

**ANALISIS SENTIMEN PENGGUNA YOUTUBE TENTANG ROHINGYA
MENGUNAKAN ALGORITMA SVM (*SUPPORT VECTOR MACHINE*)**

SKRIPSI



Oleh:

HIDAYAT

2020503014

PROGRAM STUDI TEKNOLOGI INFORMASI

FAKULTAS SAINS DAN TEKNOLOGI UNIVERSITAS IBRAHMY

SITUBONDO

2024



**ANALISIS SENTIMEN PENGGUNA YOUTUBE TENTANG ROHINGYA
MENGUNAKAN ALGORITMA SVM (*SUPPORT VECTOR MACHINE*)**

SKRIPSI

Diajukan untuk Memenuhi Salah Satu Persyaratan dalam Menyelesaikan Program
Sarjana (S-1) pada Program Studi Teknologi Informasi Fakultas Sains dan
Teknologi Universitas Ibrahimy



Oleh :

HIDAYAT

2020503014

**PROGRAM STUDI TEKNOLOGI INFORMASI
FAKULTAS SAINS DAN TEKNOLOGI UNIVERSITAS IBRAHIMY
SITUBONDO**

2024

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama : **Hidayat**

NPM : 2020503014

Program Studi : S-1 Teknologi Informasi

Fakultas : Sains dan Teknologi

Menyatakan dengan sungguh-sungguh bahwa skripsi ini sepenuhnya merupakan hasil penelitian atau karya saya sendiri, kecuali pada bagian-bagian yang dirujuk sebagai referensi dan disebutkan dalam daftar pustaka. Jika di kemudian hari terbukti bahwa tugas akhir atau skripsi ini merupakan hasil jiplakan atau plagiasi, saya siap menerima konsekuensi atas perbuatan tersebut.

Situbondo, 17 Agustus 2024

Saya yang menyatakan,



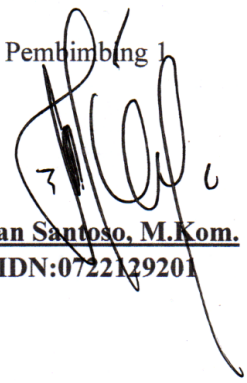
Hidayat

PERSETUJUAN PEMBIMBING SKRIPSI

Nama : **Hidayat**
NPM : 2020503014
Judul : **Analisis Sentimen Pengguna Youtube Menggunakan Algoritma SVM (Support Vector Machine)**

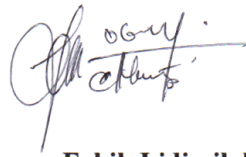
Telah disetujui oleh:

Pembimbing 1



Firman Santoso, M.Kom.
NIDN:0722129201

Pembimbing 2



Lukman Fakhri Lidimilah, M.Kom.
NIDN:0715099001

HALAMAN PENGESAHAN

SKRIPSI

ANALISIS SENTIMEN PENGGUNA YOUTUBE TENTANG ROHINGYA
MENGUNAKAN ALGORITMA SVM (SUPPORT VECTOR MACHINE)

HIDAYAT
2020503014

telah dipertahankan didepan dewan penguji Sidang/Munaqosyah Skripsi pada hari
Rabu, Tanggal 08 Agustus 2024 sebagai salah satu syarat memperoleh gelar
Sarjana (S.Kom) pada Fakultas Sanis dan Teknologi Uneversitas Ibrahimi.

Tim Penguji,

Ketua Sidang,

Abdus Samad, M.Kom.
NIDN: 0709099006

Sekretaris Sidang,

A. Hamdani, M.Kom.
NIDN: 0730118806

Penguji I,

Abd. Ghofur M.Kom.
NIDN: 0711088303

Penguji II,

Achmad Baijuri, M.Kom.
NIDN: 071209860

Mengetahui
Dekan,

Abd. Ghofur M.Kom.
NIDN: 0711088303

MOTTO

إِنَّ مَعَ الْعُسْرِ يُسْرًا

“sesungguhnya sesudah kesulitan itu ada kemudahan.” (Q.S Al-Insyirah : 6)

خَيْرُ النَّاسِ أَنْفَعُهُمْ لِلنَّاسِ

“sebaik-baik manusia adalah orang yang paling bermanfaat bagi manusia.”
(HR. Thabrani Dalam Kitab Al-Ausath no.5787)



HALAMAN PERSEMBAHAN

Puji syukur yang mendalam kami panjatkan ke hadirat Allah Subhanahu Wa Ta'ala, yang dengan limpahan rahmat dan karunia-Nya yang tak terhingga, telah memberi kami kekuatan dan petunjuk untuk menyelesaikan penelitian ini dengan penuh kesuksesan. Keberhasilan yang kami capai ini tidak lepas dari bimbingan dan anugerah-Nya yang mulia. Dengan rasa syukur dan rendah hati, kami persembahkan hasil kerja ini kepada:

1. Orang tua saya. Bapak Abdul Kadir dan Ibu Rosnah yang selalu mendukung saya serta selalu memotivasi dan mendo'akan saya. Terima kasih atas segalanya.
2. Kakak saya tercinta beserta istrinya. Yudi Saputra dan Rodiana Ningsih yang selalu mendukung dan mendo'akan adik tercintanya sampai kapanpun.
3. Keluarga besar saya yang selalu mendukung saya hingga saat ini dan sampai kapanpun.
4. Rektor, dekan, para dosen dan seluruh civitas akademika yang tidak bisa kami sebutkan satu persatu namanya, terima kasih semua bantuannya.
5. Teman seperjuangan Mahasiswa Fakultas Sain dan Teknologi yang terus menerus berusaha membantu dalam pengembangan cara berpikir dan berargumentasi kami.
6. Keluarga besar Asrama Sunan Bonang Nomor 36 yang senantiasa memberi semangat agar penulisan skripsi ini lekas selesai.
7. Seluruh keluarga rayon IKSASS Kalimantan. Terima kasih telah mengajarkan arti pengabdian yang sesungguhnya
8. Keluarga besar KESPA yang selalu menyemangati dan memotivasi, terima kasih atas segalanya. Canda tawa kalian sangat berharga.

KATA PENGANTAR

Segala puji hanya layak diberikan kepada Tuhan Yang Maha Esa. Dengan Rahmat-Nya, peneliti berhasil menyelesaikan penyusunan skripsi ini tanpa kendala yang signifikan. Sholawat serta salam selalu tercurah limpahkan kepada Baginda Nabi Muhammad SAW, sebagai contoh sempurna yang telah diciptakan oleh Allah dan menjadi teladan dalam kata dan tindakannya. Semoga kita semua mendapatkan pertolongan beliau di hari akhirat.

Dalam upaya penyelesaian skripsi ini, peneliti telah menerima banyak bantuan dan bimbingan dari berbagai pihak. Oleh karena itu peneliti mengucapkan terima kasih kepada:

1. KHR. Ach. Azaim Ibrahimi selaku Pengasuh Pondok Pesantren Salafiyah Syafi'iyah Sukorejo Situbondo.
2. KHR. Ach. Fadlail, S.H, M.H selaku Rektor Universitas Ibrahimi Situbondo.
3. Bapak Abd Ghofur, M.Kom selaku Dekan Fakultas Sains dan Teknologi Universitas Ibrahimi Situbondo.
4. Bapak Firman Santoso, M,kom selaku Ka. Prodi Teknologi Informasi Universitas Ibrahimi Situbondo dan dosen pembimbing I yang telah memberikan arahan dan bimbingan.
5. Bapak Lukman Fakhri Lidimilah, M.Kom, selaku dosen pembimbing II yang telah memberikan arahan dan bimbingan.

Semoga amal baik yang telah diberikan kepada peneliti mendapatkan balasan yang sebaik mungkin dari Allah SWT.

Situbondo, 17 Agustus 2024

Peneliti

Hidayat



ABSTRAK

Hidayat. 2024. Analisis Sentimen Pengguna YouTube Menggunakan Algoritma SVM (*Support Vector Machine*). Skripsi, Program Studi Teknologi Informasi, Universitas Ibrahimi. Pembimbing: (I) Firman Santoso, M.Kom., (II) Lukman Fakhri Lidimilah, M.Kom.

Pengungsi Rohingya yang berdatangan ke Indonesia telah menjadi isu kontroversial yang menimbulkan beragam reaksi dari masyarakat umum. Isu ini memunculkan spektrum respons yang luas, dari dukungan hingga penolakan, yang mencerminkan berbagai persepsi dan gagasan publik mengenai situasi yang terjadi. Untuk memperoleh pemahaman yang mendalam mengenai fluktuasi respons masyarakat terhadap kedatangan pengungsi Rohingya, analisis sentimen menjadi sangat penting. Melalui analisis sentimen, kita dapat mengukur dan memahami bagaimana masyarakat merespons isu ini secara emosional dan kognitif. Pemilihan metode analisis sentimen yang tepat adalah kunci untuk mendapatkan hasil yang akurat dan terpercaya. Penelitian ini bertujuan untuk melakukan analisis sentimen terhadap keterikatan etnis Rohingya dengan menerapkan metode support vector machine (SVM). Metode SVM dipilih karena kemampuannya yang efektif dalam klasifikasi teks dan analisis sentimen, yang memungkinkan pemisahan data dengan margin maksimum untuk menghasilkan prediksi yang lebih baik. Tujuan utama dari penelitian ini adalah untuk mengevaluasi dan mengimplementasikan metode SVM dalam konteks analisis sentimen terkait kedatangan pengungsi Rohingya. Untuk itu, kami menggunakan dataset yang terdiri dari 2000 komentar YouTube yang berkaitan dengan isu ini. Data tersebut dibagi ke dalam tiga skenario pelatihan dan pengujian dengan rasio yang berbeda: 70:30, 80:20, dan 90:10. Pembagian data ini bertujuan untuk menilai dampak dari berbagai proporsi data pelatihan dan pengujian terhadap kinerja model. Hasil penelitian menunjukkan bahwa metode SVM menghasilkan nilai akurasi yang sangat tinggi, yakni 98% pada ketiga rasio data yang telah ditentukan. Temuan ini mengindikasikan bahwa metode SVM sangat efektif dalam menganalisis sentimen terhadap isu pengungsi Rohingya dan memberikan hasil yang konsisten dengan tingkat akurasi yang sangat memuaskan.

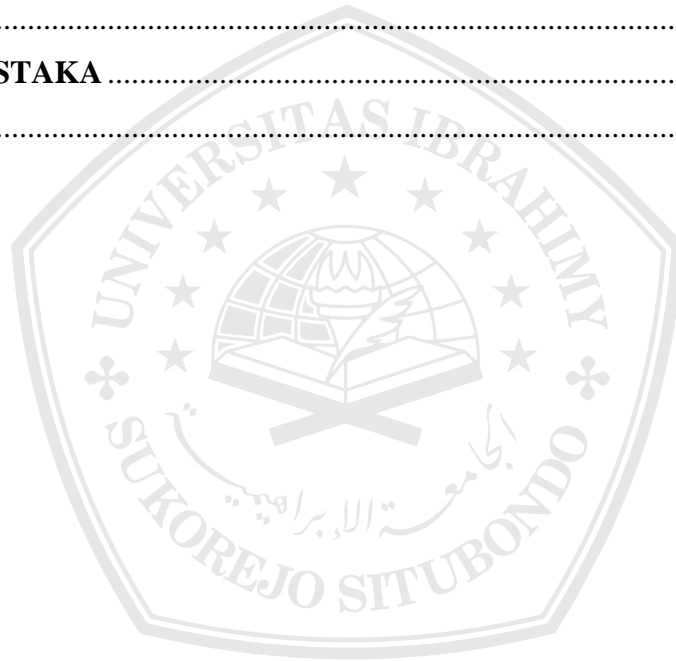
Kata kunci: Analisis Sentimen, SVM, YouTube, Rohingya

DAFTAR ISI

ANALISIS SENTIMEN PENGGUNA YOUTUBE TENTANG ROHINGYA MENGGUNAKAN ALGORITMA SVM (SUPPORT VECTOR MACHINE)..	i
PERNYATAAN KEASLIAN TULISAN	ii
PERSETUJUAN PEMBIMBING SKRIPSI.....	iii
HALAMAN PENGESAHAN	iv
MOTTO.....	v
HALAMAN PERSEMBAHAN.....	vi
KATA PENGANTAR	vii
ABSTRAK.....	viii
DAFTAR ISI.....	ix
DAFTAR TABEL.....	xii
DAFTAR GAMBAR	xiii
DAFTAR SEGMENT.....	xiv
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Identifikasi Masalah	3
1.3 Tujuan Penelitian	4
1.4 Rumusan Masalah	4
1.5 Batasan Masalah.....	4
1.6 Manfaat Penelitian	5
1.7 Sistematika Pembahasan	5
BAB II TINJAUAN PUSTAKA.....	7
2.1. Penelitian Terdahulu	7
2.2. Landasan Teori	10
2.2.1 Text mining	10
2.2.2 Analisis.....	11
2.2.3 Sentimen	11
2.2.4 Analisis Sentimen.....	11
2.2.5 Komentari.....	12
2.2.6 YouTube	12

2.2.7	<i>Support Vector Machine (SVM)</i>	12
2.2.8	<i>TF-IDF (Term Frequency — Inverse Document Frequency)</i>	13
2.3.	Perangkat Lunak	15
2.3.1	Google Chrome	15
2.3.2	Visual Studio Code	15
BAB III METODE PENELITIAN		16
3.1	Desain Penelitian	16
3.1.1.	Pendekatan Penelitian	16
3.1.2.	Jenis Penelitian	17
3.2	Teknik Pengumpulan Data	17
3.2.1.	Sumber Data	17
3.2.2.	Studi Pustaka	18
3.2.3.	Alat dan Bahan	18
3.2.4.	Crawling Data	19
3.3	Preprocessing Data	19
3.3.1.	Case folding	20
3.3.2.	Cleaning	21
3.3.3.	Tokenizing	21
3.3.4.	Stopword	21
3.3.5.	Stemming	22
3.4	Pelabelan	22
3.5	Pembobotan dan Klasifikasi	23
3.5.1.	Pembobotan	23
3.5.2.	Klasifikasi	23
BAB IV HASIL DAN PEMBAHASAN		24
4.1	Deskripsi Data	24
4.2	Proses Preprocessing Data	26
4.2.1.	Case Folding	28
4.2.2.	Cleaning	30
4.2.3.	Tokenizing	33
4.2.4.	Stopword	38

4.2.5. <i>Stemming</i>	43
4.3 Proses Pelabelan	48
4.4 Pembobotan TF-IDF dan Klasifikasi SVM.....	53
4.5 Evaluasi Model.....	57
4.5.1. Akurasi.....	57
4.5.2. Precision, Recall, dan F1-Score	58
4.5.3. Macro Average dan Weighted Average.....	58
BAB V KESIMPULAN DAN SARAN	59
5.1 Kesimpulan.....	59
5.2 Saran.....	60
DAFTAR PUSTAKA	61
LAMPIRAN.....	64



DAFTAR TABEL

Tabel 4. 1 Data Yang Dirapikan.....	27
Tabel 4. 2 Sampel Hasil Case Folding	29
Tabel 4. 3 Sampel Hasil Cleaning.....	32
Tabel 4. 4 Sampel Hasil Tokenizing	35
Tabel 4. 5 Sampel Hasil Stopword.....	40
Tabel 4. 6 Sampel Hasil Stemming.....	45
Tabel 4. 7 Sampel Hasil Pelabelan.....	50
Tabel 4. 8 Hasil Klasifikasi Ratio 90:10	57
Tabel 4. 9 Hasil Klasifikasi Ratio 80:20	57
Tabel 4. 10 Hasil Klasifikasi Ratio 70:30	57



DAFTAR GAMBAR

Gambar 3. 1 Tahapan Preprocessing Data	20
Gambar 4. 1 Sampel Hasil Crawling Data	25
Gambar 4. 2 Install NLTK	27



DAFTAR SEGMENT

Segment 4. 1 Crawling Data.....	24
Segment 4. 2 Case Folding.....	28
Segment 4. 3 Cleaning.....	31
Segment 4. 4 Tokenizing.....	33
Segment 4. 5 Stopword	38
Segment 4. 6 Stemming.....	43
Segment 4. 7 Pelabelan	48
Segment 4. 8 Pembobotan dan Klasifikasi.....	54



BAB I

PENDAHULUAN

1.1 Latar Belakang

Etnis Rohingya tinggal di wilayah Rakhine di Myanmar, yang berada di sisi barat laut dekat dengan Bangladesh. Penolakan kewarganegaraan oleh pemerintah Myanmar membuat etnis Rohingya terpuruk. Akibatnya, warga Rohingya ditindas, dipenjara, dan dibunuh. Situasi ini memaksa mereka untuk mengungsi ke negara-negara sekitar. Akibatnya, ribuan Rohingya terpaksa mengungsi ke negara-negara tetangga untuk mencari keselamatan [1].

Pengusiran etnis Rohingya terjadi pada tahun 2012 akibat konflik dengan penganut agama Buddha (Budhisme) [2]. Konflik ini memperburuk situasi bagi komunitas Rohingya yang sudah rentan. Kekerasan dan ketegangan meningkat, menyebabkan banyak Rohingya kehilangan tempat tinggal mereka. Pemerintah Myanmar tidak memberikan perlindungan, malah membiarkan atau bahkan mendukung pengusiran ini. Dengan keadaan ini etnis Rohingya terpaksa mengungsi dan tujuan utama dalam pengungsian ini adalah negara tetangga yakni Bangladesh, Thailand, Malaysia, dan Indonesia.

Masalah pengungsi Rohingya yang menjadikan Indonesia sebagai negara transit telah dibahas secara luas. Perdebatan di antara masyarakat Indonesia mengenai tindakan yang harus diambil terkait kedatangan pengungsi Rohingya semakin memanas. Ada yang mendukung penerimaan pengungsi dengan alasan kemanusiaan. Namun, ada pula yang khawatir tentang dampak sosial dan ekonomi

dari kedatangan mereka. Diskusi ini mencerminkan kompleksitas isu pengungsi Rohingya di Indonesia [3].

Media yang berfungsi sebagai penyebar informasi tentang konflik tersebut sangat berpengaruh terhadap persepsi publik, opini masyarakat, dan bahkan agenda politik [4]. Salah satu media yang populer dalam penyebaran video ataupun web video sharing adalah platform youtube, penggunaan platform youtube ini telah mencapai 93,8 % dari pengguna internet di Indonesia [5].

Oleh karena itu, analisis sentimen sangat berperan penting untuk memahami perspektif Masyarakat [6]. Analisis ini membantu memahami respon positif atau negatif dalam data teks yang diproses [7]. Dalam penelitian kali ini, algoritma *SVM (Support Vector Machine)* merupakan algoritma yang sangat cocok untuk digunakan. Algoritma *SVM (Support Vector Machine)* dipilih karena memiliki akurasi yang tinggi. Dengan demikian, analisis sentimen menggunakan *SVM (Support Vector Machine)* dapat memberikan gambaran yang lebih jelas tentang pandangan masyarakat.

Seperti penelitian yang telah dilakukan, perbandingan algoritma *SVM (Support Vector Machine)* dan algoritma *Naïve Bayes* menunjukkan bahwa algoritma *SVM (Support Vector Machine)* memiliki akurasi 99,50%, sedangkan algoritma *Naïve Bayes* menghasilkan akurasi 99,25% [8]. Penelitian lain juga membandingkan algoritma *SVM (Support Vector Machine)* dengan algoritma *Decision Tree*. Hasilnya, algoritma *SVM (Support Vector Machine)* memiliki akurasi sebesar 93,7%. Sementara itu, algoritma *Decision Tree* menghasilkan akurasi sebesar 83%

[9]. Hal ini menunjukkan keunggulan algoritma *SVM (Support Vector Machine)* dalam berbagai konteks analisis.

Dari beberapa berita pengungsi Rohingya yang tersebar, penulis mengumpulkan data komentar dari beberapa video di YouTube. Data tersebut diambil dari lima kanal berita, yaitu tvOneNews, KOMPASTV, Liputan6, METRO TV, dan CNN Indonesia. Setiap kanal memiliki banyak video terkait berita pengungsi Rohingya. Komentar-komentar ini memberikan wawasan tentang pandangan masyarakat. Dengan mengumpulkan data dari berbagai sumber, analisis dapat dilakukan secara lebih komprehensif.

Dari masing-masing kanal tersebut, tentunya banyak komentar netizen yang berbeda, baik itu berkomentar positif ataupun negatif. Supaya data teks komentar video YouTube dari beberapa kanal tersebut menjadi informasi yang mudah dipahami, penulis akan melakukan penelitian yang berjudul "Analisis Sentimen Pengguna YouTube tentang Rohingya Menggunakan *SVM (Support Vector Machine)*". Penelitian ini bertujuan untuk mengevaluasi serta mengimplementasikan metode *SVM (Support Vector Machine)* terhadap analisis sentimen pengguna YouTube tentang Rohingya. Dengan demikian, diharapkan hasil penelitian ini dapat memberikan wawasan yang lebih mendalam tentang pandangan dan perasaan pengguna YouTube terhadap isu Rohingya.

1.2 Identifikasi Masalah

Berdasarkan konteks di atas, ada beberapa permasalahan yang dapat diidentifikasi. Berikut permasalahan yang teridentifikasi dalam penelitian ini, yaitu:

- a. Sulitnya pengklasifikasian data komentar kedalam 2 bagian yaitu: Positif dan negatif tanpa bantuan sistem.
- b. Penerapan algoritma *SVM (Support Vector Machine)* sebagai kalsifikasi dalam analisis sentimen.

1.3 Tujuan Penelitian

Berdasarkan rumusan masalah diatas maka dapat ditemukan solusi atau jawabanya, adapun tujuan penelitian adalah sebagai berikut:

- a. Penerapan metode *SVM (Support Vector Machine)* dalam analisis sentimen komentar youtube.
- b. Mengetahui tingkat akurasi klasifikasi dalam penerapan metode *SVM (Support Vector Machine)*

1.4 Rumusan Masalah

Berdasarkan uraian diatas baik dari latar belakang atau pada identifikasi masalah, maka rumusan masalahnya adalah sebagai berikut:

- a. Bagaimana penerapan metode *SVM (Support Vector Machine)* pada proses klasifikasi komentar youtube?
- b. Bagaimana tingkat akurasi yang di peroleh dalam penerapan metode *SVM (Support Vector Machine)* pada klasifikasi komentar youtube?

1.5 Batasan Masalah

Agar tidak terjadi penyimpangan dalam proses penelitian ini maka akan ditetapkan Batasan masalah, berikut merupakan Batasan masalah dalam penelitian ini:

- a. Data komentar di kelompokkan kedalam bagian yaitu positif dan negative.
- b. Didalam penklasifikasian hanya menggunakan metode *SVM (Support Vector Machine)*
- c. Data komentar yang diambil dari 5 chanel YouTube, yakni chanel tvOneNews, KOMPASTV, Liputan6, METRO TV, dan CNN Indonesia.

1.6 Manfaat Penelitian

Manfaat penelitian merupakan hasil yang dapat dirasakan oleh semua pihak yang terkait didalam proses penelitian. Berikut adalah manfaat penelitian:

- a. Bagi penulis

Manfaat bagi penulis yakni memberikan pengalaman dan menambah wawasan tentang text mining dalam analisis sentiment komentar youtube.

- b. Bagi pengguna youtube dan youtuber atau konten creator

bagi para viewer memberikan informasi dalam menilai sebuah konten berdasarkan ekstraksi data komentar youtube. Dan bagi konten creator membantu untuk dapat memudahkan dalam menyimpulkan ribuan komentar. Dan dapat dilihat kualitas dari konten yang dibuat.

1.7 Sistematika Pembahasan

Penyusunan sistematika penulisan dilakukan dengan maksud mempermudah dan memperjelas tujuan dari bab yang akan dibahas yaitu:

BAB 1 : PENDAHULUAN

Pada bab ini menjelaskan tentang latar belakang, identifikasi masalah, tujuan penelitian, rumusan masalah, batasan masalah, manfaat penelitian, dan sistematika pembahasan.

BAB 2 : TINJAUAN PUSTAKA

Pada bab ini dibahas tentang tinjauan penelitian terdahulu serta definisi-definisi yang berkaitan dengan permasalahan yang sedang dibahas.

BAB 3 : METODE PENELITIAN

Pada bab ini menjelaskan tentang desain penelitian yang digunakan, teknik pengumpulan data, preprocessing data, preprocessing data, pelabelan dan pembobotan serta kalsifikasi.

BAB 4 : HASIL DAN PEMBAHASAN

Pada bab ini menjelaskan tentang spesifikasi hasil pembahsan yang diperoleh dari penelitian ini.

BAB 5 : KESIMPULAN DAN SARAN

Pada bab ini membahas tentang kesimpulan dan saran dari hasil pembahsan yang telah dibuat.

BAB II

TINJAUAN PUSTAKA

2.1. Penelitian Terdahulu

“Analisis Sentimen Komentar YouTube TvOne Tentang Ustadz Abdul Somad Dideportasi Dari Singapura Menggunakan Algoritma SVM”

Penelitian ini penulis meneliti tentang analisis sentiment tentang Ustadz Abdul Somad di deportasi dari singapura melalui channel Youtube tvOne news. Channel Youtube tvOne news sendiri memiliki jumlah pengikut sebanyak 9,91 juta subscriber per tanggal 1 Desember 2022 sejak bergabung pada tanggal 4 desember 2014 dan total viewers videonya berjumlah 6,132,720,633 di Youtube. dengan jumlah tersebut menggambarkan bahwa channel ini menduduki peringkat ke 8 dengan subscribe terbanyak dari 12 stasiun televisi swasta di Indonesia. Banyak mendapatkan respon dari masyarakat yang menontonnya menjadikan channel ini sebagai salah satu channel terpopuler di indonesia tentunya dengan tingkat antusiasme yang tinggi, membuat channel ini mendapatkan begitu banyak respon dari masyarakat. Hal ini akan berguna bagi Ustadz Abdul Somad untuk mendapatkan informasi yang ada berupa ulasan komentar chanel Youtube tvOne untuk mendapatkan feedback berupa komentar positif maupun negatif dari masyarakat secara langsung tentang permasalahan yang terjadi terhadapnya. Hal ini tentunya akan menyulitkan jika harus menganalisis setiap komentar secara manual, karena komentar berisikan ratusan hingga ribuan tanggapan dari setiap penonton di video tersebut. Komentar yang diberikan oleh seseorang tidak selalu dapat

dimengerti oleh orang lain. Maka diperlukan bidang teknologi computer yang bertujuan untuk memahami makna teks tersebut [10].

“ANALISIS SENTIMEN KOMENTAR PADA GOOGLE PLAY STORE, TWITTER DAN YOUTUBE TERHADAP APLIKASI MYPERTAMINA DENGAN SUPPORT VECTOR MACHINE”

Pada penelitian ini penulis mengawali dengan menceritakan Indonesia merupakan nomor lima pengguna kendaraan paling banyak di dunia. Yang mana hal itu menyebabkan besarnya permintaan bahan bakar minyak (BBM) di Indonesia. Uji coba pembatasan pembelian BBM bersubsidi telah dilakukan oleh pemerintah dengan menerapkan MyPertamina untuk mengontrol pembelian BBM bersubsidi mulai juli 2022, agar dapat meringankan beban keuangan negara. Pada September 2019 diluncurkan terdapat 179.854 konsumen aktif, kemudian meningkat signifikan pada tahun 2021 sebanyak 5.975.478 namun terjadi penurunan pada bulan Januari-April 2022 sebanyak 4.923.805 konsumen aktif [6]. konsumen memberikan ulasan tentang aplikasi MyPertamina pada kolom komentar Google Play Store, Youtube maupun Twitter. Ulasan tersebut dikemas dalam bentuk komentar positif, dan negatif. Komentar yang didapat pada google play store sebanyak 18.000, twitter 20.000 dan youtub 6.400. Dari data komentar ini dapat dianalisis menggunakan text mining [11].

“Analisis Sentimen Komentar Youtube Mengenai Vaksin Covid-19 Menggunakan Support Vector Machine”

Pada penelitian ini penulis menjelaskan tentang penggunaan youtube yang tidak hanya menjadi platform bagi para *content creator* namun juga sebagai tempat

bagi pengguna lain saling bertukar pikiran melalui kolom komentar. Selain fitur komentar, Youtube juga menyediakan fitur rekomendasi yang membantu pengguna untuk menemukan video yang sesuai dengan minat pengguna. Salah satu video yang banyak dibicarakan saat ini yaitu berita mengenai perkembangan vaksinasi Covid-19 yang ada di Indonesia. Pada akhir bulan November 2021, Indonesia telah mencapai vaccine rate sebesar 34% dari seluruh penduduk di Indonesia. Jumlah tersebut menunjukkan kurang lebih sebanyak 95 juta penduduk dari 233 juta penduduk telah melakukan vaksinasi. Sebagai informasi, saat ini terdapat 9 jenis vaksin yang telah disetujui oleh BPOM diantaranya adalah Sinovac, Bio Farma, Astra Zeneca, Sinopharm, Moderna, Pfizer, Sputnik V, Janssen, dan Convidecia. Namun kebijakan yang diambil pemerintah tidak selalu dapat diterima oleh masyarakat, hal ini juga terjadi pada awal penerapan kebijakan vaksinasi ini. Tidak semua elemen masyarakat menyetujui atau mendukung berjalannya proses vaksinasi di Indonesia. Bahkan tidak sedikit yang berpendapat negatif mengenai vaksin Covid-19. Agar dapat lebih memahami sentimen masyarakat terhadap kebijakan vaksinasi Covid-19, perlu dilakukan analisis terhadap opini atau komentar yang banyak ditulis melalui komentar Youtube. Analisis sentimen pada komentar Youtube dilakukan dengan cara mengekstraksi, mengolah dan memahami data yang mulanya berupa teks yang tidak terstruktur secara otomatis. Analisis ini nantinya digunakan untuk penilaian pendapat atau opini orang terhadap topik tertentu baik itu positif atau negative [12].

2.2. Landasan Teori

2.2.1 *Text mining*

Text mining adalah cabang dari data mining yang berfokus pada penemuan pola atau hubungan tertentu dalam dokumen teks. Ini bertujuan untuk mencerminkan isi atau fitur khusus dari dokumen tersebut. Proses ini melibatkan ekstraksi informasi dari data teks dan sumber lainnya. Sumber data biasanya berasal dari dokumen. Tujuan utama *text mining* adalah untuk mengidentifikasi kata-kata yang menggambarkan isi dokumen tersebut [13].

Text mining menggunakan berbagai teknik dan algoritma untuk mengolah, mengeksplorasi, dan menganalisis teks. Teknik-teknik ini beragam dan mencakup banyak pendekatan berbeda. Beberapa teknik yang umum digunakan dalam *text mining* meliputi:

- a. *Preprocessing*: Tahap pra-pemrosesan yang melibatkan Langkah-langkah seperti menghapus tanda baca, mengubah teks menjadi huruf kecil, menghilangkan kata-kata umum (*stopword*), dan melakukan normalisasi seperti *Stemming*
- b. Pemberian bobot (*Weighting*): Proses ini melibatkan pemberian bobot pada kata-kata dalam teks. Bobot ini digunakan untuk menunjukkan seberapa pentingnya kata tersebut dalam dokumen. Salah satu metode yang sering digunakan adalah *TF-IDF*. *TF-IDF* merupakan singkatan dari *Term Frequency-Inverse Document Frequency*.
- c. Klasifikasi: Klasifikasi adalah proses penting dalam *text mining* yang bertujuan untuk mengelompokkan teks ke dalam kategori atau label

berdasarkan pola atau fitur yang ada. Proses ini memanfaatkan berbagai algoritma untuk mencapai tujuan klasifikasi yang akurat, yang penting untuk aplikasi seperti analisis sentimen dan pengelolaan informasi besar.

2.2.2 Analisis

Analisis adalah tahap dimana data-data dari komentar yang telah diproses dan diteliti secara detail dan sesuai dengan prosedur yang ditetapkan [14]. Kegiatan ini merupakan salah satu proses untuk mencari masalah dan data-data yang diperlukan didalam proses penelitian.

2.2.3 Sentimen

Merupakan suara masyarakat yang dituangkan kedalam komentar sebagai wujud untuk memberikan pendapat baik itu persetujuan atau penolakan terhadap konten youtube [15]. Sentiment ini merupakan sumber data yang akan diolah pada penelitian kali ini yang nantinya akan diproses dan akan melewati beberapa tahapan didalamnya.

2.2.4 Analisis Sentimen

Juga disebut penambangan opini, adalah bidang studi yang menganalisis opini, sentimen, evaluasi, penilaian, sikap, dan emosi orang. Bidang studi ini meneliti entitas seperti produk, layanan, organisasi, individu, masalah, peristiwa, topik, dan atribut mereka. Analisis sentimen bertujuan untuk memahami bagaimana orang merasakan atau berpikir tentang berbagai hal. Hal ini dapat diterapkan dalam berbagai konteks untuk mengukur persepsi publik. Secara keseluruhan, ini mewakili ruang masalah yang besar dan kompleks [16].

2.2.5 Komentar

Komentar merupakan elemen yang signifikan dalam mempengaruhi kepuasan pengguna [17]. Tanggapan atau ulasan dapat berupa pendapat, saran, atau ketidaksetujuan terhadap suatu hal. Dalam konteks penelitian ini, komentar dikaji secara khusus dalam hubungannya dengan respons terhadap konten video di platform YouTube.

2.2.6 YouTube

YouTube merupakan platform media sosial yang menawarkan beragam konten, termasuk hiburan, musik, dan berita terkini. Selain itu, YouTube juga dimanfaatkan secara luas dalam bidang pendidikan, terutama selama masa pandemi COVID-19 [18]. YouTube juga dimanfaatkan sebagai sumber berita yang menyediakan akses ke informasi terbaru dari berbagai belahan dunia. Platform ini tidak hanya menyajikan konten pendidikan, tetapi juga menjadi saluran utama bagi berbagai jenis berita dan liputan aktual.

2.2.7 *Support Vector Machine (SVM)*

Support Vector Machine (SVM) adalah pendekatan klasifikasi yang dikembangkan pada tahun 1990an dan semakin populer karena kinerjanya yang baik. SVM adalah generalisasi dari pengklasifikasi margin maksimal yang membutuhkan pemisahan kelas dengan batas linier. Ekstensi dari ini adalah pengklasifikasi vektor pendukung, yang lebih fleksibel dan dapat diterapkan pada lebih banyak kasus. SVM dimaksudkan untuk klasifikasi biner tetapi dapat diperluas untuk lebih dari dua kelas dan terkait erat dengan metode statistik seperti

regresi logistik biner, namun juga dapat diperluas untuk lebih dari dua kelas, dan memiliki hubungan erat dengan metode statistik lainnya seperti regresi logistik [19].

Latih model svm menggunakan fungsi optimasi dan prediksi sentiment berikut merupakan rumus fungsi optimasi dan prediksi sentiment:

a. Fungsi Optimasi

$$\min_{w,b} \frac{1}{2} \|w\|^2 \text{ dengan syarat } y_i(w \cdot x_i + b) \geq 1 \forall i$$

Dimana:

y_i adalah label kelas (+1 atau -1).

x_i adalah vektor fitur dari data ke-i.

b. Prediksi Sentimen

$$f(x) = w \cdot x + b$$

Di mana $f(x)$ adalah nilai yang digunakan untuk menentukan kelas sentimen. Jika $f(x) >$, maka data diklasifikasikan ke dalam satu kelas (misalnya, positif). Jika $f(x) <$, maka data diklasifikasikan ke dalam kelas lain (misalnya, negatif).

2.2.8 TF-IDF (Term Frequency — Inverse Document Frequency)

Salah satu fitur pembobotan yang paling umum dan paling sering digunakan adalah TF-IDF, yang sangat akurat dan mudah diingat. Nilai kontribusi TF-IDF meningkat dengan jumlah kata yang muncul dalam dokumen yang lebih sering, tetapi dengan jumlah kata yang lebih sedikit muncul dalam dokumen yang lebih

sering, nilai kontribusi tersebut berkurang. *Term Frekuensi (TF)* dan *Frekuensi Dokumen Inverse (IDF)* adalah komponen dari *TF-IDF* [20].

TF-IDF (Term Frequency-Inverse Document Frequency) adalah ukuran yang digunakan untuk menilai pentingnya sebuah term dalam sebuah dokumen relatif terhadap koleksi dokumen atau korpus. Ini adalah produk dari dua metrik: *Term Frequency (TF)* dan *Inverse Document Frequency (IDF)*.

a. Komponen Rumus *TF-IDF*

$TF(t_k, d_j)$: Term Frequency dari term t_k dalam dokumen d_j . Ini mengukur seberapa sering term t_k muncul dalam dokumen d_j .

$IDF(t_k)$ Inverse Document Frequency dari term t_k . Ini mengukur seberapa penting term tersebut secara keseluruhan dalam korpus dokumen.

b. Definisi Rumus

$$TF - IDF(t_k, d_j) = TF(t_k, d_j) \times IDF(t_k)$$

Penjelasan :

TF: Term Frequency memberikan nilai seberapa sering term t_k muncul dalam dokumen d_j . Semakin sering term muncul dalam dokumen, semakin tinggi nilai TF-nya.

IDF: Inverse Document Frequency mengurangi bobot term yang sering muncul di banyak dokumen dan meningkatkan bobot term yang jarang muncul. Ini membantu dalam mengidentifikasi term yang unik dan signifikan untuk setiap dokumen.

2.3. Perangkat Lunak

2.3.1 Google Chrome

Google Chrome adalah sebuah peramban web (web browser) yang dikembangkan oleh Google. Peramban ini terkenal karena kecepatan, kemudahan penggunaan, serta kemampuannya untuk mengintegrasikan berbagai layanan Google. Karena hal itu Google Chrome sangat populer dikalangan komunitas pengguna internet [21]. Google Chrome juga dikenal karena dukungannya terhadap standar web terbaru dan keamanan yang kuat. Peramban ini tersedia untuk berbagai sistem operasi seperti Windows, macOS, Linux, dan juga untuk perangkat mobile seperti Android dan iOS.

2.3.2 Visual Studio Code

Visual Studio Code (VS Code) adalah sebuah editor kode yang sangat populer, dikembangkan oleh Microsoft. VS Code dirancang untuk menjadi ringan, cepat, dan sangat dapat disesuaikan, sehingga cocok digunakan untuk pengembangan perangkat lunak dan pemrograman secara umum [22]. Beberapa fitur utama dari VS Code termasuk dukungan untuk banyak bahasa pemrograman, integrasi dengan Git untuk pengelolaan kode sumber, ekstensi yang dapat diunduh untuk memperluas fungsionalitas, serta kemampuan debugging yang kuat. VS Code tersedia untuk Windows, macOS, dan Linux.

BAB III

METODE PENELITIAN

3.1 Desain Penelitian

3.1.1. Pendekatan Penelitian

Dalam memahami jenis penelitian, penting bagi kita untuk mengenali peran masing-masing dari sebuah jenis penelitian. Pada penelitian ini, fokus utamanya terbagi pada dua metode utama: kuantitatif dan pencarian pustaka (*library search*). Metode kuantitatif melibatkan penggunaan data numerik untuk menguji hipotesis serta mengidentifikasi pola atau hubungan dalam fenomena yang diteliti [23]. Dengan pendekatan ini, peneliti dapat mengumpulkan data yang dapat diukur secara objektif untuk memvalidasi teori atau menjawab pertanyaan penelitian dengan cara yang sistematis dan terstruktur.

Di sisi lain, pencarian pustaka menjadi aspek penting dalam penelitian ini dengan memanfaatkan sumber literatur yang ada untuk mengumpulkan dan menyintesis informasi yang relevan terkait topik penelitian. Dengan cara ini, peneliti dapat memperluas wawasan tentang subjek yang dipelajari dengan memanfaatkan hasil-hasil penelitian sebelumnya dan sudut pandang yang berbeda dari para ahli. Pendekatan ini memungkinkan untuk mendukung argumen penelitian dengan landasan teori yang kuat dan memperkaya pemahaman tentang konteks lebih luas dari topik yang diteliti.

Kombinasi kedua pendekatan ini memberikan kesempatan bagi peneliti untuk menghasilkan analisis yang mendalam dan berbasis data tentang isu-isu yang

dihadapi dalam penelitian. Dengan menggunakan metode kuantitatif dan pencarian pustaka secara bersamaan, peneliti dapat menggabungkan kekuatan dari dua pendekatan yang berbeda untuk menyusun narasi yang komprehensif dan mendalam tentang fenomena yang sedang diteliti.

3.1.2. Jenis Penelitian

Jenis penelitian yang digunakan dalam skripsi ini adalah penelitian deskriptif. Penelitian deskriptif bertujuan untuk menggambarkan atau mendeskripsikan suatu fenomena, dalam hal ini adalah sentimen pengguna YouTube terhadap isu Rohingya. Penelitian deskriptif tidak mencari hubungan kausal antara variabel, tetapi lebih pada memberikan gambaran yang jelas tentang fenomena yang sedang diteliti. Tujuan Penelitian Deskriptif yakni:

- a. Mengidentifikasi dan menggambarkan sentimen pengguna YouTube terhadap isu Rohingya.
- b. Mengklasifikasikan sentimen tersebut ke dalam kategori positif atau negatif, menggunakan algoritma *SVM (Support Vector Machine)*.

3.2 Teknik Pengumpulan Data

3.2.1. Sumber Data

Data yang digunakan pada penelitian ini merupakan data komentar youtube yang diambil dengan bantuan API youtube melalui *google colab* yang kemudian akan di pisah menjadi sentimen positif dan negatif. Data komentar youtube tersebut diambil dari 5 chanel YouTube, yakni chanel tvOneNews, KOMPASTV, Liputan6, METRO TV dan CNN Indonesia.

3.2.2. Studi Pustaka

Studi pustaka adalah proses sistematis untuk mengumpulkan, mengevaluasi, dan mensintesis literatur yang relevan terkait dengan topik penelitian tertentu. Pendekatan ini melibatkan pencarian dan analisis terhadap sumber-sumber teks seperti buku, jurnal ilmiah, artikel, dan dokumen lainnya yang telah dipublikasikan sebelumnya. Tujuan dari studi pustaka adalah untuk memahami perkembangan pengetahuan yang ada, mengidentifikasi kerangka teoritis yang relevan, serta mengevaluasi pendekatan dan metodologi yang telah digunakan dalam penelitian sebelumnya terkait topik yang sedang diteliti.

3.2.3. Alat dan Bahan

a. Google Colab

Google Colab merupakan platform yang berfungsi untuk menulis dan mengeksekusi kode Python secara online dalam lingkungan berbasis cloud. Keunggulan menggunakan Google Colab termasuk kemampuan untuk menggunakan GPU secara gratis (dalam beberapa batasan) yang mempercepat pemrosesan data yang besar seperti crawling data komentar YouTube.

b. API YouTube

API YouTube adalah antarmuka pemrograman aplikasi (API) yang disediakan oleh Google untuk mengakses fitur-fitur dan data dari platform YouTube. Penggunaan API YouTube sangat berguna untuk layanan yang terhubung secara langsung dengan konten video dan interaksi pengguna di YouTube. API ini menyediakan cara yang terstruktur dan aman untuk

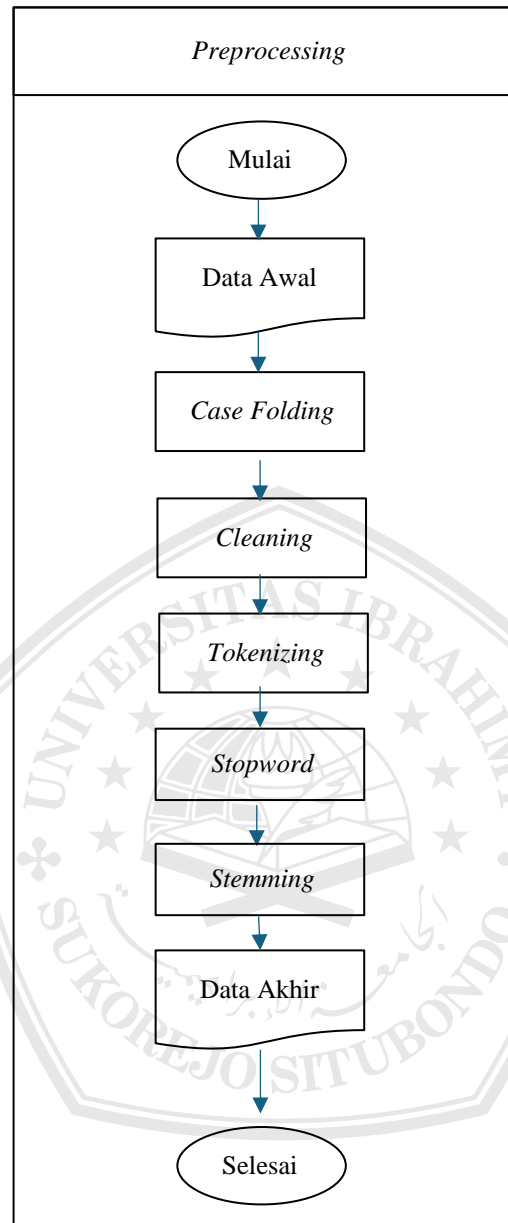
mengelola dan mengintegrasikan informasi dari platform YouTube. Dengan API ini dapat mengimplementasikan berbagai fitur, seperti mengambil komentar, statistik video, dan informasi pengguna, untuk meningkatkan fungsionalitas.

3.2.4. *Crawling Data*

Dalam penelitian ini, peneliti menggunakan API YouTube untuk mengumpulkan data komentar YouTube yang diperlukan melalui Google Colab. Bahasa pemrograman yang digunakan untuk proses ini adalah Python. Proses pengumpulan data ini disebut dengan istilah "*crawling*". Peneliti memanfaatkan sourcecode Python khusus untuk melakukan crawling data komentar YouTube

3.3 *Preprocessing Data*

Preprocessing adalah tahapan awal dalam analisis data di mana data mentah atau tidak terstruktur dipersiapkan dan dimodifikasi menjadi bentuk yang lebih terstruktur. Dalam konteks analisis teks, preprocessing melibatkan beberapa tahapan. Tahapan preprocessing bisa dilihat pada gambar 3.1.



Gambar 3. 1 Tahapan Preprocessing Data

3.3.1. *Case folding*

Penggunaan *case folding* adalah proses mengubah semua karakter menjadi huruf kecil. Metode ini berguna untuk mengstandarisasi teks dan memudahkan perbandingan. Dengan *case folding*, perbedaan huruf besar dan kecil diabaikan,

sehingga memungkinkan konsistensi dalam analisis teks. *Case folding* mempermudah proses pencarian dan manipulasi teks dalam berbagai aplikasi, mulai dari pemrosesan bahasa alami hingga pemrograman.

3.3.2. *Cleaning*

Pada taha *cleaning* teks dibersihkan dari tanda baca seperti delimiter, dan simbol-simbol lain yang tidak diperlukan. Langkah ini penting dalam pra-pemrosesan teks karena membantu menyederhanakan konten dan meningkatkan kualitas analisis. *Cleaning* juga membantu mengurangi noise atau gangguan dalam data teks sehingga memperoleh hasil yang lebih akurat dalam pemrosesan berikutnya.

3.3.3. *Tokenizing*

Tahapan *tokenizing* dilakukan untuk memecah kalimat menjadi beberapa kata yang terpisah. Proses ini penting karena memungkinkan analisis lebih lanjut pada tingkat kata-kata dan pemahaman yang lebih mendalam terhadap teks yang sedang diproses. Dengan memisahkan teks menjadi unit-unit kata, analisis seperti klasifikasi atau analisis sentimen dapat dilakukan secara lebih efektif. Tahapan *tokenizing* memungkinkan pemrosesan lebih lanjut dengan algoritma pemrosesan bahasa alami atau analisis teks lainnya.

3.3.4. *Stopword*

Tahap *stopword* berperan dalam menghilangkan kata-kata yang memiliki tingkat informasi yang rendah dari teks. Langkah ini esensial dalam pra-

pemrosesan teks karena membantu mengurangi noise dan meningkatkan relevansi informasi yang tersedia untuk analisis lebih lanjut.

3.3.5. *Stemming*

Tahap *stemming* memiliki tujuan untuk menghapus imbuhan pada awal atau akhir kata sehingga kata tersebut menjadi kata dasar. Proses ini memperoleh kata dasar dari variasi kata yang ada, membantu dalam mengurangi kompleksitas dan memperbaiki konsistensi dalam analisis teks. Dengan mengaplikasikan stemming, kemungkinan kemunculan kata yang serupa namun berbeda bentuknya dapat diperkecil, sehingga meningkatkan akurasi dan efisiensi dalam pemrosesan teks.

3.4 Pelabelan

Dalam konteks analisis sentimen, "pelabelan" mengacu pada proses atau tugas memberikan label atau kategori pada teks atau data berdasarkan sentimen atau emosi yang terkandung di dalamnya. Ini sering dilakukan untuk mengklasifikasikan apakah suatu teks atau komentar mengandung sentimen positif atau negatif.

Pada tahap pelabelan, peneliti menggunakan Visual Studio Code untuk otomatisasi proses ini dengan Python. Peneliti memanfaatkan kode untuk mengklasifikasikan sentimen teks secara akurat berdasarkan algoritma yang telah dikembangkan. Metode ini membantu mempermudah dan mempercepat analisis sentimen terhadap data besar secara efisien

3.5 Pembobotan dan Klasifikasi

3.5.1. Pembobotan

Pembobotan *TF-IDF* yang dilakukan menggunakan dataset yang telah melalui proses preprocessing dan pelabelan pada tahap sebelumnya. Tujuan dari pembobotan ini adalah menentukan nilai terhadap kata dalam suatu dokumen dan mengidentifikasi kata-kata kunci yang mempengaruhi dokumen tersebut. Dalam pembobotan *TF-IDF* peneliti melakukan dengan menggunakan bahasa pemrograman *python* dengan memanfaatkan visual studio code sebagai media pengaplikasiannya.

3.5.2. Klasifikasi

Pada tahap akhir penelitian ini, dilakukan klasifikasi sentimen menggunakan algoritma *SVM (Support Vector Machine)* dengan menggunakan bahasa pemrograman Python di lingkungan Visual Studio Code. SVM digunakan untuk memisahkan dan mengklasifikasikan data teks berdasarkan sentimennya, dengan langkah-langkah yang telah dipersiapkan sebelumnya seperti preprocessing dan pelabelan. Dalam penelitian ini klasifikasi dilakukan sebanyak tiga kali dengan ratio yang berbeda. Ratio pertama dilakukan dengan perbandingan 90:10 yakni 90% data testing dan 10% data training, yang kedua menggunakan perbandingan 80:20 yakni 80% data testing dan 20% data training dan yang ketiga menggunakan perbandingan 70:30 yakni 70% data testing dan 30% data training.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Deskripsi Data

Data yang digunakan dalam penelitian ini adalah komentar dari beberapa kanal YouTube berita Indonesia yang membahas isu Rohingya. Komentar-komentar ini dikumpulkan menggunakan API YouTube melalui Google Colab. Kanal-kanal yang dipilih antara lain tvOneNews, KOMPASTV, Liputan6, METRO TV, dan CNN Indonesia. Setiap masing-masing kanal youtube yang dipilih, komentar yang di kumpulkan sebanyak 400 komentar, dengan setiap video diambil 100 komentar. Jumlah data komentar keseluruhan sebanyak 2000 data komentar. Segmen 4.1 merupakan segmen program untuk crawling data komentar dan sampel hasil crawling bisa dilihat pada gambar 4.1.

Segmen 4. 1 *Crawling Data*

```
import googleapiclient.discovery
import pandas as pd

api_service_name = "youtube"
api_version = "v3"
DEVELOPER_KEY = "AIzaSyC01aAXdjYhqjhaU_w0QKs-18WQ80CPIk"

youtube = googleapiclient.discovery.build(
    api_service_name, api_version, developerKey=DEVELOPER_KEY)

request = youtube.commentThreads().list(
    part="snippet",
    videoId="3mtGKBVy2zw",
    maxResults=100
)
response = request.execute()
```

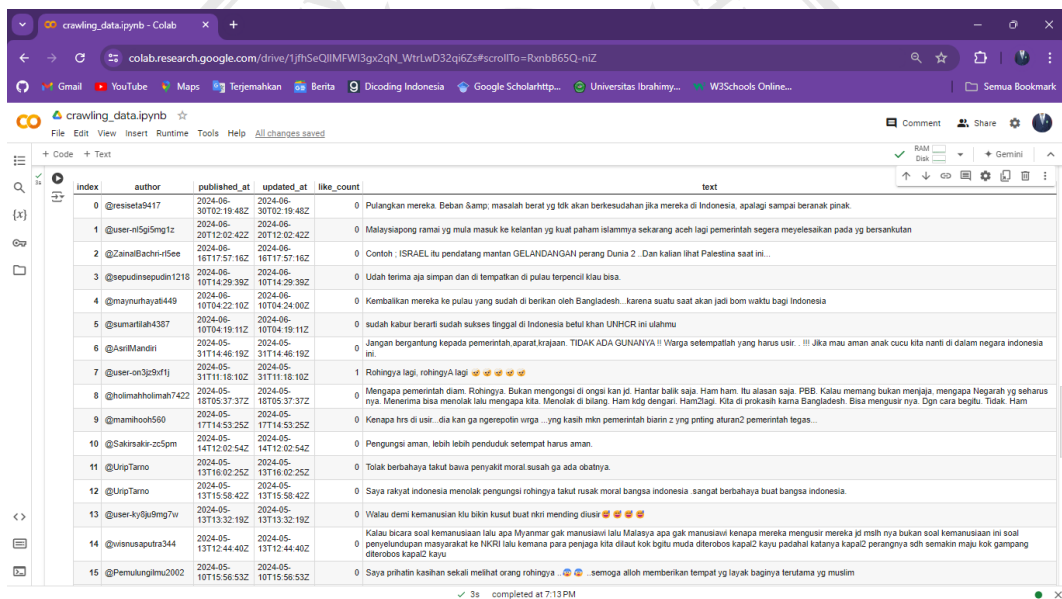
Segmen Lanjutan 4.1

```
comments = []

for item in response['items']:
    comment = item['snippet']['topLevelComment']['snippet']
    comments.append([
        comment['authorDisplayName'],
        comment['publishedAt'],
        comment['updatedAt'],
        comment['likeCount'],
        comment['textDisplay']
    ])

df = pd.DataFrame(comments, columns=['author', 'published_at',
'updated_at', 'like_count', 'text'])

df.head(100)
```



index	author	published_at	updated_at	like_count	text
0	@resiet9417	2024-09-30T02:19:48Z	2024-06-30T02:19:48Z	0	Pulangan mereka. Beban &mp; masalah berat yg tk akan berkesudahan jika mereka di Indonesia, apalagi sampai beranak-pinak
1	@user-ni5g5mg1z	2024-09-20T12:02:42Z	2024-06-20T12:02:42Z	0	Malaysiapong ramal yg mula masuk ke kelantan yg kuat paham islamnya sekarang acch lagi pemerintah segera meyelesaikan pada yg bersangkutan
2	@ZainalBachriH5ee	2024-09-16T15:57:16Z	2024-06-16T15:57:16Z	0	Contoh : ISRAEL itu pendatang mantan GELANDANGAN perang Dunia 2 . Dan kalian lihat Palestina saat ini ..
3	@sepuinsepuin1218	2024-09-10T14:29:39Z	2024-06-10T14:29:39Z	0	Utah terima aja sipan dan di tempatkan di pulau terpencil klu bisa.
4	@maynuyahab449	2024-09-10T04:22:10Z	2024-06-10T04:24:00Z	0	Kembalikan mereka ke pulau yang sudah di berikan oleh Bangladesh. karena suatu saat akan jadi bom waktu bagi Indonesia
5	@sumartiah4387	2024-09-10T04:19:11Z	2024-06-10T04:19:11Z	0	sudah kabar berarti sudah sukses tinggal di Indonesia beluk klan UMHCR ini ulahmu
6	@AsrilMandiri	2024-09-31T14:46:19Z	2024-06-31T14:46:19Z	0	Jangan bergantung kepada pemerintah, aparat,krajaan. TIDAK ADA GUNANYA !! Warga setempat yang harus usir. !! Jika mau aman anak cucu kita nanti di dalam negara Indonesia ini.
7	@user-on3j3wrf1j	2024-05-31T11:18:10Z	2024-05-31T11:18:10Z	1	Rohingya lagi rohingya lagi 🙄🙄🙄🙄🙄
8	@holimahholimah7422	2024-05-15T05:37:37Z	2024-05-15T05:37:37Z	0	Mengapa pemerintah dan. Rohingya. Bukan mengongki di ongi kan jd. Hantar balik saja. Ham ham. Itu akan saja. PBB. Kalau memang bukan menaja. mengapa Negaraah yg seharus nya. Menerima bisa menajak lalu mengapa kita. Menolak di bilang. Ham kdg dengat. HamTaqi. Kda di prokashh kama Bangladesh. Bisa mengusir nya. Dgn cara beghu. Tidak. Ham
9	@mamboh560	2024-05-17T14:53:25Z	2024-05-17T14:53:25Z	0	Kenapa hrs di usir...dia kan ga ngerepotin warga...yng kasih mkn pemerintah biarin z yng pnting aturan2 pemerintah legas...
10	@Sakirakir-cc5pm	2024-05-14T12:02:54Z	2024-05-14T12:02:54Z	0	Pengungsi aman, lebih lebih penduduk setempat harus aman.
11	@UripTano	2024-05-13T16:02:25Z	2024-05-13T16:02:25Z	0	Tolak berbahaya takut bawa penyakit moral susah ga ada obatnya.
12	@UripTano	2024-05-13T15:58:42Z	2024-05-13T15:58:42Z	0	Saya rakyat Indonesia menolak pengungsi rohingya takut rusak moral bangsa Indonesia. sangat berbahaya buat bangsa Indonesia.
13	@user-ky9jdm97w	2024-05-13T13:32:19Z	2024-05-13T13:32:19Z	0	Walaupun demi kemanusiaan klu bikin kusut buat niki mending diusir 🙄🙄🙄🙄
14	@vivanusaputra344	2024-05-13T12:44:40Z	2024-05-13T12:44:40Z	0	Kalau bicara soal kemanusiaan lalu apa Myanmar gak manusiawi lalu Malaysia apa gak manusiawi kenapa mereka mengusir mereka jd milih nya bukan soal kemanusiaan ini soal penyelundupan masyarakat ke NKRI lalu kenapa para penjaga kita dilaut kok bgtu muda diterobos kapal2 kayu padahal katanya kapal2 perangnya sdh semakin maju kok gampang diterobos kapal2 kayu
15	@Pemulangimu2002	2024-05-10T15:56:53Z	2024-05-10T15:56:53Z	0	Saya prihatin kashan sekali melihat orang rohingya. 🙄🙄...semoga Allah memberikan tempat yg layak baginya terutama yg muslim

Gambar 4. 1 Sampel Hasil *Crawling* Data

Segmen 4.1 merupakan kode python dengan tujuan untuk mengambil komentar dari video YouTube dan menyimpannya dalam DataFrame menggunakan pustaka pandas. Pertama, modul `googleapiclient.discovery` dan `pandas` diimpor untuk membangun klien API YouTube dan mengelola data, masing-masing. Variabel

`api_service_name`, `api_version`, dan `DEVELOPER_KEY` diatur untuk mengonfigurasi akses API YouTube.

Dengan objek klien yang dibangun, kode membuat permintaan untuk mengambil komentar dari video tertentu menggunakan metode `youtube.commentThreads().list`. Permintaan ini mencakup parameter untuk menentukan bagian informasi komentar yang diambil, ID video, dan batas jumlah komentar. Respons dari API disimpan dalam variabel `response`.

Data komentar kemudian dikumpulkan dari respons dan dimasukkan ke dalam `list comments`. Setiap komentar diambil dan disusun dengan informasi seperti nama penulis, tanggal publikasi, tanggal pembaruan, jumlah suka, dan teks komentar. Data ini akhirnya diubah menjadi `DataFrame` `pandas` dan hingga 100 baris pertama ditampilkan untuk analisis lebih lanjut.

4.2 Proses Preprocessing Data

Sebelum memulai proses preprocessing, pastikan untuk menginstal *NLTK* (*Natural Language Toolkit*) melalui terminal atau command prompt dengan perintah “`pip install nltk`” dan “`pip install nltk scikit-learn contractions`” untuk mendukung proses preprocessing. Untuk informasi lebih lanjut dalam menginstall NLTK, silakan lihat pada gambar 4.2. NLTK merupakan sebuah library Python yang populer digunakan untuk memproses dan menganalisis teks bahasa alami.

```
C:\WINDOWS\system32\cmd. X + -
Microsoft Windows [Version 10.0.22621.3880]
(c) Microsoft Corporation. All rights reserved.

C:\Users\HIDAYAT>pip install nltk
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: nltk in c:\users\hidayat\appdata\roaming\python\python312\site-packages (3.8.1)
Requirement already satisfied: click in c:\users\hidayat\appdata\roaming\python\python312\site-packages (from nltk) (8.1.7)
Requirement already satisfied: joblib in c:\users\hidayat\appdata\roaming\python\python312\site-packages (from nltk) (1.4.2)
Requirement already satisfied: regex<=2021.8.3 in c:\users\hidayat\appdata\roaming\python\python312\site-packages (from nltk) (2024.5.15)
Requirement already satisfied: tqdm in c:\users\hidayat\appdata\roaming\python\python312\site-packages (from nltk) (4.66.4)
Requirement already satisfied: colorama in c:\users\hidayat\appdata\roaming\python\python312\site-packages (from click->nltk) (0.4.6)

[notice] A new release of pip is available: 24.1.1 -> 24.1.2
[notice] To update, run: python.exe -m pip install --upgrade pip

C:\Users\HIDAYAT>pip install nltk scikit-learn contractions
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: nltk in c:\users\hidayat\appdata\roaming\python\python312\site-packages (3.8.1)
Requirement already satisfied: scikit-learn in c:\users\hidayat\appdata\roaming\python\python312\site-packages (1.5.0)
Requirement already satisfied: contractions in c:\users\hidayat\appdata\roaming\python\python312\site-packages (0.1.73)
Requirement already satisfied: click in c:\users\hidayat\appdata\roaming\python\python312\site-packages (from nltk) (8.1.7)
Requirement already satisfied: joblib in c:\users\hidayat\appdata\roaming\python\python312\site-packages (from nltk) (1.4.2)
Requirement already satisfied: regex<=2021.8.3 in c:\users\hidayat\appdata\roaming\python\python312\site-packages (from nltk) (2024.5.15)
Requirement already satisfied: tqdm in c:\users\hidayat\appdata\roaming\python\python312\site-packages (from nltk) (4.66.4)
Requirement already satisfied: numpy>=1.19.5 in c:\users\hidayat\appdata\roaming\python\python312\site-packages (from scikit-learn) (2.0.0)
Requirement already satisfied: scipy>=1.6.0 in c:\users\hidayat\appdata\roaming\python\python312\site-packages (from scikit-learn) (1.14.0)
Requirement already satisfied: threadpoolctl>=3.1.0 in c:\users\hidayat\appdata\roaming\python\python312\site-packages (from scikit-learn) (3.5.0)
Requirement already satisfied: textsearch>=0.0.21 in c:\users\hidayat\appdata\roaming\python\python312\site-packages (from contractions) (0.0.24)
Requirement already satisfied: anyascii in c:\users\hidayat\appdata\roaming\python\python312\site-packages (from textsearch>=0.0.21->contractions) (0.3.2)
Requirement already satisfied: pyahocorasick in c:\users\hidayat\appdata\roaming\python\python312\site-packages (from textsearch>=0.0.21->contractions) (2.1.0)
Requirement already satisfied: colorama in c:\users\hidayat\appdata\roaming\python\python312\site-packages (from click->nltk) (0.4.6)

[notice] A new release of pip is available: 24.1.1 -> 24.1.2
[notice] To update, run: python.exe -m pip install --upgrade pip
```

Gambar 4. 2 Install NLTK

Setelah menginstall NTLK, selanjutnya merapikan data mentah yang diambil pada tahap crawling sebelumnya dan menjadikan data mentah tersebut menjadi format json, dengan tujuan agar data mentah tersebut dapat di proses pada tahap preprocessing. Sampel data yang sudah dirapikan bisa dilihat pada tabel 4.1.

Tabel 4. 1 Data Yang Dirapikan

```
[
  {
    "text": "Jgn pikirin Rohingya org asli peribumi msh byk yg k
laparan k susah usirr tuh org Rohingya lama kaya org yaman tinggal d
Indonesia yaman tinggal d Indonesia semoga bp perabowo tegas"
  },
  {
    "text": "Pulangkan saja"
  },
  {
    "text": "Kalo dilamain ntar minta tanah"
  },
  {
  }
```

Tabel Lanjutan 4.1

```
"text": "SAYA UNSUBSCRIBE HOST TIDAK PAKAI KERUDUNG"  
  
  },  
  {  
    "text": "YANG BENER UMAT ISLAM DIRANGKUL  
DIBANTU. KALAU GA TAU TATA KRAMA DIAJARI ITULAH  
YANG NABI MUHAMMAD AJARKAN."  
  
  },  
  {  
    "text": "Biarkan sajah kaya di taelad mati kelaparan kan kita ga  
ngundang"  
  
  },  
  {  
    "text": "Pemerintah gak punya nyali untuk mengusir Rohingya dan  
nekan UNHCR yg jadi kaki tangan barat..."  
  
  },  
}
```

4.2.1. Case Folding

Case folding merupakan proses pertama dalam proses preprocessing. Data yang digunakan pada proses ini adalah data yang sudah dirapikan dan berformat json. Segmen 4.2 merupakan segmen program case folding dan sampel hasil case folding bisa dilihat pada table 4.2

Segmen 4. 2 Case Folding

```
import json  
  
# Load the JSON file  
file_path = 'HASIL/Data_Mentah.json'  
with open(file_path, 'r', encoding='utf-8') as file:  
    data = json.load(file)
```

Segmen Lanjutan 4.2

```
# Perform case folding
for comment in data:

    comment['text'] = comment['text'].lower()

# Save the modified data back to a new JSON file
output_path = 'HASIL/labeled_comments_case_folded.json'
with open(output_path, 'w', encoding='utf-8') as file:
    json.dump(data, file, ensure_ascii=False, indent=4)

print(f"Case folded data saved to {output_path}")
```

Tabel 4. 2 Sampel Hasil Case Folding

```
[
  {
    "text": "jgn pikirin rohingya org asli peribumi msh byk yg k
laparan k susah usirr tuh org rohingya lama kaya org yaman tinggal d
indonesia yaman tinggal d indonesia semoga bp perabowo tegas"
  },
  {
    "text": "pulangkan saja"
  },
  {
    "text": "kalo dilamain ntar minta tanah"
  },
  {
    "text": "saya unsubscribe host tidak pakai kerudung"
  },
  {
    "text": "yang bener umat islam dirangkul dibantu. kalau ga tau tata
krama diajari itulah yang nabi muhammad ajarkan."
  },
  {
    "text": "biarkan sajah kaya di taelad mati kelaparan kan kita ga
ngundang"
  },
  {
    "text": "pemerintah gak punya nyali untuk mengusir rohingya dan
nekan unhcr yg jadi kaki tangan barat..."
  },
]
```

Kode python pada segmen 4.2 ini digunakan untuk memproses data komentar dalam format JSON dengan melakukan *case folding* dan menyimpannya ke file JSON baru. Pertama, kode ini mengimpor modul `json` yang diperlukan untuk memuat dan memanipulasi data JSON. Kemudian, file JSON yang terletak di path `HASIL/Data_Mentah.json` dibuka dan isinya dimuat ke dalam variabel `data` menggunakan encoding UTF-8.

Selanjutnya, kode ini melakukan “case folding” pada teks komentar dalam data. Dengan menggunakan loop `for`, setiap komentar dalam list `data` diiterasi, dan teks komentar diubah menjadi huruf kecil dengan metode `.lower()`. Proses ini membantu menstandarkan teks komentar sehingga perbedaan antara huruf besar dan kecil tidak mempengaruhi analisis lebih lanjut.

Setelah memproses data, kode ini menyimpan hasil modifikasi ke file JSON baru dengan path `HASIL/labeled_comments_case_folded.json`. File baru dibuka dengan mode tulis dan encoding UTF-8, lalu data yang telah dimodifikasi disimpan menggunakan `json.dump`, dengan format indentasi untuk memudahkan pembacaan.

4.2.2. Cleaning

Setelah melakukan proses *case folding*, selanjutnya adalah proses *cleaning*. Data yang digunakan dalam tahapan *cleaning* merupakan data hasil akhir dari proses *case folding*. Segmen 4.3 merupakan segmen program.

Segmen 4. 3 Cleaning

```
import json
import re

# Function to clean the text
def clean_text(text):
    # Remove punctuation
    text = re.sub(r'[^\w\s]', '', text)
    # Remove extra spaces
    text = re.sub(r'\s+', ' ', text).strip()
    return text

# Load the JSON file
file_path = 'HASIL/labeled_comments_case_folded.json'
with open(file_path, 'r', encoding='utf-8') as file:
    data = json.load(file)

# Perform cleaning
for comment in data:
    comment['text'] = clean_text(comment['text'])

# Save the cleaned data back to a new JSON file
output_path = 'HASIL/labeled_comments_cleaned.json'
with open(output_path, 'w', encoding='utf-8') as file:
    json.dump(data, file, ensure_ascii=False, indent=4)

print(f"Cleaned data saved to {output_path}")
```

Kode pada segmen 4.3 ini dimulai dengan mengimpor modul json dan re, yang digunakan untuk memanipulasi data JSON dan melakukan operasi pencocokan pola pada teks, masing-masing. Fungsi clean_text didefinisikan untuk membersihkan teks komentar dengan menghapus tanda baca dan spasi berlebih. Fungsi ini pertama-tama menghilangkan semua tanda baca menggunakan ekspresi reguler, lalu menghapus spasi tambahan dan memastikan tidak ada spasi yang berlebih di awal atau akhir teks.

Selanjutnya, kode membuka file JSON yang berlokasi di 'HASIL/labeled_comments_case_folded.json' dan memuat isinya ke dalam variabel data dengan encoding UTF-8. Setelah data dimuat, fungsi `clean_text` diterapkan pada setiap teks komentar dalam data menggunakan loop `for`. Proses ini membersihkan teks dengan menghilangkan tanda baca dan spasi berlebih.

Setelah teks dibersihkan, data yang telah dimodifikasi disimpan ke file JSON baru di path 'HASIL/labeled_comments_cleaned.json'. File dibuka dengan mode tulis dan encoding UTF-8, lalu data disimpan dengan format JSON yang mudah dibaca menggunakan `json.dump`, dengan pengaturan `ensure_ascii=False` untuk karakter non-ASCII dan `indent=4` untuk indentasi yang rapi. Akhirnya, pesan konfirmasi ditampilkan untuk memberitahukan bahwa data yang telah dibersihkan telah disimpan dengan sukses. sampel hasil dari proses cleaning bisa dilihat pada table 4.3.

Tabel 4. 3 Sampel Hasil Cleaning

```
[
  {
    "text": "jgn pikirin rohingya org asli peribumi msh byk yg k laparan
k susah usirr tuh org rohingya lama kaya org yaman tinggal d indonesia
yaman tinggal d indonesia semoga bp perabowo tegas"
  },
  {
    "text": "pulangkan saja"
  },
  {
    "text": "kalo dilamain ntar minta tanah"
  },
  {
    "text": "saya unsubscribe host tidak pakai kerudung"
  },
  {
```

Tabel Lanjutan 4.3

```
"text": "yang bener umat islam dirangkul dibantu kalau ga tau tata  
krama diajari itulah yang nabi muhammad ajarkan"  
},  
{  
  "text": "biarkan sajah kaya di taelad mati kelaparan kan kita ga  
ngundang"  
},  
{  
  "text": "pemerintah gak punya nyali untuk mengusir rohingya dan  
nekan unhcr yg jadi kaki tangan barat"  
},
```

4.2.3. *Tokenizing*

Selanjutnya proses tokenizing, fungsi NLTK (Natural Language Toolkit) digunakan pada tahap ini. Data yang digunakan pada proses ini merupakan data yang telah melalui proses sebelumnya yakni proses cleaning. Segmen 4.4 merupakan segmen program.

Segmen 4. 4 Tokenizing

```
import json  
import re  
import nltk  
  
# Make sure you have the NLTK data required  
nltk.download('punkt')  
  
# Function to tokenize the text  
def tokenize_text(text):  
    return nltk.word_tokenize(text)  
  
# Load the JSON file  
file_path = 'HASIL/labeled_comments_cleaned.json'  
with open(file_path, 'r', encoding='utf-8') as file:  
    data = json.load(file)  
  
# Perform cleaning and tokenizing
```

```
for comment in data:

    comment['tokens'] = tokenize_text(comment['text'])

# Save the tokenized data back to a new JSON file
output_path = 'HASIL/labeled_comments_tokenized.json'
with open(output_path, 'w', encoding='utf-8') as file:
    json.dump(data, file, ensure_ascii=False, indent=4)

print(f"Tokenized data saved to {output_path}")
```

Segmen 4.4 ini mengimpor modul `json`, `re`, dan `nltk` yang diperlukan untuk memanipulasi data JSON, melakukan pembersihan teks, dan melakukan tokenisasi teks. Kode memastikan bahwa data yang diperlukan oleh NLTK untuk tokenisasi diunduh dengan menjalankan `nltk.download('punkt')`.

Selanjutnya, fungsi `tokenize_text` didefinisikan untuk membagi teks menjadi token kata menggunakan `nltk.word_tokenize`. Kemudian, file JSON yang berlokasi di `HASIL/labeled_comments_cleaned.json` dibuka dan isinya dimuat ke dalam variabel `data`. Setelah data dimuat, fungsi tokenisasi diterapkan pada setiap teks komentar di dalam data, dan hasil tokenisasi disimpan di bawah kunci baru `tokens` untuk setiap komentar.

Terakhir, data yang telah ditokenisasi disimpan ke file JSON baru di path `HASIL/labeled_comments_tokenized.json`. File tersebut dibuka dengan mode tulis dan encoding UTF-8, lalu data disimpan menggunakan `json.dump` dengan pengaturan `ensure_ascii=False` dan `indent=4` untuk format yang terindenter. Pesan konfirmasi ditampilkan untuk memberitahukan bahwa data yang telah ditokenisasi telah disimpan dengan sukses. Sampel hasil tokenizing bisa pada tabel 4.4.

Tabel 4. 4 Sampel Hasil Tokenizing

```
[
  {
    "text": "jgn pikirin rohingya org asli peribumi msh byk yg k laparan
k susah usirr tuh org rohingya lama kaya org yaman tinggal d indonesia
yaman tinggal d indonesia semoga bp perabowo tegas",
    "tokens": [
      "jgn",
      "pikirin",
      "rohingya",
      "org",
      "asli",
      "peribumi",
      "msh",
      "byk",
      "yg",
      "k",
      "laparan",
      "k",
      "susah",
      "usirr",
      "tuh",
      "org",
      "rohingya",
      "lama",
      "kaya",
      "org",
      "yaman",
      "tinggal",
      "d",
      "indonesia",
      "yaman",
      "tinggal",
      "d",
      "indonesia",
      "semoga",
      "bp",
      "perabowo",
      "tegas"
    ]
  }
]
```

Tabel Lanjutan 4.4

```
    },  
    {  
      "text": "pulangkan saja",  
      "tokens": [  
  
        "pulangkan",  
        "saja"  
      ]  
    },  
    {  
      "text": "kalo dilamain ntar minta tanah",  
      "tokens": [  
        "kalo",  
        "dilamain",  
        "ntar",  
        "minta",  
        "tanah"  
      ]  
    },  
    {  
      "text": "saya unsubscribe host tidak pakai kerudung",  
      "tokens": [  
        "saya",  
        "unsubscribe",  
        "host",  
        "tidak",  
        "pakai",  
        "kerudung"  
      ]  
    },  
    {  
      "text": "yang bener umat islam dirangkul dibantu kalau ga tau tata  
krama diajari itulah yang nabi muhammad ajarkan",  
      "tokens": [  
        "yang",  
        "bener",  
        "umat",  
        "islam",
```

Tabel Lanjutan 4.4

```
"dirangkul",
"dibantu",
"kalau",
"ga",
"tau",
"tata",
"krama",
"diajari",
"itulah",
"yang",
"nabi",
"muhammad",
"ajarkan"
]
},
{
  "text": "biarkan sajah kaya di taelad mati kelaparan kan kita ga
ngundang",
  "tokens": [
    "biarkan",
    "sajah",
    "kaya",
    "di",
    "taelad",
    "mati",
    "kelaparan",
    "kan",
    "kita",
    "ga",
    "ngundang"
  ]
},
{
  "text": "pemerintah gak punya nyali untuk mengusir rohingya dan
nekan unhcr yg jadi kaki tangan barat",
```

Tabel Lanjutan 4.4

```
"tokens": [  
    "pemerintah",  
    "gak",  
    "punya",  
    "nyali",  
    "untuk",  
    "mengusir",  
    "rohingya",  
    "dan",  
    "nekan",  
    "unher",  
    "yg",  
    "jadi",  
    "kaki",  
    "tangan",  
    "barat"  
]  
},
```

4.2.4. *Stopword*

Pada proses stopwords NLTK juga digunakan dalam proses ini. Data yang digunakan merupakan data hasil dari proses tokenizing sebelumnya. Segmen 4.5 merupakan segmen program stopwords.

Segmen 4. 5 Stopword

```
import json  
import nltk  
from nltk.corpus import stopwords  
  
# Download the necessary NLTK data  
nltk.download('stopwords')  
  
# Function to remove stopwords from the text  
def remove_stopwords(tokens):  
    stop_words = set(stopwords.words('indonesian'))  
    return [word for word in tokens if word not in stop_words]
```

Tabel Lanjutan 4.5

```
# Load the JSON file
file_path = 'HASIL/labeled_comments_tokenized.json'
with open(file_path, 'r', encoding='utf-8') as file:
    data = json.load(file)

# Perform stopwords removal
for comment in data:
    comment['tokens'] = remove_stopwords(comment['tokens'])

# Save the processed data back to a new JSON file
output_path = 'HASIL/labeled_comments_processed.json'
with open(output_path, 'w', encoding='utf-8') as file:
    json.dump(data, file, ensure_ascii=False, indent=4)

print(f"Processed data saved to {output_path}")
```

Pertama, kode ini mengimpor modul `json`, `nlk`, dan `stopwords` dari pustaka NLTK yang diperlukan untuk memanipulasi data JSON, melakukan pengolahan teks, dan menghapus kata-kata berhenti. Kode memastikan bahwa data yang diperlukan oleh NLTK untuk mengakses daftar kata-kata berhenti diunduh dengan menjalankan `nlk.download('stopwords')`.

Fungsi `remove_stopwords` didefinisikan untuk menghapus kata-kata berhenti dari daftar token. Fungsi ini pertama-tama memuat kata-kata berhenti dalam bahasa Indonesia menggunakan `stopwords.words('indonesian')` dan menyimpannya dalam set `stop_words`. Kemudian, fungsi ini mengembalikan daftar kata yang tidak termasuk dalam set kata-kata berhenti.

Selanjutnya, kode membuka file JSON yang terletak di `HASIL/labeled_comments_tokenized.json` dan memuat isinya ke dalam variabel `data` dengan encoding UTF-8. Setelah data dimuat, fungsi `remove_stopwords`

diterapkan pada setiap daftar token dalam data, dan hasilnya diperbarui di bawah kunci ``tokens`` untuk setiap komentar.

Data yang telah diproses disimpan ke file JSON baru di path ``HASIL/labeled_comments_processed.json``. File tersebut dibuka dengan mode tulis dan encoding UTF-8, lalu data disimpan menggunakan ``json.dump`` dengan pengaturan ``ensure_ascii=False`` dan ``indent=4`` untuk format yang terindenter. Sampel hasil stopwords bisa dilihat pada tabel 4.5.

Tabel 4. 5 Sampel Hasil Stopword

```
[
  {
    "text": "jgn pikirin rohingya org asli peribumi msh byk yg k laparan
k susah usirr tuh org rohingya lama kaya org yaman tinggal d indonesia
yaman tinggal d indonesia semoga bp perabowo tegas",
    "tokens": [
      "jgn",
      "pikirin",
      "rohingya",
      "org",
      "asli",
      "peribumi",
      "msh",
      "byk",
      "yg",
      "k",
      "laparan",
      "k",
      "susah",
      "usirr",
      "tuh",
      "org",
      "rohingya",
      "kaya",
      "org",
      "yaman",
      "tinggal",
      "d",
```

Tabel Lanjutan 4.5

```
"indonesia",
"yaman",
"tinggal",
"d",
"indonesia",
"semoga",
"bp",
"perabowo"
]
},
{
"text": "pulangkan saja",
"tokens": [
  "pulangkan"
]
},
{
"text": "kalo dilamain ntar minta tanah",
"tokens": [
  "kalo",
  "dilamain",
  "ntar",
  "tanah"
]
},
{
"text": "saya unsubscribe host tidak pakai kerudung",
"tokens": [
  "unsubscribe",
  "host",
  "pakai",
  "kerudung"
]
},
{
"text": "yang bener umat islam dirangkul dibantu kalau ga tau tata
krama diajari itulah yang nabi muhammad ajarkan",
"tokens": [
  "bener",
  "umat",
```

Tabel Lanjutan 4.5

```
"islam",
"dirangkul",
"dibantu",
"ga",
"tau",
"tata",
"krama",
"diajari",
"nabi",
"muhammad",
"ajarkan"
]
},
{
"text": "biarkan sajah kaya di taelad mati kelaparan kan kita ga
ngundang",
"tokens": [
"biarkan",
"sajah",
"kaya",
"taelad",
"mati",
"kelaparan",
"ga",
"ngundang"
]
},
{
"text": "pemerintah gak punya nyali untuk mengusir rohingya dan
nekan unhcr yg jadi kaki tangan barat",
"tokens": [
"pemerintah",
"gak",
"nyali",
"mengusir",
"rohingya",
"nekan",
"unhcr",
"kaki",
"tangan",
"barat"
]
},
}
```

4.2.5. Stemming

Proses ini merupakan proses terakhir dalam preprocessing. Dalam proses ini NLTK juga di fungsikan. Data yang digunakan merupakan data hasil proses sebelumnya yakni stopwords. Segmen 4.6 merupakan segmen program stemming dan sampel hasil stemming bisa dilihat pada tabel 4.6.

Segmen 4. 6 Stemming

```
import json
import nltk
from nltk.stem import PorterStemmer

# Download the necessary NLTK data
nltk.download('punkt')

# Function to stem tokens
def stem_tokens(tokens):
    stemmer = PorterStemmer()
    return [stemmer.stem(token) for token in tokens]

# Load the JSON file
file_path = 'HASIL/labeled_comments_processed.json'
with open(file_path, 'r', encoding='utf-8') as file:
    data = json.load(file)

# Perform stemming on the tokenized text
for comment in data:
    # Ensure the text is tokenized before stemming
    if 'tokens' in comment:
        comment['tokens'] = stem_tokens(comment['tokens'])

# Save the stemmed data back to a new JSON file
output_path = 'HASIL/labeled_comments_stemmed.json'
with open(output_path, 'w', encoding='utf-8') as file:
    json.dump(data, file, ensure_ascii=False, indent=4)

print(f"Stemmed data saved to {output_path}")
```

Segmen 4.6 ini mengimpor modul ``json`` dan ``nlk``, serta ``PorterStemmer`` dari pustaka NLTK, yang diperlukan untuk memanipulasi data JSON, melakukan stemming pada teks, dan melakukan pemrosesan bahasa alami. Kode memastikan bahwa data yang diperlukan untuk tokenisasi diunduh dengan menjalankan ``nlk.download('punkt')``. Fungsi ``stem_tokens`` didefinisikan untuk menerapkan stemming pada daftar token. Fungsi ini membuat objek ``PorterStemmer`` dan menerapkan metode ``stem`` pada setiap token dalam daftar, mengembalikan daftar token yang telah distem

Selanjutnya, kode membuka file JSON yang terletak di ``HASIL/labeled_comments_processed.json`` dan memuat isinya ke dalam variabel ``data`` dengan encoding UTF-8. Setelah data dimuat, fungsi ``stem_tokens`` diterapkan pada daftar token dalam setiap komentar. Kode memeriksa terlebih dahulu apakah kunci ``tokens`` ada dalam komentar sebelum menerapkan stemming untuk memastikan bahwa hanya data yang sudah ditokenisasi yang diproses.

Data yang telah distem disimpan ke file JSON baru di path ``HASIL/labeled_comments_stemmed.json``. File dibuka dengan mode tulis dan encoding UTF-8, dan data disimpan menggunakan ``json.dump`` dengan pengaturan ``ensure_ascii=False`` dan ``indent=4`` untuk format JSON yang terindenter. Pesan konfirmasi ditampilkan untuk memberitahukan bahwa data yang telah distem telah disimpan dengan sukses.

Tabel 4. 6 Sampel Hasil Stemming

```
[
  {
    "text": "jgn pikirin rohingya org asli peribumi msh byk yg k laparan
k susah usirr tuh org rohingya lama kaya org yaman tinggal d indonesia
yaman tinggal d indonesia semoga bp perabowo tegas",
    "tokens": [
      "jgn",
      "pikirin",
      "rohingya",
      "org",
      "asli",
      "peribumi",
      "msh",
      "byk",
      "yg",
      "k",
      "laparan",
      "k",
      "susah",
      "usirr",
      "tuh",
      "org",
      "rohingya",
      "kaya",
      "org",
      "yaman",
      "tinggal",
      "d",
      "indonesia",
      "yaman",
      "tinggal",
      "d",
      "indonesia",
      "semoga",
      "bp",
      "perabowo"
    ]
  }
]
```

Tabel Lanjutan 4.6

```
},  
{  
  "text": "pulangkan saja",  
  "tokens": [  
    "pulangkan"  
  ]  
},  
{  
  "text": "kalo dilamain ntar minta tanah",  
  "tokens": [  
    "kalo",  
    "dilamain",  
    "ntar",  
    "tanah"  
  ]  
},  
{  
  "text": "saya unsubscribe host tidak pakai kerudung",  
  "tokens": [  
    "unsubscrib",  
    "host",  
    "pakai",  
    "kerudung"  
  ]  
},  
{  
  "text": "yang bener umat islam dirangkul dibantu kalau ga tau tata  
krama diajari itulah yang nabi muhammad ajarkan",  
  "tokens": [  
    "bener",  
    "umat",  
    "islam",  
    "dirangkul",  
    "dibantu",  
    "ga",  
    "tau",  
    "tata",
```

Tabel Lanjutan 4.6

```
"krama",
"diajari",
"nabi",
"muhammad",
"ajarkan"
]
},
{
  "text": "biarkan sajah kaya di taelad mati kelaparan kan kita ga
ngundang",
  "tokens": [
    "biarkan",
    "sajah",
    "kaya",
    "taelad",
    "mati",
    "kelaparan",
    "ga",
    "ngundang"
  ]
},
{
  "text": "pemerintah gak punya nyali untuk mengusir rohingya dan
nekan unhcr yg jadi kaki tangan barat",
  "tokens": [
    "pemerintah",
    "gak",
    "nyali",
    "mengusir",
    "rohingya",
    "nekan",
    "unhcr",
    "yg",
    "kaki",
    "tangan",
    "barat"
  ]
},
}
```

4.3 Proses Pelabelan

Proses pelabelan dilakukan menggunakan metode berbasis kamus (dictionary-based), metode ini merupakan metode yang sederhana, di mana sentimen ditentukan dengan menghitung kemunculan kata-kata positif dan negatif dalam komentar. Segmen 4.7 merupakan segmen program pelabelan.

Segmen 4.7 Pelabelan

```
import json

# Membaca file JSON
with open('HASIL/labeled_comments_stemmed.json', 'r') as file:
    data = json.load(file)

# Daftar kata negatif dan positif
negative_words = [
    'usir', 'pulang', 'bubarkan', 'tidak', 'susah', 'lapar', 'beban', 'menolak',
    'gk', 'negatif', 'buruk', 'jelek'
]
positive_words = [
    'bantu', 'islam', 'nabi', 'muhammad', 'ajarkan', 'positif', 'baik', 'bagus'
]

# Fungsi untuk menentukan label sentimen
def label_sentiment(tokens):
    negative_count = sum(token in negative_words for token in tokens)
    positive_count = sum(token in positive_words for token in tokens)

    if negative_count >= positive_count:
        return 'Negative'
    else:
        return 'Positive'

# Melabeli data
for entry in data:
    entry['sentiment'] = label_sentiment(entry['tokens'])
```

Segmen Lanjutan 4.7

```
# Menyimpan hasil ke file JSON baru
with open('HASIL/labeled_comments_with_sentiment.json', 'w') as file:
    json.dump(data, file, ensure_ascii=False, indent=4)

print("Pelabelan selesai, file disimpan sebagai
labeled_comments_with_sentiment.json")
```

Skrip python pada segmen 4.7 dirancang untuk melakukan pelabelan sentimen pada data komentar yang disimpan dalam format JSON. Pertama, skrip membuka dan membaca file JSON yang berisi data komentar yang telah diproses sebelumnya. Data ini diambil dari file bernama `labeled_comments_stemmed.json` yang terletak di direktori `HASIL`. Setelah membaca data, skrip menginisialisasi dua daftar kata, yaitu kata-kata negatif dan positif, yang digunakan untuk menentukan sentimen dari setiap komentar.

Kemudian, skrip mendefinisikan fungsi `label_sentiment` yang bertugas untuk menghitung jumlah kata negatif dan positif dalam setiap komentar. Fungsi ini menggunakan list comprehension untuk menghitung berapa banyak kata dalam daftar `tokens` yang sesuai dengan kata-kata dalam daftar `negative_words` dan `positive_words`. Berdasarkan perbandingan antara jumlah kata negatif dan positif, fungsi ini mengembalikan label sentimen yang sesuai, yaitu 'Negative' jika kata negatif lebih banyak atau sama banyaknya dengan kata positif, dan 'Positive' sebaliknya.

Terakhir, skrip menerapkan fungsi pelabelan ini ke setiap entri dalam data dan menambahkan label sentimen ke dalam setiap entri dengan kunci 'sentiment'.

Setelah semua data diberi label, skrip menyimpan hasil akhir ke dalam file JSON baru dengan nama `labeled_comments_with_sentiment.json` di direktori `HASIL`. Skrip mengakhiri proses dengan mencetak pesan yang menandakan bahwa pelabelan telah selesai dan file baru telah disimpan. Sampel hasil pelabelan bisa dilihat pada tabel 4.7.

Tabel 4. 7 Sampel Hasil Pelabelan

```
[
  {
    "text": "jgn pikirin rohingya org asli peribumi msh byk yg k laparan
k susah usirr tuh org rohingya lama kaya org yaman tinggal d indonesia
yaman tinggal d indonesia semoga bp perabowo tegas",
    "tokens": [
      "jgn",
      "pikirin",
      "rohingya",
      "org",
      "asli",
      "peribumi",
      "msh",
      "byk",
      "yg",
      "k",
      "laparan",
      "k",
      "susah",
      "usirr",
      "tuh",
      "org",
      "rohingya",
      "kaya",
      "org",
      "yaman",
      "tinggal",
      "d",
    ]
  }
]
```

Tabel Lanjutan 4.7

```
"indonesia",
"yaman",
"tinggal",
"d",
"indonesia",
"semoga",
"bp",
"perabowo"
],
"sentiment": "Negative"
},
{
"text": "pulangkan saja",
"tokens": [
  "pulangkan"
],
"sentiment": "Negative"
},
{
"text": "kalo dilamain ntar minta tanah",
"tokens": [
  "kalo",
  "dilamain",
  "ntar",
  "tanah"
],
"sentiment": "Negative"
},
{
"text": "saya unsubscribe host tidak pakai kerudung",
"tokens": [
  "unsubscrib",
  "host",
  "pakai",
  "kerudung"
],
"sentiment": "Negative"
```

Tabel Lanjutan 4.7

```
    },  
    {  
      "text": "yang bener umat islam dirangkul dibantu kalau ga tau tata  
krama diajari itulah yang nabi muhammad ajarkan",  
      "tokens": [  
        "bener",  
        "umat",  
        "islam",  
        "dirangkul",  
        "dibantu",  
        "ga",  
        "tau",  
        "tata",  
        "krama",  
        "diajari",  
        "nabi",  
        "muhammad",  
        "ajarkan"  
      ],  
      "sentiment": "Positive"  
    },  
    {  
      "text": "biarkan sajah kaya di taelad mati kelaparan kan kita ga  
ngundang",  
      "tokens": [  
        "biarkan",  
        "sajah",  
        "kaya",  
        "taelad",  
        "mati",  
        "kelaparan",  
        "ga",  
        "ngundang"  
      ],  
      "sentiment": "Negative"  
    },  
  ],  
}
```

Tabel Lanjutan 4.7

```
{
  "text": "pemerintah gak punya nyali untuk mengusir rohingya dan
nekan unhr yg jadi kaki tangan barat",
  "tokens": [
    "pemerintah",
    "gak",
    "nyali",
    "mengusir",
    "rohingya",
    "nekan",
    "unhr",
    "yg",
    "kaki",
    "tangan",
    "barat"
  ],
  "sentiment": "Negative"
},
```

4.4 Pembobotan TF-IDF dan Klasifikasi SVM

Pembobotan TF-IDF (Term Frequency-Inverse Document Frequency) digunakan untuk menilai pentingnya kata dalam dokumen relatif terhadap koleksi dokumen. Nilai TF-IDF tinggi menunjukkan bahwa kata tersebut sering muncul dalam dokumen tertentu namun jarang muncul di dokumen lain, menunjukkan kata tersebut penting untuk dokumen tersebut.

Setelah melakukan pembobotan akan dilakukan klasifikasi menggunakan metode svm. Dalam klasifikasi terdapat data pengujian dan data pelatihan yakni dengan ratio 90:10 menggunakan `test_size=0.9` , 80:20 menggunakan `test_size=0.8` dan 70:30 menggunakan `test_size=0.7`. Data pelatihan digunakan untuk mempelajari pola sentiment dan data pengujian digunakan untuk mengevaluasi

kinerja dan akurasinya. Segmen 4.8 merupakan segmen program pembobotan serta klasifikasi dan sampel hasil klasifikasi bisa dilihat pada tabel 4.8 untuk ratio 90:10, tabel 4.9 untuk ratio 80:10 dan tabel 4.10 untuk ratio 70:30.

Segmen 4. 8 Pembobotan dan Klasifikasi

```
import pandas as pd
import json
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.model_selection import train_test_split
from sklearn.svm import SVC
from sklearn.metrics import classification_report, accuracy_score

# Load data from JSON file
file_path = 'HASIL/labeled_comments_with_sentiment.json'
with open(file_path, 'r', encoding='utf-8') as file:
    data = json.load(file)

# Convert data to DataFrame
df = pd.DataFrame(data)

# Display the first few rows of the dataframe
print(df.head())

# Extract features and labels
X = df['text']
y = df['sentiment']

# Perform TF-IDF vectorization
vectorizer = TfidfVectorizer()
X_tfidf = vectorizer.fit_transform(X)

# Split the data into training and testing sets with a ratio
X_train, X_test, y_train, y_test = train_test_split(X_tfidf, y,
test_size=0.9, random_state=42)

# Train the SVM model
svm_model = SVC(kernel='linear')
svm_model.fit(X_train, y_train)
```

Segmen Lanjutan 4.8

```
# Predict on the test data
y_pred = svm_model.predict(X_test)

# Evaluate the model
accuracy = accuracy_score(y_test, y_pred)
report = classification_report(y_test, y_pred)

print(f"Accuracy: {accuracy}")
print("Classification Report:")
print(report)
```

Skrip Python ini dirancang untuk membangun dan mengevaluasi model pembelajaran mesin yang mengklasifikasikan sentimen komentar menggunakan data yang telah dilabeli. Langkah pertama dalam skrip adalah memuat data dari file JSON bernama `labeled_comments_with_sentiment.json`, yang berisi komentar bersama label sentimennya. Data ini dibaca dengan menggunakan modul `json` dan diubah menjadi sebuah DataFrame dengan pustaka `pandas`. Setelah konversi selesai, skrip menampilkan beberapa baris pertama dari DataFrame untuk memastikan bahwa data telah diimpor dengan benar dan siap untuk diproses lebih lanjut.

Dengan data yang telah siap, skrip melanjutkan untuk menyiapkan fitur dan label untuk pemodelan. Kolom 'text' diambil sebagai fitur (X) yang akan digunakan untuk pelatihan model, sementara kolom 'sentiment' diambil sebagai label (y) yang akan diprediksi oleh model. Fitur teks kemudian diproses menggunakan TF-IDF vectorization yang diterapkan oleh `TfidfVectorizer` dari pustaka `scikit-learn`.

Proses ini mengubah teks komentar menjadi representasi numerik yang dapat dipahami oleh algoritma pembelajaran mesin.

Data yang telah diubah menjadi bentuk TF-IDF kemudian dibagi menjadi dua set: set pelatihan dan set pengujian, dengan proporsi 90% untuk pengujian dan 10% untuk pelatihan untuk kedua ratio yang lain cukup mengubah nominal `tets_size` sesuai penjelasan diatas, menggunakan fungsi ``train_test_split``. Pembagian ini memungkinkan evaluasi model yang lebih baik dengan memastikan bahwa model diuji pada data yang belum pernah dilihat sebelumnya. Selanjutnya, model Support Vector Machine (SVM) dengan kernel linear dilatih menggunakan data pelatihan. Proses pelatihan ini melibatkan pengoptimalan model untuk memaksimalkan akurasi prediksi pada data pelatihan.

Setelah model dilatih, skrip menggunakan model yang telah terlatih untuk memprediksi label sentimen pada data pengujian. Hasil prediksi dibandingkan dengan label asli dari data pengujian untuk mengevaluasi kinerja model. Skrip kemudian menghitung akurasi prediksi menggunakan ``accuracy_score`` dan menghasilkan laporan klasifikasi dengan ``classification_report``. Kedua metrik ini memberikan wawasan tentang seberapa baik model dalam mengklasifikasikan sentimen komentar.

Terakhir, skrip mencetak hasil evaluasi model, termasuk nilai akurasi dan laporan klasifikasi. Laporan ini mencakup informasi detail seperti presisi, recall, dan skor F1 untuk masing-masing kelas sentimen, memberikan gambaran menyeluruh tentang kekuatan dan kelemahan model dalam tugas klasifikasi.

Dengan output ini, pengguna dapat menilai efektivitas model dalam mengenali dan mengklasifikasikan sentimen komentar berdasarkan data yang telah dilabeli.

Tabel 4. 8 Hasil Klasifikasi Ratio 90:10

	Precision	Recall	F1-Score	Support
Negative	0.98	1.00	0.99	1762
Positive	0.00	0.00	0.00	38
Accuracy			0.98	1800
Macro avg	0.49	0.50	0.49	1800
Weighted avg	0.96	0.98	0.97	1800

Tabel 4. 9 Hasil Klasifikasi Ratio 80:20

	Precision	Recall	F1-Score	Support
Negative	0.98	1.00	0.99	1566
Positive	1.00	0.02	0.04	34
Accuracy			0.98	1600
Macro avg	0.49	0.50	0.49	1600
Weighted avg	0.96	0.98	0.97	1600

Tabel 4. 10 Hasil Klasifikasi Ratio 70:30

	Precision	Recall	F1-Score	Support
Negative	0.98	1.00	0.99	1372
Positive	1.00	0.11	0.19	28
Accuracy			0.98	1400
Macro avg	0.99	0.55	0.59	1400
Weighted avg	0.98	0.98	0.98	1400

4.5 Evaluasi Model

4.5.1. Akurasi

Semua rasio menunjukkan akurasi yang sangat tinggi, yaitu 98%. Meskipun angka ini terlihat menjanjikan, akurasi yang tinggi mungkin tidak mencerminkan

performa model secara menyeluruh. Hal ini terutama disebabkan oleh adanya ketidakseimbangan kelas yang signifikan dalam dataset. Ketidakseimbangan ini dapat mempengaruhi kemampuan model dalam mengklasifikasikan kelas minoritas dengan baik. Oleh karena itu, penting untuk mempertimbangkan metrik evaluasi tambahan untuk mendapatkan gambaran yang lebih akurat mengenai kinerja model.

4.5.2. Precision, Recall, dan F1-Score

Kelas Negatif: Precision dan Recall mendekati 1.0 untuk semua rasio, menandakan bahwa model sangat efektif dalam mengidentifikasi instance negatif dengan akurat.

Kelas Positif: Precision dan Recall menunjukkan performa yang sangat rendah. Khususnya, Recall mendekati 0 untuk rasio 90:10 dan 80:20, serta hanya mencapai 0.11 untuk rasio 70:30. Hal ini menunjukkan bahwa model kurang mampu mendeteksi instance positif secara efektif..

4.5.3. Macro Average dan Weighted Average

Macro Average: Nilai Precision, Recall, dan F1-Score yang rendah, sekitar 0.50, menunjukkan bahwa performa model tidak seimbang antara kelas negatif dan positif.

Weighted Average: Nilai yang lebih tinggi dibandingkan macro average mengindikasikan dominasi kelas dengan jumlah instance yang lebih besar (kelas negatif) dalam metrik ini.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Model *SVM (Support Vector Machine)* yang diterapkan dalam penelitian ini menunjukkan akurasi yang sangat tinggi, mencapai 98%. Meskipun angka ini tampak menjanjikan, penting untuk menginterpretasikannya dengan hati-hati. Tingginya akurasi ini mungkin tidak mencerminkan performa model secara keseluruhan, terutama karena adanya ketidakseimbangan kelas yang signifikan dalam dataset. Dalam situasi seperti ini, model cenderung lebih akurat dalam memprediksi kelas yang dominan, tetapi bisa jadi kurang efektif dalam memprediksi kelas yang lebih sedikit jumlahnya.

Evaluasi metrik seperti *precision*, *recall*, dan *F1-score* mengungkapkan bahwa nilai macro average berada pada tingkat yang cukup rendah, sekitar 0.50, yang menunjukkan ketidakseimbangan performa antara kelas positif dan negatif. Selain itu, perbedaan antara nilai macro average dan weighted average mengindikasikan bahwa weighted average lebih tinggi, yang mencerminkan dominasi kelas mayoritas dalam penilaian performa model. Ini menegaskan bahwa meskipun akurasi keseluruhan tampak tinggi, model mungkin tidak sepenuhnya efektif dalam menangani semua kelas secara seimbang, sehingga penting untuk mempertimbangkan metrik evaluasi yang lebih menyeluruh.

5.2 Saran

Disarankan untuk menggunakan teknik penanganan ketidakseimbangan kelas seperti oversampling pada kelas minoritas atau undersampling pada kelas mayoritas untuk memperbaiki kemampuan model dalam mendeteksi instance positif. Serta menambah jumlah data komentar dari berbagai sumber atau platform lain untuk memperkaya dataset dan meningkatkan generalisasi model dalam analisis sentiment.



DAFTAR PUSTAKA


- [1] B. Budaya, “Dampak Kewarganegaraan Etnis Rohingya Di Myanmar Terhadap Pelanggaran Hak Asasi Manusia Dan Negara Sekitar,” *J. Ilm. Huk.*, vol. 11, no. 1, pp. 106–120, 2017, [Online]. Available: <https://maksigama.wisnuwardhana.ac.id/index.php/maksigama/article/view/44>
- [2] Wa Ode Rizka Fauzia Ali, “Sejarah Konflik Muslim Rohingya,” *J. Pengabd. Masy. Indones. Sejah.*, vol. 1, no. 3, pp. 43–62, 2022, doi: 10.59059/jpmis.v1i3.6.
- [3] A. A. Apsarini, L. Sofwan, M. N. Kurnia, and S. F. Muzakki, “Tindakan Indonesia Dalam Melindungi Pengungsi Rohingya Berdasarkan Perspektif Etika Utilitarianisme,” *J. Kaji. Kontemporer Huk. dan Masy.*, vol. 02, no. 01, pp. 1–25, 2024, doi: 10.11111/dassollen.xxxxxxx.
- [4] S. Mamis, A. S. Rustan, S. H. Arnus, and H. Basri, “Framing Media Dan Dinamika Opini Publik Terkait Pengungsi Rohingya Di Indonesia: Perspektif UNHCR Dan Otoritas Aceh Dalam Sorotan Kritis,” *Wardah*, vol. 24, no. 2, pp. 137–163, 2024, doi: 10.19109/wardah.v24i2.21384.
- [5] S. A. Anan Septia Machidhar, Maya Sekar Wangi, “EFEKTIVITAS PENGGUNAAN MEDIA SOSIAL YOUTUBE SEBAGAI MEDIA PENYEBARAN INFORMASI PADA CHANNEL YOUTUBE DISKOMINFO BOYOLALI,” 2022.
- [6] A. Liawati *et al.*, “ANALISIS SENTIMEN KOMENTAR POLITIK DI MEDIA SOSIAL X DENGAN PENDEKATAAN DEEP LEARNING,” vol. 7, no. 6, pp. 3557–3563, 2023.
- [7] Y. A. Saputra, “SENTIMEN MAHASISWA TERHADAP MATERI PEMBELAJARAN KOMPUTER: STUDI KASUS DAN ANALISIS,” vol. 1, no. 3, pp. 1–16, 2024.
- [8] B. A. Maulana and M. J. Fahmi, “Sentiment Analysis of Pluang Applications With Naive Bayes and Support Vector Machine (SVM) Algorithm Analisis Sentimen Terhadap Aplikasi Pluang Menggunakan Algoritma Naive Bayes dan Support Vector Machine (SVM),” vol. 4, no. April, pp. 375–384, 2024.
- [9] N. Zelina and A. Afiyati, “Analisis Sentimen Ulasan Pengguna Aplikasi M- Banking Menggunakan Algoritma Support Vector Machine dan Decision Tree,” vol. 7, no. 1, pp. 31–37, 2024.
- [10] D. Mualfah, Ramadhoni, R. Gunawan, and D. Mulyadipa Suratno, “Analisis Sentimen Komentar YouTube TvOne Tentang Ustadz Abdul Somad Dideportasi Dari Singapura Menggunakan Algoritma SVM,” *J. Fasilkom*, vol. 13, no. 01, pp. 72–80, 2023, doi: 10.37859/jf.v13i01.4920.
- [11] A. Nursalim, R. Novita, I. Systems, and S. Program, “SENTIMENT

- ANALYSIS OF COMMENTS ON GOOGLE PLAY STORE , TWITTER AND YOUTUBE TO THE MYPERTAMINA APPLICATION WITH SUPPORT VECTOR ANALISIS SENTIMEN KOMENTAR PADA GOOGLE PLAY STORE , TWITTER DAN YOUTUBE TERHADAP APLIKASI MYPERTAMINA DENGAN SUPPORT,” vol. 4, no. 6, pp. 1305–1312, 2023.
- [12] A. Wijayanto, “Analisis Sentimen Komentar Youtube Mengenai Vaksin Covid-19 Menggunakan Support Vector Machine,” *J. PILAR Teknol. J. Ilm. Ilmu Ilmu Tek.*, vol. 7, no. 1, pp. 24–31, 2022, doi: 10.33319/piltek.v7i1.118.
- [13] A. C. Kamilla, N. Priyani, R. Priskila, and V. H. Pranatawijaya, “ANALISIS SENTIMEN FILM AGAK LAEN DENGAN KECERDASAN BUATAN: TEXT MINING METODE NAÏVE BAYES CLASSIFIER,” vol. 8, no. 3, pp. 2923–2928, 2024.
- [14] G. Fanani, I. Riadi, and A. Yudhana, “Analisis Forensik Aplikasi Michat Menggunakan Metode Digital Forensics Research Workshop,” vol. 6, no. April, pp. 1263–1271, 2022, doi: 10.30865/mib.v6i2.3946.
- [15] P. Arsi, L. N. Hidayati, and A. Nurhakim, “Komparasi Model Klasifikasi Sentimen Issue Vaksin Covid-19 Berbasis Platform Instagram,” vol. 6, pp. 459–466, 2022, doi: 10.30865/mib.v6i1.3509.
- [16] B. Liu, *Sentiment Analysis and Opinion Mining*. in Synthesis digital library of engineering and computer science. Morgan & Claypool, 2012. [Online]. Available: <https://books.google.co.id/books?id=Gt8g72e6MuEC>
- [17] A. N. Husna and P. Rianto, “Membaca Komentar di Media Sosial Sebagai Hiburan Reading Comments on Social Media for Entertainment,” *J. Mhs. Komun. Cantrik*, vol. 1, pp. 29–40, 2021.
- [18] N. Made, I. Priyanti, and U. Musamus, “PENERAPAN MODEL PEMBELAJARAN PROBLEM BASED LEARNING BERBANTUAN MEDIA YOUTUBE,” vol. 4, no. 1, pp. 96–101, 2023.
- [19] R. T. Gareth James, Daniela Witten, Trevor Hastie and An, *An Introduction to Statistical Learning with Applications in R*, vol. 7, no. 10. 2000. doi: 10.2174/0929867003374372.
- [20] C. H. Yutika, A. Adiwijaya, and S. Al Faraby, “Analisis Sentimen Berbasis Aspek pada Review Female Daily Menggunakan TF-IDF dan Naïve Bayes,” *J. Media Inform. Budidarma*, vol. 5, no. 2, p. 422, 2021, doi: 10.30865/mib.v5i2.2845.
- [21] J. August, I. No, and M. Goyal, “Available Online at www.ijarcs.info International Journal of Advanced Research in Computer Science A Review on K-Mode Clustering Algorithm,” vol. 8, no. 7, pp. 725–729, 2017.
- [22] M. Nanda Syarif, N. Pambudiyatno, W. Utomo, J. I. Jemur Andayani No, and K. Siwalankerto Kec Wonocolo, “Rancangan Sistem

- Presensi Dan Rekapitulasi Jurnal Kegiatan Ojt Menggunakan Visual Studio Code Berbasis Web Di Airnav Cabang Matsc,” *Pros. Semin. Nas. Inov. Teknol. Penerbangan Tahun*, p. 2023, 2023.
- [23] Z. Afif, D. S. Azhari, M. Kustati, and N. Sepriyanti, “Penelitian Ilmiah (Kuantitatif) Beserta Paradigma , Pendekatan , Asumsi Dasar, Karakteristik, Metode Analisis Data Dan Outputnya,” *Innov. J. Soc. Sci. Res.*, vol. 3, no. 3, pp. 682–693, 2023, [Online]. Available: <https://j-innovative.org/index.php/Innovative%0APenelitian>



KARTU BIMBINGAN TUGAS AKHIR / SKRIPSI		
TANGGAL	CATATAN	PARAF

<p style="text-align: center;">KARTU BIMBINGAN TUGAS AKHIR / SKRIPSI FAKULTAS SAINS & TEKNOLOGI UNIVERSITAS IBRAHIMY TAHUN AKADEMIK 2023/2024</p>  <p>NPM : <u>202050304</u> Nama : <u>Hi Bayet</u> Program Studi : <u>Teknologi Informatika</u> Judul TA / Skripsi : <u>Analisis Semantik Pengguna Youtube Terhadap Kualitas Menggunakan Algoritma SVM (Support Vector Machine).</u></p> <p style="text-align: center;">= CATATAN =</p> <ol style="list-style-type: none"> 1. Dalam penyusunan Laporan TA / Skripsi, mahasiswa harus berkonsultasi dengan pembimbingnya secara bertahap. 2. Pada setiap konsultasi, kartu bimbingan harus dibawa dan diisi oleh pembimbing 3. Mahasiswa wajib Konsultasi selama penyusunan Laporan TA / Skripsi ke pembimbing Minimal 6 x 4. Waktu bimbingan dimulai sejak tahapan proposal sampai laporan kegiatan 5. Skedul TA / Skripsi dapat dilihat pada buku panduan penyusunan Laporan Kegiatan.

Data

	author	published_at	updated_at	like_count	text
0	@bambanghadiarso154	2024-03-12T07:48:00Z	2024-03-12T07:48:00Z	0	Bubarkan UNHCR di Indonesia dan usir rohingnya...
1	@budionoluhur6058	2024-03-11T19:55:13Z	2024-03-11T19:55:13Z	0	Pulangkan saja kembali ke asalnya.....
2	@user-qy6jl4yi7g	2024-03-11T13:25:12Z	2024-03-11T13:25:12Z	0	Kalau pun diterima tapi tak boleh sembarangan....
3	@KoesplusBunga-pk8be	2024-03-11T08:15:38Z	2024-03-11T08:15:38Z	0	PENJAJAH YG DIRANCANG ROINGYA SUDAH MENGAJAR DI...
4	@user-ln8mw6pk1w	2024-03-10T21:09:56Z	2024-03-10T21:09:56Z	0	Tidak ada tempat orang luar di Indonesia ini k...
...
95	@BuTatik-zi4fi	2024-01-30T00:41:51Z	2024-01-30T00:41:51Z	0	Gak usah diterima. Apakah kamu mau mereka sep...
96	@user-vy7vh8gy2h	2024-01-29T23:48:28Z	2024-01-29T23:48:28Z	0	Kita berbicara soal kemanusiaan ubtuk menampun...
97	@user-yl3mc1cx1j	2024-01-29T17:27:42Z	2024-01-29T17:27:42Z	0	Untuk nambaj si Uara pemilu.. lumayan .seng...
98	@ahmadlanda8937	2024-01-29T15:47:12Z	2024-01-29T15:47:12Z	0	contohnya sudah ada di Malaysia,, di kasih ha...
99	@ahmadlanda8937	2024-01-29T15:42:02Z	2024-01-29T15:42:02Z	0	jangan ditampung tapi pulangkan saja ke asalny...
	author	published_at	updated_at	like_count	text
0	@kabar_baik	2024-03-11T14:28:12Z	2024-03-11T14:28:12Z	0	Update terbaru berita ini apa ya?
1	@aynsisca862	2024-03-10T01:50:42Z	2024-03-10T01:50:42Z	0	UNHCR pikirannya kemana demi uang nafsu ingin ...
2	@muyaniyani1473	2024-03-09T00:31:12Z	2024-03-09T00:31:12Z	0	Yg jelas ada politik internasional di cox baza...
3	@muyaniyani1473	2024-03-07T04:28:06Z	2024-03-07T04:28:06Z	0	Disediakan penampungan 300000 hektar dihuni 1....
4	@user-dt5xi7vs1d	2024-03-07T02:19:24Z	2024-03-07T02:19:24Z	0	Apa ini hanya akal bulus ameriaka srail iingg...
...
95	@syafa529	2023-12-16T06:43:05Z	2023-12-16T06:43:05Z	0	Makan berak tidur babi
96	@yadisusilo2370	2023-12-16T02:10:48Z	2023-12-16T02:10:48Z	0	Intinya jangan sok jadi pahlawan....rakyat se...
97	@yadisusilo2370	2023-12-16T02:09:20Z	2023-12-16T02:09:20Z	0	Suruh orang UNHCR tampung,kasih .makan semua.....
98	@joyojoyo7857	2023-12-15T17:38:52Z	2023-12-15T17:38:52Z	0	Buat seperatis aja .. Intuk bngkitan GAm ..
99	@sugatirahayu8357	2023-12-15T05:19:02Z	2023-12-15T05:19:02Z	0	Usir Rohingnya kepegunungan Gintung

100 rows × 5 columns

