN)	29 (FP)
Positive (1)	139 (FN)	204 (TP)

Dari tabel diatas, dapat dijelaskan bahwa:

- Sebanyak 128 data label negatif diprediksi benar sebagai label negatif (TN)
- Sebanyak 204 data label positif diprediksi benar sebagai label positif (TP)
- Sebanyak 29 data label positif diprediksi salah sebagai label negatif (FN)
- Sebanyak 139 data label negatif diprediksi salah sebagai label positif (FP)

Berdasarkan dari tabel *confuison matrix* digunakan dasar perhitungan metrik dalam menentukan performa model diantaranya *accuracy, precision, recall , f1-score* dan *Area Under Curve.* berikut proses perhitungan peneliti jabarkan sebagai berikut:

a. *accuracy*

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Accuracy = \frac{204 + 128}{204 + 128 + 139 + 29}$$

$$Accuracy = \frac{332}{500} \times 100\% = 66.4\%$$

b. *precision*

$$Precision = \frac{(TP)}{(TP + FP)}$$

$$Precision = \frac{204}{204 + 139} \times 100$$

$$Precision = 87.55$$

c. *recall*

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

$$Recall = \frac{204}{204 + 29} \times 100$$

$$Recall = 59.48$$

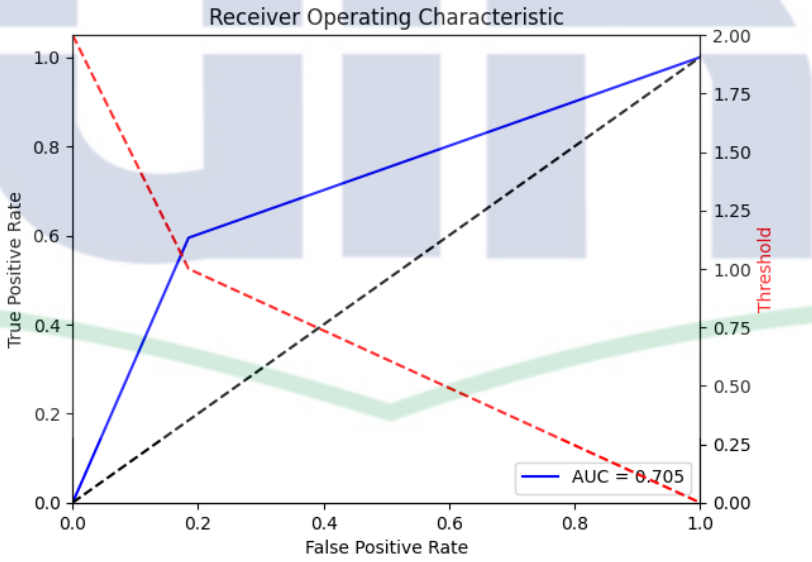
d. *f1-score*

$$FI - score = \frac{(2 \times recall \times precision)}{recall + precision}$$

$$FI - score = \frac{(2 \times 99.13 \times 84.79)}{99.13 + 84.79} \times 100$$

$$FI - score = 70.83$$

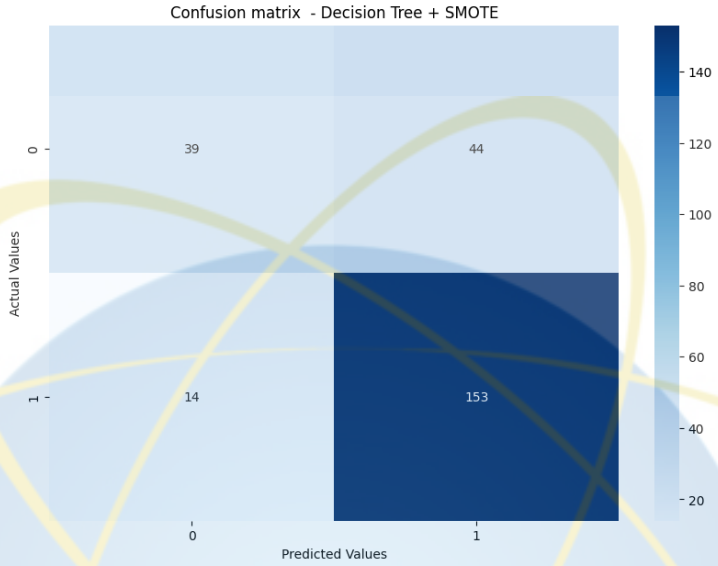
e. *Area Under Curve*



Gambar 5. 22 *Area Under Curve Decision Tree* dengan SMOTE skenario 2

5.2.6 Confusion Matrix Decision Tree dengan SMOTE

Berikut hasil dari *connfusion matrix* algoritma Decision Tree dari 10% data uji atau sebesar 250 data uji dengan SMOTE.



Gambar 6. 23 *Heatmap Confusion Matrix Decision Tree* dengan SMOTE skenario 3

Dari *confusion matrix* tersebut menunjukan bahwa mayoritas data berhasil di prediksi dengan benar atau sesuai dengan data aktualnya. Untuk keterangan lebih lanjut dapat dilihat pada tabel dibawah ini:

Tabel 4. 12 Hasil *Confusion Matrix Decision Tree* dengan SMOTE skenario 3

Predicted / Actual	Negative (0)	Positive (1)
Negative (0)	39 (TN)	44 (FP)
Positive (1)	14 (FN)	153 (TP)

Dari tabel diatas, dapat dijelaskan bahwa:

- Sebanyak 39 data label negatif diprediksi benar sebagai label negatif (TN)
- Sebanyak 153 data label positif diprediksi benar sebagai label positif (TP)

- Sebanyak 14 data label positif diprediksi salah sebagai label negatif (FN)
- Sebanyak 44 data label negatif diprediksi salah sebagai label positif (FP)

Berdasarkan dari tabel *confuison matrix* digunakan dasar perhitungan metrik dalam menentukan performa model diantaranya *accuracy, precision, recall , f1-score* dan *Area Under Curve.* berikut proses perhitungan peneliti jabarkan sebagai berikut:

a. *accuracy*

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Accuracy = \frac{153 + 39}{153 + 39 + 44 + 14}$$

$$Accuracy = \frac{192}{250} \times 100\% = 76.8\%$$

b. *precision*

$$Precision = \frac{(TP)}{(TP + FP)}$$

$$Precision = \frac{153}{153 + 44} \times 100$$

$$Precision = 77.66$$

c. *recall*

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

$$Recall = \frac{153}{153 + 14} \times 100$$

$$Recall = 91.62$$

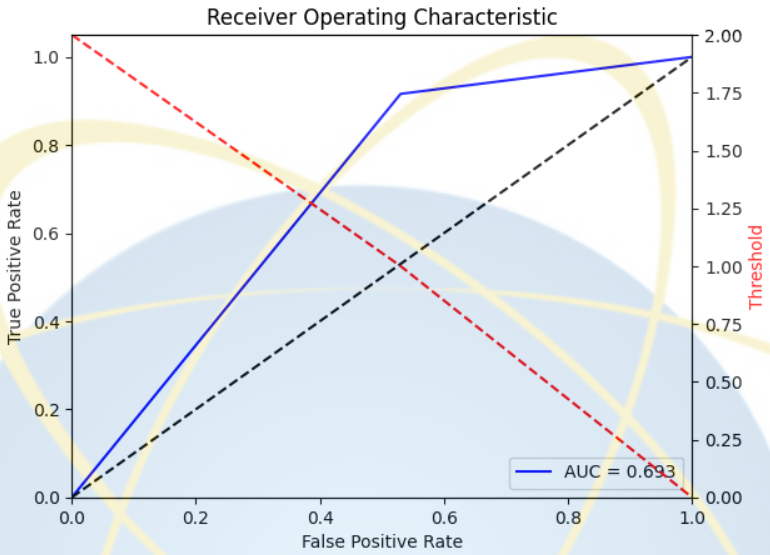
d. *f1-score*

$$FI - score = \frac{(2 \times recall \times precision)}{recall + precision}$$

$$FI - score = \frac{(2 \times 91.62 \times 77.66)}{91.62 + 77.66}$$

$$FI - score = 84.07$$

e. *Area Under Curve*



Gambar 7. 24 *Area Under Curve Decision Tree* dengan SMOTE skenario 3

5.3 Hasil Klasifikasi

Setelah melakukan tahap evaluasi performa dengan *confuion matrix* dalam skenario yang berbeda beda, maka didapatkan hasil pada tabel dibawah ini:

Tabel 5. 13 Rekapitulasi *Confusion Matrix* SVM

Evaluasi	Skenario 1	Skenario 2	Skenario 3
	Data latih 75%, 25% data uji	Data latih 80%, 20% data uji	Data latih 90%, 10% data uji
Accuracy	80.16%	80.2%	81.2%
Precision	84.3%	84.66%	85.33%
Recall	87.44%	87.17%	88.7%
F1-Score	85.84%	85.8%	86.98%
AUC	0,758	0,761	0,759

Tabel 5. 14 Rekapitulasi *Confusion Matrix Decision Tree*

Evaluasi	Skenario 1	Skenario 2	Skenario 3
	Data latih 75%, 25% data uji	Data latih 80%, 20% data uji	Data latih 90%, 10% data uji
Accuracy	77.28%	77.8%	75.2%
Precision	79.63%	78.29%	76.5%
Recall	90.0%	93.59%	93.79%
F1-Score	84.5%	85.26%	84.26%
AUC	0,701	0,684	0,620

Tabel 5. 15 Rekapitulasi *Confusion Matrix* SVM dengan SMOTE

Evaluasi	Skenario 1	Skenario 2	Skenario 3
	Data latih 75%, 25% data uji	Data latih 80%, 20% data uji	Data latih 90%, 10% data uji
Accuracy	76.0%	78.2%	76.8%
Precision	85.35%	85.67%	85.63%
Recall	78.86%	81.92%	80.79%
F1-Score	81.84%	83.76%	83.14%
AUC	0,744	0,760	0,740

Tabel 5. 16 Rekapitulasi *Confusion Matrix Decision Tree* dengan SMOTE

Evaluasi	Skenario 1	Skenario 2	Skenario 3
	Data latih 75%, 25% data uji	Data latih 80%, 20% data uji	Data latih 90%, 10% data uji
Accuracy	77.6%	66.4%	76.8%
Precision	79.96%	87.55%	77.66%
Recall	90.0%	59.48%	91.62%

F1-Score	84.68%	70.83%	84.07%
AUC	0,701	0,705	0.693

Berdasarkan tabel diatas dapat disimpulkan bahwa hasil akurasi terbesar dalam klasifikasi SVM terdapat dalam skenario 3 dengan data latih 90% dan 10% data uji, dengan nilai sebesar 81.2%. Dan sementara dalam klasifikasi dengan algoritma Decision Tree akurasi terbesar terdapat dalam skenario 2 dengan 80% data latih dan 20% data uji, dengan nilai sebesar 77.8%. setelah itu dalam klasifikasi SVM dan Decision Tree dengan SMOTE, dalam klasifikasi SVM mengalami penurunan dalam tingkat *accuracyy, recall, f1-score dan AUC*, namun dalam precision justru meningkat beberapa persen saja dalam pemakaian SMOTE tersebut, namun dalam Decision Tree dengan SMOTE dalam skenario 1 dan skenario 3 terdapat peningkatan akurasi yaitu untuk skenario 1 sebesar 0,32% dengan tingkat akurasi 77,8%. sedangkan untuk skenario 3 terdapat peningkatan akurasi 1,6%% dengan nilai 76,8%. Hal ini menunjukan bahwa pengaruh SMOTE berdampak kepada algoritma Decision Tree pada beberapa skeanrio, sehingga dalam klasifikai dengan SMOTE Decision Tree dibandingkan dengan SVM, dapat terlihat perbedaannya.

5.3.1 Hasil Klasifikasi Sentimen dengan SVM

Dibawah ini adalah tabel hasil klasifikasi sentimen dengan memilih tingkat akurasi yang lebih besar dalam evaluasi model dengan confusion matrix yaitu memakai data uji berjumlah 250 data yang berhasil diprediksi dengan benar maupun tidak dengan menggunakan algoritma SVM.

Tabel 6. 17 Hasil Klasifikasi SVM skenario 3

Keterangan	**SVM (Skenario 3)**	
	Prediksi	**Aktual**
Positif	183	176
Negatif	67	74

5.3.2 Hasil Klasifikasi Sentimen dengan Decision Tree

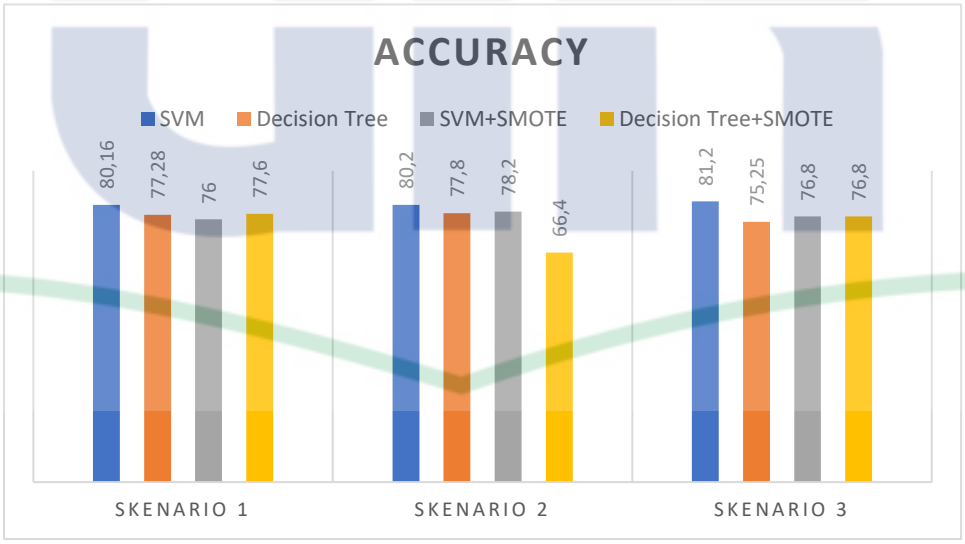
Dibawah ini adalah tabel hasil klasifikasi sentimen dengan memilih tingkat akurasi yang lebih besar dalam evaluasi model dengan confusion matrix yaitu memakai data uji berjumlah 500 data yang berhasil diprediksi dengan benar maupun tidak dengan menggunakan algoritma Decision Tree.

Tabel 7. 18 Hasil Klasifikasi Decision Tree skenario *2*

Keterangan	**Decision Tree (Skenario 2)**	
	Prediksi	**Aktual**
Positif	409	343
Negatif	91	157

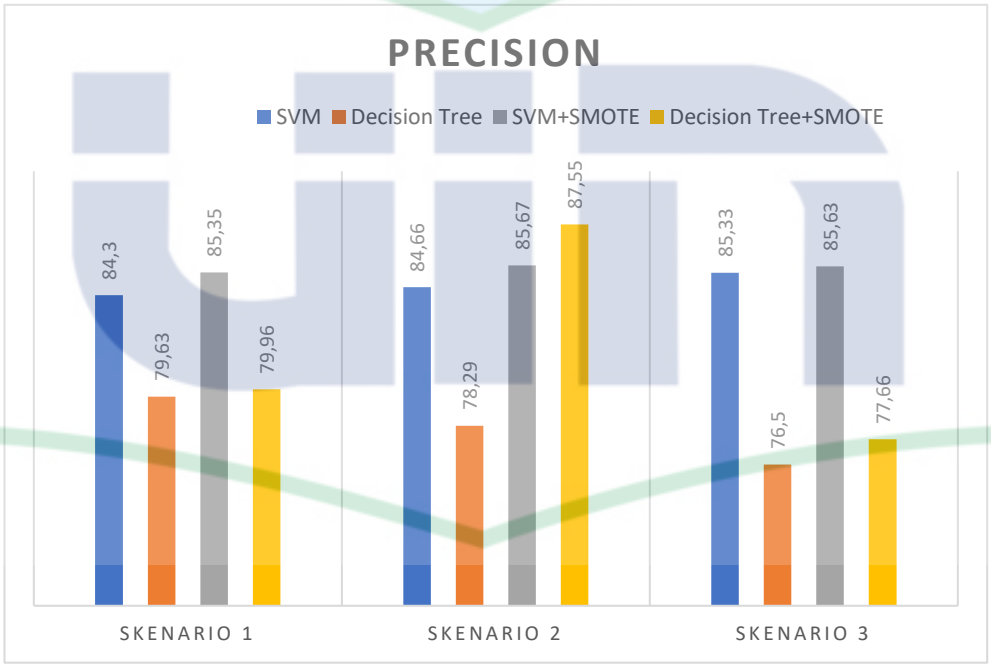
5.4 Hasil Perbandingan

Tahap sebelumnya telah dilakukan hasil klasifikasi dari algoritma SVM dan Algoritma Decision Tree dengan SMOTE. Selanjutnya perbandingan dari hasil performa confusion matrix dengan meliputi *accuracy*, *precision*, *recall*, *f1-score* dan *area under curve* (AUC) terhadap kinerja ketiga skenario dalam melakukan klasifikasi analisis sentimen.



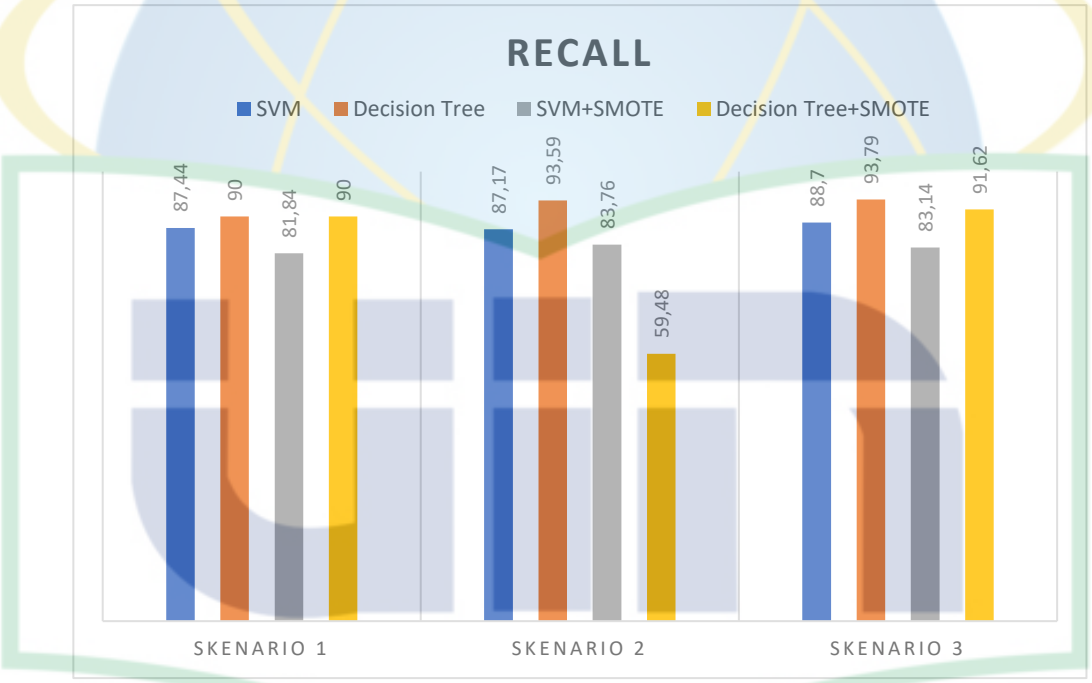
Gambar 8. 25 Grafik *Accuracy*

Hasil pengujian berdasarkan grafik menunjukkan bahwa hasil akurasi dari algoritma SVM, Decision Tree, SVM dengan SMOTE dan Decision Tree dengan SMOTE. Dalam klasifikasi diatas terdapat hasil yang beragam disetiap skenario. Dalam klasifikasi dengan menggunakan algoritma SVM cenderung meningkat dalam setiap skenario nya, sedangkan dalam klasifikasi dengan Decision Tree cenderung tidak stabil dalam hasil akurasi nya. Tingkat akurasi terbesar diperoleh SVM dengan Skenario 3 sebesar 81,2% dan terendah pada skenario 1 dengan akurasi 80,16%. Untuk algoritma Decision Tree diperoleh dalam skenario 2 sebesar 77,8% dan terendah pada skenario 3 dengan nilai akurasi 75,25%. Selanjutnya terdapat perubahan yang signifikan oleh SVM Decision Tree terhadap penggunaan SMOTE, pada SVM dalam tingkat akurasi cenderung menurun dalam penggunaan SMOTE ini, dapat dilihat pada grafik diatas. Sedangkan dalam Decision Tree terdapat beberapa peningkatan dalam skenario 1 dan 3, dengan penambahan 0,32% pada skenario 1 dan 1,6% pada skenario 3. Sehingga dapat disimpulkan bahwasannya penggunaan SMOTE pada Decision Tree dapat menambahkan akurasi pada beberapa skenario pada Decision Tree. Tingkat akurasi ini mengidentifikasikan bahwa seberapa akurat model dalam mengklasifikasikan data sentimen dengan tepat.



Gambar 9. 26 Grafik *Precision*

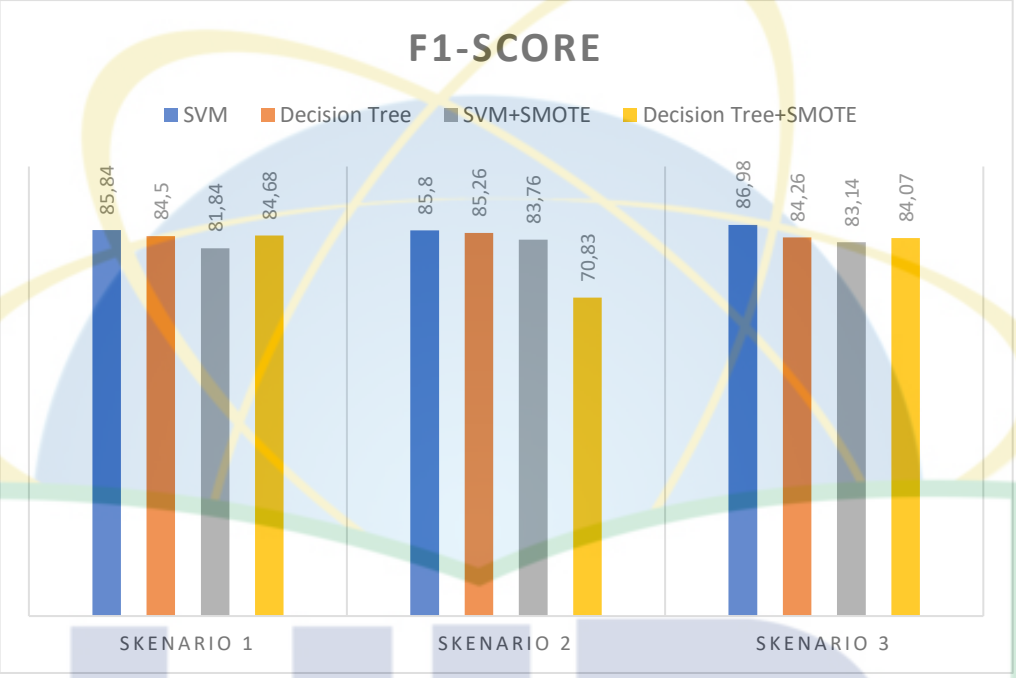
Berdasarkan hasil pengujian diatas menunjukkan hasil dari *precision* menggunakan algoritma SVM dan Decision Tree ditambah dengan penggabungan SMOTE pada algoritma tersebut. Dalam algoritma SVM nilai *precision* cenderung meningkat disetiap skenario, nilai *precision* terbesar pada skenario 3 sebesar 85,33%, sedangkan untuk Decision Tree dalam setiap skenario cenderung menurun. Dalam hasil *precision* terbesar yang dihasilkan *Decision Tree* dihasilkan pada skenario 1 dengan nilai 79,63%. Selanjutnya dalam penggunaan SMOTE dalam proses pengklasifikasian tersebut dalam SVM dan Decision Tree nilai *precision* meningkat dalam setiap skenario nya. Skenario tertinggi pada SVM dengan SMOTE terdapat pada skenario 2 dengan nilai 85,67%. Untuk *Decision Tree* dengan SMOTE terdapat pada skenario 2 dengan nilai 87,55. Tingkat *precision* ini menunjukkan bahwa, dengan nilai *preicion* yang tinggi pada percobaa tersebut, model mampu mengklasifikasikan data sentimen positif dengan baik.



Gambar 10. 27 Grafik *Recall*

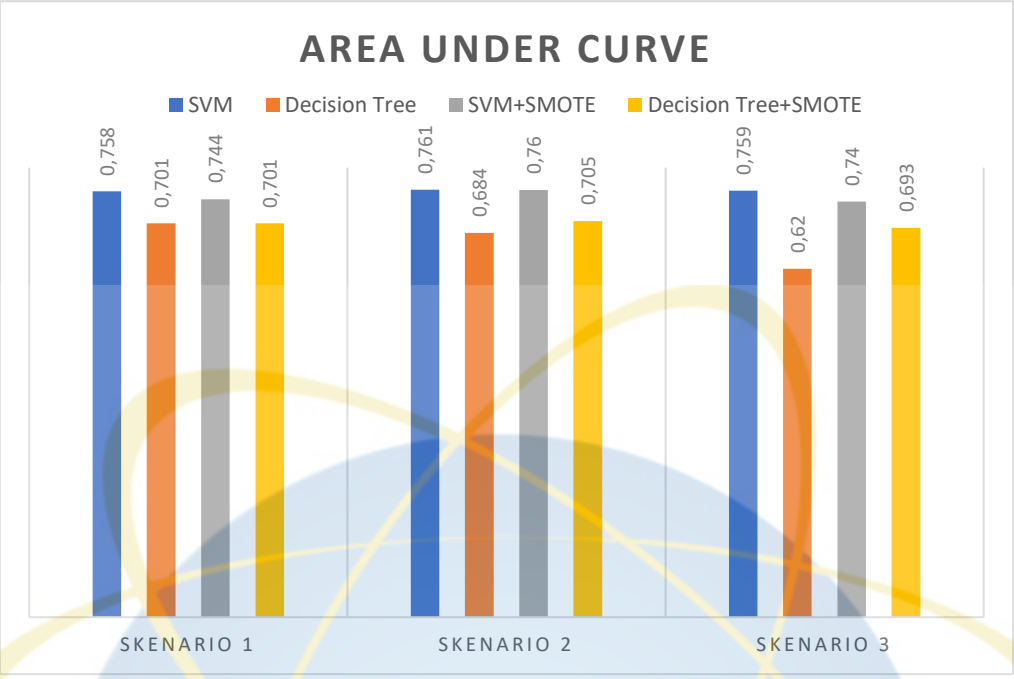
Berdasarkan hasil pengujian diatas menunjukkan hasil dari *recall* menggunakan algoritma SVM dan Decision Tree ditambah dengan penggabungan SMOTE pada algoritma tersebut. Dalam algoritma SVM nilai *recall* terbesar diperoleh dalam skenraio ke-3 dengan nilai 88,7%, sedangkan untuk Decision Tree nilai *recall* terbesar yang dihasilkan pada skenario 3 dengan nilai 93,79%. Selanjutnya dalam penggunaan SMOTE dalam proses

pengklasifikasian tersebut dalam SVM cenderung menurun dalam tiap skenario diatas dengan nilai tertinggi dalam skenario 2 sebesar 83,76, sedangkan pada Decision Tree pada skenario 2 dan 3 nilai *recall* menurun sedangkan skenario 1 nilai *recall* tetap. Tingkat *recall* ini menunjukkan bahwa, nilai dari rata-rata nilai *recall* tersebut model mampu mengklasifikasikan data uji dengan baik. *recall* berfungsi untuk mengetahui tingkat keberhasilan model dalam menemukan kembali sebuah informasi pada suatu data.



Gambar 11. 28 Grafik *F1-Score*

Berdasarkan hasil pengujian diatas menunjukkan hasil dari *f1-score* menggunakan algoritma SVM dan Decision Tree ditambah dengan penggabungan SMOTE pada algoritma tersebut. Dalam algoritma SVM nilai *f1-score* terbesar diperoleh dalam skenraio ke-3 dengan nilai 86,98%, sedangkan untuk *Decision Tree* nilai *f1-score* terbesar yang dihasilkan pada skenario 2 dengan nilai 85,26%. Selanjutnya dalam penggunaan SMOTE dalam proses pengklasifikasian tersebut dalam SVM cenderung menurun nilai *f1-score* nya, sedangkan pada *Decision Tree* dalam skenario 1 nilai *f1-score* meningkat 0,13%. dan dalam skeanrio 2 dan 3 nilai *f1-score* menurun. Tingkat *f1-score* ini menunjukkan, nilai dari rata-rata dari *precision* dan *recall* yang dibobotkan . dengan nilai *f1-score* tersebut menunjukan bahwa model mampu memiliki performa dengan baik dalam melakukan pengklasifikasian analisis sentimen.



Gambar 12. 29 Grafik *Area Under Curve*

Berdasarkan hasil pengujian diatas menunjukkan hasil dari *Area Under Curve* (AUC) menggunakan algoritma SVM dan *Decision Tree* ditambah dengan penggabungan SMOTE pada algoritma tersebut dapat disimpulkan bahwa dalam algoritma SVM nilai *AUC* terbesar diperoleh dalam skenario ke-2 dengan nilai 0,761 (Fair), sedangkan untuk Decision Tree nilai *f1-score* terbesar yang dihasilkan pada skenario 1 dengan nilai 0,701 (Fair). Dalam hal ini penggunaan SVM lebih baik dibandingkan Decision Tree. Selanjutnya dalam penggunaan SMOTE dalam proses pengklasifikasian tersebut dalam SVM cenderung menurun tingkat AUC nya dengan nilai terbesar pada skenario 2 dengan nilai 0,760 (Fair) hanya menurun 0,1% saja , sedangkan pada Decision Tree pada skenario 1 nilai AUC tetap dan skenario 2 dan 3 nilai *AUC* meningkat. Skenario 2 dengan nilai 0,705 (Fair), skenario 3 dengan nilai 0,693 (Pool). Tingkat nilai *AUC* ini menunjukkan bahwa nilai dari rata-rata dari *AUC* yang mempunyai score mendekati nilai 1.0 dapat dikatakan bahwa performa dari klasifikasi kualiatas pemodelan sangat baik dalam melakukan evaluasi performa.

BAB VI
PENUTUP

6.1 Kesimpulan

Berdasarkan kesimpulan dalam penelitian ini yang telah peneliti lakukan, diperoleh dalam menanalisis sentimen dengan Algoritma *Support Vector Machine* (SVM) dan *Decision Tree* dengan *Synthetic Minority Oversampling Technique* (SMOTE) berhasil diimplementasikan dalam melakukan pengklasifikasian sentimen terhadap Pengesahan RKUHP di *twitter* sebagai berikut :

1. Hasil performa Algoritma SVM dan Decision Tree menunjukan bahwa klasifikasi menggunakan SVM lebih unggul dibandingkan *Decision Tree*. Hal ini ditunjukkan dengan hasil *accuracy* sebesar 81,2%, *precision* 85,33% *recall* 88,7%, *f1-score* 86,98% dengan AUC 0,759 yang berarti tingkat akurat model dalam pengklasifikasian tergolong baik, serta *model quality* pada algoritma *Support Vector Machince* tergolong *Fair* (Cukup).
2. Dalam penggunaan SMOTE pada klasifikasi SVM dan Decision Tree terdapat perubahan sebelum dan sesudah penggunaan SMOTE dalam tingkat akurasi dan AUC diantara keduanya, dalam SVM mengalami penurunan dalam tingkat akurasi, sedangkan Decision Tree terdapat peningkatan beberapa persen dalam akurasi tersebut. Untuk nilai akurasi tertinggi dalam SVM dengan SMOTE didapatkan pada skenario 2 80% data latih dan 20% data uji dengan nilai akurasi sebesar 78,2%, sedangkan untuk Decision Tree dengan SMOTE terdapat peningkatan pada skenario 1 dan 3, Adapun nilai akurasi sebelum melakukan proses SMOTE pada skenario 1 sebesar 77,28% dan setelah proses SMOTE 77,6%. Namun dalam penggunaan SMOTE ini hasil klasifikasi SVM masih unggul dalam pengklasifikasian dalam analisis sentimen tersebut dengan tingkat akurasi serta *Area under curve* pada klasifikasi SVM tanpa SMOTE lebih besar dibanding dengan SMOTE dengan hasil 0,761, sedangkan *Decision Tree* dengan SMOTE lebih besar dengan nilai 0,705, yang berarti bahwa tingkat kualitas performa model tergolong *fair* dalam parameter *model quality* pada AUC.

6.2 Saran

Dari hasil penelitian ini, penulis menyadari bahwa masih terdapat banyak kekurangan yang tidak sesuai dengan harapan penulis. Oleh karena itu, ada beberapa saran yang perlu untuk diberikan kepada penelitian selanjutnya agar mendapatkan hasil yang lebih maksimal dan lebih baik kedepannya yaitu :

1. Melakukan perbandingan metode SMOTE yang lain seperti BorderlineSMOTE-SVM, SMOTE-NC dan Borderline SMOTE.
2. Melakukan perbandingan terkait jenis jenis metode dalam *Decision Tree* CART seperti ID3, C.45, C.50, CHAID, dan MARS.
3. Melakukan perbandingan terkait jenis kernel *trick* pada metode *Support Vector Machine* seperti *Polynomial, Radial Basic Function,* dan *Sigmoid.*
4. Mennggunakan metode Algoritma klasifikasi yang lain seperti *Unsupervised Learning* dan membandingkan dengan teknik *undersampling.*
5. Mengikutsertakan *tweet* berupa emoji untuk mendukung penelitian yang lebih akurat seperti marah, senang, sedih, dan lain-lain.

DAFTAR PUSTAKA

Abdulrohim, U., & Kom, S. (n.d.). *Natural Language Processing*.

Akmal Iftikar, M. (n.d.). *Analisis Sentimen Twitter: Penanganan Pandemi Covid-19 Menggunakan Metode Hybrid Naïve Bayes, Decision Tree, dan Support Vector Machine*.

Arsi, P., Hidayati, L. N., & Nurhakim, A. (2022). Komparasi Model Klasifikasi Sentimen Issue Vaksin Covid-19 Berbasis Platform Instagram. *JURNAL MEDIA INFORMATIKA BUDIDARMA*, *6*(1), 459. https://doi.org/10.30865/mib.v6i1.3509

Asshiddiqi, M. F., & Lhaksmana, K. M. (n.d.). *Perbandingan Metode Decision Tree dan Support Vector Machine untuk Analisis Sentimen pada Instagram Mengenai Kinerja PSSI*.

Aziz, A. (2022). Analisis Sentimen Identifikasi Opini Terhadap Produk, Layanan dan Kebijakan Perusahaan Menggunakan Algoritma TF-IDF dan SentiStrength. In *Jurnal Sains Komputer & Informatika (J-SAKTI* (Vol. 6, Issue 1). Abdul Aziz.

Azizah, H., Rintyarna, B. S., & Cahyanto, T. A. (2022). Sentimen Analisis Untuk Mengukur Kepercayaan Masyarakat Terhadap Pengadaan Vaksin Covid-19 Berbasis Bernoulli Naive Bayes. *BIOS : Jurnal Teknologi Informasi Dan Rekayasa Komputer*, *3*(1), 23–29. https://doi.org/10.37148/bios.v3i1.36

Bhatia, P. (2019). *Data Mining and Data Warehousing*. www.cambridge.org

Cahyaningtyas, C., Nataliani, Y., & Widiasari, I. R. (2021a). Analisis sentimen pada rating aplikasi Shopee menggunakan metode Decision Tree berbasis SMOTE. *AITI: Jurnal Teknologi Informasi*, *18*(Agustus), 173–184.

Choesni Herlingga, A., Putra, I., Prismana, E., Prehanto, D. R., & Dermawan, D. A. (2020). Algoritma Stemming Nazief & Adriani Dengan Metode Cosine Similarity Untuk Chatbot Telegram Terintegrasi Dengan E-layanan. *Journal of Informatics and Computer Science*, *02*.

Faisal, F., & Rustamaji, M. (2021). Pembaruan Pilar Hukum Pidana Dalam RUU KUHP. *Jurnal Magister Hukum Udayana (Udayana Master Law Journal)*, *10*(2), 291. https://doi.org/10.24843/jmhu.2021.v10.i02.p08

Faisal, M. R., & Kartini, D. (2020). *DNA Sequence Classification View project IT Asset Management View project*. https://www.researchgate.net/publication/359619425

Fitrah, F. A. (2021). Perbandingan Hukum terkait Pembentukan Pasal Penghinaan terhadap Peradilan, Perzinahan, dan Santet dalam RKUHP Indonesia. *SIGn Jurnal Hukum*, *2*(2), 122–137. https://doi.org/10.37276/sjh.v2i2.93

Ginantra, N. L. W. S. R., Yanti, C. P., Prasetya, G. D., Sarasvananda, I. B. G., & Wiguna, I. K. A. G. (2022). Analisis Sentimen Ulasan Villa di Ubud Menggunakan Metode Naive Bayes, Decision Tree, dan K-NN. *Jurnal Nasional Pendidikan Teknik Informatika (JANAPATI)*, *11*(3), 205–215. https://doi.org/10.23887/janapati.v11i3.49450

Hafidz, N., & Yanti Liliana, D. (2021). Klasifikasi Sentimen pada Twitter Terhadap WHO Terkait Covid-19 Menggunakan SVM, N-Gram, PSO. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, *5*(2), 213–219. https://doi.org/10.29207/resti.v5i2.2960

Hairani, H., Saputro, K. E., & Fadli, S. (2020). K-means-SMOTE for handling class imbalance in the classification of diabetes with C4.5, SVM, and naive Bayes. *Jurnal Teknologi Dan Sistem Komputer*, *8*(2), 89–93. https://doi.org/10.14710/jtsiskom.8.2.2020.89-93

Hidayatullah, M. R., & Warih Maharani. (2022). Depression Detection on Twitter Social Media Using Decision Tree. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, *6*(4), 677–683. https://doi.org/10.29207/resti.v6i4.4275

Hutami, W. P., Wijayanto, H., & Sulvianti, I. D. (2022). Penerapan Support Vector Machine dengan SMOTE Untuk Klasifikasi Sentimen Pemberitaan Omnibus Law Pada Situs CNNIndonesia.com. *Xplore: Journal of Statistics*, *11*(1), 26–35. https://doi.org/10.29244/xplore.v11i1.852

Ihda Aulia Rahmah, S. H. (2022). *Pro Kontra Pengesahan RKUHP*. PERSEKUTUAN PERDATA DONI BUDIONO & REKAN. https://pdb-lawfirm.id/pro-kontra-pengesahan-rkuhp/

Informatika, S., & Polinema, A. (n.d.-a). IMPLEMENTASI ANALISIS SENTIMEN TWITTER MENGENAI OPINI MASYARAKAT TERHADAP RKUHP TAHUN 2019. *SIAP)*, 2020.

Iskandar, J. W., & Nataliani, Y. (2021). Perbandingan Naïve Bayes, SVM, dan k-NN untuk Analisis Sentimen Gadget Berbasis Aspek. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, *5*(6), 1120–1126. https://doi.org/10.29207/resti.v5i6.3588

Keputusan Dirjen Penguatan Riset dan Pengembangan Ristek Dikti, S., Nikmatul Kasanah, A., Pujianto, U., Elektro, T., Teknik, F., & Negeri Malang, U. (2017). Terakreditasi SINTA Peringkat 2 Penerapan Teknik SMOTE untuk Mengatasi Imbalance Class dalam Klasifikasi Objektivitas Berita Online Menggunakan Algoritma KNN. *Masa Berlaku Mulai*, *1*(3), 196–201.

Kevin, V., Que, S., Analisis, :, Transportasi, S., Iriani, A., & Purnomo, H. D. (2020). Analisis Sentimen Transportasi Online Menggunakan Support Vector Machine Berbasis Particle Swarm Optimization (Online Transportation Sentiment Analysis Using Support Vector Machine Based on Particle Swarm Optimization). In *Jurnal Nasional Teknik Elektro dan Teknologi Informasi* | (Vol. 9, Issue 2). www.tripadvisor.com,

Kulkarni, A., & Shivananda, A. (2019). Natural Language Processing Recipes. In *Natural Language Processing Recipes*. Apress. https://doi.org/10.1007/978-1-4842-4267-4

Liu, B. (2019). *Sentiment Analysis and Opinion Mining*. Morgan & Claypool Publishers.

Morgan Peters. (2020). *Data Analysis From Scratch With Python_ Beginner Guide using Python, Pandas, NumPy, Scikit-Learn, IPython, TensorFlow and Matplotlib (PDFDrive)*.

Pattiiha, F. S., & Hendry, H. (2022). Perbandingan Metode K-NN, Naïve Bayes, Decision Tree untuk Analisis Sentimen Tweet Twitter Terkait Opini Terhadap PT PAL Indonesia. *JURIKOM (Jurnal Riset Komputer)*, *9*(2), 506. https://doi.org/10.30865/jurikom.v9i2.4016

Puspita, R., & Widodo, A. (2021). Perbandingan Metode KNN, Decision Tree, dan Naïve Bayes Terhadap Analisis Sentimen Pengguna Layanan BPJS. *Jurnal Informatika Universitas Pamulang*, *5*(4), 646. https://doi.org/10.32493/informatika.v5i4.7622

Putri, M. I., & Kharisudin, I. (2022). Penerapan Synthetic Minority Oversampling Technique (SMOTE) Terhadap Analisis Sentimen Data Review Pengguna Aplikasi Marketplace

Tokopedia. *PRISMA, Prosiding Seminar Nasional Matematika*, *5*, 759–766. https://journal.unnes.ac.id/sju/index.php/prisma/

Putu, I., Wirayasa, M., Made, I., Wirawan, A., Pradnyana, A., Kunci, K., Stemming, :, Bali, B., & Bastal, A. (n.d.). *ALGORITMA BASTAL: ADAPTASI ALGORITMA NAZIEF & ADRIANI UNTUK STEMMING TEKS BAHASA BALI* (Vol. 8).

Rahman Isnain, A., Indra Sakti, A., Alita, D., & Satya Marga, N. (2021). SENTIMEN ANALISIS PUBLIK TERHADAP KEBIJAKAN LOCKDOWN PEMERINTAH JAKARTA MENGGUNAKAN ALGORITMA SVM. *JDMSI*, *2*(1), 31–37. https://t.co/NfhnfMjtXw

Rahma Yustihan, S., & Pandu Adikara, P. (2021). *Analisis Sentimen berbasis Aspek terhadap Data Ulasan Rumah Makan menggunakan Metode Support Vector Machine (SVM)* (Vol. 5, Issue 3). http://j-ptiik.ub.ac.id

Ramasamy, L. K., Kadry, S., Nam, Y., & Meqdad, M. N. (2021). Performance analysis of sentiments in Twitter dataset using SVM models. *International Journal of Electrical and Computer Engineering*, *11*(3), 2275–2284. https://doi.org/10.11591/ijece.v11i3.pp2275-2284

Rezalina, O. (n.d.). *PERBANDINGAN ALGORITMA STEMMING NAZIEF & ADRIANI, PORTER DAN ARIFIN SETIONO UNTUK DOKUMEN TEKS BAHASA INDONESIA*.

Rizaty, M. A. (2022). *Pengguna Twitter di Indonesia Capai 18,45 Juta pada 2022*. Dataindonesia.Id. https://dataindonesia.id/digital/detail/pengguna-twitter-di-indonesia-capai-1845-juta-pada-2022

Ryanto, S. A., Richasdy, D., & Astuti, W. (2022). Partner Sentiment Analysis for Telkom University on Twitter Social Media Using Decision Tree (CART) Algorithm. *JURNAL MEDIA INFORMATIKA BUDIDARMA*, *6*(4), 1940. https://doi.org/10.30865/mib.v6i4.4533

Sari, F. V., & Wibowo, A. (2019). ANALISIS SENTIMEN PELANGGAN TOKO ONLINE JD.ID MENGGUNAKAN METODE NAÏVE BAYES CLASSIFIER BERBASIS KONVERSI IKON EMOSI. *Jurnal SIMETRIS*, *10*(2).

Sari, R. (2020). Analisis Sentimen Pada Review Objek Wisata Dunia Fantasi menggunakan Algoritma K-Nearest Neighbor (K-NN). *Jurnal Sains Dan Manajemen*, *8*(1). www.tripadvisor.com.

Susanto, E. B., Paminto Agung Christianto, Mohammad Reza Maulana, & Sattriedi Wahyu Binabar. (2022). Analisis Kinerja Algoritma Naïve Bayes Pada Dataset Sentimen Masyarakat Aplikasi NEWSAKPOLE Samsat Jawa Tengah. *Jurnal CoSciTech (Computer Science and Information Technology)*, *3*(3), 234–241. https://doi.org/10.37859/coscitech.v3i3.4343

Susanto, V. Y. (2022). *Penuh Pro Kontra, RUU KUHP Resmi Disahkan DPR*. Business Insight. https://insight.kontan.co.id/news/penuh-pro-kontra-ruu-kuhp-resmi-disahkan-dpr

Wati, R., & Ernawati, S. (2021). *Analisis Sentimen Persepsi Publik Mengenai PPKM Pada Twitter Berbasis SVM Menggunakan Python*. https://netlytic.org

Zenvita, D., Utami, A., & Judul, : (2021). ANALISIS SENTIMEN OPINI MASYARAKAT PADA MEDIA SOSIAL TWITTER TERHADAP DUKUNGAN PENGESAHAN RUU PENGHAPUSAN KEKERASAN SEKSUAL (PKS) MENGGUNAKAN ALGORITMA SUPPORT VECTOR MACHINE (SVM) DAN NAÏVE BAYES CLASSIFIER (NBC). In *Jurnal Teknik Informatika* (Vol. 13, Issue 1).

Adhi Putra, A. D. (2021). Analisis Sentimen pada Ulasan pengguna Aplikasi Bibit Dan Bareksa dengan Algoritma KNN. *JATISI (Jurnal Teknik Informatika Dan Sistem Informasi)*, *8*(2), 636–646. https://doi.org/10.35957/jatisi.v8i2.962

Andreyestha, A., & Azizah, Q. N. (2022). Analisa Sentimen Kicauan Twitter Tokopedia Dengan Optimalisasi Data Tidak Seimbang Menggunakan Algoritma SMOTE. *Infotek : Jurnal Informatika Dan Teknologi*, *5*(1), 108–116. https://doi.org/10.29408/jit.v5i1.4581

Arsi, P., Hidayati, L. N., & Nurhakim, A. (2022). Komparasi Model Klasifikasi Sentimen Issue Vaksin Covid-19 Berbasis Platform Instagram. *JURNAL MEDIA INFORMATIKA BUDIDARMA*, *6*(1), 459. https://doi.org/10.30865/mib.v6i1.3509

Arsi, P., Wahyudi, R., & Waluyo, R. (2021). Optimasi SVM Berbasis PSO pada Analisis

Sentimen Wacana Pindah Ibu Kota Indonesia. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, *5*(2), 231–237. https://doi.org/10.29207/resti.v5i2.2698

Artikel, R., Luh, N., Chandra, P., Rahman, R. A., Venyutzky, R., & Rakhmawati, N. A. (2021). *Analisis Klasifikasi Sentimen Terhadap Sekolah Daring pada Twitter Menggunakan Supervised Machine Learning*. *7*(April), 47–58.

Asshiddiqi, M. F., & Lhaksmana, K. M. (2020). *Perbandingan Metode Decision Tree dan Support Vector Machine untuk Analisis Sentimen pada Instagram Mengenai Kinerja PSSI*.

Fazrin, F., Nurul Prastiwi, O., & Andeswari, R. (2022). *Perbandingan Algoritma K-Nearest Neighbor dan Logistic Regression pada Analisis Sentimen terhadap Vaksinasi Covid-19 pada Media Sosial Twitter*. *10*(2), 1596–1604.

Gunawan, B., Sasty, H., #2, P., Esyudha, E., & #3, P. (2018). *JEPIN (Jurnal Edukasi dan Penelitian Informatika) Sistem Analisis Sentimen pada Ulasan Produk Menggunakan Metode Naive Bayes*. *4*(2), 17–29. www.femaledaily.com

Hendriyana, Karo, I. M. K., & Dewi, S. (2022). *ANALISIS PERBANDINGAN ALGORITMA SUPPORT VECTOR MACHINE, NAIVE BAYES, DAN REGRESI LOGISTIK UNTUK MEMPREDIKSI DONOR DARAH*. *8*(2), 121–126.

Imaduddin, A. H. (2022). *Sejarah Panjang Pengesahan RKUHP Lebih dari 5 Dekade*. Tempo.Co. https://nasional.tempo.co/read/1668881/sejarah-panjang-pengesahan-rkuhp-lebih-dari-5-dekade

Imron, A. (2019). *ANALISIS SENTIMEN TERHADAP TEMPAT WISATA DI KABUPATEN REMBANG MENGGUNAKAN METODE NAIVE BAYES CLASSIFIER*.

Isa. (2022). *Media Asing Soroti RI Bakal Pidana Pasangan Kumpul Kebo di RKUHP Baru*. CNNIndonesia. https://www.cnnindonesia.com/internasional/20221206080852-106-883330/media-asing-soroti-ri-bakal-pidana-pasangan-kumpul-kebo-di-rkuhp-baru

Keputusan Dirjen Penguatan Riset dan Pengembangan Ristek Dikti, S., Nikmatul Kasanah, A., Pujianto, U., Elektro, T., Teknik, F., & Negeri Malang, U. (2017). Terakreditasi

SINTA Peringkat 2 Penerapan Teknik SMOTE untuk Mengatasi Imbalance Class dalam Klasifikasi Objektivitas Berita Online Menggunakan Algoritma KNN. *Masa Berlaku Mulai*, *1*(3), 196–201.

Liang, S. (2021). Comparative Analysis of SVM, XGBoost and Neural Network on Hate Speech Classification. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, *5*(5), 896–903. https://doi.org/10.29207/resti.v5i5.3506

Liu, B. (2019). *Sentiment Analysis and Opinion Mining*. Morgan & Claypool Publishers.

Maula Chamzah, S., Lestandy, M., Kasan, N., Nugraha, A., Raya Tlogomas No, J., & Timur, J. (2022). Penerapan Synthetic Minority Oversampling Technique (SMOTE) untuk Imbalance Class pada Data Text Menggunakan KNN. In *Syntax: Jurnal Informatika* (Vol. 11, Issue 02).

Maziida, S. R. (2018). Klasifikasi Penyakit Diabetes Mellitus Dengan Menggunakan Perbandingan Algoritma J48 dan Random Forest (Studi Kasus : Rumah Sakit Muhammadiyah Lamongan). *Journal of Chemical Information and Modeling*, *53*(9), 1689–1699.

Pamuji, F. Y., Dwi, S., & Putri, A. (2021). *Komparasi Metode SMOTE dan ADASYN Untuk Penanganan Data Tidak Seimbang MultiClass*. 331–338.

Permana, A. P., Ainiyah, K., & Holle, K. F. H. (2021). Analisis Perbandingan Algoritma Decision Tree, kNN, dan Naive Bayes untuk Prediksi Kesuksesan Start-up. *JISKA (Jurnal Informatika Sunan Kalijaga)*, *6*(3), 178–188. https://doi.org/10.14421/jiska.2021.6.3.178-188

Putri, M. I., & Kharisudin, I. (2022). Penerapan Synthetic Minority Oversampling Technique (SMOTE) Terhadap Analisis Sentimen Data Review Pengguna Aplikasi Marketplace Tokopedia. *PRISMA, Prosiding Seminar Nasional Matematika*, *5*, 759–766. https://journal.unnes.ac.id/sju/index.php/prisma/

Ramadhanti, D. V., Santoso, R., & Widiharih, T. (2023). Perbandingan Smote Dan Adasyn Pada Data Imbalance Untuk Klasifikasi Rumah Tangga Miskin Di Kabupaten Temanggung Dengan Algoritma K-Nearest Neighbor. *Jurnal Gaussian*, *11*(4), 499–505.

https://doi.org/10.14710/j.gauss.11.4.499-505

Ramadhon, M. I. (2020). *Analisis Sentimen Terhadap Pemindahan Ibu Kota Indonesia Pada Media Sosial Twitter Menggunakan Metode Algoritma K-Nearest Neighbor (K-Nn)*.

Rizkia, S., Budi, E., Si, S. S., S, D. P. S., & Pd, M. (2019). *Analisis Sentimen Kepuasan Pelanggan Terhadap Internet Provider Indihome di Twitter Menggunakan Metode Decision Tree dan Pembobotan TF-IDF*. *6*(2), 9683–9693.

Ruben, B., & Sumigar, F. (2021). *Pelanggaran Berat HAM dalam RUU KUHP: Tinjauan dari Hukum Internasional Gross Violations of Human Rights in the Criminal Code Bill: an Overview from International Law*. www.komnasham.go.id/index.php/news/2019/9/5/1133/

Saputro, E., & Rosiyadi, D. (2022). Penerapan Metode Random Over-Under Sampling Pada Algoritma Klasifikasi Penentuan Penyakit Diabetes. *Bianglala Informatika*, *10*(1), 42–47. https://doi.org/10.31294/bi.v10i1.11739

Sir, Y. A., & Soepranoto, A. H. H. (2022). Pendekatan Resampling Data Untuk Menangani Masalah Ketidakseimbangan Kelas. *Jurnal Komputer Dan Informatika*, *10*(1), 31–38. https://doi.org/10.35508/jicon.v10i1.6554

Solia, R. A., Magriasti, L., Negeri, U., Sebagai Bagian, P., Politik, K., & Kebijakan, M. (2022). *Rima Arfa Solia, Lince Magriasti| Partisipasi Politik Mahasiswa Universitas Negeri Padang sebagai Bagian dari Kekuatan Politik dalam Mempengaruhi Kebijakan RKUHP Jurnal Mahasiswa Ilmu Administrasi Publik (JMIAP) PARTISIPASI POLITIK MAHASISWA UNIVERSITAS NE*. *2*(4), 10–19.

Trivusi. (2022a). *Penjelasan Lengkap Algoritma Support Vector Machine (SVM)*. https://www.trivusi.web.id/2022/04/algoritma-svm.html

Trivusi. (2022b). *Yuk Kenali Apa itu Algoritma K-Nearest Neighbors (KNN)*. https://www.trivusi.web.id/2022/06/algoritma-knn.html

Utami, E., & Dwi Hartanto, A. (n.d.). *ANALISIS SENTIMEN MASYARAKAT TERHADAP PELAKSANAAN P3K GURU DENGAN ALGORITMA NAIVE BAYES DAN DECISION*

TREE (THE ANALYSIS OF COMMUNITY SENTIMENT ON THE IMPLEMENTATION OF GOVERMENT EMPLOYEES WITH WORK AGREEMENT (P3K) TEACHERS WITH NAIVE BAYES ALGORITHM AND DECISION TREE).

Wati, R., & Ernawati, S. (2021). *Analisis Sentimen Persepsi Publik Mengenai PPKM Pada Twitter Berbasis SVM Menggunakan Python*. https://netlytic.org

WIJAYANTI, N. P. Y. T., N. KENCANA, E., & SUMARJAYA, I. W. (2021). Smote: Potensi Dan Kekurangannya Pada Survei. *E-Jurnal Matematika*, *10*(4), 235. https://doi.org/10.24843/mtk.2021.v10.i04.p348