

**IMPLEMENTASI METODE *LEXICON BASED* DAN *SUPPORT VECTOR MACHINE*
(SVM) DALAM MENGANALISA SENTIMEN PUBLIK TERHADAP PARTAI
POLITIK DI INDONESIA**

TUGAS AKHIR

Sebagai syarat untuk memperoleh gelar sarjana S-1 di Program Studi Informatika, Jurusan
Informatika, Fakultas Teknik Industri, Universitas Pembangunan Nasional “Veteran”
Yogyakarta



Disusun oleh :

Reytama Syahdewa Purba

123200021

**PROGRAM STUDI INFORMATIKA
JURUSAN INFORMATIKA
FAKULTAS TEKNIK INDUSTRI
UNIVERSITAS PEMBANGUNAN NASIONAL “VETERAN”
YOGYAKARTA
2024**

**HALAMAN PENGESAHAN PEMBIMBING
IMPLEMENTASI METODE LEXICON BASED DAN SUPPORT VECTOR
MACHINE (SVM) DALAM MENGANALISA SENTIMEN PUBLIK TERHADAP
PARTAI POLITIK DI INDONESIA**

Disusun Oleh:

Reytama Syahdewa Purba

123200021

Telah di uji dan dinyatakan lulus oleh pembimbing

Pada tanggal :

Menyetujui,
Pembimbing

Dr. Heriyanto, A.Md, S.Kom, M.Cs.

NIDN. 05080066703

Menyetujui,
Koordinator Program Studi

Dessyanto Boedi P., S.T., M.T.

NIDN. 0505127501

**HALAMAN PENGESAHAN PENGUJI
IMPLEMENTASI METODE LEXICON BASED DAN SUPPORT VECTOR
MACHINE (SVM) DALAM MENGANALISA SENTIMEN PUBLIK TERHADAP
PARTAI POLITIK DI INDONESIA**

Disusun Oleh:

Reytama Syahdewa Purba

123200021

Telah di uji dan dinyatakan lulus oleh penguji

Pada tanggal :

Menyetujui,

Penguji 1

Penguji II

Dr. Heriyanto, A.Md, S.Kom, M.Cs

NIDN. 0508067703

Mangaras Yanu F, S.T., M.Eng

NIDN. 0521018201

Penguji III

Penguji IV

Wilis Kaswidjanti, S.Si., M.Kom

NIDN. 0513047601

Dr. Heru Cahya R. S.Si, M.T.

NIDN. 0505127501

SURAT PERNYATAAN KARYA ASLI TUGAS AKHIR

Sebagai mahasiswa Program Studi Informatika Fakultas Teknik Industri Universitas Pembangunan Nasional “Veteran” Yogyakarta, yang bertanda tangan dibawah ini, saya:

Nama : Reytama Syahdewa Purba
NIM : 123200021

Menyatakan bahwa karya ilmiah saya yang berjudul:

IMPLEMENTASI METODE LEXICON BASED DAN SUPPORT VECTOR MACHINE (SVM) DALAM MENGANALISA SENTIMEN PUBLIK TERHADAP PARTAI POLITIK DI INDONESIA

Merupakan karya asli saya dan belum pernah dipublikasikan dimanapun. Apabila di kemudian hari, karya saya disinyalir bukan merupakan karya asli saya, maka saya bersedia menerima konsekuensi apa pun yang diberikan Program Studi Informatika Fakultas Teknik Industri Universitas Pembangunan Nasional “Veteran” Yogyakarta kepada saya.

Demikian surat pernyataan ini saya buat dengan sebenarnya.

Dibuat di :

Pada tanggal :

Yang menyatakan

Reytama Syahdewa Purba
NIM. 123200021

PERNYATAAN BEBAS PLAGIASI

Saya yang bertanda tangan di bawah ini :

Nama : Reytama Syahdewa Purba
NIM : 123200021
Fakultas/Prodi : Teknik Industri/Informatika

Dengan ini saya menyatakan bahwa judul Tugas Akhir

IMPLEMENTASI METODE LEXICON BASED DAN SUPPORT VECTOR MACHINE (SVM) DALAM MENGANALISA SENTIMEN PUBLIK TERHADAP PARTAI POLITIK DI INDONESIA

Adalah hasil kerja saya sendiri dan benar bebas dari plagiasi kecuali cuplikan serta ringkasan yang terdapat di dalamnya telah saya jelaskan sumbernya (Sitasi) dengan jelas. Apabila pernyataan ini terbukti tidak benar maka saya bersedia menerima sanksi sesuai peraturan Mendiknas RI No. 17 Tahun 2010 dan Peraturan Perundang-undangan yang berlaku.

Demikian surat pernyataan ini saya buat dengan penuh tanggung jawab.

Yogyakarta,
Yang menyatakan

Reytama Syahdewa Purba
NIM. 123200021

ABSTRAK

Komentar masyarakat yang kerap muncul di media sosial, sering kali memicu konflik serta perdebatan yang dapat berujung pada permusuhan di antara para pengguna. Hal ini bisa menjadi dampak yang merugikan, terutama jika terus dibiarkan. Dalam konteks komentar masyarakat terhadap partai politik, situasi ini menciptakan risiko ketegangan sosial yang lebih besar, yang sebaiknya dikelola atau diatasi untuk mencegah dampak negatif lebih lanjut. Analisis sentimen menjadi jalan keluar yang bisa dilakukan untuk mengenali variasi dan *pattern* yang sering mengakibatkan mispersepsi dari komentar masyarakat terhadap partai politik pada sosial media X. Dimana Jejaring sosial seperti Twitter, umumnya digunakan dalam analisis sentimen untuk menentukan persepsi publik. Penelitian ini menggunakan metode *lexicon based* yang dikombinasikan dengan metode *Support Vector Machine* (SVM). Analisis sentimen pada penelitian ini terdiri dari tiga kelas yaitu kelas positif, kelas negatif, dan kelas netral. Model yang telah dibangun dilakukan pengujian dengan tabel *confusion matrix*. Dari tabel *confusion matrix* tersebut akan diperoleh nilai akurasi, presisi, serta *recall*.

Hasil pengujian menunjukkan bahwa metode SVM dalam menganalisis sentimen publik memberikan kinerja yang cukup baik dalam memprediksi sentimen, baik positif maupun negatif, terhadap partai politik di Indonesia. Dengan demikian, penelitian ini berhasil menunjukkan bahwa metode yang digunakan dapat mencapai hasil yang cukup optimal merujuk pada hasil evaluasi model pada penelitian ini mencapai tingkat akurasi sebesar 82,11%, dengan nilai *recall* 82,11% dan *precision* 82,52% dengan menggunakan kernel *Radial Basis Function* (RBF) sebagai kernel dengan hasil akurasi paling optimal dibandingkan kernel lainnya yaitu Linear, Sigmoid, dan juga Polynomial setelah dilakukannya proses *gridsearch* untuk mencari parameter dan kernel terbaik.

Kata kunci : Partai Politik, Analisis Sentimen, *Lexicon Based*, *Support Vector Machine*.

KATA PENGANTAR

Bismillahirrahmanirahim, Bismillahirrahmanirrahim, segala puji dan syukur penulis haturkan kepada Allah SWT atas rahmat dan hidayah-Nya yang tiada henti, sehingga penulis dapat menyelesaikan penelitian dengan judul “Implementasi Metode *Lexicon Based* dan *Support Vector Machine* (SVM) Dalam Menganalisis Sentimen Publik Terhadap Partai Politik di Indonesia”. Selama penyusunan tugas akhir ini, penulis menyadari banyak sekali pihak yang sudah berkontribusi dalam memberikan dukungan baik dalam bentuk dukungan moril maupun dukungan materil, sehingga penulis mampu menyelesaikan tugas akhir ini dengan lancar. Maka penulis mengucapkan terimakasih yang sebesar - besarnya kepada :

1. Allah SWT yang senantiasa memberikan ketabahan, ketenangan, Kesehatan, kemudahan berfikir dan Kesehatan dalam proses penyusunan tugas akhir ini.
2. Papa dan mama, sebagai kedua orangtua penulis yang selalu ingin menunaikan kewajibannya untuk senantiasa memberikan nasihat dan dukungan yang tidak bisa dihitung lagi jumlahnya, serta selalu meyakinkan bahwa setiap proses dalam hidup ini adalah salah satu bagian dalam proses pendewasaan.
3. Bapak Dr. Heriyanto, A.Md, S.Kom, M.Cs. selaku dosen pembimbing yang telah memberikan kemudahan, arahan dan bimbingannya dengan penuh ketegasan dan tanggung jawab agar penulis bisa untuk segera menyelesaikan masa studinya sesegera mungkin.
4. Seluruh dosen, karyawan, dan staff tata usaha prodi Informatika UPN “Veteran” Yogyakarta yang telah mendukung dan memberikan bantuan kepada penulis selama menempuh pendidikan.
5. Ahlida Sabila Larasati, wanita yang sampai saat ini menemani dan mendampingi penulis dalam setiap tahapan dan proses dalam menyusun tugas akhir yang senantiasa sabar untuk membantu dan memberikan dukungan positif sehingga penulis bisa menyelesaikan tugas akhir ini.
6. Saudara Muhammad Hafizh Akmal, merupakan kerabat tak sedarah penulis yang tak bosan – bosan untuk memberikan kritikan saran untuk penulis selama menyelesaikan proses perkuliahan sehingga penulis bisa *touch down* sampai titik ini.
7. Saudara Rifqi Maulana, merupakan kerabat tak sedarah penulis yang senantiasa mengarahkan penulis untuk selalu memberikan yang terbaik dalam penyusunan tugas akhir ini agar bisa menyelesaikan tugas akhir ini dengan penuh tanggung jawab.
8. Saudara Hugo Vale Poetratama, merupakan kerabat tak sedarah penulis yang selalu memberikan semangat, keceriaan, dan energi positif baik dalam proses penyusunan tugas akhir ini ataupun selama penulis menempuh pendidikan selama perkuliahan.
9. Saudari Nurhikmahwati Husni, yang telah berbaik hati untuk memberikan penulis arahan pada saat sedang tidak memiliki arah dalam proses penyusunan tugas akhir ini.
10. Rekan – rekan Informatika Angkatan 2020 yang tidak bisa penulis ucapkan semuanya, terimakasih yang sebesar – besarnya penulis ucapkan atas semua waktu, proses, dan semua pengorbanannya selama penulis menempuh masa kuliah, semua memori tersebut akan selalu menjadi *core memory* bagi penulis dan akan selalu menjadi bahan evaluasi untuk penulis.

Dalam penyusunan Laporan Tugas Akhir ini masih terdapat beberapa kekurangan di dalamnya, oleh karena itu saran dan kritik yang bersifat membangun dari semua pihak sangat diharapkan, tidak lupa harapan penulis semoga Laporan Tugas Akhir ini dapat bermanfaat bagi pembaca serta dapat menambah ilmu pengetahuan.

Yogyakarta,

Penulis

DAFTAR ISI

COVER.....	i
HALAMAN PENGESAHAN PEMBIMBING.....	ii
HALAMAN PENGESAHAN PENGUJI.....	iii
SURAT PERNYATAAN KARYA ASLI TUGAS AKHIR.....	iv
PERNYATAAN BEBAS PLAGIASI.....	v
ABSTRAK.....	vi
KATA PENGANTAR.....	vii
DAFTAR ISI.....	ix
DAFTAR GAMBAR.....	xi
DAFTAR TABEL.....	xii
BAB 1 PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	3
1.3 Batasan Masalah.....	3
1.4 Tujuan Penelitian.....	4
1.5 Manfaat Penelitian.....	4
1.6 Metodologi Penelitian dan Pengembangan Sistem.....	4
1.6.1 Metode Penelitian.....	4
1.6.2 Metode Pengembangan Sistem.....	5
1.7 Sistematika Penulisan.....	6
BAB II TINJAUAN LITERATUR.....	7
2.1 Analisis Sistem.....	7
2.2 <i>X</i>	7
2.3 Analisis Sentimen.....	7
2.4 Data Preprocessing.....	8
2.4 <i>Scrapping</i>	9
2.5 <i>Term Frequency-Invers Document Frequency (TF-IDF)</i>	9
2.6 Support Vector Machine.....	10
2.7 Lexicon Based.....	13
2.8. Evaluasi Model.....	13
2.9 Studi Pustaka.....	15
BAB III METODOLOGI PENELITIAN DAN PENGEMBANGAN SISTEM.....	21
3.1 Metodologi Penelitian.....	21
3.1.1 Pengumpulan Data.....	21

3.1.2 <i>Preprocessing</i>	22
3.1.3 <i>Labeling</i>	28
3.1.4 Pembobotan TF-IDF	29
3.1.5 Analisis Sentimen dengan Model SVM	35
3.1.6 Pembuatan Model Sentimen	39
3.1.7 Pengujian	40
3.2 Metode Pengembangan Sistem	41
3.2.1 <i>Requirements Analysis</i>	41
3.2.2 <i>System And Software Design</i>	42
3.2.3 Rancangan Interface	43
3.2.4 <i>Implementation</i>	46
3.2.5 <i>System Testing</i>	46
BAB IV HASIL DAN PEMBAHASAN	48
4.1 Impelementasi	48
4.1.1 Implementasi Model Klasifikasi	48
4.1.2 Implementasi Pengembangan Sistem	60
4.2 Hasil.....	64
4.2.1 Proses Pengujian Model	64
4.2.2 Proses Evaluasi Model.....	67
4.2.3 Pengujian Pengembangan Sistem	68
4.3 Pembahasan	69
BAB V KESIMPULAN DAN SARAN	71
5.1 Kesimpulan.....	71
5.2 Saran.....	71
DAFTAR PUSTAKA	72
LAMPIRAN.....	74

DAFTAR GAMBAR

Gambar 2. 1 SVM dalam mencari <i>hyperplane</i> optimal (Nugroho et al., 2003)	10
Gambar 3. 1 Metodologi Penelitian	21
Gambar 3. 2 <i>Flowchart Preprocessing</i>	23
Gambar 3. 3 <i>Flowchart Cleansing</i>	23
Gambar 3. 4 <i>Flowchart Case Folding</i>	24
Gambar 3. 5 <i>Flowchart Kamus Slang</i>	25
Gambar 3. 6 <i>Flowchart Negasi</i>	26
Gambar 3. 7 <i>Flowchart Stopword Removal</i>	27
Gambar 3. 8 <i>Flowchart Stemming</i>	28
Gambar 3. 9 <i>Flowchart TF-IDF</i>	30
Gambar 3. 10 <i>Flowchart Klasifikasi Model SVM</i>	35
Gambar 3. 11 <i>Flowchart Pembuatan Model Sentimen</i>	40
Gambar 3. 12 <i>Arsitektur Sistem</i>	43
Gambar 3. 13 <i>Rancangan Halaman Beranda</i>	44
Gambar 3. 14 <i>Rancangan Halaman Dataset</i>	44
Gambar 3. 15 <i>Rancangan Halaman Dataset After Sentiment Preprocessing</i>	45
Gambar 3. 16 <i>Rancangan Dataset After Sentiment Labelling</i>	45
Gambar 3. 17 <i>Rancangan Halaman Evaluasi</i>	46
Gambar 3. 18 <i>Rancangan Halaman Prediksi</i>	46
Gambar 4. 1 <i>Halaman Beranda</i>	61
Gambar 4. 2 <i>Halaman Dataset Fresh</i>	61
Gambar 4. 3 <i>Halaman Dataset After Preprocessing</i>	62
Gambar 4. 4 <i>Halaman Dataset Labelling</i>	62
Gambar 4. 5 <i>Halaman Evaluasi</i>	63
Gambar 4. 6 <i>Halaman Prediksi</i>	64

DAFTAR TABEL

Tabel 2. 1 Confusion Matrix	14
Tabel 2. 2 Studi Pustaka.....	15
Tabel 2. 3 Lanjutan Studi Pustaka	16
Tabel 2. 4 Lanjutan Studi Pustaka	17
Tabel 2. 5 Lanjutan Studi Pustaka	18
Tabel 2. 6 Lanjutan Studi Pustaka	19
Tabel 2. 7 Lanjutan Studi Pustaka	20
Tabel 3. 1 Kata Kunci X	22
Tabel 3. 2 Contoh Hasil <i>Cleansing</i>	24
Tabel 3. 3 Contoh Hasil <i>Case Folding</i>	24
Tabel 3. 4 Contoh Hasil Penerapan Kamus Slang	25
Tabel 3. 5 Contoh Hasil Negasi	26
Tabel 3. 6 Contoh Hasil <i>Stopword Removal</i>	27
Tabel 3. 7 Contoh Hasil <i>Stemming</i>	28
Tabel 3. 8 Contoh Hasil Pelabelan Data	29
Tabel 3. 9 Contoh Dokumen Perhitungan.....	31
Tabel 3. 10 Contoh Hasil Perhitungan TF	31
Tabel 3. 11 Lanjutan Contoh Hasil Perhitungan Nilai TF	32
Tabel 3. 12 Contoh Hasil Perhitungan Nilai IDF	32
Tabel 3. 13 Lanjutan Contoh Hasil Perhitungan Nilai IDF	33
Tabel 3. 14 Contoh Hasil Perhitungan TF-IDF	33
Tabel 3. 15 Contoh Hasil Perhitungan TF-IDF	34
Tabel 3. 16 Nilai Matriks K	36
Tabel 3. 17 Nilai Matriks Hessian	36
Tabel 3. 18 Hasil Perhitungan E_i	36
Tabel 3. 19 Hasil Perhitungan $\delta\alpha_i$	36
Tabel 3. 20 Tabel Perhitungan α_i baru	37
Tabel 3. 21 Hasil Perhitungan E_i iterasi ke-2	37
Tabel 3. 22 Hasil Perhitungan $\delta\alpha_i$ Iterasi ke-2	37
Tabel 3. 23 Hasil Perhitungan α_i baru	38
Tabel 3. 24 Nilai Perhitungan Data Uji	39
Tabel 3. 25 Tabel Rancangan Pengujian.....	41
Tabel 3. 26 Kebutuhan <i>Hardware</i>	42
Tabel 3. 27 Kebutuhan <i>Software</i>	42
Tabel 3. 28 Rancangan Pengujian <i>Black Box</i>	47
Tabel 4. 1 Contoh Hasil Pelabelan Manual.....	54
Tabel 4. 2 Lanjutan Contoh Hasil Pelabelan Manual	55
Tabel 4. 3 <i>Training Labeling Lexicon Based</i>	64
Tabel 4. 4 <i>Training Labelling Manual</i>	65
Tabel 4. 5 <i>Splitting Data</i>	65
Tabel 4. 6 Hasil Sentimen Data <i>Training</i>	65
Tabel 4. 7 Hasil Sentimen Data <i>Testing</i>	65
Tabel 4. 8 Inisiasi Parameter.....	66
Tabel 4. 9 Hasil Pelatihan Model SVM	66
Tabel 4. 10 <i>Training Kernel SVM</i>	66
Tabel 4. 11 Hasil Pengujian <i>Confusion Matrix</i>	67
Tabel 4. 12 Hasil Pengujian <i>Black Box</i>	69

BAB 1 PENDAHULUAN

1.1 Latar Belakang

Sosial media memiliki peranan penting dalam proses sebaran informasi harian yang saat ini semakin massif digunakan oleh manusia saat ini. Sebagai salah satu sosial media yang penggunaannya mencapai 24,85 Juta per April tahun 2024 di Indonesia, X yang dulunya bernama Twitter menjadi salah satu raksasa sosial media yang penggunaannya banyak menjadikannya sebagai salah satu opsi penyebaran informasi yang cukup efektif dan efisien berdasarkan riset oleh Radio Republik Indonesia (2024). Munculnya komentar masyarakat terhadap isu yang sedang hangat dibicarakan kerap terjadi karena adanya konten atau wacana yang dibentuk dan disebarluaskan melalui media sosial, terutama dalam konteks pembahasan partai politik. Ketika mosi tersebut dibagikan, banyak pengguna media sosial merasa terpengaruh untuk memberikan tanggapan yang bisa bersifat emosional. Komentar-komentar ini sering kali mencerminkan sikap atau pendapat mereka terhadap partai politik. (Sapar, 2024).

Komentar masyarakat yang kerap muncul di media sosial, sering kali memicu konflik serta perdebatan yang dapat berujung pada permusuhan di antara para pengguna. Hal ini bisa menjadi dampak yang merugikan, terutama jika terus dibiarkan. Dalam konteks komentar masyarakat terhadap partai politik, situasi ini menciptakan risiko ketegangan sosial yang lebih besar, yang sebaiknya dikelola atau diatasi untuk mencegah dampak negatif lebih lanjut (Adam, 2024). Analisis sentimen menjadi jalan keluar yang bisa dilakukan untuk mengenali variasi dan pattern yang sering mengakibatkan mispersepsi dari komentar masyarakat terhadap partai politik pada sosial media X yang dimana Jejaring sosial seperti *Twitter*, umumnya digunakan dalam analisis sentimen untuk menentukan persepsi publik (Samsir, et al., 2021).

Dilakukan penelitian dengan judul Analisis Sentimen Masyarakat pada Media Sosial *Twitter* Terhadap Partai Politik Peserta Pemilu 2019 Menggunakan *Naïve Bayes Classifier* (Aprillia et, al., 2019) yang penelitian tersebut memiliki tujuan untuk dapat mengetahui tingkat akurasi metode *Naïve Bayes Classifier* dalam mengklasifikasikan sentimen masyarakat terhadap partai politik peserta pemilu 2019, serta dapat mengidentifikasi sentimen masyarakat terhadap partai politik peserta pemilu 2019. Nilai akurasi yang berhasil didapat kan dengan mengimplementasikan metode *Naïve Bayes* sebagai metode klasifikasinya adalah 78.03%. Selanjutnya pernah dilakukan penelitian dengan menerapkan metode *supervised learning* lainnya dengan jumlah data 2105 tweet yang bertujuan untuk membandingkan metode SVM dan juga KNN dengan menganalisa sentimen publik terhadap vaksin *Sinovac* dengan mendapatkan akurasi model SVM setinggi 73% dan metode KNN sebesar 60% (Anna, et al., 2021). Penelitian lain yang membandingkan metode supervised learning dilakukan oleh Fajar Sodik Pamungkas dengan hasil SVM memiliki akurasi yang lebih tinggi daripada *Naive Bayes* dan KNN dengan rata-rata akurasinya sebesar 90,01% pada SVM dengan kernel linear, 79,20% pada Naive. Dalam analisis sentimen, proses pengolahan dataset agar siap uji tentu memerlukan tahapan – tahapan yang mewajibkan data tersebut harus sesuai dengan kebutuhan output nantinya. Pelabelan sentimen merupakan tahapan yang tidak bisa untuk lewati. Salah satu metode pelabelan yang sering dilakukan dalam analisis sentimen adalah pelabelan manual. Namun, pelabelan secara manual cenderung sulit terlepas dari subjektivitas dari sipelabel yang mengakibatkan kondisi data setelah bisa saja terpengaruh oleh *human error*. Sehingga

diperlukan pendekatan lain untuk mengatasi kekurangan tersebut. Salah satu cara untuk menyelesaikan proses pelabelan sentimen agar lebih kredibel adalah dengan pengimplementasian kamus *Lexicon*. Pernah dilakukan penelitian terhadap partai politik peserta pemilu di Indonesia yang pada penelitian tersebut menggunakan ini menggunakan pendekatan *lexicon resource* berbahasa Indonesia sehingga memiliki keunggulan tidak perlu melakukan translasi kedalam Bahasa Inggris (Fanhar, 2019). Selain itu juga penerapan kamus *lexicon* juga memiliki keunggulan dalam menganalisis komentar singkat pada proses prediksi dikarenakan data inputan belum tentu ada label sentimennya (Arsya, et al., 2019). Menambahkan metode *lexicon based*, pada sisi pelabelan dapat membuat proses pelabelan dataset *support vector machine* tidak lagi dilakukan secara manual melainkan diproses oleh metode *lexicon based* dan *dictionary* pada *lexicon* dapat ditambahkan seiring dengan perkembangan konten data pada media sosial x. Dengan begitu proses transfer dataset pada *support vector machine* akan menjadi lebih efisien dengan akurasi pelabelan mencapai 90% dengan catatan data *dictionary* pada setiap kelas sesuai dengan arti dan esensi yang ada pada Kamus Besar Bahasa Indonesia. (Danar, 2019).

Penelitian lain juga pernah dilakukan oleh Natasya Eldha Oktaviana, yang menguji kemampuan metode *lexicon based* untuk melakukan pelabelan data dengan hasil evaluasi model 68% accuracy, 56% nilai precision, 75% nilai recall dan 64% nilai f-measure. Sementara tanpa *lexicon* didapatkan 48% nilai accuracy, 46% nilai precision, 58% nilai recall, dan 52% nilai f-measure. Dari penelitian tersebut secara keseluruhan nilai paling tinggi didapatkan oleh metode pelabelan dengan menggunakan metode *lexicon based*. Rindu Hafil Muhammadi dengan judul *Combination of Support Vector Machine and Lexicon Based Algorithm in Twitter Sentiment Analysis* tahun 2022 mendapatkan hasil akurasi 81.73% yang dimana penelitian tersebut dilakukan dengan pengimplementasian metode *lexicon based* sebagai metode pelabelan.

Sebagai perbandingan dengan metode pelabelan lain, penelitian dilakukan oleh Yessy Asri, tahun 2022, yang menerapkan metode *vader* sebagai metode pelabelan yang mendapatkan nilai akurasi model sebanyak 70%. Dalam penelitian dilakukan perbandingan metode pelabelan antara metode *TextBlob* dan juga metode *Vader* dan didapatkan hasil 63,94% untuk sentimen negatif dan 36,05% sentimen positif untuk metode *TextBlob* dan berhasil 56,28% sentimen negatif dan 43,71 untuk sentimen positif untuk metode *Vader*. Sehingga dapat disimpulkan metode *lexicon based* dalam proses pelabelan memiliki performa yang lebih baik dibandingkan metode pelabelan lainnya.

Dalam proses analisis sentimen beberapa metode sudah pernah diterapkan untuk melakukan klasifikasi dan prediksi sentimen dengan metode pelatihan data berlabel (*supervised learning*). Penelitian yang sama pernah dilakukan Bayes dan 62,10% pada KNN dengan jumlah K adalah 20 dan menggunakan kernel optimal.

Penelitian lain yang menunjukkan optimalnya performa metode klasifikasi dengan metode *Support Vector Machine (SVM)* dilakukan dengan judul Analisis Sentimen Terhadap Penerapan Sistem E-Tilang Pada Media Sosial Twitter (Dea et, al.,2023), penelitian tersebut mendapatkan hasil dengan kesimpulan pengimplementasi algoritma *support vector machine (SVM)* pada penelitian ini menunjukkan tingkat keakurasian sebesar 74.20%, precision sebesar

83.33% dan recall 5.28%. Penelitian lain dengan judul Analisis Sentimen Terhadap Bantuan Langsung Tunai (BLT) Bahan Bakar Minyak (BBM) Menggunakan *Support Vector*

Machine didapatkan hasil klasifikasi sentimen menggunakan metode SVM dengan tingkat accuracy 85,98%, rata-rata nilai precision 82,25%, nilai rata-rata recall 66,35%, dan nilai rata-rata i 73,44%. (Rizky et, al., 2023).

Dari penelitian – penelitian sebelumnya, metode support vector machine memiliki hasil yang lebih baik dibandingkan metode lainnya. Mengingat SVM memiliki banyak kernel dan parameter yang bisa dicoba untuk mengetahui akurasi model yang paling optimal. Adapun cara yang bisa digunakan untuk mencari parameter terbaik adalah dengan menerapkan metode Searchgrid, yang bertujuan untuk menemukan hyper parameter yang paling optimal dengan cara mencoba berbagai kombinasi parameter. (Fatimah, 2021). Dalam penerapannya, algoritma Searchgrid akan bekerja sama dengan teknik cross validation pada data training sebagai metrik kinerja yang berguna untuk menghindari classifier menghasilkan model prediksi yang overfitting. Hal ini bertujuan untuk menentukan kombinasi dari tiap hyper parameter yang memberikan hasil optimal untuk classifier dan membuat classifier dapat melakukan prediksi pada data testing dengan akurat (Trianto, 2022).

Dengan penelitian - penelitian dan uraian diatas dapat disimpulkan bahwa penelitian dengan metode pelabelan lexicon based mendapatkan hasil yang lebih tinggi dibandingkan metode pelabelan lainnya. Serta penerapan metode pelatihan data berlabel (supervised learning) menjadikan metode support vector machine menjadi metode pengklasifikasi diskriminatif dengan Hyperplane pemisah (Arief, 2019) yang bisa digunakan untuk melakukan klasifikasi sentimen dengan hasil yang cukup baik. Serta penelitian yang dilakukan dengan menerapkan metode support vector machine didapatkan hasil yang lebih baik jika dibandingkan dengan metode supervised learning lainnya dalam penanganan analisis sentimen dengan hasil klasifikasi multiclass. Hal tersebut merupakan pertimbangan serta alasan mengapa pada penelitaian kali ini akan dilakukan analisa sentimen dengan penerapan metode *Lexicon Based* dan juga *Metode Support Vector Machine (SVM)* terhadap opini masyarakat pada social media X dalam memandang partai politik di Indonesia.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan, dan mempertimbangkan penelitian – penelitian sebelumnya, rumusan masalah pada penelitian ini adalah bagaimana penerapan metode *Lexicon Based* sebagai pelabelan dan metode *Support Vector Machine (SVM)* sebagai metode untuk melakukan klasifikasi dapat digunakan untuk menganalisis sentimen publik terhadap partai politik di Indonesia, serta sejauh mana metode *Support Vector Machine (SVM)* sebagai metode klasifikasi mampu menghasilkan akurasi yang optimal dalam proses analisis sentimen.

1.3 Batasan Masalah

Agar batasan masalah yang diteliti memiliki arah dan fokus serta menghasilkan sasaran yang jelas, maka dibuatlah batasan masalah untuk menghindari adanya perluasan pembahasan kedepannya adalah sebagai berikut :

1. Data penelitian yang digunakan merupakan data yang dihasilkan dari sosial media X dengan cara *scrapping*, skala cakupan *keyword* “Partai Politik, parpol, partai indonesia, partai dan tagar #parpol”, yang di *crawl* pada sosial media X dimulai pada tahun 2020 sampai dengan tahun 2024.

2. Data yang digunakan dalam penelitian memakai format .csv
3. Kelas untuk klasifikasi sentimen yaitu dengan 3 kelas yaitu kelas positif, kelas negatif serta kelas netral.
4. Jumlah data yang digunakan adalah 5111 Data, dengan rasio pembagian data nantinya menggunakan skala perbandingan, 90% Data *training* dan 10% data *testing*.
5. Metode pembobotan kata yang digunakan adalah metode *Term Frequency – Inverse Document Frequency* (TF-IDF).
6. Metode klasifikasi yang digunakan yaitu metode *Support Vector Machine* (SVM) dengan menggunakan *Lexicon Based* sebagai metode pelabelan.

1.4 Tujuan Penelitian

Tujuan penelitian ini adalah untuk menerapkan metode *Lexicon Based* sebagai pelabelan sentimen dan *Support Vector Machine* (SVM) sebagai metode klasifikasi untuk menganalisis sentimen publik terhadap partai politik di Indonesia. Serta, untuk mengetahui kemampuan metode *Support Vector Machine* (SVM) dalam menghasilkan akurasi yang optimal.

1.5 Manfaat Penelitian

Dengan dilakukannya penelitian ini, diharapkan memiliki manfaat untuk beberapa aspek kehidupan, yaitu :

1. Mengetahui pengaruh dalam penerapan Metode *Lexicon Based* dan Metode *Support Vector Machine* dalam menyelesaikan permasalahan analisis sentimen opini publik terhadap partai politik di Indonesia.
2. Menjadikan bahan evaluasi untuk lembaga atau instansi terkait, mengenai pandangan publik terhadap partai politik di Indonesia
3. Menjadi referensi pengguna *Lexicon Based*, dan metode *Support Vector Machine* untuk analisis sentimen pada penelitian selanjutnya

1.6 Metodologi Penelitian dan Pengembangan Sistem

Metode penelitian ini menggunakan metode penelitian kuantitatif. Metode penelitian kuantitatif merupakan penelitian empiris dimana data dalam bentuk sesuatu yang dapat dihitung atau angka (Punch, 1988). Berikut merupakan tahapan-tahapan penelitian yang dilakukan :

1.6.1 Metode Penelitian

2. Pengumpulan Data
Pada penelitian ini, data yang akan digunakan merupakan data yang didapatkan dari proses crawling data dari Aplikasi X. Data yang telah di crawling akan disimpan dalam bentuk .csv.
3. *Preprocessing* Data
Text preprocessing bertujuan untuk mengubah bentuk data tidak terstruktur menjadi data terstruktur. Pada tahapan ini akan dilakukan *preprocessing* data termasuk penghapusan *case folding*, *cleansing*, pengimplemntasian kamus slang, *negation handling*, *stopwords removal*, dan *stemming*.
4. Pelabelan

Data yang sudah dilakukan tahap *preprocessing* kemudian dilakukan pelabelan sentimen antara kelas positif dan negatif dengan mengimplementasikan pelabelan dengan *Lexicon*.

5. Ekstraksi Fitur

Pada tahap ekstraksi fitur akan dilakukan pembobotan kata dengan menggunakan metode TF-IDF, pada tahap ini implementasi sistem yang akan dibuat menggunakan Bahasa pemrograman Python.

6. Klasifikasi SVM

Pembuatan model metode SVM yang digunakan dan menguji performanya untuk mendapatkan hasil analisis sentiment menggunakan *library python* yaitu *Scikit-learn*.

7. Pengujian

Pada tahap pengujian, model yang telah dikembangkan akan diuji kinerjanya dengan menggunakan *confusion matrix* untuk menentukan *Accuracy, Precision, Recall*.

1.6.2 Metode Pengembangan Sistem

Metode *Waterfall* adalah salah satu model dalam SDLC yang sering diterapkan dalam pengembangan perangkat lunak atau sistem informasi. Model ini mengadopsi pendekatan yang terstruktur dan berurutan, di mana setiap tahap dilakukan secara sistematis. Proses dimulai dari tahap perencanaan hingga tahap pemeliharaan (*maintenance*), dengan setiap langkah diselesaikan sebelum melanjutkan ke langkah berikutnya. (Abdul Wahid, 2020). Tahapan metode *waterfall* adalah sebagai berikut:

1. *Requirements Analysis and Definition*

Merupakan tahapan awal yang melibatkan identifikasi dan pemahaman yang mendalam terhadap kebutuhan. Tujuan utamanya yaitu mengumpulkan persyaratan fungsional dan non-fungsional yang nantinya akan menjadi dasar dari pengembangan sistem.

2. *System and Software Design*

Tahapan perancangan sistem ini mengalokasikan kebutuhan-kebutuhan sistem pada perangkat keras maupun perangkat lunak dengan memberntuk arsitektur sistem secara keseluruhan.

3. *Implementation*

Pada tahap ini, perancangan perangkat lunak direalisasikan sebagai serangkaian program.

4. *System Testing*

Merupakan tahap pengujian terhadap sistem yang telah dibuat yang bertujuan untuk mengetahui apakah sistem yang dibuat udah siap digunakan atau belum.

1.7 Sistematika Penulisan

Sistematika penulisan pada penelitian ini adalah sebagai berikut :

BAB I PENDAHULUAN

Pada bagian pendahuluan membahas mengenai latar belakang masalah, perumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian, serta sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Tinjauan Pustaka merupakan bagian yang memuat mengenai dasar teori yang digunakan untuk analisis serta perancangan sistem dan juga implementasi pada penelitian ini. Selain itu juga digunakan sebagai bahan referensi serta pondasi untuk memperkuat argumentasi pada penelitian ini.

BAB III METODOLOGI PENELITIAN DAN PENGEMBANGAN SISTEM

Pada bab ini membahas mengenai metodologi penelitian, analisis sistem dan perancangan sistem analisis sentimen.

BAB IV HASIL, PENGUJIAN, DAN PEMBAHASAN

Bab ini menyajikan hasil dari penelitian yang berisi hasil implementasi dari perancangan yang telah dibuat pada bab sebelumnya. Selain itu berisi pengujian terhadap hasil penelitian beserta pembahasannya.

BAB V KESIMPULAN DAN SARAN

Bab ini berisi kesimpulan dari hasil penelitian serta saran yang diajukan oleh penulis untuk pengembangan pada penelitian selanjutnya.

BAB II

TINJAUAN LITERATUR

2.1 Analisis Sistem

Dalam menganalisis sebuah sistem, dibutuhkan data yang akan digunakan sebagai bahan analisis penelitian berupa data-data sentimen yang diambil dari media sosial X. Data sentimen ini diperoleh dengan cara *crawling* data yang ada pada media sosial X yang memanfaatkan web scraping, proses *crawling* data nantinya akan berdasarkan *keyword* pencarian “Partai Politik” pada sosial media X.

2.2 X

Twitter menjadi layanan penghubung antar setiap orang. Setiap orang bisa bercerita tentang apapun mengenai aktivitas dan cara pandangnya. Pengguna lain siapapun itu dapat menyimak informasi tersebut secara kontinu. Dalam hal sosial politik, twitter juga ampuh digunakan sebagai media mengeluarkan pendapat atau melakukan kampanye (Aprillia, 2019). Sebagai salah satu media sosial yang sering digunakan sehari-hari, X yang dulunya bernama twitter kini telah berkembang menjadi sumber informasi yang kaya dan beragam. Twitter telah hampir memiliki 600 juta pengguna dan lebih dari 250 juta pesan per hari, dan dengan cepat akan menjaditambang data bagi organisasi untuk mendapatkan informasi mengenai reputasi dan merk mereka dengan mengekstraksi dan menganalisis sentimen melalui tweet para pengguna X (Satrio et al., 2019).

2.3 Analisis Sentimen

Analisis sentimen merupakan salah satu metode yang digunakan untuk memahami opini, emosi, atau evaluasi dari teks. Menurut Liu (2012) analisis sentimen bertujuan untuk menentukan sentimen yang terkandung dalam teks, yang dapat diklasifikasikan menjadi kategori positif, negatif, atau netral. Proses ini melibatkan pemrosesan bahasa alami (NLP) untuk menginterpretasikan makna dari kata-kata dan konteks di sekitarnya. Liu juga mencatat bahwa analisis sentimen dapat dilakukan dengan berbagai pendekatan, termasuk pendekatan berbasis kamus (*dictionary-based*) dan pendekatan berbasis machine learning.

Analisis sentimen sebagai salah satu pemrosesan bahasa alami yang digunakan untuk memantau pendapat publik mengenai produk atau topik tertentu. Juga dikenal sebagai penggalian opini, analisis sentimen membangun sistem untuk mengumpulkan dan menganalisis opini yang diungkapkan melalui posting blog, komentar, ulasan, atau tweet (Vinodini, 2012). Secara garis besar analisis sentimen itu sendiri bertujuan untuk menentukan perilaku atau opini dari seseorang dengan memperhatikan suatu topik tertentu, perilaku tersebut bisa mengindikasikan alasan, opini, maupun kondisi kecenderungan (Permatasari et al., 2021).

Analisis sentimen didefinisikan juga sebagai opini, perasaan emosi, dan penilaian positif atau negatif (Manek et al., 2017). Analisis sentimen adalah sebuah riset secara komputasi pada opini atau emosi yang diekspresikan secara tekstual oleh seseorang atau kelompok (Vinodhini, 2012). Analisis sentimen merupakan turunan proses dari text mining, yang secara khusus melakukan analisis dokumen teks atau data tekstual. Umumnya proses analisis dokumen teks akan mengelompokkan sentimen dalam wujud positif, negatif maupun netral. Dengan melakukan analisis sentimen bisa ditemukan sebuah informasi yang berharga dari sebuah data yang tidak terstruktur, dimana informasi yang diberikan

berupa opini tersebut dapat diklasifikasi ke dalam sentimen positif atau negatif berdasarkan emosional dari opini yang diberikan (Bagus, 2024)

2.4 Data Preprocessing

Tahapan *preprocessing* merupakan tahapan awal untuk mempersiapkan dokumen agar lebih mudah untuk diproses, tahapan *preprocessing* sebelum proses klasterisasi meliputi *case folding*, *tokenizing*, *filtering*, dan *stemming* (Amalia et al., 2018). Tujuan dari *preprocessing* adalah untuk memastikan ketersediaan data yang akan digunakan pada proses klasifikasi sentimen telah bebas *noise*. Berdasarkan hal tersebut, mengubah data yang sebelumnya tidak terstruktur memerlukan proses perubahan menjadi data yang terstruktur untuk diproses pada langkah berikutnya. Berikut adalah tahapan dari *preprocessing*.

a. *Cleansing* dan *Case Folding*

Cleansing merupakan proses pembersihan data yang bertujuan untuk menghilangkan elemen-elemen yang tidak dibutuhkan pada sebuah dokumen. *Cleansing* bertujuan untuk memperbaiki kualitas data. Pada penelitian ini tahapan *cleansing* berfungsi untuk menghilangkan tanda baca, menghilangkan *emoticon*, serta menghilangkan angka-angka. Tujuan utama dilakukan proses *cleansing* adalah menghilangkan *noise*, mengingat teks asli yang diperoleh dari platform X sering mengandung banyak *noise*.

Case folding merupakan tahap mengubah karakter menjadi huruf kecil, di mana semua huruf kapital diubah menjadi huruf kecil. Selain itu, tahap pembersihan juga mencakup penghapusan angka, simbol, *emoticon* dan tanda baca yang bertujuan untuk mengurangi beban pemrosesan.

b. *Normalisasi*

Normalisasi merupakan tahapan untuk mengimplementasikan kamus slang dan kamus negasi sebagai parameter yang mengubah kalimat yang terdapat pada dataset. Seperti contohnya mengubah kata “gaje” menjadi “tidak jelas”. Dan kata “tidak” pada susunan frasa “tidak_jelas” akan dihapus, dan kata jelas akan dicari antonimnya yaitu “kabur”, “buram”

c. *Stopwords Removal*

Stopword removal merupakan proses untuk mengeliminasi kata-kata yang terdapat dalam daftar *stoplist* atau kamus *stopword*. Contoh kata yang termasuk dalam *stoplist* meliputi 'dan', 'dari', 'di', 'yang', dan sebagainya. Kata-kata ini dianggap tidak berfungsi dalam klasifikasi, sehingga dihapus (Dr. S. Vijayarani, Ms. J. Ilamathi, 2015). Kata-kata dalam *stoplist* umumnya tidak memiliki makna yang signifikan.

d. *Stemming*

Proses *stemming* bertujuan untuk mengubah kata menjadi bentuk dasar atau *root word*. Pada tahap ini, imbuhan seperti awalan (*prefiks*), akhiran (*sufiks*), dan sisipan dihilangkan. Imbuhan yang dihapus mencakup berbagai bentuk, seperti meng-, kan-, me-, di-, peng-, i-, an-, a-, dan lainnya. Dalam penelitian ini, algoritma yang digunakan untuk *stemming* dalam bahasa Indonesia adalah Algoritma Nazief dan Adriani, yang telah diimplementasikan dalam *library* Sastrawi. Proses ini memerlukan kamus yang berisi kata-kata dasar yang diperoleh dari *library* Sastrawi.

2.4 Scrapping

Web scraping merupakan sebuah teknik untuk melakukan ekstraksi data dan informasi berdasarkan suatu *website* yang disimpan dalam format tertentu (Aditama et al., n.d.). Teknik *scraping* memungkinkan konten utama yang terdapat pada situs dapat diekstraksi, dihimpun, kemudian dapat diproses. Scraping merupakan proses ekstraksi dari sebuah dokumen semi terstruktur pada internet, biasanya berupa halaman *web* yang ditulis menggunakan bahasa markup seperti *HTML* atau *XHTML*, yang kemudian dokumen tersebut dianalisis untuk mengekstrak data tertentu yang dapat dimanfaatkan untuk tujuan lain (Setiawan et al., 2020). Cara kerja *web scraping* adalah dengan mengakses halaman pada *web*, menentukan data yang dalam halaman tersebut, melakukan ekstraksi, dan transformasi bila diperlukan, kemudian menyimpan data tersebut menjadi dataset terstruktur (Boeing & Waddell, n.d.)

2.5 Term Frequency-Invers Document Frequency (TF-IDF)

K-Nearest TF-IDF atau Term Frequency Inverse Document Frequency merupakan metode pembobotan dengan menggabungkan metode TF dan IDF, metode ini memberikan bobot hubungan suatu kata terhadap dokumen (Wahyuni et al., 2017). Proses Frekuensi dokumen yang mengandung kata tersebut menunjukkan seberapa umum kata tersebut. Sehingga bobot hubungan antara sebuah kata dan sebuah dokumen akan tinggi apabila frekuensi kata tersebut tinggi di dalam dokumen dan frekuensi keseluruhan dokumen yang mengandung kata tersebut yang rendah pada kumpulan dokumen. Langkah-langkah pembobotan TF-IDF tertera pada persamaan dibawah,

1. Menghitung *term frequency* $tf_{t,d}$
2. Menghitung *weighting term frequency* (W_{tf})

$$W_{tf_{t,d}} = \begin{cases} 1 + \log 10tf_{t,d}, & \text{if } tf_{t,d} > 0 \\ 0, & \text{otherwise} \end{cases} \dots\dots\dots(2.1)$$

3. Menghitung *document frequency* (df)
4. Menghitung *inverse document frequency* (idf)

$$idf_t = \log\left(\frac{D}{df_t}\right) \dots\dots\dots(2.2)$$

5. Menghitung nilai bobot TF-IDF

$$W_{t,d} = W_{tf_{t,d}} \times idf_{d,t} \dots\dots\dots(2.3)$$

Keterangan:

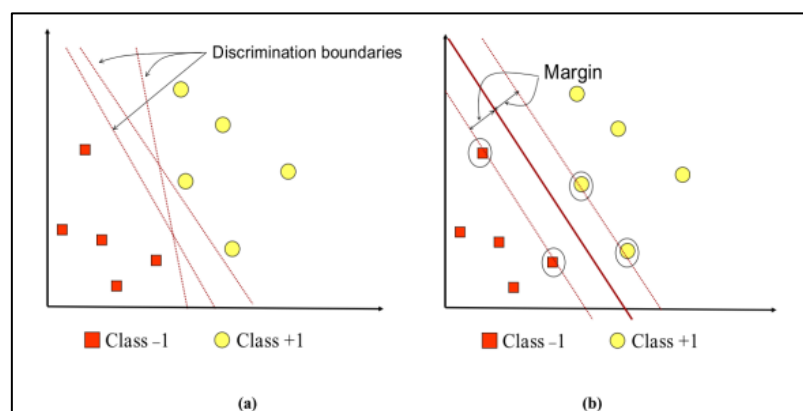
- $tf_{t,d}$: frekuensi term
- $W_{tf_{t,d}}$: bobot frekuensi term
- $idf_{d,t}$: IDF dari term t dalam kumpulan korpus d
- D : banyaknya dokumen
- $W_{t,d}$: bobot TF-IDF

2.6 Support Vector Machine

Support Vector Machine (SVM) merupakan seperangkat metode pembelajaran yang menganalisis data dan mengenali pola. Algoritma ini dapat digunakan untuk klasifikasi dan analisis regresi (Giovani et al., 2020) Tujuan dari SVM adalah memberikan bobot pada frekuensi kemunculan kata dan mengklasifikasikan teks dengan label positif atau negatif. Keunggulan SVM terletak pada kemampuannya menangani ruang input berdimensi tinggi dan menggunakan ruang dokumen vektor untuk pemrosesan data teks (Indrayuni, 2016). Support Vector Machine dikembangkan dengan tujuan agar bisa memecahkan masalah klasifikasi dikarenakan metode ini memiliki kemampuan yang tergolong lebih baik pada generalisasian data jika dibandingkan dengan metode-metode yang sudah ada sebelumnya (Gun et al., 1998).

Support Vector Machine (SVM) adalah teknik yang tergolong baru untuk prediksi, baik dalam kasus klasifikasi maupun regresi. SVM merupakan kumpulan metode pembelajaran terarah (*supervised learning*) yang berfungsi untuk menganalisis data dan mengenali pola, serta digunakan dalam proses klasifikasi dan analisis regresi. (Buntoro et al., 2017). Proses klasifikasi dengan Support Vector Machine terbagi dalam dua proses diantaranya proses training (latih) dan testing (uji). Metode ini dikembangkan pertamakali oleh Guyon, Vapnik dan Guyon pada tahun 1992 dalam *event Annual Workshop on Computational Learning Theory* yang merupakan sistem pembelajaran yang pengklasifikasiannya memakai hipotesis ruang dimana terdapat linear function dalam sebuah fitur *space* dimensi tinggi. Metode ini merupakan metode dengan sistem pembelajaran yang memanfaatkan ruang berupa fungsi-fungsi linear pada suatu fitur ruang yang mempunyai dimensi tinggi serta di training dengan menggunakan algoritma pembelajaran yang berdasar pada teori optimasi dengan menggunakan learning bias dalam pengimplementasiannya.

Pada metode SVM diupayakan memperoleh fungsi pemisah (*hyperplane*) paling baik diantara fungsi-fungsi yang jumlahnya tidak terbatas untuk memisahkan dua buah kelas pada ruang input. Fungsi pemisah terbaik antar masing-masing kelas bisa ditemukan dengan cara mengukur *margin hyperplane* serta mencari titik maksimalnya. *Hyperplane* yang dimaksud bisa berupa *line* pada *two dimension* serta bisa berupa *flat plane* pada *multiple plane*.



Gambar 2. 1 SVM dalam mencari *hyperplane* optimal (Nugroho et al., 2003)

Gambar 2.1 menunjukkan konsep dasar SVM, di mana data terbagi menjadi dua kelas: kotak merah mewakili kelas negatif (-1) dan lingkaran kuning mewakili kelas positif (+1). Pada gambar 2.1a, terdapat beberapa garis pemisah antara kedua kelas, sedangkan gambar 2.1b memperlihatkan hyperplane pemisah terbaik yang terletak di tengah antara kedua kelas. Titik-titik kotak dan lingkaran yang ada di dalam lingkaran hitam disebut sebagai support vector machine (Nugroho et al., 2003).

Berikut merupakan rumus umum yang digunakan pada *support vector machine* untuk proses klasifikasi dengan fungsi Keputusan

$$f(x) : f(x) = \text{sgn} \sum_{i=1}^n a_i y_i K(x, x_i) + b \dots \dots \dots (2.4)$$

Keterangan :

- $f(x)$: Fungsi SVM terhadap data masukan (x)
- x : Data masukan SVM
- sgn : Fungsi signum (fungsi yang menghasilkan nilai tanda dari bilangan x)
- n : Banyaknya data
- a_i : Nilai bobot setiap titik data
- $\sum_{i=1}^n a_i$: Sigma a_i , i dari 1 sampai dengan n
- $K(x, x_i)$: Fungsi kernel
- b : Parameter *hyperplane* yang dicari

Metode pendekatan yang sering disebut sebagai kernel mempunyai potensi dalam mengatasi masalah yang timbul dalam ruang fitur (*feature space*), serta mempengaruhi Tingkat akurasi yang dapat dicapai. Fungsi kernel pada SVM berperan dalam penyelesaian kasus non linier. Penggunaan fungsi kernel adalah cara untuk mendapatkan hasil dari klasifikasi yang terbaik. *Support Vector Machine* menunjukkan kinerja yang efektif pada data yang tidak linier dengan menggunakan *trick* kernel. *Trick* kernel berperan dalam melakukan pemetaan dari ruang input yang memiliki dimensi rendah ke ruang yang berdimensi lebih tinggi. Menurut (Nanda et al., 2018) ada berbagai fungsi kernel yang dapat digunakan dalam menyelesaikan masalah yang muncul pada metode SVM, yaitu sebagai berikut :

1. Linear Kernel

$$K(x_i, x) = x_i^T x \dots \dots \dots (2.5)$$

Dengan x_i merupakan data latih (*training*), x merupakan data uji

2. Polynomial Kernel

$$K(x_i, x) = (\gamma(x_i^T x) + r)^p \dots \dots \dots (2.6)$$

Dengan x_i merupakan data latih (*training*), x merupakan data uji, p merupakan derajat polinomial. Nilai derajat pada kernel polinomial berperan dalam mengontrol fleksibilitas dari hasil klasifikasi. Penggunaan derajat polinomial yang terlalu tinggi pada kernel menyebabkan *overfitting*, sehingga pada pengujian dengan data uji menyebabkan penurunan akurasi.

3. RBF (*Radial Basic Function*) Kernel

$$K(x_i, x) = \exp(-\gamma|x_i - x|^2), \gamma > 0 \dots \dots \dots (2.7)$$

Dengan x_i merupakan data latih (*training*), x merupakan data uji, γ merupakan *scalling* parameter dari jarak *Euclidan*.

4. Sigmoid Kernel

$$K(x_i, x) = \tanh(\gamma(x_i^T x) + r) \dots \dots \dots (2.8)$$

Dengan x_i merupakan data latih (*training*), x merupakan data uji, r merupakan koefisien. Menurut (Al-Mejibli et al., 2020) penggunaan *gamma* yang terlalu tinggi akan menyebabkan penurunan Tingkat akurasi pada klasifikasi tetapi tetap bergantung pada jumlah fitur yang digunakan, semakin banyak fitur maka *gamma* akan cenderung kecil, begitupun sebaliknya.

Pengolahan data latih menggunakan algoritma *sequential training* dikarenakan merupakan metode yang efisien tanpa memerlukan banyak waktu (Vijayakumar, 1998), dengan Langkah-langkah perhitungan sebagai berikut :

1. Inisialisasi terhadap berbagai parameter, seperti α_i, γ, C , dan ε .

Keterangan :

- α_i = alfa, untuk mencari *support vector*
- γ = konstanta *gamma* untuk mengontrol kecepatan
- C = variabel *slack*
- ε = epsilon digunakan untuk mencari nilai

2. Hitung *matrix Hessian* yang didapat dari perkalian antar kernel polinomial dan y yang adalah vector bernilai 1 dan -1. Persamaan *matrix Hessian* yaitu :

$$D_{ij} = y_i y_j (K(x_i, x_j) + \lambda^2) \dots \dots \dots (2.9)$$

Keterangan :

- D_{ij} = nilai *matrix Hessian* data ke-ij
- y_i = kelas data ke-i
- y_j = kelas data ke-j
- x_i = data ke-i
- x_j = data ke-j
- $K(x_i, x_j)$ = fungsi kernel
- λ = lamda

3. Hitung interasi data i sampai j

$$E_i = \sum_j^i \alpha_j D_{ij} \dots \dots \dots (2.10)$$

$$\delta \alpha_i = \min (\max[\gamma(1 - E_i), \alpha_i], C - \alpha_i) \dots \dots \dots (2.11)$$

$$\alpha_i = \alpha_i + \delta \alpha_i \dots \dots \dots (2.12)$$

Keterangan :

- E_i = *variable slack*
- D_{ij} = nilai *matrix hessian* data ke-ij
- α_i = alfa ke-i
- $\delta \alpha_i$ = delta alfa ke-i
- γ = deta alfa ke-i
- C = nilai toleransi

4. Melakukan tiga langkah sebelumnya secara berulang hingga mencapai batas maksimum

5. Melakukan proses *sequential learning* dari langkah pertama sampai ke empat akan menghasilkan nilai dari vector pendukung (SV), dimana nilai $SV = \alpha_i > thresholdSV$). Kemudian dilakukan perhitungan pada nilai bias b yang diperoleh pada persamaan 2.13

$$b = -\frac{1}{2}(\sum_{i=0}^N \alpha_i y_i K(x_i, x^-) + \sum_{i=0}^N \alpha_i y_i K(x_i, x^+)) \dots \dots \dots (2.13)$$

Keterangan :

- b = bias/*intercept*
- i = data ke-i
- n = banyaknya data
- α_i = alfa ke-i
- y_i = kelas data ke-i
- $K(x_i, x^-)$ = fungsi kernel data *negative*
- $K(x_i, x^+)$ = fungsi kernel data *positive*

2.7 Lexicon Based

Lexicon Based merupakan tipe dari fitur dengan didasari oleh knowledge atau pengetahuan yang memiliki focus pada memperoleh lexicon berdasarkan opini dari teks dan kemudian mengidentifikasi polaritas lexicon tersebut. Lexicon merupakan kumpulan istilah yang diketahui (Desai dan Mehta, 2017). Fitur ini memberikan bobot dengan membutuhkan bantuan lexicon/kamus untuk mengklasifikasikan dokumen menjadi sentimen yang positif atau sentimen yang negatif (Tiara, Sabariah, dan Effendy, 2015). *Lexicon* adalah kumpulan kata yang sering digunakan untuk menyampaikan sentimen positif atau negatif diberi nilai tertentu pada setiap kata. Barasa adalah kamus data sentimen dalam Bahasa Indonesia yang memuat kata-kata beserta nilai numerik yang merepresentasikan sentimen yang diasosiasikan dengan kata tersebut.

Metode *lexicon based* menggunakan kamus sebagai referensi sumber bahasanya. Dalam metode ini akan dilakukan proses klasifikasi sentimen dari setiap opini sehingga bisa dikelompokkan sesuai kelas positif maupun negatif. Pada *lexicon based* akan dihitung sentimen suatu teks dari polaritas frase atau kata yang terdapat dalam suatu teks (Turney, 2002). Pada metode ini dibutuhkan kamus (lexicon) kata-kata dengan polaritas yang sudah ditetapkan. Adapun beberapa contoh lexicon yang ada seperti : Opinion Lexicon (Hu et al., 2004), SentiWordNet (Esuli et al., 2006), AFINN Lexicon (Nielsen, 2011), LoughranMcDonald Lexicon, NRC-Hashtag (Mohammad et al., 2013), General Inquirer Lexicon3 (Stone et al., 1963). Dalam metode lexicon based setiap opini yang ada pada dataset akan dilakukan analisa satu per satu. Komentar yang mengandung kata yang terdaftar dalam kamus kata opini kemudian akan dilabeli sebagai komentar opini.

S positif adalah bobot kalimat yang diperoleh dari penjumlahan skor polaritas kata-kata opini positif, sedangkan S negatif adalah bobot kalimat yang diperoleh dari penjumlahan skor polaritas kata-kata opini negatif. Dengan demikian, dari persamaan nilai sentimen dalam satu kalimat, kita dapat memperoleh rumus untuk menentukan orientasi sentimen berdasarkan perbandingan jumlah nilai positif, negatif, dan netral.

2.8. Evaluasi Model

Confusion matrix adalah proses yang digunakan untuk menganalisis keakuratan model klasifikasi yang bertujuan mengidentifikasi data dari kelas yang berbeda (Afrillia et al., 2022). Pengujian menggunakan confusion matrix ini menghitung nilai true positive, false positive, true negative, dan false negative, yang selanjutnya dapat digunakan untuk

mengukur akurasi, presisi, dan recall. Dengan mengukur tingkat akurasi, kita dapat menilai seberapa baik performa model klasifikasi tersebut. Bentuk confusion matrix adalah tabel yang terdiri dari empat kombinasi berbeda antara nilai prediksi dan nilai aktual. Metode ini memiliki empat kemungkinan yang merepresentasikan hasil klasifikasi, yaitu TP (*True Positive*), TN (*True Negative*), FP (*False Positive*), dan FN (*False Negative*). Tabel *confusion matrix* dapat dilihat pada Tabel 2.1 berikut:

Tabel 2. 1 Confusion Matrix

Aktual	Negative	Neutral	Positive
Negative	TN (<i>True Negative</i>)	FL2 (<i>False Neutral 2</i>)	FP2 (<i>False Positive</i>)
Neutral	FN2 (<i>False Negative2</i>)	TL(<i>True Neutral</i>)	FP2 (<i>False Positive2</i>)
Positive	FN (<i>False Negative</i>)	FL (<i>False Neutral</i>)	TP (<i>True Positive</i>)
Jumlah	N'	L'	P'

Keterangan :

- TN (*True Negative*) : Prediksi benar bernilai negatif
- TP (*True Positive*) : Prediksi benar bernilai positif
- TL (*True Neutral*) : Prediksi benar bernilai netral
- FN (*False Negative*) : Negatif terprediksi positif
- FP (*False Positive*) : Positif terprediksi negatif
- FL (*False Neutral*) : Positif terprediksi netral
- FN2 (*False Negative 2*) : Netral terprediksi negatif
- FP2 (*False Positive 2*) : Netral terprediksi positif
- FL2 (*False Neutral 2*) : Negatif terprediksi netral
- N' : Jumlah prediksi negatif
- L' : Jumlah prediksi netral
- P' : Jumlah prediksi positif

Berdasarkan Tabel 2.1 *confusion matrix* diatas dapat digunakan dalam perhitungan *performance matrix* yang bertujuan untuk mengukur model yang digunakan yang dapat memperoleh nilai *accuracy*, *recall*, dan *precision*.

1. Accuracy

Akurasi merupakan nilai yang menunjukkan kedekatan antar nilai prediksi dan nilai aktual. Perhitungan akurasi dengan cara membagi jumlah data yang akan diklasifikasi secara tepat dengan total *sample data testing* yang diuji. Berikut merupakan rumus perhitungan nilai akurasi:

$$Accuracy = \frac{TP+TN+TL}{TP+FP+FN+TN} \times 100\% \dots\dots\dots(2.14)$$

2. Recall

Recall merupakan perbandingan jumlah data yang dilakukan prediksi pada kelas positif benar dengan jumlah data yang diharapkan berada pada kelas positif. *Recall* dikatakan sebagai Tingkat keberhasilan model dalam menemukan informasi. Berikut merupakan rumus perhitungan nilai *recall* :

$$Recall = \frac{TP}{TP+FN+FL} \dots\dots\dots(2.15)$$

3. Precision

Precision merupakan 15ingkat akurasi antar informasi yang diinginkan pengguna serta respon

sistem. Nilai presisi menunjukkan data positif yang diklasifikasi dengan tepat kemudian dilakukan pembagian dengan jumlah data positif yang diklasifikasi. Berikut merupakan rumus perhitungan nilai presisi :

$$Precision = \frac{TP}{TP+FP+FP2} \dots\dots\dots(2.16)$$

2.9 Studi Pustaka

Penelitian di bawah ini merupakan penelitian-penelitian yang telah dilakukan sebelumnya dan berkaitan dengan penelitian tugas akhir ini, sehingga menjadi referensi dalam penelitian ini. Berikut dapat dilihat pada Tabel 2.2, Tabel 2.3, Tabel 2.4, Tabel 2.5, Tabel 2.6, dan Tabel 2.7 berikut.

Tabel 2. 2 Studi Pustaka

No	Penulis	Judul	Metode	Hasil
1.	Ibnu Fanhar Nur. F, Anisa Herdiani, S.T., M.T, Widi Astuti, S.T., M.Kom, 2019	Analisis Sentimen Berbasis Leksikon InSet Terhadap Partai Politik Peserta Pemilu 2019 Pada Media Sosial Twitter	SVM dan <i>Lexicon Based</i> .	Penelitian ini menggunakan metode Lexicon dan SVM dapat digunakan dalam melakukan analisis sentimen. Penggunaan metode Lexicon Baesd dan SVM mendapatkan hasil 61%
2.	Aprillia Rizki Adiati, Anisa Herdiani, S.T., M.T., Widi Astuti, S.T., M.Kom.	Analisis Sentimen Masyarakat pada Media Sosial Twitter Terhadap Partai Politik Peserta Pemilu 2019 Menggunakan Naïve Bayes Classifier.	<i>Lexicon Based</i> dan <i>Support Vector Machine</i> .	Penelitian ini menggunakan metode <i>Lexicon Based</i> dan juga SVM untuk membandingkan algoritma KNN dan juga <i>Naïve Bayes</i>
3.	Angelina Puput Giovani, Ardiansyah, Tuti Haryanti, Laela Kurniawati, Windu Gata, 2020	Analisis Sentimen Aplikasi Ruang Guru Di Twitter Menggunakan Algoritma Klasifikasi	<i>Naive Bayes</i> , <i>Support Vector Machine</i> , <i>K-Nearest Neighbour (K-NN)</i> , dan <i>feature selection</i> dengan algoritma <i>Particle Swarm Optimization (PSO)</i>	Berdasarkan hasil analisis terkait sentimen terhadap partai politik peserta pemilu 2019 dengan menggunakan Naïve Bayes Classifier untuk menentukan kelas prediksi, didapatkan hasil sentimen negatif sebesar 46,13% dan sentimen positif sebesar 53,87% dengan akurasi sebesar 78,03%.
4.	Styawati, Nirwana Hendrastuty,	Analisis Sentimen Masyarakat Terhadap Program Kartu	<i>Support Vector Machine</i>	Hasil pada penelitian ini yaitu klasifikasi

Tabel 2. 3 Lanjutan Studi Pustaka

No	Penulis	Judul	Metode	Hasil
	Auliya Rahman Isnain,Ari Yanti Rahmadhani, 2021	Prakerja Pada <i>Twitter</i> Dengan Metode <i>Support Vector Machine</i>		menggunakan metode <i>Support Vector Machine</i> yang dibagi dalam tiga kelas netral sebanyak 98,34%, kelas negatif sebanyak 0,99%, dan kelas positif sebanyak 0,66%. Hasil evaluasi yang dihasilkan dengan kernel linear mendapatkan akurasi 98.67%, <i>precision</i> 98%, <i>recall</i> 99%, dan F1-Score 98%,
5.	Rindu Hafil Muhammadi, Tri Ginanjar Laksana, Amalia Beladinna Arifa, 2022	Combination of Support Vector Machine and Lexicon Based Algorithm in Twitter Sentiment Analysis	<i>Lexicon based</i> dan <i>Support Vector Machine</i>	Hasil penelitian menunjukkan bahwa aplikasi optimasi terbaik dalam model ini adalah algoritma PSO berbasis SVM dengan nilai akurasi sebesar 78,55% dan AUC sebesar 0,853. Penelitian ini berhasil mendapatkan algoritma yang efektif dan terbaik dalam mengklasifikasikan komentar positif dan komentar negatif terkait dengan aplikasi Ruang Guru.
6.	Dede Sandi, Ema Utami, Kusnawi, 2023	Analisis Sentimen Publik Terhadap Elektabilitas Ganjar Pranowo di Tahun Politik 2024 di <i>Twitter</i> dengan Algoritma KNN dan <i>Naïve Bayes</i>	KNN dan <i>Naïve Bayes</i>	Hasil perbandingan dari kedua pemodelan yang diujikan menunjukkan bahwa dengan K-Nearest Neighbor lebih akuratif dengan accuracy 99% dibanding accuracy 96% dari <i>Naïve Bayes</i> . Persentase sentiment dengan perbandingan 96,6% sentimen positif dan 3,4% sentimen negatif menyimpulkan bahwa sebagian besar masyarakat masih mendominasi sentimen positif terhadap elektabilitas Ganjar Pranowo sebagai capres di tahun politik 2024.

Tabel 2. 4 Lanjutan Studi Pustaka

No	Penulis	Judul	Metode	Hasil
7.	Arfina Handayani, Ilka Zufria, 2023	Analisis Sentimen Terhadap Bakal Capres RI 2024 di Twitter Menggunakan Algoritma SVM	<i>Support Vector Machine</i>	Pada penelitian ini proses analisis yang dilakukan memiliki rata-rata akurasi untuk ketiga calon adalah sebesar 78%. Hasil akurasi terhadap penerapan metode <i>Support Vector Machine</i> dipengaruhi oleh beberapa hal, salah satunya adalah jumlah dataset dan komposisi jumlah sentimen positif dan negative.
8.	Muhammad Fernanda Naufal Fathoni ,Eva Yulia Puspaningrum, Andreas Nugroho Sihananto, 2024	Perbandingan Performa Labeling Lexicon InSet dan VADER pada Analisa Sentimen Rohingya di Aplikasi X dengan SVM	<i>Lexicon dan Support Vector Machine</i>	Hasil pada penelitian ini menunjukkan bahwa hasil kombinasi <i>lexicon InSet SVM</i> menghasilkan rata rata akurasi 87,83% dan pada kombinasi <i>lexicon VADER SVM</i> menghasilkan rata rata akurasi sebesar 87,66%. Dengan hal tersebut, <i>lexicon InSet</i> lebih unggul 0.17% dibandingkan <i>lexicon VADER</i> .
9.	Yessy Asri,Widya Nita Suliyanti, Dwina Kuswardani, Muhamad Fajri, 2022	Pelabelan Otomatis <i>Lexicon Vader</i> dan Klasifikasi <i>Naive Bayes</i> dalam menganalisis sentimen data ulasan PLN Mobile	<i>Lexicon Vader dan Naïve bayes</i>	Hasil pada penelitian ini menunjukkan bahwa hasil pengimplementasian <i>Vader</i> dan <i>Naïve Bayes</i> mendapatkan hasil 70%.
10.	Natasya Eldha Oktaviana, Yuita Arum Sari, Indriati, 2022	Analisis Sentimen Terhadap Kebijakan Kuliah Daring Selama Pandemi Menggunakan Pendekatan <i>Lexicon Based Features</i> Dan <i>Support Vector Machine</i>	<i>Lexicon Based Features dan Support Vector Machine</i>	Hasil evaluasi sistem pada penelitian ini data komentar pengguna Twitter memperoleh peningkatan setelah menggunakan <i>Lexicon Based Features</i> yaitu dengan nilai <i>accuracy</i> 0,6, nilai <i>precision</i> 0,56, nilai <i>recall</i> 0,75, dan <i>fmeasure</i> 0,64 dibandingkan proses klasifikasi tanpa <i>Lexicon Based Feature</i> yang hanya menghasilkan nilai <i>accuracy</i> 0,48, nilai <i>precision</i> 0,46, nilai <i>recall</i> 0,58, dan nilai <i>fmeasure</i> 0,52. Dimana nilai evaluasi tersebut

Tabel 2. 5 Lanjutan Studi Pustaka

No	Penulis	Judul	Metode	Hasil
				meningkat sebanyak 12% daripada metode SVM yang tidak menggunakan pendekatan <i>Lexicon Based Features</i> .
11.	Isram Rasalb, 2023	Penerapan Algoritma Naive Bayes Dan Svm Untuk Analisis Sentimen Boy Band Bts Pada Media Sosial Twitter	<i>Naive Bayes</i> Dan SVM	Berdasarkan Penelitian yang telah dilakukan penulis hasil dari analisis sentimen dengan menggunakan <i>Machine Learning</i> yang membandingkan 2 algoritma yaitu <i>Naive Bayes</i> dan <i>Support Vector Machine</i> telah berhasil dilakukan. Pada analisis sentimen ini didapatkan hasil analisis sentimen positif sebesar 78.8% dan analisis sentimen negatif sebesar 21.2%. Kemudian algoritma yang mendapatkan hasil akurasi tertinggi pada analisis yang dilakukan adalah algoritma <i>Support Vector Machine</i> dengan nilai akurasi sebesar 81%. Sedangkan untuk algoritma <i>Naive Bayes</i> menghasilkan nilai akurasi sebesar 79%.
12.	Fadhilah Fazrin, Oktariani Nurul Pratiwi, Rachmadita Andreswari	Perbandingan Algoritma K-Nearest Neighbor dan Logistic Regression pada Analisis Sentimen terhadap Vaksinasi Covid-19 pada Media Sosial Twitter dengan Pelabelan Vader dan Textblob	Metode pelabelan <i>Vader</i> , dan metode Metode klasifikasi yang digunakan <i>K-Nearest Neighbor</i>	Hasil perbandingan sentimen positif dan sentimen negatif pada kedua algoritma adalah sentimen negatif lebih besar dari sentimen positif. Terdapat 64% sentimen negatif pada hasil klasifikasi menggunakan <i>TextBlob</i> dan terdapat 56% sentimen negatif pada hasil klasifikasi menggunakan <i>VaderSentiment</i> .

Tabel 2. 6 Lanjutan Studi Pustaka

No	Penulis	Judul	Metode	Hasil
13.	Anna Baita, Yoga Pristyanto, Nuri cahyono (2022)	Analisis Sentimen Mengenai Vaksin Sinovac menggunakan Algoritma Support Vector Machine (SVM) dan KNN.	Support Vector Machine (SVM) dan KNN.	Hasil Klasifikasi dengan menggunakan algoritma SVM ditunjukkan dengan hasil <i>accuracy</i> 73% dan 60% untuk klasifikasi KNN.
14.	Fajar Sodik Pamungkas,Iqbal Kharisudina(2021)	Analisis Sentimen dengan SVM, NAIVE BAYES dan KNN untuk Studi Tanggapan Masyarakat Indonesia Terhadap Pandemi Covid-19 pada Media Sosial Twitter	Support Vector Machine (SVM) dan KNN, dan Naive Bayes.	diperoleh hasil bahwa algoritma SVM memiliki akurasi yang lebih tinggi daripada Naive Bayes dan KNN dengan rata - rata akurasinya
15.	Aprillia Rizki Adiati, Anisa Herdiani, S.T., M.T., Widi Astuti, S.T., M.Kom.	Pemanfaatan Metode Multiclass-SVM pada Model Klasifikasi Pesan Bencana Banjir di Twitter	Metode <i>Naive Bayes</i> , dan <i>Lexicon Based</i>	Total jumlah data yang telah diolah sebanyak 3000 data untuk 3 kelas label data. Pendekatan metode One Versus One (OVO) untuk dua kelas label data dan One Versus All (OVA) untuk lebih dari dua kelas label pada algoritma SVM. Pada penelitian ini telah diterapkan algoritma Support Vector Machine untuk multiclass dengan metode OVA. Hasil evaluasi tersebut adalah nilai akurasi hingga 87.03%.
16.	Dea Oktavia, Yudhi Raymond Ramadahan, Minarto.	Analisis Sentimen Terhadap Penerapan Sistem E -Tilang Pada Media Sosial Twitter Menggunakan Algoritma Support Vector Machine (SVM).	Support Vector Machine (SVM).	Hasil sentimen masyarakat pada sosial media Twitter mengenai penerapan e-Tilang tergolong netral. Implementasi algoritma Support Vector Machine (SVM) pada penelitian ini menunjukkan tingkat keakurasian sebesar 74.20%, precision sebesar 83.33% dan recall 5.28%.
17.	Rizky Rahman Salam,Muhammad Fajri Jamil,Yusril Ibrahim,Rahmaddeni, Soni, Herianto	Analisis Sentimen Terhadap Bantuan Langsung Tunai (BLT) Bahan Bakar Minyak (BBM) Menggunakan <i>Support Vector Machine</i> .	<i>Support Vector Machine</i>	Dari hasil perhitungan, terlihat bahwa proses klasifikasi sentimen menggunakan metode SVM didapatkan tingkat <i>accuracy</i> 85,98%, rata-rata nilai precision 82,25% ,nilai rata-rata

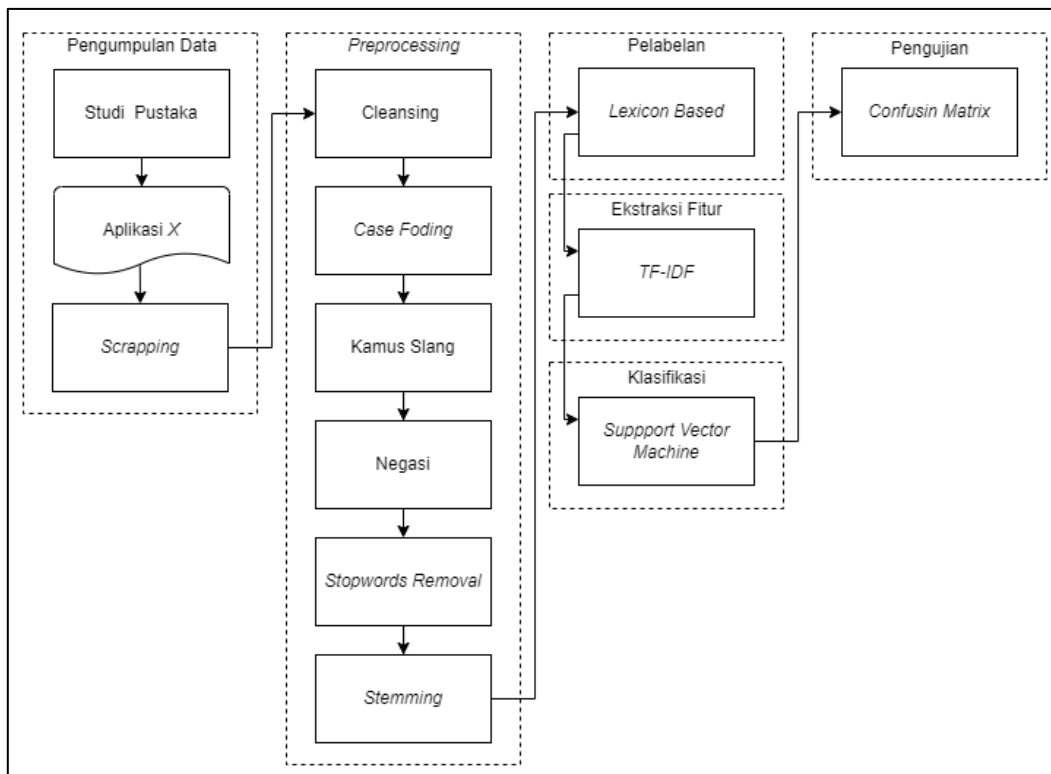
Tabel 2. 7 Lanjutan Studi Pustaka

No	Penulis	Judul	Metode	Hasil
				recall66,35%, dan nilai rata-rata f-measure 73,44%. Hasil yang diperoleh menunjukkan bahwa sentimen negatif lebih banyak daripada sentimen positif, dengan masing-masing persentase 78.61% dan 21.34%.
18.	Fatihah Rahmadayana (2021)	<i>Sentiment Analysis of Work from Home Activity using SVM with Randomized Search Optimization</i>	<i>Support Vector Machine</i>	Mendapatkan hasil paling baik dengan menerapkan akurasi paling baik dengan menggunakan menambahkan <i>gridsearch</i> dengan akurasi 83.62 dan tanpa <i>gridsearch</i> diangka 82.49%.
19.	Trianto, Anita Muliawati (2020)	Penerapan <i>BorderlineSMOTE</i> dan <i>Grid Search</i> pada <i>Bagging-SVM</i> untuk Klasifikasi Penyakit Diabetes	<i>Support Vector Machine</i> dan <i>SMOTE</i>	Hasil penelitian menunjukkan bahwa model yang dibentuk dengan algoritma SVM, bagging, <i>borderlineSMOTE</i> dan <i>grid search</i> mendapat akurasi sebesar 92,1%, nilai <i>precision</i> sebesar 95,51% untuk kelas sehat dan 86,12% untuk kelas diabetes, nilai <i>recall</i> sebesar 92,32% untuk kelas sehat dan 91,66% untuk kelas diabetes, dan nilai <i>f1score</i> sebesar 93,39% untuk kelas sehat dan 88,81% untuk kelas diabetes.

BAB III METODOLOGI PENELITIAN DAN PENGEMBANGAN SISTEM

3.1 Metodologi Penelitian

Pada sub bab ini akan dibahas metodologi yang digunakan dalam penelitian, yang bertujuan untuk menganalisis sentimen masyarakat di media sosial X terkait ulasan terhadap partai politik di Indonesia dengan menggunakan metode *Support Vector Machine*. Metodologi ini didasarkan pada hasil studi pustaka yang relevan dengan penelitian ini. Data yang digunakan diperoleh melalui *scraping* dari aplikasi X untuk mengumpulkan tweet yang diperlukan. Langkah-langkah penelitian dapat dilihat pada Gambar 3.1 berikut.



Gambar 3.1 Metodologi Penelitian

3.1.1 Pengumpulan Data

Pada pengumpulan data dilakukan untuk memperoleh data yang nantinya akan diolah menjadi informasi yang mudah dipahami dan dapat digunakan sebagai dasar pengambilan keputusan. Dalam penelitian ini, data dikumpulkan dengan melakukan crawling dari aplikasi X.

a. Studi Pustaka

Studi pustaka merupakan langkah awal dalam proses penelitian. Kegiatan ini bertujuan untuk mencari dan memperoleh informasi terkait metode, topik, serta masalah yang menjadi fokus penelitian. Studi pustaka ini didasarkan pada referensi dari jurnal, makalah, prosiding, buku, dan sumber terpercaya lainnya. Studi pustaka dilakukan untuk memperkuat argumentasi dalam penelitian yang sedang dilakukan.

b. X

Pada penelitian ini data yang digunakan diambil dari aplikasi X. Aplikasi X, sebelumnya dikenal sebagai twitter, merupakan *platform* media sosial yang memungkinkan pengguna untuk berbagi pesan singkat yang disebut "*tweet*." Aplikasi ini digunakan secara luas oleh berbagai kalangan, mulai dari individu hingga organisasi besar, untuk menyampaikan pendapat, berita, dan informasi lainnya. Data dalam penelitian ini diambil dari *tweet* di aplikasi X, dikarenakan *platform* ini memberikan peluang untuk menganalisis sentimen masyarakat terhadap partai politik di Indonesia dan mengamati reaksi pengguna terhadap berbagai topik, termasuk ulasan tentang layanan tertentu.

c. *Scraping*

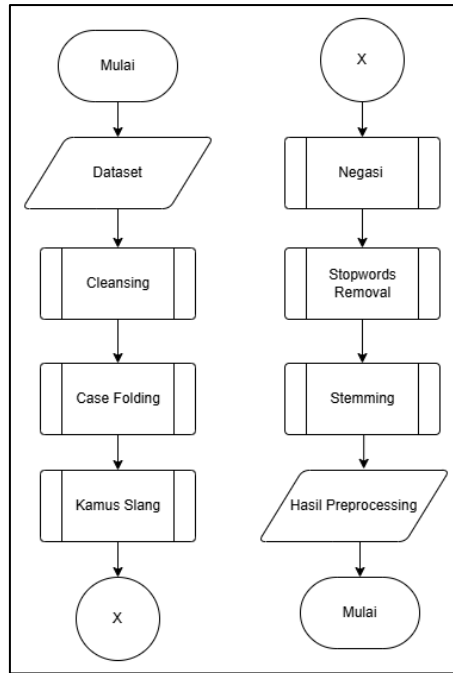
Pengumpulan data dilakukan dengan cara *scraping* pada aplikasi X. Data yang berhasil dikumpulkan akan dilakukan proses *pre-processing*. Pada penelitian ini data yang digunakan merupakan data dengan menggunakan Bahasa Indonesia dengan total data yang digunakan berjumlah 5111 *tweet* dengan menggunakan 5 kata kunci yang tertera pada Tabel 3.1

Tabel 3. 1 Kata Kunci X

No.	Kata Kunci	Jumlah Data
1.	Partai Politik	1155
2.	#parpol	1255
3.	Parpol	1018
4.	Partai Indonesia	891
5.	Partai	792
	Jumlah	5111

3.1.2 *Preprocessing*

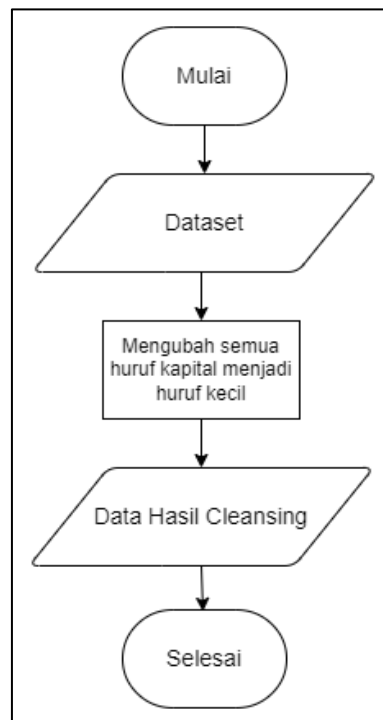
Preprocessing data adalah tahap penting dalam pengolahan data yang bertujuan untuk membersihkan dan mempersiapkan data agar siap digunakan dalam analisis lebih lanjut. Pada tahap ini, berbagai langkah dilakukan, seperti menghapus data yang tidak relevan (misalnya, tanda baca, angka, dan simbol khusus), mengubah teks menjadi format standar (seperti konversi huruf besar menjadi huruf kecil), menghilangkan kata-kata umum yang tidak memiliki makna khusus (*stopwords*), serta melakukan stemming untuk mengembalikan kata ke bentuk dasarnya. Proses *preprocessing* ini sangat penting untuk meningkatkan akurasi model dalam memahami dan menganalisis data secara efektif, terutama dalam tugas-tugas yang melibatkan analisis teks seperti analisis sentimen. Berikut merupakan *flowchart preprocessing* data dapat dilihat pada Gambar 3.2 *Preprocessing* Data.



Gambar 3. 2 Flowchart Preprocessing

a. *Cleansing*

Proses ini bertujuan untuk membersihkan data dengan menghapus elemen-elemen yang tidak relevan, seperti tanda baca, angka, emotikon, URL, atau karakter khusus lainnya yang tidak memiliki makna signifikan untuk analisis teks. Cleansing membantu memfokuskan data pada elemen-elemen yang relevan untuk analisis lebih lanjut.



Gambar 3. 3 Flowchart Cleansing

Flowchart pada Gambar 3.3 merupakan alur tahapan dari proses *cleansing* tahapan *preprocessing*. Berikut contoh hasil dari tahap *cleansing* and *case folding* dapat dilihat pada Tabel 3.2.

Tabel 3. 2 Contoh Hasil *Cleansing*

No.	Data sebelum <i>Cleansing</i>	Data sesudah <i>Cleansing</i>
1.	Pada dasarnya mereka tidak suka kalo Presiden dr @PDIP...meski jelas2 Islam kalo dari @PDIP pasti dibilang kafir...Partai yang paling tidak bisa menerima kalo Presiden dari @PDIP adalah PKS...kalo Ormas dari FPI HTI dkk	Pada dasarnya mereka tidak suka <u>kalo</u> presiden dari PDIP meski jelas <u>islam</u> kalo dari <u>PDIP</u> pasti dibilang kafir partai yang paling tidak bisa menerima kalo <u>Presiden</u> dari <u>PDIP</u> adalah <u>PKS</u> kalo Ormas dari FPI HTI dkk

b. *Case Folding*

Dalam tahap pre-processing tahap ini bertujuan untuk mengganti huruf kapital yang terdapat pada data menjadi huruf kecil. Setelah melewati proses ini, data dengan huruf kecil akan disimpan, untuk meneruskan ke proses selanjutnya.



Gambar 3. 4 Flowchart *Case Folding*

Flowchart pada Gambar 3.4 merupakan alur tahapan dari proses *case folding* tahapan *preprocessing*. Berikut contoh hasil dari tahap *case folding* dapat dilihat pada Tabel 3.3.

Tabel 3. 3 Contoh Hasil *Case Folding*

No.	Data sebelum <i>Case folding</i>	Data sesudah <i>Case folding</i>
1.	Pada dasarnya mereka tidak suka <u>kalo</u> presiden dari PDIP meski jelas <u>islam</u> kalo dari <u>PDIP</u> pasti dibilang kafir partai yang paling tidak bisa menerima kalo <u>Presiden</u> dari <u>PDIP</u> adalah <u>PKS</u> kalo Ormas dari FPI HTI dkk	<u>pada</u> dasarnya mereka tidak suka <u>kalo</u> presiden dari pdip meski jelas <u>islam</u> kalo dari <u>pdip</u> pasti dibilang kafir partai yang paling tidak bisa menerima kalo <u>presiden</u> dari <u>pdip</u> adalah <u>pkS</u> kalo ormas dari fpi hti dkk

c. Kamus Slang

Proses ini memisahkan kalimat atau teks menjadi elemen-elemen yang lebih kecil, yang biasanya berupa kata serta memasukkan kamus slang. Penggunaan kamus slang memiliki keuntungan dalam mengartikan kata – kata *gaul* yang cenderung susah diartikan kedalam Bahasa Indonesia dan memerlukan parameter untuk mengubah definisi atau pengertian untuk mengetahui konteks kata yang terdapat pada dataset. Berikut merupakan tahapan dari proses normalisasi dapat dilihat pada Gambar 3.5.



Gambar 3. 5 Flowchart Kamus Slang

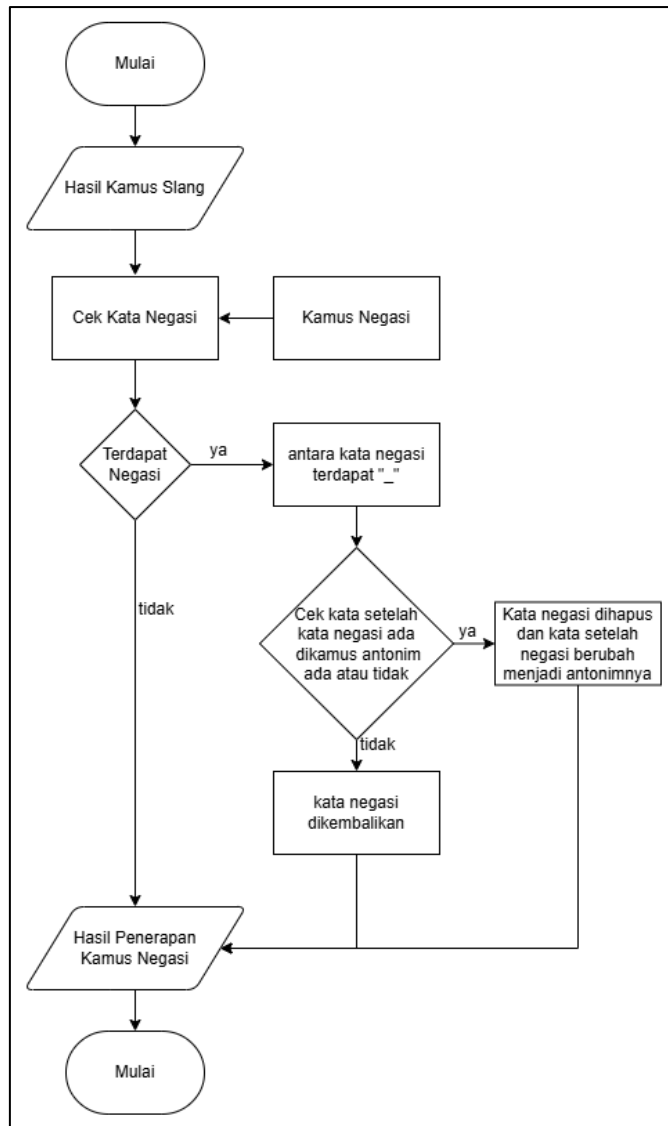
Flowchart pada Gambar 3.5 merupakan alur tahapan dari proses penerapan kamus slang pada tahapan *preprocessing*. Berikut contoh hasil dari tahap normalisasi dapat dilihat pada Tabel 3.4.

Tabel 3. 4 Contoh Hasil Penerapan Kamus Slang

No.	Sebelum Kamus Slang	Sesudah Kamus Slang
1.	pada dasarnya mereka tidak suka kalo presiden dari pdip meski jelas islam kalo dari pdip pasti dibilang kafir partai yang paling tidak bisa menerima kalo presiden dari pdip adalah pks kalo ormas dari fpi hti dkk	pada dasarnya mereka tidak suka kalo presiden dari partai meski jelas islam kalau dari partai pasti dibilang kafir yang paling tidak bisa menerima kalo presiden dari partai adalah partai kalo organisasi dari komunitas komunitas dan kawan kawan

d. Negasi

Langkah ini akan mencakup penanganan negasi dengan mengubah kata-kata negasi seperti "tidak" menjadi "tidak_" sehingga memungkinkan pencarian kata-kata yang memiliki kesamaan atau sinonim, tanpa mengubah makna sentimennya. Berikut merupakan contoh hasil dari proses penerapan negasi. Berikut merupakan tahapan preprocessing negasi dapat dilihat pada Gambar 3.6.



Gambar 3. 6 Flowchart Negasi

Flowchart pada Gambar 3.6 merupakan alur tahapan dari proses *negation handling* pada tahapan *preprocessing*. Berikut contoh hasil dari tahap negasi dapat dilihat pada Tabel 3.5.

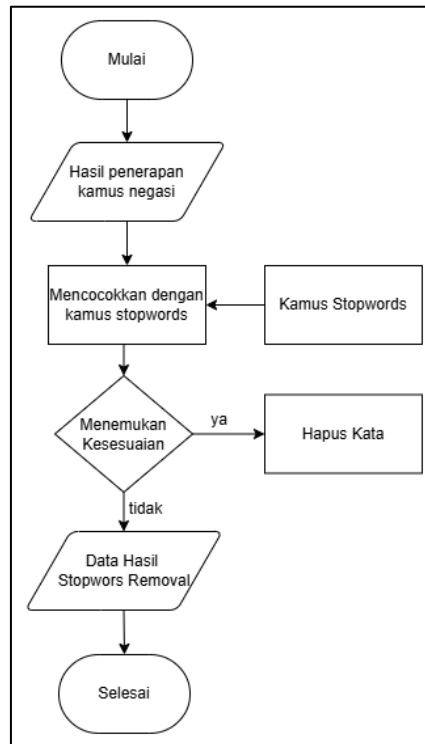
Tabel 3. 5 Contoh Hasil Negasi

No.	Sebelum Penerapan Negasi	Sesudah Penerapan Negasi
1.	pada dasarnya mereka tidak suka kalo presiden dari partai meski jelas islam kalau dari partai pasti dibilang kafir yang paling tidak bisa menerima kalo presiden dari partai adalah partai kalo organisasi dari komunitas komunitas dan kawan kawan	pada dasarnya mereka tidak suka kalau presiden dari partai meski jelas islam kalau dari partai pasti dibilang kafir partai yang paling tidak bisa menerima kalau presiden dari partai adalah kalo organisasi dari komunitas komunitas dan kawan kawan

e. *Stopwords Removal*

Pada tahap ini, kata-kata umum yang sering muncul tetapi tidak memiliki makna khusus dalam konteks analisis, seperti "dan," "atau," "dengan," akan dihilangkan. Kata-kata ini disebut *stopwords* dan biasanya tidak memberikan kontribusi signifikan dalam

penentuan makna teks. Berikut merupakan tahapan dari proses stopword removal dapat dilihat pada Gambar 3.7.



Gambar 3. 7 Flowchart Stopword Removal

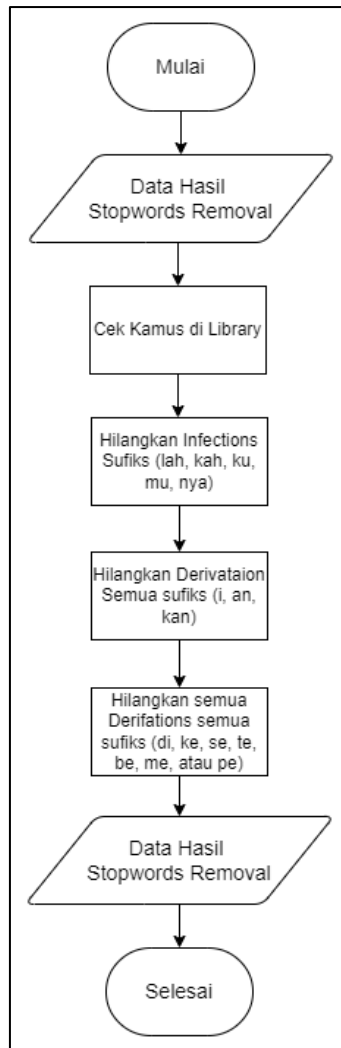
Flowchart pada Gambar 3.7 merupakan alur tahapan dari proses *stopword removal* pada tahapan *preprocessing*. Berikut contoh hasil dari tahap *stopword removal* dapat dilihat pada Tabel 3.6.

Tabel 3. 6 Contoh Hasil Stopword Removal

No.	Sebelum Stopword Removal	Sesudah Stopword Removal
1.	pada dasarnya mereka tidak_suka kalau presiden dari partai meski jelas islam kalau dari partai pasti dibilang kafir partai yang paling tidak_bisa menerima kalau presiden dari partai adalah kalo organisasi dari komunitas komunitas dan kawan kawan	dasarnya mereka tidak_suka presiden dari partai meski jelas islam dari partai pasti dibilang kafir partai paling tidak_bisa menerima presiden dari partai adalah organisasi dari komunitas komunitas dan kawan kawan

f. *Stemming*

Proses *stemming* merupakan proses mengubah kata-kata dalam teks menjadi bentuk dasarnya atau kata dasar, dengan membuang awalan, akhiran atau sisipan. Seperti *inflection suffixes* (-lah, -kah, -tah, -pun, -ku, dsb), *derivational suffix* (-an, -kan, -i), dan *derivational prefix* (-be, -di, -ke, -me, -pe, -se, -te). Dalam penelitian ini proses *stemming* menggunakan algoritma Nazief & Adriani untuk teks berbahasa Indonesia. Berikut merupakan tahapan dari proses *stemming* dapat dilihat pada Gambar 3.8.



Gambar 3. 8 Flowchart Stemming

Flowchart pada Gambar 3.8 merupakan alur tahapan dari proses *stemming* pada tahapan *preprocessing*. Berikut contoh hasil dari tahap *stemming* dapat dilihat pada Tabel 3.7.

Tabel 3. 7 Contoh Hasil Stemming

No.	Sebelum Stemming	Sesudah Stemming
1.	dasarnya mereka tidak_suka presiden dari partai meski jelas islam dari partai pasti dibilang kafir partai paling tidak_bisa menerima presiden dari partai adalah organisasi dari komunitas komunitas dan kawan kawan	dasar mereka tidak suka presiden dari partai meski jelas islam dari partai pasti bilang kafir partai paling tidak bisa terima presiden dari partai adalah organisasi dari komunitas komunitas kawan kawan

3.1.3 Labeling

Data yang telah dilakukan proses *preprocessing* data, selanjutnya akan diberi label dengan menggunakan pelabelan *lexicon*. Pelabelan dengan menggunakan pendekatan *lexicon* merupakan langkah penting dalam implementasi metode *lexicon-based* dan *Support Vector Machine* (SVM) untuk menganalisis sentimen publik terhadap partai politik di Indonesia. Metode ini menggunakan daftar kata (*lexicon*) yang memiliki nilai sentimen tertentu, baik positif, negatif, maupun netral, untuk menentukan kategori dari setiap teks yang akan dianalisis.

Dalam tahap ini, setiap kata yang terdapat dalam data dicocokkan dengan daftar kata pada *lexicon*. Jika kata tersebut ditemukan dalam *lexicon*, maka akan diberikan nilai sentimen sesuai dengan yang tertera pada daftar tersebut. Misalnya, kata-kata seperti "bagus" atau "terpuji" dapat diberi label positif, sedangkan kata seperti "buruk" atau "mengecewakan" diberi label negatif. Skor total kemudian dihitung berdasarkan nilai dari setiap kata dalam teks. Jika skor total lebih besar dari nol, maka teks diberi label sentimen positif; jika kurang dari nol, diberi label negatif; dan jika skor adalah nol, maka dianggap netral.

Setelah pelabelan menggunakan metode *lexicon*, hasil pelabelan ini kemudian digunakan untuk melatih model SVM. SVM akan menggunakan data yang telah dilabeli tersebut untuk belajar membedakan pola dan karakteristik teks positif, negatif, atau netral. Dengan pelabelan menggunakan *lexicon*, model SVM dapat lebih efektif dalam mengklasifikasikan sentimen pada data baru, dikarenakan sudah dilatih menggunakan data yang dilabeli secara manual atau semi-otomatis berdasarkan *lexicon*. Penggabungan metode *lexicon-based* dengan SVM ini memanfaatkan keunggulan dari kedua pendekatan yaitu interpretasi langsung dari *lexicon* dan kemampuan pembelajaran dari SVM sehingga menghasilkan model yang lebih akurat untuk analisis sentimen publik terhadap partai politik di Indonesia. Berikut merupakan contoh pelabelan data dapat dilihat pada Tabel 3.8 berikut :

Tabel 3. 8 Contoh Hasil Pelabelan Data

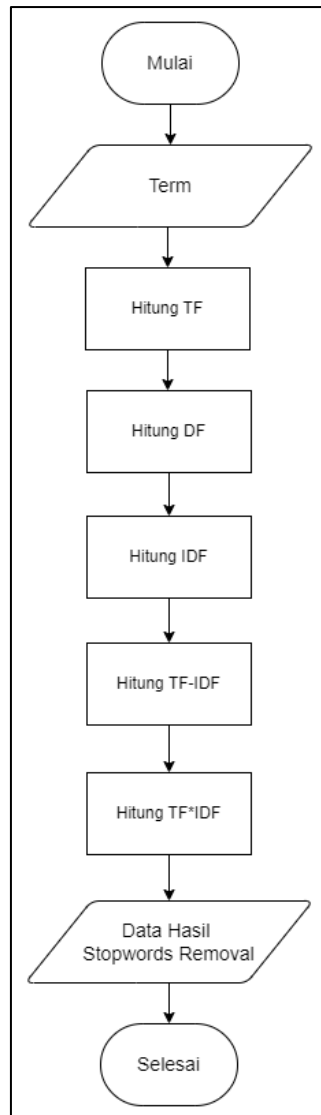
No.	Ulasan	Lexicon	Manual
1.	turun aksi protes pilpres curang akibat penting kuasa partai politik	-1	-1
2.	ngalir partai politik pilkada pilpres coba ungkap biar seru	0	0
3.	proses kaderisasi dalam partai politik harus terus gulir	1	0
4.	rakyat seperti sudah skeptis dengan parpol penting bisa makan saja	-1	-1

Pada Tabel 3.8 merupakan contoh ulasan yang sudah diberi label, dimana label positif akan direpresentasikan dengan nilai 1, label negatif dengan nilai -1, dan label netral dengan label 0.

3.1.4 Pembobotan TF-IDF

Pembobotan kata dengan menggunakan metode Term Frequency-Inverse Document Frequency (TF-IDF) adalah teknik yang digunakan untuk mengekstraksi fitur dalam analisis teks, termasuk analisis sentimen. TF-IDF adalah cara untuk mengukur seberapa penting sebuah kata dalam sebuah dokumen relatif terhadap keseluruhan kumpulan dokumen (korpus). Metode ini memberikan bobot lebih tinggi pada kata-kata yang sering muncul dalam suatu dokumen, tetapi jarang ditemukan dalam dokumen lain di korpus, karena dianggap lebih signifikan.

Pada penelitian ini, setiap kata dalam dokumen akan diberi bobot menggunakan nilai TF-IDF. Kata-kata dengan bobot TF-IDF tinggi dianggap lebih relevan dan memiliki peran lebih besar dalam mendeskripsikan dokumen tersebut. Pembobotan ini membantu mengurangi dampak dari kata-kata umum yang sering muncul tetapi tidak memberikan banyak informasi (*stopwords*), sementara memberi fokus lebih pada kata-kata unik yang lebih menggambarkan isi dari dokumen. Berikut merupakan tahapan dari perhitungan pembobotan kata dengan TF-IDF dapat dilihat pada Gambar 3.9.



Gambar 3.9 Flowchart TF-IDF

Gambar 3.9 menunjukkan tahapan perhitungan TF-IDF, yang dimulai dengan memilih *term* yang akan dihitung. Selanjutnya, dilakukan penghitungan frekuensi kemunculan kata dalam dokumen (TF), diikuti dengan menghitung jumlah dokumen yang mengandung kata tersebut (DF). Setelah itu, nilai IDF dihitung, dan tahap terakhir adalah menghitung nilai TF-IDF dengan mengalikan hasil TF dan IDF.

1. Menghitung *Term Frequency* (TF)

Term Frequency (TF) adalah frekuensi kemunculan suatu istilah dalam sebuah dokumen. Semakin sering istilah tersebut muncul dalam dokumen, semakin tinggi bobotnya. Berikut merupakan contoh data yang akan digunakan untuk perhitungan manual TF-IDF dapat dilihat pada Tabel 3.9.

Tabel 3. 9 Contoh Dokumen Perhitungan

ID	Dokumen	Sentimen	Label
D1	turun aksi protes pilpres curang akibat penting kuasa partai politik	Negatif	-1
D2	ngalir partai politik pilkada pilpres coba ungkap biar seru	Netral	0
D3	proses kaderisasi dalam partai politik harus terus gulir	Positif	1
D4	rakyat seperti sudah skeptis dengan parpol penting bisa makan saja	Negatif	-1
U1	dah bagus bubar saja partai politik batal semua caleg bikin korupsi meroket ahli tata negara abal abal sekolah dimana iq berapa	?	?

Selanjutnya yaitu menghitung nilai *Term Frequency* (TF) dari masing-masing dokumen yang telah dilakukan *Text Preprocessing* pada tiap *term* yang ada. Proses perhitungan TF dengan memberikan nilai 1 apabila *term* tersebut terdapat pada komentar dan nilai 0 dan sebaliknya. Proses dilakukan pada dokumen 1(D1) hingga dokumen 4 (D4). Hasil TF dapat dilihat pada Tabel 3.10 dan Tabel 3.11 sebagai berikut :

Tabel 3. 10 Contoh Hasil Perhitungan TF

No.	Term	TF				
		D1	D2	D3	D4	U1
1.	abal	0	0	0	0	2
2.	ahli	0	0	0	0	1
3.	akibat	1	0	0	0	0
4.	aksi	1	0	0	0	0
5.	bagus	0	0	0	0	1
6.	batal	0	0	0	0	1
7.	berapa	0	0	0	0	1
8.	biar	0	1	0	0	0
9.	bikin	0	0	0	0	1
10.	bisa	0	0	0	1	0
11.	bubar	0	0	0	0	1
12.	caleg	0	0	0	0	1
13.	coba	0	1	0	0	0
14.	curang	1	0	0	0	0
15.	dah	0	0	0	0	1
16.	dalam	0	0	0	0	1
17.	dengan	0	0	0	1	0
18.	dimana	0	0	0	0	1
19.	gulir	0	0	1	0	0
20.	harus	0	0	1	0	0
21.	iq	0	0	0	0	1
22.	kaderisasi	0	0	1	0	0
23.	korupsi	0	0	0	0	1
24.	kuasa	1	0	0	0	0
25.	makan	0	0	0	1	0
26.	meroket	0	0	0	0	1
27.	negara	0	0	0	0	1
28.	ngalir	0	1	0	0	0
29.	parpol	0	0	0	1	0
30.	partai	1	1	1	0	1
31.	penting	1	0	0	1	0

Tabel 3. 11 Lanjutan Contoh Hasil Perhitungan Nilai TF

No.	Term	TF				
		D1	D2	D3	D4	U1
32.	pilkada	0	1	0	0	0
33.	pilpres	0	0	0	1	0
34.	politik	1	1	1	0	1
35.	proses	0	0	1	0	0
36.	protes	0	0	0	0	1
37.	rakyat	0	0	0	1	0
38.	saja	0	0	0	1	1
39.	sekolah	0	0	0	0	1
40.	semua	0	0	0	0	1
41.	seperti	0	0	0	1	0
42.	seru	0	1	0	0	0
43.	skeptis	0	1	0	0	0
44.	sudah	0	0	0	1	0
45.	tata	0	0	0	0	1
46.	terus	0	0	1	0	0
47.	turun	1	0	0	0	0
48.	ungkap	0	1	0	0	0

Dapat dilihat pada Tabel 3.10 dan Tabel 3.11 merupakan *term* yang telah diproses perhitungan TF, dengan diberi nilai 1 apabila *term* tersebut terdapat pada dokumen, dan nilai 0 apabila *term* tersebut tidak ada dalam dokumen.

2. Menghitung Nilai *Inverse Document Frequency* (IDF)

Setelah perhitungan *Term Frequency* (TF) selesai, langkah berikutnya adalah menghitung *Inverse Document Frequency* (IDF), yaitu menghitung seberapa banyak *term* muncul di seluruh dokumen. Rumus IDF dihitung menggunakan persamaan:

$$idf_t = \log \frac{N}{df_t}$$

Berikut merupakan hasil perhitungan IDF dapat dilihat pada Tabel 3.12 dan Tabel 3.13

Tabel 3. 12 Contoh Hasil Perhitungan Nilai IDF

No.	Term	TF					DF	Nd/df(t)	IDF
		D1	D2	D3	D4	U1			
1.	abal	0	0	0	0	2	1	3	2,099
2.	ahli	0	0	0	0	1	1	3	2,099
3.	akibat	1	0	0	0	0	1	3	2,099
4.	aksi	1	0	0	0	0	1	3	2,099
5.	bagus	0	0	0	0	1	1	3	2,099
6.	batal	0	0	0	0	1	1	3	2,099
7.	berapa	0	0	0	0	1	1	3	2,099
8.	biar	0	1	0	0	0	1	3	2,099
9.	bikin	0	0	0	0	1	1	3	2,099
10.	bisa	0	0	0	1	0	1	3	2,099
11.	bubar	0	0	0	0	1	1	3	2,099
12.	caleg	0	0	0	0	1	1	3	2,099
13.	coba	0	1	0	0	0	1	3	2,099
14.	curang	1	0	0	0	0	1	3	2,099
15.	dah	0	0	0	0	1	1	3	2,099
16.	dalam	0	0	0	0	1	1	3	2,099
17.	dengan	0	0	0	1	0	1	3	2,099
18.	dimana	0	0	0	0	1	1	3	2,099
19.	gulir	0	0	1	0	0	1	3	2,099

Tabel 3. 13 Lanjutan Contoh Hasil Perhitungan Nilai IDF

No.	Term	TF					DF	Nd/df(t)	IDF
		D1	D2	D3	D4	U1			
20.	harus	0	0	1	0	0	1	3	2,099
21.	iq	0	0	0	0	1	1	3	2,099
22.	kaderisasi	0	0	1	0	0	1	3	2,099
23.	korupsi	0	0	0	0	1	1	3	2,099
24.	kuasa	1	0	0	0	0	1	3	2,099
25.	makan	0	0	0	1	0	1	3	2,099
26.	meroket	0	0	0	0	1	1	3	2,099
27.	negara	0	0	0	0	1	1	3	2,099
28.	ngalir	0	1	0	0	0	1	3	2,099
29.	parpol	0	0	0	1	0	1	3	2,099
30.	partai	1	1	1	0	1	4	1,2	1,182
31.	penting	1	0	0	1	0	2	2	1,693
32.	pilkada	0	1	0	0	0	1	3	2,099
33.	pilpres	0	0	0	1	0	1	3	2,099
34.	politik	1	1	1	0	1	4	1,2	1,182
35.	proses	0	0	1	0	0	1	3	2,099
36.	protes	0	0	0	0	1	1	3	2,099
37.	rakyat	0	0	0	1	0	1	3	2,099
38.	saja	0	0	0	1	1	2	2	1,693
39.	sekolah	0	0	0	0	1	1	3	2,099
40.	semua	0	0	0	0	1	1	3	2,099
41.	seperti	0	0	0	1	0	1	3	2,099
42.	seru	0	1	0	0	0	1	3	2,099
43.	skeptis	0	1	0	0	0	1	3	2,099
44.	sudah	0	0	0	1	0	1	3	2,099
45.	tata	0	0	0	0	1	1	3	2,099
46.	terus	0	0	1	0	0	1	3	2,099
47.	turun	1	0	0	0	0	1	3	2,099
48.	ungkap	0	1	0	0	0	1	3	2,099

Pada Tabel 3.12 dan Tabel 3.13 merupakan hasil perhitungan nilai IDF atau seberapa sering suatu *term* muncul pada dokumen-dokumen tersebut.

3. Menghitung Nilai TF-IDF

Setelah didapatkan nilai TF, DF, dan IDF maka langkah selanjutnya yaitu menghitung nilai TF-IDF, melalui persamaan berikut :

$$W_{dt} = tf_{dt} \cdot idf_t$$

Dengan W merupakan hasil dari perhitungan TF dikalikan dengan IDF (TF*IDF) Maka, hasil perhitungan TF-IDF dari dokumen diatas dapat dilihat pada Tabel 3.14 dan Tabel 3.15.

Tabel 3. 14 Contoh Hasil Perhitungan TF-IDF

No.	Term	W_{dt}			
		D1	D2	D3	D4
1.	abal	0	0	0	0
2.	ahli	0	0	0	0
3.	akibat	2,098612	0	0	0
4.	aksi	2,098612	0	0	0
5.	bagus	0	0	0	0
6.	batal	0	0	0	0
7.	berapa	0	0	0	0
8.	biar	0	2,098612	0	0
9.	bikin	0	0	0	0
10.	bisa	0	0	0	2,098612
11.	bubar	0	0	0	0

Tabel 3. 15 Contoh Hasil Perhitungan TF-IDF

No.	Term	W_{dt}			
		D1	D2	D3	D4
12.	caleg	0	0	0	0
13.	coba	0	2,098612	0	0
14.	curang	2,098612	0	0	0
15.	dah	0	0	0	0
16.	dalam	0	0	0	0
17.	dengan	0	0	0	2,098612
18.	dimana	0	0	0	0
19.	gulir	0	0	2,098612	0
20.	harus	0	0	2,098612	0
21.	iq	0	0	0	0
22.	kaderisasi	0	0	2,098612	0
23.	korupsi	0	0	0	0
24.	kuasa	2,098612	0	0	0
25.	makan	0	0	0	2,098612
26.	meroket	0	0	0	0
27.	negara	0	0	0	0
28.	ngalir	0	2,098612	0	0
29.	parpol	0	0	0	2,098612
30.	partai	1,182322	1,182322	1,182322	0
31.	penting	1,693147	0	0	1,693147
32.	pilkada	0	2,098612	0	0
33.	pilpres	0	0	0	2,098612
34.	politik	1,182322	1,182322	1,182322	0
35.	proses	0	0	2,098612	0
36.	protes	0	0	0	0
37.	rakyat	0	0	0	2,098612
38.	saja	0	0	0	1,693147
39.	sekolah	0	0	0	0
40.	semua	0	0	0	0
41.	seperti	0	0	0	2,098612
42.	seru	0	2,098612	0	0
43.	skeptis	0	2,098612	0	0
44.	sudah	0	0	0	2,098612
45.	tata	0	0	0	0
46.	terus	0	0	2,098612	0
47.	turun	2,098612	0	0	0
48.	ungkap	0	2,098612	0	0
	KELAS	-1	0	1	-1

Berdasarkan hasil perhitungan TF-IDF selanjutnya dituliskan dalam bentuk notasi *vector* yang akan digunakan untuk melakukan *training* dengan menggunakan metode SVM.

$$X1 = \{ 0, 0, 2.098612289, 2.098612289, 0, 0, 0, 0, 0, 0, 0, 0, 0, 2.098612289, 0, 0, 0, 0, 0, 0, 0, 0, 2.098612289, 0, 0, 0, 0, 0, 1.182321557, 1.693147181, 0, 0, 1.182321557, 0, 0, 0, 0, 0, 0, 0, 0, 0, 2.098612289, 0 \}$$

$$X2 = \{ 0, 0, 0, 0, 0, 0, 0, 2.098612289, 0, 0, 0, 0, 2.098612289, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 2.098612289, 0, 1.182321557, 0, 2.098612289, 0, 1.182321557, 0, 0, 0, 0, 0, 0, 0, 2.098612289, 2.098612289, 0, 0, 0, 0, 2.098612289 \}$$

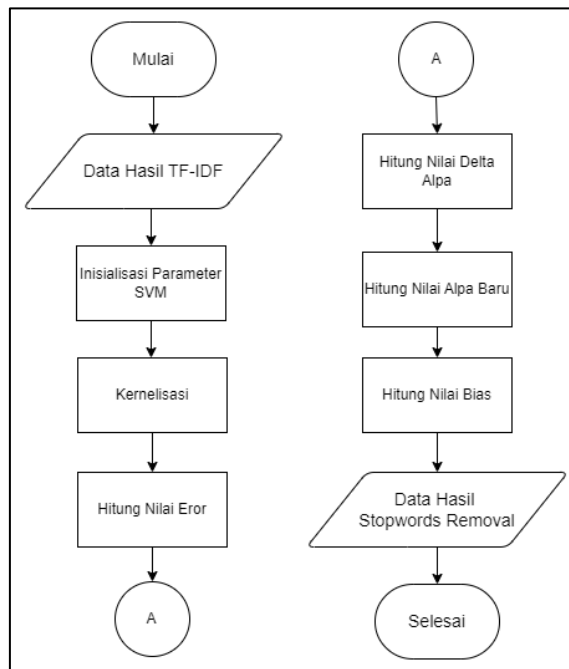
$$X3 = \{ 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 2.098612289, 2.098612289,$$

0, 2.098612289, 0, 0, 0, 0, 0, 0, 0, 0, 1.182321557, 0, 0,0, 1.182321557, 2.098612289, 0,0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 2.098612289, 0, 0 }

$X_4 = \{ 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 2.098612289, 0, 0, 0, 0, 0, 0, 2.098612289, 0, 0, 0, 0, 0, 0, 2.098612289, 0, 0, 0, 2.098612289, 0, 0, 0, 2.098612289, 0, 1.693147181, 0, 2.098612289, 0, 0, 0, 2.098612289, 1.693147181, 0, 0, 2.098612289, 0, 0, 2.098612289, 0, 0, 0, 0 \}$

3.1.5 Analisis Sentimen dengan Model SVM

Pada tahap ini yaitu menentukan hasil analisis sentiment berdasarkan data yang telah diproses sebelumnya. Untuk mendapatkan hasil analisis sentiment menggunakan metode *Support Vector Machine* (SVM). Berikut merupakan *flowchart* dari proses analisis sentiment dengan metode SVM dapat dilihat pada Gambar 3.10.



Gambar 3. 10 *Flowchart* Klasifikasi Model SVM

1. Melakukan inisialisasi parameter :
 - a. $C = 1$
 - b. $\lambda = 0,5$
 - c. $\gamma = 0,001$
 - d. $\varepsilon = 0,0001$
 - e. $\alpha = 0$
2. Setelah proses inisialisasi parameter yaitu mencari nilai kernel pada matriks K beserta matriks *hessian* yang berukuran $n \times n$ yaitu sebanyak data *training*. Berikut merupakan contoh perhitungannya.

$$k(x_i, x_j) = (x_i \cdot x_j + 1)^2$$

$$k(x_i, x_j) = ((0 \times 0) + (0 \times 0) + (2,098 \times 0) + (2,098 \times 0) + (0 \times 0) + (0 \times 0) + (0 \times 0) + (0 \times 2,098) + (0 \times 0) + (0 \times 0) + (0 \times 0) + (0 \times 0) + (0 \times 2,098) + (2,098 \times 0) + (0 \times 0) + (0 \times 0) + (0 \times 0) + (0 \times 0) + (0 \times 0) + (0 \times 0) + (0 \times 0) + (0 \times 0) + (0 \times 0) + (0 \times 0) + (2,098 \times 0) + (0 \times 0) + (0 \times 0) + (0 \times 0) + (0 \times 2,098) + (0 \times 0) + (1,1823 + 1,1823) + (0 \times 0) + (0 \times 0) + (0 \times 0) + (0 \times 0) +$$

$$(0 \times 0) + (0 \times 0) + (0 \times 0) + (0 \times 2,098) + (0 \times 2,098) + (0 \times 0) + (0 \times 0) + (0 \times 0) + (2,098 \times 0) + (0 \times 2,098) + 1)^2$$

$$k(x_i, x_j) = 14,41$$

Perhitungan tersebut terus dilakukan pada dokumen lainnya sehingga memperoleh hasil matriks K seperti pada Tabel 3.16.

Tabel 3. 16 Nilai Matriks K

Dokumen	X1	X2	X3	X4
X1	822,74	14,41	14,40786	14,95
X2	14,41	1198,89	14,40786	1
X3	14,40786	14,40786	458,49	1
X4	14,95	1	1	1410,96

Selanjutnya untuk memperoleh nilai matriks hessian, yaitu dilakukan perhitungan dengan proses perhitungan sebagai berikut

$$D_{ij} = y_i y_j (K(x_i, x_j) + \lambda^2)$$

$$D_{12} = (1)(0)((14.41) + 0,5^2))$$

$$D_{ij} = -14.66$$

Lakukan berulang perhitungan tersebut pada dokumen lainnya sehingga mendapatkan hasil matriks hessian yang terdapat pada Tabel 3.17 berikut :

Tabel 3. 17 Nilai Matriks Hessian

Dokumen	X1	X2	X3	X4
X1	822,99	-14,66	0	-15,2
X2	-14,66	1199,14	0	1,25
X3	0	0	0	0
X4	-15,2	1,25	0	1411,21

- Setelah proses perhitungan nilai matriks hessian, selanjutnya yaitu mencari nilai eror. Berikut persamaan yang digunakan

$$E_i = \sum_{j=1}^n a_j D_{ij}$$

$$E_i = (0 \times 822,99) + (0 \times -14,66) + (0 \times 0) + (0 \times -15,2)$$

$$E_i = 0$$

Ulangi perhitungan tersebut pada dokumen lainnya sehingga mendapatkan hasil nilai E_i yang terdapat pada Tabel 3.18 berikut:

Tabel 3. 18 Hasil Perhitungan E_i

Dokumen	E_i
D1	0
D2	0
D3	0
D4	0

- Selanjutnya mencari nilai $\delta\alpha_i$ melalui persamaan $\delta\alpha_i = \min\{\max[\gamma(1-E_i), \alpha_i], C - \alpha_i\}$. Berikut merupakan contoh perhitungannya: $\delta\alpha_i = \min\{\max[0,001(1-0), 0]1-0\}$

$$\delta\alpha_i = 0,001.$$

Ulangi perhitungan tersebut pada dokumen lainnya sehingga mendapatkan nilai $\delta\alpha_i$ seperti pada Tabel 3.19 berikut :

Tabel 3. 19 Hasil Perhitungan $\delta\alpha_i$

Dokumen	$\delta\alpha_i$
X1	0,001
X2	0,001
X3	0,001
X4	0,001

5. Tahap selanjutnya yaitu mencari nilai baru dengan menggunakan persamaan $\alpha_i = \alpha_i + \delta\alpha_i$
 $= 0 + 0,001$
 $= 0,001$

Ulangi perhitungan tersebut pada dokumen lainnya sehingga mendapatkan nilai α_i seperti pada Tabel 3.20 berikut :

Tabel 3. 20 Tabel Perhitungan α_i baru

Dokumen	α_i baru
X1	0,001
X2	0,001
X3	0,001
X4	0,001

6. Selanjutnya yaitu mengulang perhitungan pada tahap 3 sampai tahap 5 sampai mencapai batas maksimal iterasi. Berikut merupakan perhitungan iterasi kedua.
 $E_i = (0,001 \times 1874.81) + (0,001 \times (-45.59)) + (0,001 \times 0) + (0,001 \times (-45.59))$
 $= 1,78363$

Ulangi perhitungan tersebut pada dokumen lainnya sehingga mendapatkan hasil nilai E_i yang terdapat pada Tabel 3.21.

Tabel 3. 21 Hasil Perhitungan E_i iterasi ke-2

Dokumen	E_i
X1	0,79313
X2	1,18573
X3	0
X4	1,39726

7. Tahap ke-7 yaitu menghitung nilai $\delta\alpha_i$ pada iterasi ke-2. Berikut merupakan contoh perhitungan $\delta\alpha_i$ pada iterasi ke-2
 $\delta\alpha_i = \min\{\max[\gamma(1 - E_i), \alpha_i], C - \alpha_i\}$
 $\delta\alpha_i = \min\{\max[0,001(1 - 0,79313), 0,001], 1 - 0,001\}$
 $= 0,001$

Ulangi perhitungan tersebut pada dokumen lainnya sehingga mendapatkan hasil nilai $\delta\alpha_i$ yang terdapat pada Tabel 3.22.

Tabel 3. 22 Hasil Perhitungan $\delta\alpha_i$ Iterasi ke-2

Dokumen	$\delta\alpha_i$
D1	0,001
D2	0,001
D3	0,001
D4	0,001

$$\alpha_i = \alpha_i + \delta\alpha_i$$

$$= 0,001 + 0,001$$

$$= 0,002$$

Ulangi perhitungan tersebut pada dokumen lainnya sehingga mendapatkan hasil nilai $\delta\alpha_i$ yang terdapat pada Tabel 3.23.

Tabel 3. 23 Hasil Perhitungan α_i baru

Dokumen	α_i baru
D1	0,002
D2	0,002
D3	0,002
D4	0,002

8. Selanjutnya yaitu menghitung untuk mencari nilai bias dengan persamaan $b = -\frac{1}{2}((w \cdot x_{-1}) + (w \cdot x_{+1}))$. x_{-1} masuk pada golongan kelas negatif yang dimana memiliki nilai alpha paling besar, sedangkan x_{+1} masuk pada golongan kelas positif yang Dimana memiliki nilai alpha paling besar.

$$\begin{aligned} w \cdot x_{+1} &= (-1 \times 0.002 \times 14,40786 + (0 \times 0.002 \times 14,40786) + (1 \times 0.002 \times 458,49) \\ &\quad + (-1 \times 0.002 \times 1)) \\ &= 0,88616428 \end{aligned}$$

$$\begin{aligned} w \cdot x_{-1} &= (-1 \times 0.002 \times 14,95) + (0 \times 0.002 \times 1) + (1 \times 0.002 \times 1) \\ &\quad + (-1 \times 0.002 \times 1410,96) \\ &= -0,0279 \end{aligned}$$

Berdasarkan hasil kedua nilai tersebut, maka untuk mendapatkan nilai bias dapat dilakukan dengan perhitungan sebagai berikut:

$$\begin{aligned} b &= -\frac{1}{2}(0,88616428 + (0,0279)) \\ &= -0,42913214 \end{aligned}$$

9. Setelah didapatkan nilai bias, kemudian dilakukan proses analisis. Data uji yang digunakan pada proses klasifikasi dilakukan tahap *preprocessing* serta tahap pembobotan kata dengan TF-IDF. Berikut merupakan data hasil dari tahap *preprocessing* dan tahap pembobotan TF-IDF:

U1 : [4,197224577; 2,098612289; 0; 0; 2,098612289; 2,098612289; 2,098612289; 0; 2,098612289; 0; 2,098612289; 2,098612289; 0; 0; 2,098612289; 2,098612289; 0; 2,098612289; 0; 0; 2,098612289; 0; 2,098612289; 0; 0; 2,098612289; 2,098612289; 0; 0; 1,182321557; 0; 0; 0; 1,182321557; 0; 2,098612289; 0; 1,693147181; 2,098612289; 2,098612289; 0; 0; 0; 0; 2,098612289; 0; 0; 0]

10. Kemudian dilakukan tahap kernelisasi terhadap data uji.

$$k(x_i \cdot x_j) = (x_i \cdot x_j + 1)^2$$

$$\begin{aligned} k(x_i \cdot x_j) &= ((0 \times 4,197) + (0 \times 2,098) + (0 \times 0) + (0 \times 0) + (0 \times 2,098) + (0 \times 2,098) + \\ &\quad (0 \times 2,098) + (2,098 \times 0) + (0 \times 2,098) + (0 \times 0) + (0 \times 2,098) + (0 \times 2,098) + (2,098 \times 0) + \\ &\quad (0 \times 2,098) + (0 \times 0) + (0 \times 2,098) + (0 \times 2,098) + (0 \times 0) + (0 \times 2,098) + (0 \times 2,098) + (0 \times 0) \\ &\quad + (0 \times 2,098) + (0 \times 2,098) + (0 \times 2,098) + (0 \times 0) + (1,182 \times 1,182) + (0 \times 0) + (0 \times 2,098) \\ &\quad + (0 \times 0) + (0 \times 1,693) + (0 \times 2,098) + (0 \times 2,098) + (0 \times 0) + (2,098 \times 0) + (2,098 \times 0) + \\ &\quad (0 \times 0) + (0 \times 2,098) + (0 \times 0) + (0 \times 0) + (2,098 \times 0) + 1)^2 \end{aligned}$$

$$= k(x_i \cdot x_j) = 14,41$$

Berikut merupakan hasil kernelisasi pada seluruh data dapat dilihat pada Tabel 3.24 :

Tabel 3. 24 Nilai Perhitungan Data Uji

$k(x.x_1)$	$k(x.x_2)$	$k(x.x_3)$	$k(x.x_4)$
14,41	14,41	14,41	14,95

11. Tahap selanjutnya setelah didapatkan hasil kernelisasi pada keseluruhan data, yaitu menentukan kelas data menggunakan fungsi keputusan $f(x)$ persamaan 2.4. Fungsi $sign()$ merupakan fungsi normalisasi apabila nilai x yang dihasilkan pada fungsi $sign()$ adalah 1, maka data tersebut masuk kedalam kategori positif, jika nilai yang dihasilkan -1 data masuk kedalam kelas negatif, dan apabila nilai yang dihasilkan 0 maka data tersebut termasuk kedalam kelas netral.

$$f(x) = sign \sum_{i=1}^n a_i y_i K(x, x_i) + b$$

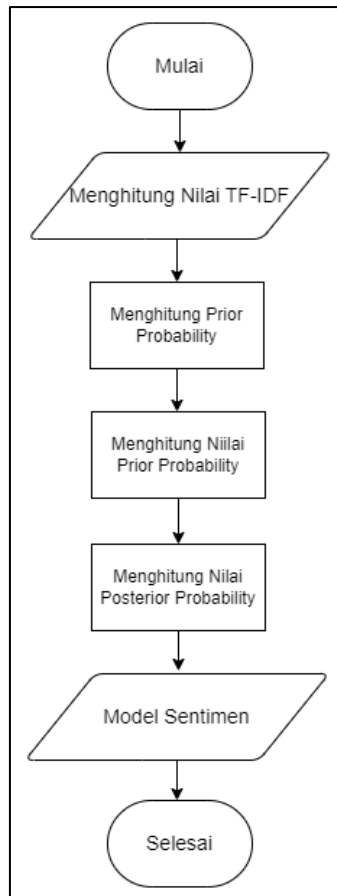
$$f(x) = sign \left((-1) \times 0.002 \times 14.41 + (0 \times 0.002 \times 14.41) + (1 \times 0.002 \times 14.41) + ((-1) \times 0.002 \times 14.96) + -0,42913 \right)$$

$$f(x) = -1$$

Sehingga hasil perhitungan fungsi keputusan untuk data uji U1 menghasilkan nilai = -1 yang berarti data uji tersebut termasuk pada kelas sentimen negatif.

3.1.6 Pembuatan Model Sentimen

Untuk mendapatkan hasil akhir dari penelitian ini, yaitu klasifikasi sentiment publik terhadap partai politik di yang terdapat pada sosial media X, langkah yang diambil adalah melakukan klasifikasi menggunakan metode Support Vector Machine (SVM). Berikut merupakan flowchart pembuatan model sentiment dapat dilihat pada Gambar 3.9.



Gambar 3. 11 *Flowchart* Pembuatan Model Sentimen

Flowchart proses klasifikasi sentimen menggunakan metode Support Vector Machine (SVM) dimulai dengan mengambil hasil dari proses term frequency-inverse document frequency (TF-IDF), yang digunakan untuk mengekstraksi fitur dari data ulasan yang akan diklasifikasikan. Setelah itu, dilakukan perhitungan prior probability sebagai langkah awal dalam memodelkan probabilitas. Kemudian, probabilitas dihitung berdasarkan fitur-fitur yang telah diekstraksi. Selanjutnya, dilakukan perhitungan posterior probability untuk menentukan kelas akhir dari data yang diklasifikasikan. Hasil akhir dari proses ini adalah pengelompokan data ke dalam kategori sentimen yang tepat, seperti positif, negatif, atau netral. Proses ini akan diulang dari awal jika ada data baru yang perlu diklasifikasikan.

3.1.7 Pengujian

Pada penelitian ini pengujian sistem dilakukan menggunakan *confusion matrix* untuk mengevaluasi performa model yang diterapkan. Pengujian ini bertujuan untuk mengukur tingkat kinerja metode *Support Vector Machine* yang digunakan dalam penelitian ini. Dari *confusion matrix*, beberapa nilai kinerja yang diperoleh meliputi akurasi, presisi, dan *recall* model. Pada penelitian ini, data dibagi menjadi 90% data latih (*training data*) dan 10% data uji (*testing data*). Pengujian dilakukan dengan menggunakan *package scikit-learn* yang terdapat pada bahasa pemrograman *Python*, di mana *package* tersebut dapat menghasilkan *confusion matrix* beserta nilai akurasi, presisi, dan *recall*. secara otomatis. Rancangan pengujian *confusion matrix* dapat dilihat pada Tabel 3.25.

Tabel 3. 25 Tabel Rancangan Pengujian

Aktual	Negative	Neutral	Positive
Negative	TN (True Negative)	FL2 (False Neutral 2)	FP2 (False Positive)
Neutral	FN2 (False Negative2)	TL(True Neutral)	FP2 (False Positive2)
Positive	FN (False Negative)	FL (False Neutral)	TP (True Positive)
Jumlah	N'	L'	P'

3.2 Metode Pengembangan Sistem

Dalam pengembangan sistem ini, metode yang digunakan adalah metode *Waterfall*. Metode *Waterfall* merupakan metode yang terdiri dari tahap *requirements analysis*, *system and software design*, *implementations*, dan *system testing*. Berikut merupakan uraian terkait dengan masing masing proses dari metode *Waterfall* yang digunakan sebagai metode pengembangan sistem :

3.2.1 Requirements Analysis

Tahapan awal ini merupakan tahapan yang berfokus pada pencarian kebutuhan apa saja yang bisa digunakan dalam proses pembuatan sistem ini diawali dengan mengidentifikasi kebutuhan keseluruhan yang akan diterapkan serta diimplementasikan menjadi sebuah perangkat lunak atau *software*. Tahap awal ini melibatkan proses pengumpulan data, di mana opini pengguna X yang terdapat pada kolom komentar sosial media X terkait dengan keyword yang digunakan dikumpulkan menggunakan teknik *scrapping*. Data-data opini yang diperoleh kemudian disimpan ke dalam suatu file untuk dianalisis lebih lanjut. Selain itu, pada tahap ini juga dilakukan pengumpulan referensi studi literatur yang relevan dengan penelitian yang dilakukan.

1. Kebutuhan Fungsional

- a. Kemampuan sistem untuk melakukan proses *preprocessing* terhadap dataset yang berhasil di *crawling* melalui sosial media X.
- b. Sistem berhasil melakukan pelabelan data dengan metode *TF-IDF*.
- c. Sistem berhasil untuk mengklasifikasikan komentar dengan kelas analisis sentimen dengan algoritma *Support Vector Machine*.
- d. Sistem mampu untuk membaca komentar dari hasil inputan user untuk keperluan pengujian sistem.
- e. Sistem mampu untuk memberikan *output* nilai akurasi dari operasi perhitungan dengan seluruh dataset yang sudah berhasil di *crawl*.

2. Kebutuhan Non-Fungsional

Untuk mengimplementasikan sistem yang akan dibangun sesuai dengan rancangan yang telah dibuat, diperlukan beberapa perangkat keras (*hardware*), perangkat lunak (*software*), dan pengguna sebagai dukungan dalam pengembangan sistem analisis. Berikut adalah analisis kebutuhan non-fungsional yang diperlukan untuk merancang sistem :

a. Kebutuhan *Hardware*

Berikut merupakan spesifikasi *hardware* yang dipakai dalam pembuatan sistem pada penelitian ini dapat dilihat pada Tabel 2.26

Tabel 3. 26 Kebutuhan Hardware

No.	Hardware	Keterangan
1.	<i>Processor</i>	AMD Ryzen 7 4700U <i>Radeon Graphics</i> (8CPU) 2.0GHz
2.	RAM	8192 MB
3.	HDD	512 GB
4.	Perangkat Input dan Output	<i>Mouse, Keyboard</i>
5.	Koneksi	<i>WiFi, Seluler</i>

b. Kebutuhan *Software*

Berikut merupakan spesifikasi *hardware* yang dipakai dalam pembuatan suatu sistem dalam penelitian ini terdapat pada Tabel 3.27 :

Tabel 3. 27 Kebutuhan Software

No.	Software	Keterangan
1.	<i>Operating System</i>	Windows 11, 64 bit
2.	<i>Visual Studio</i>	<u>Text Editor yang digunakan untuk menjalankan program</u>
3.	<i>Chrome</i>	Web browser
4.	<i>Python</i>	Bahasa pemrograman yang digunakan dalam pembuatan sistem
5.	<i>Draw.io</i>	<i>Design flow</i> sistem

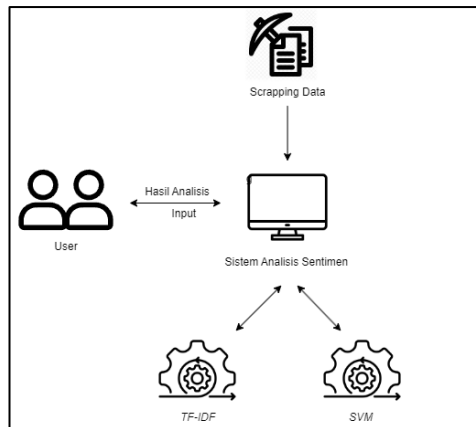
3.2.2 System And Software Design

Merupakan tahapan untuk melakukan perancangan mengenai sistem dan juga perangkat lunak yang akan dibangun dan digunakan dalam penelitian ini. Penting dan krusial merupakan alasan utama mengapa diperlukannya tahap ini. Adapun tahapannya:

1. Perancangan Arsitektur

Beberapa arsitektur sistem yang akan dikembangkan dalam penelitian ini mencakup elemen-elemen seperti pengguna (user), *web scrapper*, model SVM berbasis *Lexicon Based*, serta sistem analisis sentimen terhadap opini partai politik. Pengguna (user) dalam sistem ini adalah individu yang dapat menggunakan atau mengoperasikan sistem secara umum serta melakukan pengecekan terhadap opini untuk mengetahui polaritas analisis sentimennya.

Pada sistem analisis sentimen ini, terdapat beberapa tahap di mana data opini publik diperoleh melalui proses scraping dari platform media sosial X. Data tersebut kemudian dilabeli menggunakan kamus *Lexicon* dan melalui tahapan *preprocessing*. Setelah tahap *preprocessing* selesai, sistem melanjutkan dengan proses analisis sentimen berdasarkan model SVM yang telah dirancang, di mana model ini sebelumnya sudah melalui proses pelatihan (*training*), validasi, dan pengujian (*testing*). Jika model SVM sudah siap digunakan, sistem akan menampilkan hasil analisis sentimen untuk admin dan pengguna (user). Sistem ini juga dapat menampilkan dan menyimpan data yang berasal dari dan menuju basis data. Ilustrasi arsitektur sistem dapat dilihat pada Gambar 3.12 berikut.



Gambar 3. 12 Arsitektur Sistem

2. Perancangan Proses

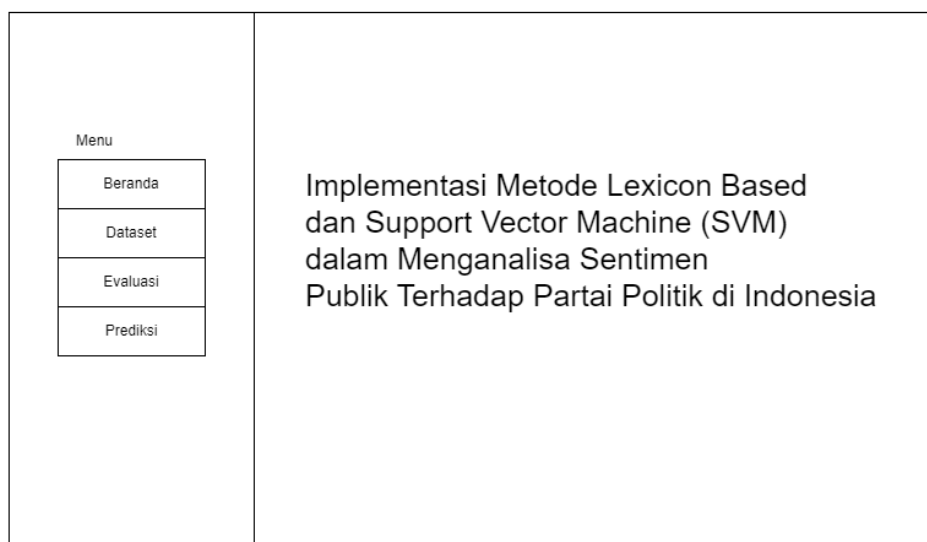
Perancangan ini menggambarkan tahapan dalam analisis sentimen pada data tweet menggunakan metode TF-IDF dan Support Vector Machine (SVM). Proses dimulai dengan mengumpulkan data tweet melalui teknik scraping. Setelah data terkumpul, dilakukan preprocessing pada data untuk membersihkannya. Kemudian, data tweet yang telah diproses diberi bobot menggunakan metode TF-IDF, yang menonjolkan kata-kata yang relevan dengan memberikan bobot lebih tinggi pada kata yang sering muncul di dokumen tertentu tetapi jarang di seluruh corpus. Hasil dari pembobotan TF-IDF ini kemudian digunakan sebagai input bagi model SVM untuk mengklasifikasikan sentimen dalam data tweet tersebut. Tahap akhir adalah klasifikasi sentimen, di mana model SVM yang telah dilatih mengategorikan tweet ke dalam sentimen seperti positif, negatif, atau netral. Setelah klasifikasi selesai, proses pun berakhir. *Flowchart* ini memberikan gambaran lengkap mengenai alur pengumpulan, pengolahan, dan analisis data *tweet* dengan metode TF-IDF dan SVM guna menentukan sentimen yang terkandung dalam tweet tersebut.

3.2.3 Rancangan Interface

Rancangan antarmuka pengguna adalah proses desain yang berfokus pada membangun model komunikasi antara pengguna dan sistem. Rancangan ini mencakup berbagai aspek, seperti tata letak dan navigasi, dengan tujuan agar pengguna dapat berinteraksi dengan sistem secara mudah dan efektif. Dalam sistem ini, terdapat dua aktor utama, yaitu admin dan pengguna. Berikut ini merupakan gambaran rancangan antarmuka pengguna, yang terbagi ke dalam beberapa bagian sebagai berikut :

1. Rancangan Halaman Beranda

Halaman ini merupakan halaman awal sistem. Rancangan halaman utama pengunjung dapat dilihat pada Gambar 3.13 berikut :



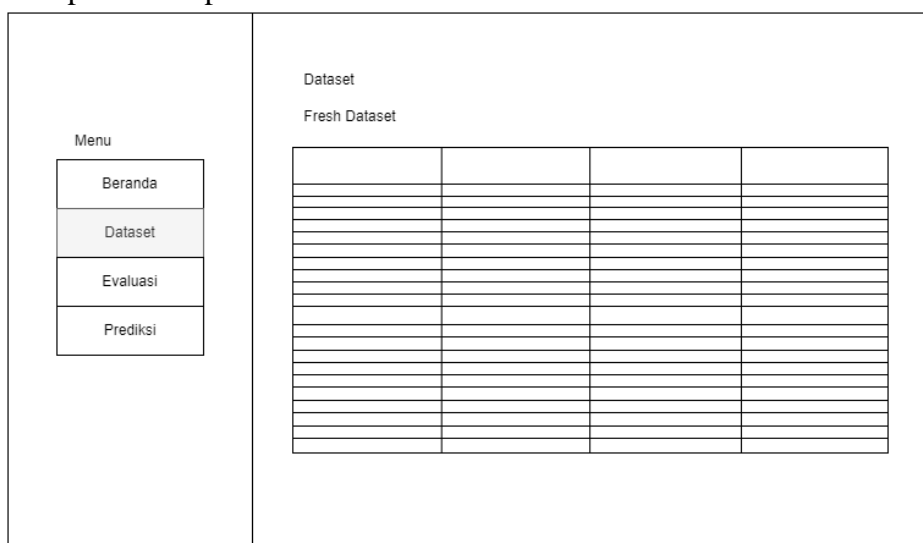
Gambar 3. 13 Rancangan Halaman Beranda

2. Rancangan Halaman Dataset

Pada halaman dataset, terdapat beberapa data yang bisa dilihat dan ditampilkan, Adapun data yang dapat dilihat :

a. Rancangan Halaman *Dataset Fresh*

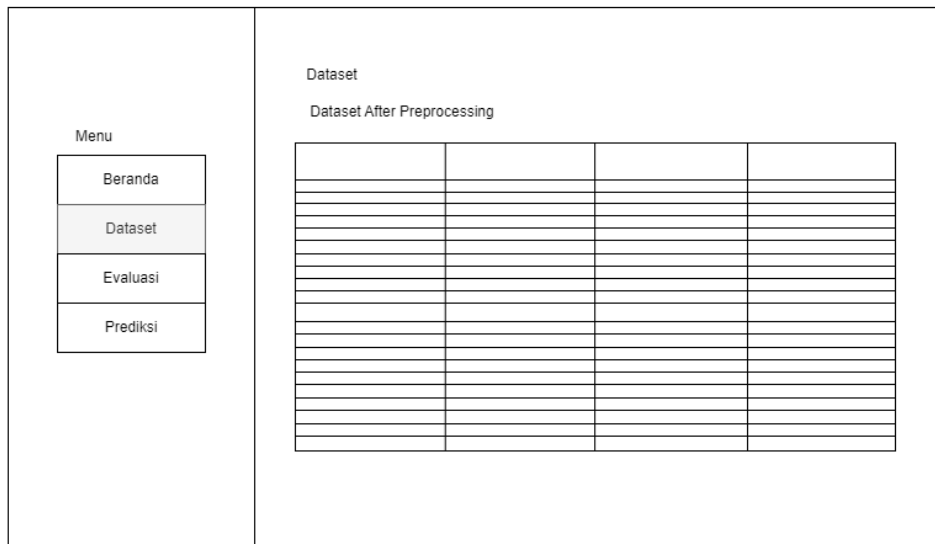
Halaman ini merupakan halaman untuk memperlihatkan dataset yang digunakan sebelum melakukan proses preprocessing apapun. Rancangan halaman *Dataset Fresh* dapat dilihat pada Gambar 3.14 berikut



Gambar 3. 14 Rancangan Halaman Dataset

b. Rancangan Halaman *Dataset After Sentiment Preprocessing*

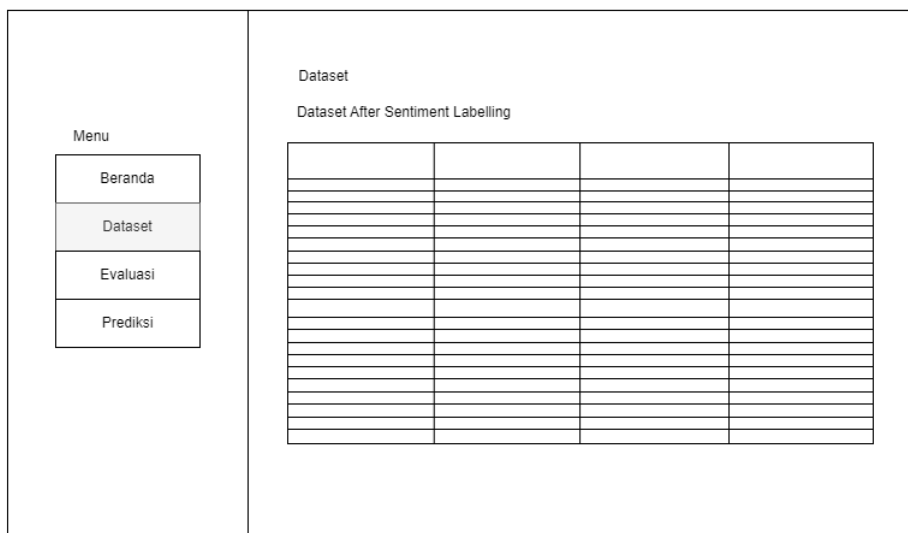
Halaman ini merupakan halaman untuk menampilkan dataset hasil proses *preprocessing* dapat dilihat pada Gambar 3.15 berikut :



Gambar 3. 15 Rancangan Halaman *Dataset After Sentiment Preprocessing*

c. Rancangan Halaman *Dataset after Sentiment Labelling*

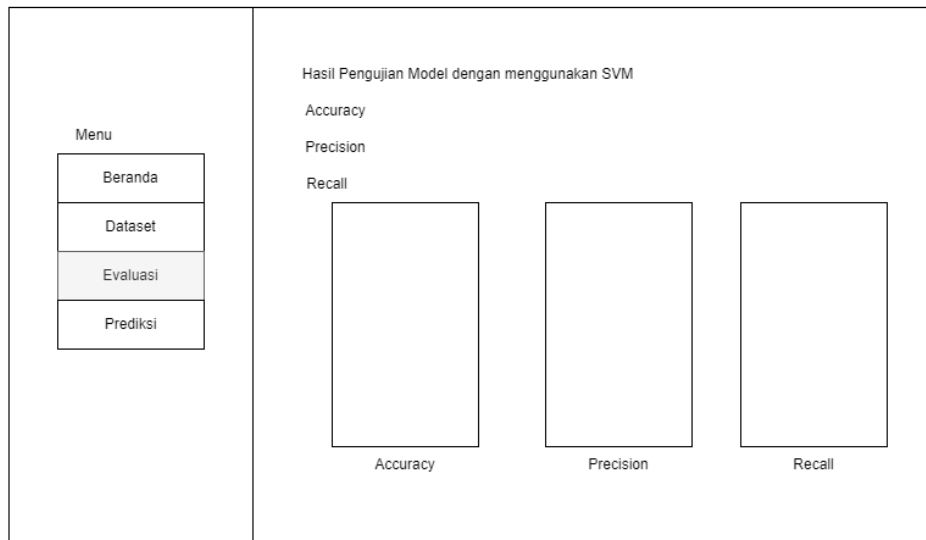
Halaman ini merupakan halaman untuk menampilkan dataset hasil pelabelan sentiment. Dapat dilihat pada Gambar 3.16 berikut :



Gambar 3. 16 Rancangan *Dataset After Sentiment Labelling*

3. Rancangan Halaman Evaluasi

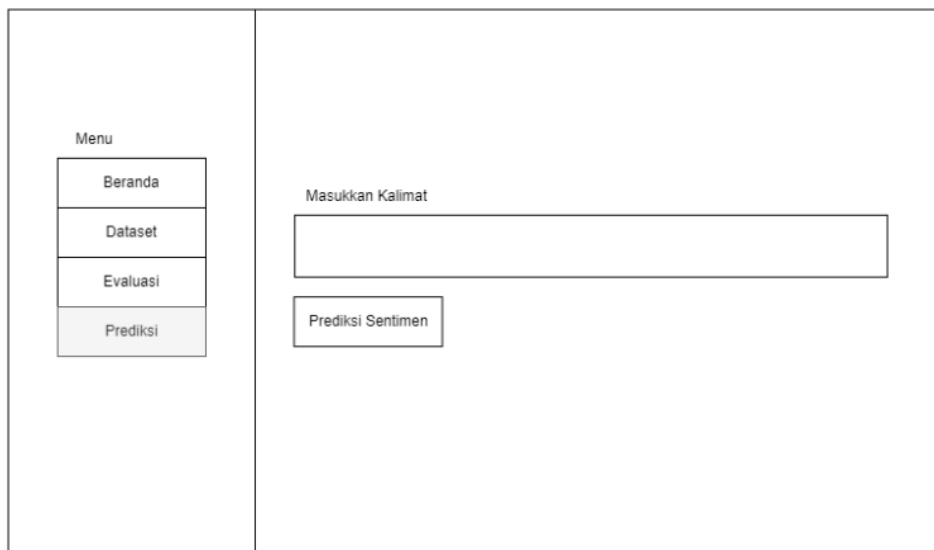
Halaman ini merupakan halaman untuk menampilkan hasil evaluasi model SVM. Dapat dilihat pada Gambar 3.17 berikut :



Gambar 3. 17 Rancangan Halaman Evaluasi

4. Rancangan Halaman Prediksi

Halaman ini merupakan halaman yang digunakan untuk melakukan prediksi sentimen. Dapat dilihat pada Gambar 3.18 berikut :



Gambar 3. 18 Rancangan Halaman Prediksi

3.2.4 Implementation

Tahap implementasi merupakan fase di mana desain dan perancangan sistem yang telah dibuat direalisasikan menjadi kode program untuk menghasilkan sebuah sistem yang berfungsi. Sistem analisis sentimen ini diimplementasikan dan dirancang sesuai dengan metode yang digunakan, berdasarkan tahapan analisis sentimen untuk

3.2.5 System Testing

Tahap akhir ini merupakan tahapan untuk pengujian sistem. Pengujian ini bertujuan untuk menguji rancangan sistem yang sudah dibangun sebelumnya agar dapat melakukan evaluasi dari model yang digunakan untuk proses klasifikasi yang sudah di jalankan. *Black box*

merupakan metode yang digunakan untuk tahapan *system testing* ini. Metode *Black box testing* memiliki tujuan untuk mengetahui efektivitas, keberhasilan serta kemampuan kerja sistem yang sudah dikembangkan. Selain itu juga digunakannya metode *Black box* bertujuan untuk memastikan fungsi setiap fitur yang terdapat pada sistem yang sudah dibangun sesuai dengan yang seharusnya. Berikut detail dari hasil pengujian sistem menggunakan metode *Black box testing* dapat dilihat pada Tabel 2.28:

Tabel 3. 28 Rancangan Pengujian *Black Box*

Role	Halaman	Detail Pengujian	Berhasil	Tidak Berhasil
User	Beranda	Menampilkan Tampilan awal sistem dirancang		
	Dataset	Menampilkan halaman dataset yang digunakan dari mulai data set mentah sampai dataset yang sudah berhasil dilabelling		
	Evaluasi	Halaman dapat menampilkan hasil evaluasi model SVM dengan menunjukkan nilai <i>accuracy</i> , <i>precission</i> , dan <i>recall</i>		
	Prediksi	Dapat menginputkan kalimat untuk diuji prediksi sentimennya		

BAB IV HASIL DAN PEMBAHASAN

4.1 Implementasi

Bab ini berisi pengimplementasi dalam penelitian yang akan menguraikan secara rinci tentang penerapan rancangan yang telah disusun sebelumnya, serta menjelaskan langkah-langkah implementasi yang dilakukan dalam membangun model klasifikasi. Selain itu, bab ini juga akan membahas proses pengembangan sistem yang digunakan dalam penelitian, yang mencakup bagaimana sistem dibangun dan diintegrasikan untuk mendukung analisis sentimen publik terhadap partai politik di Indonesia. Penjelasan akan mencakup penggunaan metode *Lexicon-Based* untuk pelabelan sentimen, penerapan model SVM untuk klasifikasi, serta pengujian dan pengembangan sistem untuk memastikan kinerja yang optimal dalam proses analisis data.

4.1.1 Implementasi Model Klasifikasi

Pada bagian ini, akan dijelaskan hasil dari implementasi model klasifikasi yang telah dibangun berdasarkan rancangan yang telah disusun sebelumnya. Pembahasan akan difokuskan pada algoritma yang diterapkan dalam proses klasifikasi, khususnya penggunaan metode *Lexicon-Based* untuk pelabelan sentimen dan model *Support Vector Machine* (SVM) yang digunakan untuk mengklasifikasikan sentimen publik terhadap partai politik di Indonesia. Selain itu, bagian ini akan menguraikan bagaimana kedua metode tersebut diintegrasikan untuk menghasilkan model yang dapat mengidentifikasi sentimen positif, netral, dan negatif dengan akurasi yang optimal.

1. Pengumpulan Data

Pengumpulan data dilakukan dengan menggunakan teknik *web crawling* pada komentar *web X* dari bulan Oktober 2020 hingga Oktober 2024. Data diambil dengan bantuan *library Tweet Harvest* dan selanjutnya disimpan dalam format file *CSV*. Berikut merupakan modul program yang digunakan dalam proses *web crawling* dapat dilihat pada Algoritma 1.

Algoritma 1 : Web Crawling

```
Input: twitter_auth_token, search_keyword, limit
Output: df (pandas Dataframe)
Function twitter_data_collection(twitter_auth_token, search_keyword, limit)
    install pandas
    update system packages
    install CA certificates, curl, and gnupg
    add nodesource repository for Node.js
    filename <- 'keyword1.csv'
    execute tweet-harvest to crawl tweets with search_keyword, save results
to filename, limit results to limit, use twitter_auth_token
    file_path <- construct file path 'tweets-data/' and filename
    df <- read CSV file from file_path using pandas
    return df
end function
```

2. Preprocessing Data

Setelah data berhasil diperoleh, data tweet akan melalui tahap preprocessing untuk membersihkan dan meningkatkan kualitas data, agar siap digunakan dalam analisis lebih

lanjut. Proses awal dari preprocessing ini mencakup impor data dengan menggunakan fungsi `read_csv()` yang berfungsi untuk membaca file data yang telah disimpan sebelumnya. Berikut merupakan modul program yang digunakan dalam proses *import dataset* dapat dilihat pada Algoritma 2.

Algoritma 2 : Import Dataset

```

Input: filename
Output: df (pandas Dataframe)
Function read_and_display_data(filename)
    import pandas library as pd
    file_path <-construct file path to "data?" and filename
    display df
    return df
end function

```

a. *Cleansing*

Setelah data diimpor, langkah selanjutnya adalah proses cleansing yang bertujuan untuk menghapus tanda baca, angka, dan simbol yang terdapat dalam data tweet, guna memastikan data yang digunakan lebih bersih dan siap untuk analisis lebih lanjut. Berikut merupakan modul program yang digunakan dalam proses *cleansing* dapat dilihat pada Algoritma 3.

Algoritma 3 : Cleansing

```

# remove RT tag
text = re.sub(r'RT\s', '', text)
# remove @_username
text = re.sub(r"@([\w]+)", " ", text)
# replace emoji decode with space
text = re.sub(r"\u[a-zA-Z0-9]{4}", " ", text)
# replace enter /n/ with space
text = re.sub(r"\n\s", " ", text)
text = re.sub(r"\n", " ", text)
# remove non-ascii
text = re.sub(r'^\x00-\x7F+', ' ', text)
# fix duplicate characters (ex: hellooooo)
text = re.sub(r'([a-zA-Z])\1\1','\1', text)
# replace url
text = re.sub(r'http[s]?://\.[a-zA-Z0-9.\//\?=%&#\-\+!]+', ' ', text)
text = re.sub(r'pic.twitter.com?[a-zA-Z0-9.\//\?=%&#\-\+!]+', ' ', text)

```

b. *Case Folding*

Case folding, yang bertujuan untuk menyamakan huruf kapital dan huruf kecil agar tidak ada perbedaan antara kata yang memiliki kapitalisasi berbeda namun memiliki makna yang sama. Berikut merupakan modul program yang digunakan dalam proses *case folding* dapat dilihat pada Algoritma 4.

Algoritma 4 : Case Folding

```

# convert to lowercase
text = text.lower()

```

c. *Kamus Slang*

Proses pengimplementasian kamus slang dilakukan untuk mengganti kata-kata slang atau tidak baku yang sering ditemukan dalam tweet dengan kata yang lebih standar, sehingga model dapat lebih akurat dalam mengklasifikasikan sentimen. Berikut

merupakan modul program yang digunakan dalam proses *slang word* dapat dilihat pada Algoritma 5.

Algoritma 5 : Kamus Slang

```
def dictionaryofslang1(text):
    words = text.split()
    standardization_words = []

    for word in words:
        if word in result_kamus_slang1:
            standardization_words.append(result_kamus_slang1[word])
        else:
            standardization_words.append(word)

    return " ".join(standardization_words)

def dictionaryofslang2(text):
    words = text.split()
    standardization_words = []

    for word in words:
        if word in result_kamus_slang2:
            standardization_words.append(result_kamus_slang2[word])
        else:
            standardization_words.append(word)

    return " ".join(standardization_words)

DATA_FRAME_DATASET["slangbersih"] =
DATA_FRAME_DATASET["databersih"].apply(dictionaryofslang1)
DATA_FRAME_DATASET["slangbersih"] =
DATA_FRAME_DATASET["slangbersih"].apply(dictionaryofslang2)
display(DATA_FRAME_DATASET["slangbersih"].tail().to_frame())
```

d. Negasi

Selanjutnya, untuk menangani kata-kata yang memiliki pengaruh negatif dalam analisis sentimen, dilakukan langkah negasi. Proses ini bertujuan untuk menangani kata-kata seperti "tidak", "bukan", yang dapat mengubah makna kalimat dan perlu dipertimbangkan secara khusus. Berikut merupakan modul program yang digunakan dalam proses *negation handling* dapat dilihat pada Algoritma 6.

Algoritma 6 : Negasi

```
def garisbawah(text):
    words = text.split()
    hurufnegasi= set(dataframe_negasi1["negasi"].values)
    skip_next = False
    new_words = []
    for i in range(len(words)):
        if skip_next:
            skip_next = False
            continue
        if words[i] in hurufnegasi and i < len(words) - 1:
            new_words.append(words[i] + "_" + words[i+1])
            skip_next = True
        else:
            new_words.append(words[i])
    return " ".join(new_words)
```

```
DATA_FRAME_DATASET["negasibersih"] =
DATA_FRAME_DATASET["slangbersih"].apply(garisbawah)
display(DATA_FRAME_DATASET["negasibersih"].tail().to_frame())
```

e. *Stopwords Removal*

Proses berikutnya adalah stopwords removal, di mana kata-kata umum yang tidak memiliki makna signifikan, seperti "dan", "di", "untuk", akan dihapus. Langkah ini bertujuan untuk mengurangi dimensi data dan meningkatkan efisiensi model dalam menganalisis sentimen. Berikut merupakan modul program yang digunakan dalam proses *stopword removal* dapat dilihat pada Algoritma 7.

Algoritma 7 : Stopwords Removal

```
komb_stopwords = stopwords.words('indonesian')
komb_stopwords.clear()
komb_stopwords.extend(dataframe_stopword1["stopwords"].values)

factory = StopWordRemoverFactory()
swsastrawi = factory.get_stop_words()

komb_stopwords = set(komb_stopwords).union(set(swsastrawi))

def drop_stopwords(text):
    return " ".join([word for word in text.split() if word not in
komb_stopwords])

DATA_FRAME_DATASET["textsetelahsw"] =
DATA_FRAME_DATASET["swapnegasibersih"].apply(drop_stopwords)
display(DATA_FRAME_DATASET["textsetelahsw"].tail().to_frame())
```

f. *Stemming*

Proses *Stemming* bertujuan untuk mengubah kata-kata turunan menjadi bentuk dasarnya, misalnya "berjalan" menjadi "jalan". Proses ini membantu mengurangi variasi kata dan membuat model lebih fokus pada makna inti dari kata tersebut. Berikut merupakan modul program yang digunakan dalam proses *stemming* dapat dilihat pada Algoritma 8.

Algoritma 8 : Stemming

```
factory = StemmerFactory()
stemmer = factory.create_stemmer()

def stem_indonesian_text(text, index):
    # logger
    if index % 100 == 0:
        logger.info(f"stemming progress: {index}/{DATA_FRAME_DATASET_LENGTH}")

    return " ".join([stemmer.stem(word) for word in text.split()])

# apply stemming with streaming logs
for index, row in DATA_FRAME_DATASET.iterrows():
    DATA_FRAME_DATASET.at[index, "textsetelahstem"] =
stem_indonesian_text(row["textsetelahsw"], index)

display(DATA_FRAME_DATASET["textsetelahstem"].tail().to_frame())
```

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

Langkah selanjutnya dalam penelitian ini adalah penerapan proses pembobotan TF-IDF (*Term Frequency-Inverse Document Frequency*), yang bertujuan untuk memberikan bobot pada setiap istilah atau term dalam data teks. Pembobotan ini dilakukan dengan menghitung nilai *Term Frequency* (TF), yang menggambarkan frekuensi kemunculan suatu kata dalam dokumen, serta *Inverse Document Frequency* (IDF), yang mengukur seberapa penting suatu kata dalam keseluruhan koleksi dokumen. Proses pembobotan ini bertujuan untuk memberikan bobot yang lebih tinggi pada kata-kata yang muncul secara spesifik dalam dokumen, tetapi jarang muncul dalam dokumen lain, sehingga dapat memberikan kontribusi yang lebih besar pada model. Modul pembobotan TF-IDF ini menggunakan fungsi `TfidfVectorizer()`, yang berfungsi untuk menghitung dan mengaplikasikan nilai-nilai TF-IDF pada setiap term dalam dataset yang digunakan, menghasilkan representasi vektor yang lebih informatif untuk analisis selanjutnya. Berikut merupakan modul program yang digunakan dalam proses vektorisasi TF-IDF dapat dilihat pada Algoritma 9.

Algoritma 9 : Vektorisasi TF-IDF

```
import pandas as pd
from sklearn.feature_extraction.text import TfidfVectorizer

# Membersihkan nilai NaN
train_df = train_df.dropna(subset=['ind_preprocessed_text'])

# Inisialisasi TfidfVectorizer
vectorizer = TfidfVectorizer()

# Fit transform data teks
x = vectorizer.fit_transform(train_df['ind_preprocessed_text'])

# Mendapatkan nama fitur
vectorizer.get_feature_names_out()

# Membuat DataFrame dari matriks TF-IDF
df = pd.DataFrame(x.todense().T,
                  index=vectorizer.get_feature_names_out(),
                  columns=[f'D{i+1}' for i in range(len(train_df))])

print(df)
df.to_csv('TF-IDF_Parpol.CSV', encoding='utf-8', index=False)
```

4. Pelabelan *Lexicon Based*

Setelah proses preprocessing selesai, data yang telah diproses akan melalui tahap pelabelan data, di mana setiap data akan diberi label berdasarkan aspek sentimen yang terkandung dalam teks tersebut. Pelabelan ini dilakukan menggunakan kamus lexicon, yang berisi daftar kata-kata beserta nilai sentimennya. Berdasarkan analisis kata-kata dalam ulasan, setiap teks akan diberi label sesuai dengan sentimen yang tercermin: ulasan yang bernilai positif akan diberi label 1, ulasan yang bernilai netral akan diberi label 0, dan ulasan yang bernilai negatif akan diberi label -1. Proses pembacaan data hasil preprocessing dilakukan menggunakan fungsi `read_csv()`, yang memungkinkan data yang telah dibersihkan dan diproses untuk dimuat dan digunakan dalam tahap selanjutnya, yaitu

pelatihan model. Berikut merupakan modul program yang digunakan dalam proses *labelling* data dapat dilihat pada Algoritma 10.

Algoritma 10 : Pelabelan *Lexicon*

```
# InSet
with open("./resultpreprocessing/inset1.txt") as f:
    inset_neg = f.read()
with open("./resultpreprocessing/inset2.txt") as f:
    inset_pos = f.read()

insetNeg = json.loads(inset_neg)
insetPos = json.loads(inset_pos)

# inisiasi vader sentiment analyzer
sia_sen = SentimentIntensityAnalyzer()

# clear default lexicon
sia_sen.lexicon.clear()

# assign InSet lexicon
sia_sen.lexicon.update(sen)

# polarity score
def get_inset_compound_score(text):
    inset_compound_score = sia_sen.polarity_scores(text) ["compound"]
    return inset_compound_score

DATA_FRAME_PREPROCESSED_DATASET["compound_score"] =
DATA_FRAME_PREPROCESSED_DATASET["ind_preprocessed_text"].apply(get_inset_compound_score)
display(DATA_FRAME_PREPROCESSED_DATASET.tail(1))

# labelling
def get_inset_label(number):
    if number < 0:
        return "negatif"
    elif number == 0:
        return "netral"
    elif number > 0:
        return "positif"
    else:
        return

DATA_FRAME_PREPROCESSED_DATASET["label"] =
DATA_FRAME_PREPROCESSED_DATASET["compound_score"].apply(get_inset_label)
display(DATA_FRAME_PREPROCESSED_DATASET.tail(1))
```

5. Pelabelan Manual

Selain menggunakan kamus, pelabelan pada proses analisis sentimen juga dapat dilakukan dengan proses manual. Namun jika dibandingkan dengan pelabelan lainnya, metode pelabelan manual masih ada beberapa kekurangan, antara lain waktu pelabelan yang cenderung panjang, serta hasil pelabelan tidak bisa dilepaskan dari subjektivitas pelabel, yang pasti memiliki makna berbeda apabila dilakukan lebih dari 1 orang pelabel, berikut hasil pelabelan manual yang dilakukan oleh 2 orang mahasiswa. Berikut merupakan contoh hasil pelabelan data manual dapat dilihat pada Tabel

Tabel 4. 1 Contoh Hasil Pelabelan Manual

Hasil Crawl	Hasil Label 1	Hasil Label 2
rakyat indonesia partai politik mana	Positif	Netral
tum partai politik bicara sembarang iya	Negatif	Netral
bocil tum partai politik iya lenge sok asik main kritik publik bocil banget	Negatif	Negatif
ridwan kamil bahagia partai politik on fire pekan pertama kampanye putar	Positif	Netral
wkwkwkwk afiliasi mafia didik kayak asih orang partai politik hc hc an	Netral	Negatif
heran buka jejak digital brutal fufufafa duga kait gibran difahami usaha adu domba fihak kalah pilpres lolos gibran cawapres partai politik milik kuasa jadi gibran cawapres eksekutif	Negatif	Negatif
tukar guling partai politik sri rahayu mundur arteria eh arteria mundur sang cucu proklamator dewan wakil rakyat ealahhh partai politik dinasti	Negatif	Netral
pekan pertama kampanye ridwan kamil bahagia partai politik on fire raih target putar	Positif	Netral
ml konglo terjun politik nikahin anak teman mama keluarga kena gede fl ml ambil alih partai politik ganti koalisi pas pilpres fl nyangkanya nikah batas tuju ml cinta benar	Positif	Negatif
bu mega kasih hc bbrapa univ ln asli univ ln bukan yang pusat bekas beliau kasih orasi ilmiah beliau lulus sama pantesnya gelar kontribusi bidang ketua partai politik presiden raffi hm	Netral	Netral
menteri partai politik kaya	Positif	Positif
takud banget acil partai partai politik	Netral	Netral
warga percaya sistem demokrasi via partai politik pilih partai politik tum resep mujarab nihil kawula maju pilih kepala daerah pileg	Positif	Netral
aju gugat ptun partai patuh uu partai politik henti legislator pilih	Netral	Netral
bablas iku bamsoet partai politik reformasi soeharto hidup hidup partai politik tidak banyak	Netral	Netral
bukti tidak suka bilang tugas partai politik hak jawabanya gampang bang negeri orang orang partai politik susah cari orang orang baik baik deh adab tingkat daerah nasional	Netral	Negatif
iya tidak juga president hak nya nihil paksa partai politik bukti nya rusak uu nya suruh partai politik	Negatif	Netral
pilih kepala daerah periode tidak perlu jalur partai politik nyalon maling ktp warga abai panggil bawaslu bantu sahkan kpu calon nyalon nyolong ktp wrga deh brkuasa tidak nyolong	Netral	Negatif
benar lolos wni partai politik nyalonin pilpres pilih sih asli	positif	

Tabel 4. 2 Lanjutan Contoh Hasil Pelabelan Manual

tidak pakai konser fun football acara partai politik sih aman rumput moga bohong gbk biar rumput sehat	positif	
duit baik tabung bikin partai politik nyewa preman sandiwara	Netral	Negatif
nihil partai politik percaya	positif	
presiden tuduh curang zalim iya perintah partai politik tum	Negatif	Negatif
nyala rahasia selesai baca wkwkwk barusan baca deh nyeritain nyala ngabisin malam sama tum partai politik ujung hamil nagih janji beliau dinikahin sama tum cerita tidak mudah singkat	Positif	Netral
sistem proporsional buka maju pikir kolot partai politik ketua partai politik partai politik tum kuasa absolut embel kaderisasi amp musyawarah nasional partai politik reformasi jabat tum batas periode	Positif	Positif
mulyono langsung loyo iya tidak punya partai politik menteri nya langsung gerak	Netral	Negatif
demokrasi ala wawas partai politik kursi parlemen harap suara usul protes aju eksekutif sesuai undang paham	Positif	Negatif
benci banget sama negara nik mama daftar anggota partai politik partai politik nihil daerah anjj	Negatif	Negatif
susup nu susup mu partai politik bumh perintah daerah jadi wantim pres darurat	Negatif	Negatif
perintah pilih kepala daerah landas kuku elite partai politik sadar adu warga waspada kotak bohong bohir hilir mudik istana siap dana wowo antartika nyata tulis janji bela negara	Positif	Negatif
partai politik mingkem cari aman	Positif	Netral
kasih partai politik partai	Netral	Netral
tidak harus hina gadai kali bang rawan pilpres pilih anies imin dukung partai politik nasdem partai partai tidak kultus individu tokoh	Negatif	Netral
ngotorin hati bikin bawa asa tidak ktulungan mcap sandera sgera deklarasi aza bikin partai politik elegan spaya btarung kayak diocehin tidak lelah be realistic om	Positif	Netral
temu sukarelawan partai politik usung klaten cagub jateng ahmad luthfi	Netral	Netral
anies framing jegal dikroyok tukang survei amp gagal partai politik anies tidak layak tidak pantas amp tidak berkwalitas jago bicara kerja bodoh	Negatif	Negatif

6. Klasifikasi *Support Vector Machine* (SVM)

Proses klasifikasi dilakukan dengan menggunakan model *Support Vector Machine* (SVM). Langkah pertama adalah memuat model *Lexicon-Based* yang telah dilatih sebelumnya untuk melakukan pelabelan sentimen pada data. Selanjutnya, data yang berisi teks yang telah dibersihkan dan diproses (vektorisasi) dimasukkan ke dalam variabel `resultlabel3`. Label sentimen dan label aspek yang ada dalam dataset digunakan sebagai target untuk pelatihan model. Proses pembagian data dilakukan dengan membagi 90% data untuk pelatihan dan 10% untuk pengujian. Model *Support Vector Classifier* (SVC) kemudian dilatih menggunakan data latih yang telah dipersiapkan. Setelah model selesai dilatih, model ini digunakan untuk memprediksi label sentimen dan label aspek pada data uji yang telah disiapkan.

A. Kernel Linear

Berikut merupakan modul program yang digunakan dalam proses klasifikasi SVM dengan menggunakan kernel linear dapat dilihat pada Algoritma 11.

Algoritma 11 : Klasifikasi SVM Kernel Linear

```
from sklearn import datasets
from sklearn.model_selection import train_test_split
from sklearn.svm import SVC
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import classification_report

# Baca dataset dari file Excel
df = pd.read_excel('resultlabel3.xlsx')

# Lakukan ekstraksi fitur TF-IDF
tfidf_vectorizer = TfidfVectorizer(max_features=2000, max_df=0.75,
min_df=5)
X_tfidf = tfidf_vectorizer.fit_transform(df['ind_preprocessed_text'])

# Bagi dataset menjadi data pelatihan (90%) dan data pengujian (10%) setelah
ekstraksi fitur TF-IDF
X_train, X_test, y_train, y_test = train_test_split(X_tfidf, df['label'],
test_size=0.1, random_state=21)

# Latih model SVM
svm_classifier = SVC(C=1, kernel='linear', probability=True)
svm_classifier.fit(X_train, y_train)
```

B. Kernel RBF

Berikut merupakan modul program yang digunakan dalam proses klasifikasi SVM dengan menggunakan kernel RBF dapat dilihat pada Algoritma 11.

Algoritma 12 : Klasifikasi SVM Kernel RBF

```
from sklearn import datasets
from sklearn.model_selection import train_test_split
from sklearn.svm import SVC
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import classification_report

# Baca dataset dari file Excel
df = pd.read_excel('resultlabel4.1.xlsx')
```

```

# Lakukan ekstraksi fitur TF-IDF
tfidf_vectorizer = TfidfVectorizer(max_features=2000, max_df=0.75,
min_df=5)
X_tfidf = tfidf_vectorizer.fit_transform(df['ind_preprocessed_text'])

# Bagi dataset menjadi data pelatihan (90%) dan data pengujian (10%) setelah
ekstraksi fitur TF-IDF
X_train, X_test, y_train, y_test = train_test_split(X_tfidf, df['label'],
test_size=0.1, random_state=21)

# Latih model SVM
svm_classifier_rbf = SVC(C=10, gamma=0.1, kernel='rbf', probability=True)
svm_classifier_rbf.fit(X_train, y_train)

y_pred = svm_classifier_rbf.predict(X_test)
classification_rep = classification_report(y_test, y_pred, target_names=['-
0.1', '0.0', '1.0'])
print("Classification Report:\n", classification_rep)

# Hitung dan tampilkan akurasi
accuracy = accuracy_score(y_test, y_pred)
accuracy_percentage = accuracy * 100
print("Accuracy: {:.2f}%".format(accuracy_percentage))

conf_matrix = confusion_matrix(y_test, y_pred)
sns.heatmap(conf_matrix, annot=True, fmt='d', cmap='Reds',
xticklabels=['Negatif', 'Netral', 'Positif'],
yticklabels=['Negatif', 'Netral', 'Positif'])
plt.title('Confusion Matrix')
plt.xlabel('Predicted')
plt.ylabel('Actual')
plt.show()

```

C. Kernel Sigmoid

Berikut merupakan modul program yang digunakan dalam proses klasifikasi SVM dengan menggunakan kernel sigmoid dapat dilihat pada Algoritma 13.

Algoritma 13 : Klasifikasi SVM Kernl Sigmoid

```

from sklearn import datasets
from sklearn.model_selection import train_test_split
from sklearn.svm import SVC
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import classification_report

# Baca dataset dari file Excel
df = pd.read_excel('resultlabel4.1.xlsx')

# Lakukan ekstraksi fitur TF-IDF
tfidf_vectorizer = TfidfVectorizer(max_features=2000, max_df=0.75,
min_df=5)
X_tfidf = tfidf_vectorizer.fit_transform(df['ind_preprocessed_text'])

# Bagi dataset menjadi data pelatihan (90%) dan data pengujian (10%) setelah
ekstraksi fitur TF-IDF
X_train, X_test, y_train, y_test = train_test_split(X_tfidf, df['label'],
test_size=0.1, random_state=21)

```

```

# Latih model SVM
svm_classifier_sigm = SVC(C=1, kernel='sigmoid', probability=True)
svm_classifier_sigm.fit(X_train, y_train)

y_pred = svm_classifier_sigm.predict(X_test)
classification_rep = classification_report(y_test, y_pred, target_names=['-
0.1', '0.0', '1.0'])
print("Classification Report:\n", classification_rep)

# Hitung dan tampilkan akurasi
accuracy = accuracy_score(y_test, y_pred)
accuracy_percentage = accuracy * 100
print("Accuracy: {:.2f}%".format(accuracy_percentage))

conf_matrix = confusion_matrix(y_test, y_pred)
sns.heatmap(conf_matrix, annot=True, fmt='d', cmap='Reds',
xticklabels=['Negatif', 'Netral', 'Positif'],
yticklabels=['Negatif', 'Netral', 'Positif'])
plt.title('Confusion Matrix')
plt.xlabel('Predicted')
plt.ylabel('Actual')
plt.show()

```

D. Kernel Polynomial

Berikut merupakan modul program yang digunakan dalam proses klasifikasi SVM dengan menggunakan kernel polynomial dapat dilihat pada Algoritma 14.

Algoritma 14: Klasifikasi SVM Kernel Polynomial

```

from sklearn import datasets
from sklearn.model_selection import train_test_split
from sklearn.svm import SVC
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import classification_report

# Baca dataset dari file Excel
df = pd.read_excel('resultlabel4.1.xlsx')

# Lakukan ekstraksi fitur TF-IDF
tfidf_vectorizer = TfidfVectorizer(max_features=2000, max_df=0.75,
min_df=5)
X_tfidf = tfidf_vectorizer.fit_transform(df['ind_preprocessed_text'])

# Bagi dataset menjadi data pelatihan (90%) dan data pengujian (10%) setelah
ekstraksi fitur TF-IDF
X_train, X_test, y_train, y_test = train_test_split(X_tfidf, df['label'],
test_size=0.1, random_state=21)

# Latih model SVM
svm_classifier_poly = SVC(C=1, kernel='poly', probability=True)
svm_classifier_poly.fit(X_train, y_train)

y_pred = svm_classifier_poly.predict(X_test)
classification_rep = classification_report(y_test, y_pred, target_names=['-
0.1', '0.0', '1.0'])
print("Classification Report:\n", classification_rep)

# Hitung dan tampilkan akurasi

```

```

accuracy = accuracy_score(y_test, y_pred)
accuracy_percentage = accuracy * 100
print("Accuracy: {:.2f}%".format(accuracy_percentage))

conf_matrix = confusion_matrix(y_test, y_pred)
sns.heatmap(conf_matrix, annot=True, fmt='d', cmap='Reds',
            xticklabels=['Negatif', 'Netral', 'Positif'],
            yticklabels=['Negatif', 'Netral', 'Positif'])
plt.title('Confusion Matrix')
plt.xlabel('Predicted')
plt.ylabel('Actual')
plt.show()

```

7. Tuning *Hyper Parameter GridSearch*

Proses ini merupakan tahapan untuk melakukan pencarian parameter terbaik untuk masing – masing kernel yang dimana bertujuan untuk mencari hasil akurasi yang paling optimal. Beberapa parameter akan di inialisasikan terlebih dahulu untuk sesuai dengan kebutuhan masing – masing kernel yang akan diuji. Berikut merupakan modul program inisiasi parameter dengan algoritma *GridSearch* dapat dilihat pada Algoritma 15.

Algoritma 15 : Algoritma *GridSearch*

```

from sklearn.svm import SVC
from sklearn.model_selection import train_test_split, GridSearchCV
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics import classification_report, precision_score,
recall_score
import pandas as pd

# Baca dataset dari file Excel
df = pd.read_excel('resultlabel4.1.xlsx')

# Lakukan ekstraksi fitur TF-IDF
tfidf_vectorizer = TfidfVectorizer(max_features=2000, max_df=0.75, min_df=5)
X_tfidf = tfidf_vectorizer.fit_transform(df['ind_preprocessed_text'])

# Bagi dataset menjadi data pelatihan (90%) dan data pengujian (10%) setelah
ekstraksi fitur TF-IDF
X_train, X_test, y_train, y_test = train_test_split(X_tfidf, df['label'],
test_size=0.1, random_state=21)

# Inisialisasi model SVM
svm = SVC()

# Mendefinisikan grid hyperparameter untuk dicoba
param_grid = {
    'C': [0.1, 1, 10, 100],
    'gamma': [1, 0.1, 0.01, 0.001],
    'kernel': ['poly', 'rbf', 'sigmoid', 'linear'],
}

# Menggunakan GridSearchCV dengan cross-validation 5-fold
grid_search = GridSearchCV(estimator=svm, param_grid=param_grid, cv=5,
verbose=2, n_jobs=-1)
grid_search.fit(X_train, y_train)

# Menampilkan hasil terbaik
print("Best parameters found: ", grid_search.best_params_)

```

```
print("Best cross-validation score: ", grid_search.best_score_)

# Mengevaluasi model terbaik pada data uji
y_pred = grid_search.predict(X_test)
print("\nClassification report for best model:\n",
      classification_report(y_test, y_pred))

# Menampilkan akurasi model terbaik pada data uji
test_score = grid_search.score(X_test, y_test)
print(f"Test accuracy score for best model: {test_score:.4f}")
precision = precision_score(y_test, y_pred, average='weighted')
recall = recall_score(y_test, y_pred, average='weighted')
print(f"Recall : {recall:.4f}")
print(f"Precision : {precision:.4f}")
```

4.1.2 Implementasi Pengembangan Sistem

Implementasi pengembangan sistem dalam penelitian ini dilakukan dengan menggunakan library *Streamlit*, yang memungkinkan pembuatan antarmuka pengguna (user interface) yang interaktif dan fungsional. Pada tahap ini, hasil dari desain sistem yang telah direncanakan sebelumnya diwujudkan dalam bentuk aplikasi yang dapat diakses dan digunakan langsung oleh pengguna. Desain sistem tidak hanya mencakup tampilan visual dari antarmuka, tetapi juga menyertakan berbagai fungsi yang memungkinkan pengguna berinteraksi dengan sistem secara efektif dan efisien. Setiap halaman yang telah dirancang menampilkan bagaimana sistem ini berfungsi dan bekerja sesuai dengan spesifikasi yang telah ditetapkan pada tahap perancangan. Dengan memanfaatkan *Streamlit*, setiap elemen antarmuka dan fungsi yang telah direncanakan dapat diimplementasikan dengan mudah dan cepat, memberikan gambaran yang jelas tentang bagaimana aplikasi ini beroperasi dan berfungsi secara keseluruhan.

1. Halaman Beranda

Halaman Beranda merupakan tampilan pertama yang muncul saat sistem dijalankan. Halaman ini menampilkan gambar yang menggambarkan topik penelitian yang diangkat, serta mencantumkan judul penelitian secara jelas. Di samping itu, halaman ini dilengkapi dengan menu samping (side menu) yang memungkinkan pengguna untuk mengakses berbagai halaman lain dalam sistem. Tampilan lengkap dari halaman beranda ini dapat dilihat pada Gambar 4.1 berikut:



Gambar 4. 1 Halaman Beranda

2. Halaman Dataset

Submenu ini menampilkan data mentah yang akan digunakan dalam penelitian. Pada halaman ini, pengguna dapat melihat data dalam bentuk aslinya sebelum diproses lebih lanjut. Selain itu, submenu ini juga menampilkan visualisasi dalam bentuk pie chart yang menggambarkan distribusi jumlah data dalam dataset. Tampilan dari halaman ini dapat dilihat pada Gambar 4.2 berikut:

a. *Dataset Fresh*

Berikut merupakan tampilan halaman *dataset fresh* dapat dilihat pada Gambar 4.2.

Fresh Dataset					
	Unnamed: 0	conversation_id_str	created_at	favorite_count	full_text
0	0	⚠️ 1840656325165883748	Mon Sep 30 23:59:07 +0000 2024	0	@lauklaut @d
1	1	⚠️ 1840781035082273022	Mon Sep 30 23:49:41 +0000 2024	2	@ferrykoto @
2	2	⚠️ 1840880607309807793	Mon Sep 30 23:45:55 +0000 2024	2	@RfgL88 dlm
3	3	⚠️ 1840899453945069820	Mon Sep 30 23:40:06 +0000 2024	1	Salah satu yar
4	4	⚠️ 1840549001621590507	Mon Sep 30 23:39:10 +0000 2024	0	@roarrngh @0
5	5	⚠️ 1840751976138854566	Mon Sep 30 23:34:06 +0000 2024	3	@tjitrosoenar
6	6	⚠️ 1840758650551545978	Mon Sep 30 23:34:04 +0000 2024	6	@peppertipe
7	7	⚠️ 1840896895235080375	Mon Sep 30 23:29:56 +0000 2024	113	10 tahun terak
8	8	⚠️ 1840751976138854566	Mon Sep 30 23:27:19 +0000 2024	4	@tjitrosoenar
9	9	⚠️ 1840892459724992747	Mon Sep 30 23:12:19 +0000 2024	0	Preman Politil

Gambar 4. 2 Halaman *Dataset Fresh*

b. *Dataset After Preprocessing*

Berikut merupakan tampilan halaman *dataset after preprocessing* dapat dilihat pada Gambar 4.3.

Dataset after preprocessing

	Unnamed: 0	conversation_id_str	created_at	favorite_count	full_text
0	0	⚠️ 1840656325165883748	Mon Sep 30 23:59:07 +0000 2024	0	@lauklaut@d
1	1	⚠️ 1840781035082273022	Mon Sep 30 23:49:41 +0000 2024	2	@ferrykoto@
2	2	⚠️ 1840880607309807793	Mon Sep 30 23:45:55 +0000 2024	2	@RfgL88 dlm
3	3	⚠️ 1840899453945069820	Mon Sep 30 23:40:06 +0000 2024	1	Salah satu yar
4	4	⚠️ 1840549001621590507	Mon Sep 30 23:39:10 +0000 2024	0	@roarrgh@0
5	5	⚠️ 1840751976138854566	Mon Sep 30 23:34:06 +0000 2024	3	@tjitrosoenar
6	6	⚠️ 1840758650551545978	Mon Sep 30 23:34:04 +0000 2024	6	@peppertipe
7	7	⚠️ 1840896895235080375	Mon Sep 30 23:29:56 +0000 2024	113	10 tahun terak
8	8	⚠️ 1840751976138854566	Mon Sep 30 23:27:19 +0000 2024	4	@tjitrosoenar
9	9	⚠️ 1840892459724992747	Mon Sep 30 23:12:19 +0000 2024	0	Preman Politil

Gambar 4. 3 Halaman *Dataset After Preprocessing*

c. *Dataset Labelling*

Berikut merupakan tampilan halaman dataset *labelling* dapat dilihat pada Gambar 4.4.

Dataset after sentiment labelling

	Unnamed: 0	conversation_id_str	created_at	favorite_count	full_text
0	0	⚠️ 1840656325165883904	Mon Sep 30 23:59:07 +0000 2024	0	@lauklaut@d
1	1	⚠️ 1840781035082273024	Mon Sep 30 23:49:41 +0000 2024	2	@ferrykoto@
2	2	⚠️ 1840880607309808128	Mon Sep 30 23:45:55 +0000 2024	2	@RfgL88 dlm
3	3	⚠️ 1840899453945070080	Mon Sep 30 23:40:06 +0000 2024	1	Salah satu yar
4	4	⚠️ 1840549001621591040	Mon Sep 30 23:39:10 +0000 2024	0	@roarrgh@0
5	5	⚠️ 1840751976138854912	Mon Sep 30 23:34:06 +0000 2024	3	@tjitrosoenar
6	6	⚠️ 1840758650551546112	Mon Sep 30 23:34:04 +0000 2024	6	@peppertipe
7	7	⚠️ 1840896895235079936	Mon Sep 30 23:29:56 +0000 2024	113	10 tahun terak
8	8	⚠️ 1840751976138854912	Mon Sep 30 23:27:19 +0000 2024	4	@tjitrosoenar
9	9	⚠️ 1840892459724993024	Mon Sep 30 23:12:19 +0000 2024	0	Preman Politil

Gambar 4. 4 Halaman *Dataset Labelling*

3. Halaman Evaluasi

Halaman Evaluasi dirancang untuk menampilkan hasil evaluasi model klasifikasi yang telah dilatih menggunakan data uji. Pada halaman ini, pengguna dapat melihat metrik

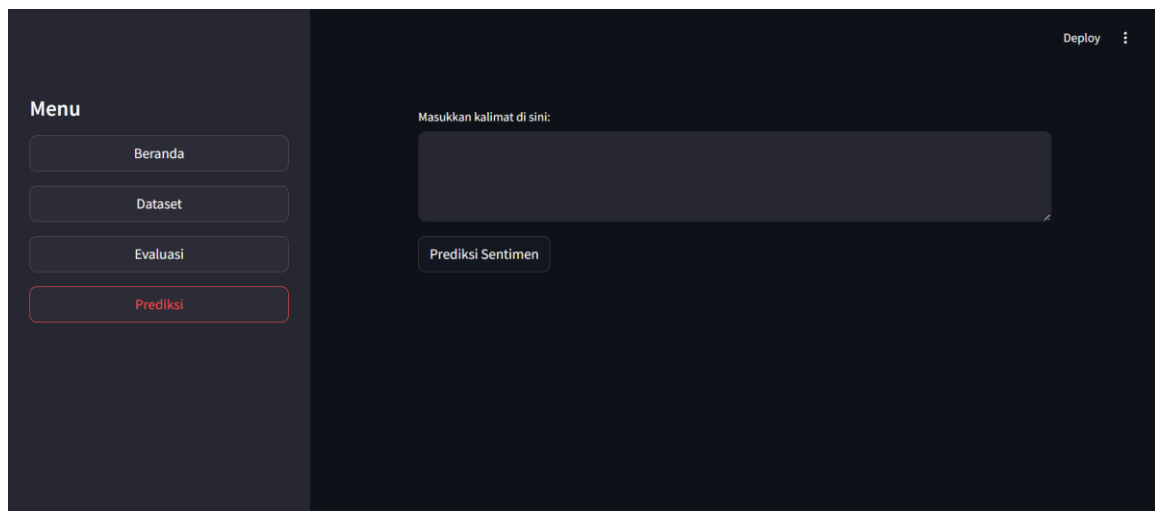
performa model seperti akurasi, precision, dan recall yang menunjukkan sejauh mana model dapat mengklasifikasikan sentimen dan aspek dengan benar. Setiap metrik ini dihitung berdasarkan hasil prediksi yang dibandingkan dengan label sebenarnya dalam data uji. Halaman ini juga menyertakan visualisasi berupa grafik atau tabel yang mempermudah pengguna untuk memahami performa model secara lebih jelas dan mendalam. Selain itu, evaluasi juga mencakup analisis kesalahan dan saran perbaikan untuk model agar lebih optimal di masa depan. Hasil evaluasi ini memberikan gambaran lengkap mengenai kinerja sistem dalam melakukan analisis sentimen dan identifikasi aspek dari komentar dapat dilihat pada Gambar 4.5 berikut.



Gambar 4. 5 Halaman Evaluasi

4. Halaman Prediksi

Halaman Prediksi dirancang untuk memungkinkan pengguna menginput data komentar tweet yang ingin diuji untuk analisis sentimennya. Pada halaman ini, pengguna dapat langsung memasukkan komentar yang relevan, dan sistem akan otomatis menganalisis sentimen yang terkandung dalam komentar tersebut, serta mengidentifikasi aspek kategori yang terkait dengan konten komentar. Halaman prediksi dapat dilihat pada Gambar 4.6 berikut :



Gambar 4. 6 Halaman Prediksi

4.2 Hasil

Bab hasil dalam penelitian ini akan menguraikan secara rinci temuan dari seluruh proses pengujian yang telah dilakukan. Bagian ini mencakup analisis hasil pengujian model klasifikasi untuk mengukur kinerja akurasi, precision, recall, dan metrik evaluasi lainnya, serta pengujian terhadap pengembangan sistem yang diterapkan dalam analisis sentimen publik terhadap partai politik di Indonesia. Uraian hasil pengujian ini bertujuan untuk memberikan pemahaman mendalam tentang seberapa baik model dan sistem yang dikembangkan dalam memenuhi tujuan penelitian serta efektivitas metode yang digunakan.

4.2.1 Proses Pengujian Model

Pada bagian ini, akan dijelaskan secara rinci mengenai pengujian model klasifikasi yang diterapkan dalam penelitian ini, termasuk hasil-hasil yang diperoleh. Pengujian dilakukan untuk mengevaluasi sejauh mana model dapat memprediksi dengan akurat sentimen dan kategori aspek dari data komentar tweet yang diberikan. Untuk menilai kinerja model klasifikasi, digunakan metode confusion matrix, yang memberikan analisis menyeluruh tentang performa model dalam mengklasifikasikan data.

1. *Training Labeling Lexicon Based*

Berikut merupakan hasil pembagian tiap kelas berdasarkan *labelling* menggunakan *lexicon based* dapat dilihat pada Tabel 4.1.

Tabel 4. 3 Training Labeling Lexicon Based

Sentimen	Jumlah
Positif	1863
Netral	1443
Negatif	1424

Berdasarkan Tabel 4.1 diatas dapat dilihat bahwa kelas sentimen positif memiliki jumlah data label terbanyak, yang dimana proses pelabelan menggunakan lexicon based dengan kamus *senti strength* mendapatkan hasil 1.867 data atau 39.37% dari data set yang ada, sentiment dengan kelas netral didapatkan jumlah sebesar 1.448 data atau 30.54% dari dataset yang ada. Dan yang terakhir pada kelas negatif didapatkan jumlah 1.427 data dengan presentase 30.09% dari total data yang ada. Hal ini disebabkan oleh setelah proses *web crawling* yang sebagian besar menghasilkan *tweet* yang bersifat dukungan atau

tanggapan positif setelah dilabeli dengan mencocokkan kamus *senti strength* pada kamus *lexicon based*.

2. Pengujian Labeling Manual

Berikut merupakan hasil pembagian tiap kelas berdasarkan *labelling* menggunakan *lexicon based* dapat dilihat pada Tabel 4.2.

Tabel 4. 4 Training Labelling Manual

Sentimen	Jumlah
Positif	1268
Netral	380
Negatif	3082

Berdasarkan Tabel 4.2 diatas dapat dilihat bahwa kelas sentimen negatif memiliki jumlah data label terbanyak berbanding terbalik dengan pelabelan yang dilakukan menggunakan *lexicon based*, yang dimana proses pelabelan manual mendapatkan hasil 3.082 data atau 65.16% untuk sentimen negatif dari data set yang ada, sentiment dengan kelas netral didapatkan jumlah sebesar 379 data atau 8.01% dari dataset yang ada. Dan yang terakhir pada kelas negatif didapatkan jumlah 1.268 data dengan presentase 26.82% dari total data yang ada. Hal ini disebabkan oleh setelah proses *web crawling* dilakukan proses pelabelan manual oleh 2 mahasiswa yang tidak bisa terlepas dari subjektivitas dalam melakukan pelabelan sentimen. Kondisi diatas merupakan kondisi yang tidak baik untuk proses pengujian model dikarenakan nantinya model akan mengalami proses *overfitting*. Proses *overfitting* ini merupakan kondisi dimana model akan terlatih dengan pola data mayoritas dan akan menjadikan data tersebut sebagai acuan untuk melakukan prediksi, sehingga nantinya akan sulit untuk mengenali pola pada data pada 2 kelas lainnya apabila dilakukan input menggunakan data baru. Sehingga pada penelitian ini akan dilakukan proses pengujian dengan menggunakan hasil proses pelabelan yang paling baik dengan menggunakan metode pelabelan *lexicon based*. Dengan menggunakan proses pelabelan *lexicon* selanjutnya data akan dipisah untuk melatih model dengan pembagian data seperti pada tabel 4.3

Tabel 4. 5 Splitting Data

Data Training 90%	DataTesting 10%	Jumlah
4257	473	4730

Dengan pembagian data seperti diatas, selanjutnya dapat dilihat sentimen yang terkandung pada data *training*. Jumlah sentimen dapat dilihat pada Tabel 4.4 berikut.

Tabel 4. 6 Hasil Sentimen Data Training

Sentimen	Jumlah
Positif	1666
Netral	1311
Negatif	1280

Dengan pembagian data seperti diatas, selanjutnya dapat dilihat sentimen yang terkandung pada data *testing*. Jumlah sentimen dapat dilihat pada Tabel 4.5 berikut.

Tabel 4. 7 Hasil Sentimen Data Testing

Sentimen	Jumlah
Positif	191
Netral	141
Negatif	131

3. Inisiasi *Hyperparameter*

Proses ini merupakan tahapan untuk mencari parameter terbaik untuk digunakan dalam proses training model nantinya. Adapun parameter yang akan digunakan dalam proses ini dapat dilihat pada Tabel 4.6 berikut.

Tabel 4. 8 Inisiasi Parameter

Nilai C	Nilai Gamma
0.1	1
1	0,1
10	0,01
100	0,001

4. Pengujian Parameter Default Kernel SVM

Pada pelatihan model SVM, penelitian ini akan menggunakan beberapa kernel untuk mencari performa terbaik dalam menghasilkan akurasi model. Selanjutnya masing masing kernel akan diuji parameter terbaiknya dengan melakukan inialisasi parameter *C* dan juga *gamma*. Hal ini bertujuan untuk mengetahui *Hyper* parameter terbaik dalam menghasilkan proses evaluasi model ketika menghasilkan akurasi. Berikut merupakan hasil pelatihan model SVM dengan menggunakan empat kernel dapat dilihat pada Tabel 4.7.

Tabel 4. 9 Hasil Pelatihan Model SVM

No.	Kernel	Akurasi	Precision	Recall
1.	Linear	81.05%	82.25%	81.05%
2.	RBF	79.16%	79.73%	79.16%
3.	Polynomial	67.58%	71.36%	67.58%
4.	Sigmoid	79.37%	80.01%	79.37%

Dari tabel diatas dapat dilihat bahwa pengujian masing masing kernel yaitu linear, RBF, polynomial, dan juga sigmoid berhasil dilakukan. Dengan parameter *default* didapatkan hasil pengujian model SVM dengan kernel terbaik pada kernel Linear dalam menghasilkan akurasi.

5. Pengujian Hyperparameter Kernel SVM

Berikut merupakan hasil training pada kernel SVM dengan menggunakan 4 kernel SVM, dapat dilihat pada Tabel 4.8.

Tabel 4. 10 Training Kernel SVM

No.	Kernel	Akurasi	Precision	Recall
1.	Linear	81.89%	82.25%	81.89%
2.	RBF	82.11%	82.52%	82.11%
3.	Polynomial	67.58%	71.36%	67.58%
4.	Sigmoid	81.47%	81.87%	81.47%

Dari Tabel 4.8 diatas dapat dilihat bahwa pengujian masing masing kernel yaitu linear, RBF, polynomial, dan juga sigmoid berhasil dilakukan. Pengujian tersebut menggunakan metode *gridsearch* yang dimana metode tersebut merupakan cara yang bisa digunakan untuk mencari *Hyper* parameter terbaik dalam proses pengujian kernel SVM. Dari penerapan metode *gridsearch* didapatkan hasil pengujian model SVM dengan kernel terbaik pada kernel RBF dalam menghasilkan akurasi.

4.2.2 Proses Evaluasi Model

Pada bagian ini, akan dijelaskan secara rinci mengenai pengujian model klasifikasi yang dikembangkan dalam penelitian ini, beserta hasil yang diperoleh. Pengujian dilakukan untuk mengevaluasi seberapa efektif model dalam memprediksi sentimen pada data komentar *tweet*. Untuk menilai kinerja model, digunakan metode *confusion matrix*. Pada penelitian ini, digunakan *confusion matrix multiclass*, mengingat adanya tiga kelas sentimen dan tiga kategori aspek yang perlu diklasifikasikan. Penggunaan *confusion matrix multiclass* ini memungkinkan evaluasi yang akurat terhadap kinerja model melalui metrik seperti akurasi, *precision*, *recall*, dan *F1-Score*.

Pengujian ini bertujuan untuk mengukur performa model dengan menghasilkan nilai akurasi, *precision*, *recall*, dan *F1-Score* untuk setiap kelas sentimen: positif, netral, dan negatif. Pengujian awal dilakukan pada model SVM untuk klasifikasi kelas sentiment dapat dilihat pada Tabel 4.9.

Tabel 4. 11 Hasil Pengujian *Confusion Matrix*

Data Aktual	Data Prediksi		
	Negatif	Netral	Positif
Negatif	107	21	11
Netral	11	127	10
Positif	12	20	156

Berdasarkan Tabel 4.9 diatas terlihat hasil dari *confusion matrix* sentiment hasil pengujian model yang terbagi menjadi tiga kelas sentimen yaitu negatif, netral, dan positif. Berdasarkan tabel tersebut, terlihat ada 107 data negatif yang benar terprediksi negatif, tetapi ada 21 data negatif terbaca netral, dan 11 data negatif terbaca positif. Lalu kelas netral terdapat 11 data netral terprediksi negatif, 127 data netral terbaca netral dan terdapat 10 data netral terbaca positif. Selanjutnya pada kelas positif terdapat 12 data positif yang terbaca negatif, 20 data positif terbaca netral, dan ada 156 data positif yang terprediksi dengan benar. Hasil ini menunjukkan adanya beberapa kesalahan prediksi pada beberapa kategori, terutama di kelas netral dan positif, yang mengindikasikan bahwa model masih mengalami kesulitan dalam membedakan beberapa kategori sentimen tertentu. Pengukuran nilai akurasi, *precision*, *recall*, dan *F1-Score* dari model SVM untuk sentimen ini diperoleh berdasarkan hasil vektorisasi menggunakan metode TF-IDF. Berikut perhitungan *accuracy*, *precision*, *recall*, serta *F1-score*. Meskipun menghasilkan performa model yang baik, tetapi kemampuan model juga terdapat kesalahan yang dapat dilihat dari data - data yang salah terprediksi dengan benar. Hal tersebut dikarekanakan kualitas data yang beragam dan cenderung memiliki banyak konteks kata yang tidak bisa dapat dipelajari dengan baik oleh SVM.

$$Accuracy = \frac{TP+TN+TL}{Total} X 100\% =$$

$$Accuracy = \frac{156+127+107}{107+21+11+11+127+10+12+20+156} X 100\% = 82.11\%$$

Selanjutnya akan dihitung *precision* untuk masing - masing kelas dengan rumus

$$Precision \text{ kelas positif} = \frac{TP}{TP+FP+FP2} = \frac{156}{156+11+10} = 0,8814\%$$

$$Precision \text{ kelas negatif} = \frac{TN}{TN+FN+FN2} = \frac{107}{107+12+11} = 0,7560\%$$

$$Precision \text{ kelas netral} = \frac{TL}{TL+FL+FL2} = \frac{127}{127+20+21} = 0,8231\%$$

$$Precision = \frac{P \text{ Precision} + N \text{ Precision} + L \text{ Precision}}{3} \times 100 = \frac{2,4605}{3} \times 100 = 82.92\%$$

Selanjutnya akan dihitung *recall* untuk masing - masing kelas dengan rumus

$$Recall \text{ kelas positif} = \frac{TP}{TP+FL+FN} = \frac{156}{156+12+20} = 0,8298\%$$

$$Recall \text{ kelas negatif} = \frac{TN}{TN+FL2+FP} = \frac{107}{107+21+11} = 0,7705\%$$

$$Recall \text{ kelas netral} = \frac{TL}{TL+FN2+FP2} = \frac{127}{127+10+11} = 0,8581\%$$

$$Recall = \frac{P \text{ Precision} + N \text{ Precision} + L \text{ Precision}}{3} \times 100 = \frac{2,4584}{3} \times 100 = 82.11\%$$

Selanjutnya akan dihitung *F1-Score* untuk masing - masing kelas dengan rumus

$$F1 - Score \text{ kelas positif} = 2 \times \frac{Precision \text{ P} \times Recall \text{ P}}{Precision \text{ P} + Recall \text{ P}} = 2 \times \frac{0,8814 \times 0,8298}{0,8814 + 0,8298} = 0,8549\%$$

$$F1 - Score \text{ kelas negatif} = 2 \times \frac{Precision \text{ N} \times Recall \text{ N}}{Precision \text{ N} + Recall \text{ N}} = 2 \times \frac{0,8231 \times 0,7705}{0,8231 + 0,7705} = 0,7960\%$$

$$F1 - Score \text{ kelas netral} = 2 \times \frac{Precision \text{ L} \times Recall \text{ L}}{Precision \text{ L} + Recall \text{ L}} = 2 \times \frac{0,7560 \times 0,8581}{0,7560 + 0,8581} = 0,8039\%$$

$$F1 - Score = \frac{F1 - Score \text{ P} + F1 - Score \text{ N} + F1 - Score \text{ L}}{3} \times 100 = \frac{2,4584}{3} \times 100 = 81.82\%$$

Berdasarkan hasil evaluasi, model klasifikasi sentimen mencapai akurasi sebesar 82.11%, yang menunjukkan bahwa 82.11% dari total prediksi sesuai dengan data yang sebenarnya. Rata-rata nilai presisi mencapai 82.92%, mengindikasikan bahwa 82.92% dari prediksi model untuk kelas positif, netral, dan negatif adalah benar. Dengan rata-rata recall sebesar 82.11%, model mampu mengidentifikasi 82.11% dari data aktual untuk setiap kelas sentimen. Secara spesifik, untuk kelas positif, presisi tercatat sebesar 0.8814, recall 0.8298, dan F1-score 0.8298. Kelas negatif memiliki presisi 0.7560, recall 0.7705, dan F1-score 0.7960. Sementara itu, kelas netral memperoleh hasil terbaik dengan presisi 0.8231, recall 0.8581, dan F1-score 0.8039. Rata-rata F1-score model sebesar 81.82% mencerminkan keseimbangan yang cukup baik antara presisi dan *recall*.

4.2.3 Pengujian Pengembangan Sistem

Pengujian pengembangan sistem dalam penelitian ini dilakukan menggunakan metode *black-box testing*, yang berfokus pada pengujian fungsionalitas sistem tanpa melihat implementasi internalnya. Dalam pengujian ini, sistem diuji untuk memastikan kemampuannya dalam menjalankan proses klasifikasi teks terhadap data komentar publik yang diperoleh dari platform media sosial X. Pengujian ini bertujuan untuk mengevaluasi sejauh mana sistem dapat melakukan analisis sentimen secara akurat. Hasil dari pengujian ini dapat dilihat pada Tabel 4.10 berikut.

Tabel 4. 12 Hasil Pengujian *Black Box*

Role	Halaman	Detail Pengujian	Hasil Pengujian
User	Beranda	Menampilkan Tampilan awal sistem dirancang	Berhasil
	Dataset	Menampilkan halaman dataset yang digunakan dari mulai data set mentah sampai dataset yang sudah berhasil dilabelling	Berhasil
	Evaluasi	Halaman dapat menampilkan hasil evaluasi model SVM dengan menunjukkan nilai <i>accuracy</i> , <i>precision</i> , dan <i>recall</i>	Berhasil
	Prediksi	Dapat menginputkan kalimat untuk diuji prediksi sentimennya	Berhasil

Hasil pengujian *black-box* yang dilakukan pada berbagai halaman sistem menunjukkan bahwa setiap halaman berfungsi sesuai dengan harapan. Pada halaman Beranda, judul penelitian tentang analisis sentimen publik terhadap partai politik di Indonesia ditampilkan dengan jelas. Halaman Dataset berhasil menampilkan informasi terkait dataset yang digunakan, termasuk rincian data yang telah melalui proses pembersihan (*text preprocessing*) dan pembagian data untuk pelatihan dan pengujian. Di halaman Evaluasi, hasil evaluasi model, seperti akurasi, *precision*, *recall*, dan metrik lainnya, ditampilkan dengan benar. Halaman Prediksi memungkinkan pengguna untuk menginput data baru berupa komentar publik yang kemudian diuji menggunakan model SVM yang telah dikembangkan, dan pengujian menunjukkan bahwa halaman ini berfungsi tanpa kesalahan. Secara keseluruhan, setiap halaman dalam sistem berfungsi dengan baik, memenuhi tujuan pengujian, dan mendukung kelancaran analisis sentimen.

4.3 Pembahasan

Setelah dilakukannya pengujian terhadap metode *Support Vector Machine* (SVM) dalam menganalisis sentimen publik terhadap partai politik di Indonesia, didapatkan hasil yang cukup baik. Hasil tersebut tak terlepas dari proses pelabelan kata menggunakan metode *Lexicon Based* serta dilakukannya proses vektorisasi kata dengan ekstrkasksi fitur *Term Frequency – Inverse Document Frequency* (TF-IDF).

Setelah dilakukan pelabelan dengan menggunakan pelabelan manual dan juga *lexicon* didapatkan hasil perbedaan hasil yang signifikan. Pada pelabelan manual didapatkan hasil 3.082 data atau 65.16% untuk sentimen negatif dari data set yang ada, sentiment dengan kelas netral didapatkan jumlah sebesar 379 data atau 8.01% dari dataset yang ada. Dan yang terakhir pada kelas negatif didapatkan jumlah 1.268 data dengan presentase 26.82% dari total data yang ada. Pada pelabelan *lexicon* didapatkan hasil 1.867 data atau 39.37% dari data set yang ada, sentimen dengan kelas netral didapatkan jumlah sebesar 1.448 data atau 30.54% dari dataset yang ada. Dan yang terakhir pada kelas negatif didapatkan jumlah 1.427 data dengan presentase 30.09% dari total data yang ada. Pengujian diatas menggunakan jumlah dataset yang sama dengan menggunakan 4730. Semakin *balance* hasil akhir pelabelan, maka akan semakin baik

kinerja model dalam melakukan evaluasi. Dikarenakan *support vector machine* merupakan metode yang cukup rentan dengan data yang tidak *balance* atau seimbang.

Selanjutnya berdasarkan hasil pengujian dan pengimplementasian yang dilakukan pada penelitian ini, Penggunaan metode *Support Vector Machine* (SVM) dengan menggunakan metode *gridsearch* dalam proses pencarian *Hyperparameter* mendapatkan hasil yang cukup baik. Dimana kernel RBF atau *Radial Base Function* merupakan kernel yang mendapatkan hasil paling optimal dalam menghasilkan akurasi model dibandingkan dengan kernel lainnya. Dengan kemampuan model menghasilkan *accuracy* diangka 82.11%, *recall* 82.11% dan *precision* diangka 82.52%. Lebih tinggi dari kernel linear, polynomial, dan juga sigmoid dengan hasil *accuracy* 81.89%, *recall* 81.89% dan *precision* 82.52% untuk kernel linear. 65.89% *accuracy*, 70.66% *precision*, dan 65.89% untuk *recall* pada kernel polynomial. 81.47% *accuracy*, 81.47% *recall*, dan 81.87% *precision* untuk kernel sigmoid.

Hasil evaluasi diatas didapatkan setelah dilakukannya penerapan metode *gridsearch* untuk mencari hyper parameter untuk masing – masing kernel. Proses pencarian parameter terbaik dilakukan dengan melakukan inisiasi nilai C dari 0.1, 1, 10, dan juga 100. Selanjutnya dilakukan juga inisiasi nilai gamma yang dimulai dari 1, 0.1, 0.01, dan juga 0.001, dengan total kernel yang akan diuji untuk parameter diatas berjumlah 4 yaitu linear, polynomial, rbf, dan juga sigmoid. Selanjutnya dilakukan *cross validation* dengan nilai 5 sebagai data yang akan diuji untuk pencarian kernel terbaik. Sehingga dari proses inisiasi diatas akan dilakukan proses pencarian parameter terbaik dengan jumlah 64 kandidat dengan 5 data yang akan digunakan. Setelah dilakukannya 320 kali kemungkinan kombinasi, ditemukan parameter yang paling baik terdapat pada kernel RBF atau *Radial Basis Function* dengan nilai C = 10 serta dengan nilai gamma 0.1.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan perancangan, analisis, penelitian, serta pembahasan yang sudah dilakukan, diperoleh beberapa kesimpulan sebagai berikut :

1. Proses penganalisaan sentimen publik terhadap partai politik di Indonesia menggunakan metode *Lexicon Based* untuk pelabelan dan *Support Vector Machine* (SVM) untuk klasifikasi berhasil dilakukan.
2. Hasil tersebut menunjukkan bahwa metode SVM dalam menganalisis sentimen publik memberikan kinerja yang cukup baik dalam memprediksi sentimen, baik positif maupun negatif, terhadap partai politik di Indonesia. Dengan demikian, penelitian ini berhasil menunjukkan bahwa metode yang digunakan dapat mencapai hasil yang cukup optimal merujuk pada hasil evaluasi model pada penelitian ini mencapai tingkat akurasi sebesar 82,11%, dengan nilai *recall* 82,11% dan *precision* 82,52% dengan menggunakan kernel *Radial Basis Function* (RBF) sebagai kernel dengan hasil akurasi paling optimal dibandingkan kernel lainnya yaitu Linear, Sigmoid, dan juga Polynomial setelah dilakukannya proses *gridsearch* untuk mencari parameter dan kernel terbaik.

5.2 Saran

Adapun saran yang bisa digunakan sebagai landasan dalam pengembangan penelitian selanjutnya adalah sebagai berikut :

1. Penelitian selanjutnya diharapkan dapat mempertimbangkan penambahan metode penyeimbang kelas seperti SMOTE (*Synthetic Minority Over-sampling Technique*). Penggunaan SMOTE akan membantu mengatasi masalah ketidakseimbangan kelas pada data, sehingga model dapat lebih baik dalam mengenali pola pada kelas yang kurang terwakili, yang pada akhirnya dapat meningkatkan kinerja model secara keseluruhan.
2. Penelitian selanjutnya bisa untuk melakukan peningkatan akurasi untuk mengeksplorasi kombinasi metode atau parameter tuning yang lebih mendalam pada model SVM maupun pendekatan lain yang relevan. Selain itu, integrasi teknik-teknik seperti *Random Search* dapat membantu menemukan parameter optimal, sehingga diharapkan dapat meningkatkan akurasi dan performa model dalam analisis sentimen publik terhadap partai politik.

DAFTAR PUSTAKA

- Adiati, A. R., Herdiani, A., & Astuti, W. (2019). Analisis Sentimen Masyarakat Pada Media Sosial Twitter Terhadap Partai Politik Peserta Pemilihan Umum 2019 Menggunakan Naive Bayes Classifier. *EProceedings ...*, 6(2).
- Asri, Y., Suliyanti, W. N., Kuswardani, D., & Fajri, M. (2022). Pelabelan Otomatis Lexicon Vader dan Klasifikasi Naive Bayes dalam menganalisis sentimen data ulasan PLN Mobile. *Petir*, 15(2), 264–275. <https://doi.org/10.33322/petir.v15i2.1733>
- Baita, A., Pristyanto, Y., & Cahyono, N. (2021). Analisis Sentimen Mengenai Vaksin Sinovac Menggunakan Algoritma Support Vector Machine (Svm) Dan K-Nearest Neighbor (Knn). *Infos*, 4(2), 42–42.
- Fazrin, F., Pratiwi, O. N., & Andreswari, R. (2023). Perbandingan Algoritma K-Nearest Neighbor dan Logistic Regression pada Analisis Sentimen terhadap Vaksinasi Covid-19 pada Media Sosial Twitter dengan Pelabelan Vader dan Textblob. *Journal E-Proceeding of Engineering*, 10(2), 1596–1604.
- Giovani, A. P., Ardiansyah, A., Haryanti, T., Kurniawati, L., & Gata, W. (2020). Analisis Sentimen Aplikasi Ruang Guru Di Twitter Menggunakan Algoritma Klasifikasi. *Jurnal Teknoinfo*, 14(2), 115. <https://doi.org/10.33365/jti.v14i2.679>
- Handayani, A., & Zufria, I. (2023). Analisis Sentimen Terhadap Bakal Capres RI 2024 di Twitter Menggunakan Algoritma SVM. *Journal of Information System Research (JOSH)*, 5(1), 53–63. <https://doi.org/10.47065/josh.v5i1.4379>
- Hendrastuty, N., Rahman Isnain, A., & Yanti Rahmadhani, A. (2021). Analisis Sentimen Masyarakat Terhadap Program Kartu Prakerja Pada Twitter Dengan Metode Support Vector Machine. *Jurnal Informatika: Jurnal Pengembangan IT*, 6(3), 150–155.
- Muhammad Fernanda Naufal Fathoni, Eva Yulia Puspaningrum, & Andreas Nugroho Sihananto. (2024). Perbandingan Performa Labeling Lexicon InSet dan VADER pada Analisa Sentimen Rohingya di Aplikasi X dengan SVM. *Modem : Jurnal Informatika Dan Sains Teknologi.*, 2(3), 62–76. <https://doi.org/10.62951/modem.v2i3.112>
- Muhammadi, R. H., Laksana, T. G., & Arifa, A. B. (2022). Combination of Support Vector Machine and Lexicon-Based Algorithm in Twitter Sentiment Analysis. *Khazanah Informatika : Jurnal Ilmu Komputer Dan Informatika*, 8(1), 59–71. <https://doi.org/10.23917/khif.v8i1.15213>
- Najib, A. C., Irsyad, A., Qandi, G. A., & Rakhmawati, N. A. (2019). Perbandingan Metode Lexicon-based dan SVM untuk Analisis Sentimen Berbasis Ontologi pada Kampanye Pilpres Indonesia Tahun 2019 di Twitter. *Fountain of Informatics Journal*, 4(2), 41. <https://doi.org/10.21111/fij.v4i2.3573>
- Nur, I. F., Herdiani, A., & Astuti, W. (2019). Analisis Sentimen Berbasis Leksikon InSet Terhadap Partai Politik Peserta Pemilu 2019 Pada Media Sosial Twitter. *E-Proceeding of Engineering*, 6(3), 1–11.
- Oktavia, D., Ramadahan, Y. R., & Minarto. (2023). Analisis Sentimen Terhadap Penerapan Sistem E-Tilang Pada Media Sosial Twitter Menggunakan Algoritma Support Vector Machine (SVM). *KLIK: Kajian Ilmiah Informatika Dan Komputer*, 4(1), 407–417. <https://doi.org/10.30865/klik.v4i1.1040>

- Pamungkas, F. S., & Kharisudin, I. (2021). Analisis Sentimen dengan SVM, NAIVE BAYES dan KNN untuk Studi Tanggapan Masyarakat Indonesia Terhadap Pandemi Covid-19 pada Media Sosial Twitter. *Prosiding Seminar Nasional Matematika*, 4, 1–7. <https://journal.unnes.ac.id/sju/prisma/article/view/45038>
- Pramayasa, K., Maysanjaya, I. M. D., & Indradewi, I. G. A. A. D. (2023). Analisis Sentimen Program Mbkm Pada Media Sosial Twitter Menggunakan KNN Dan SMOTE. *SINTECH (Science and Information Technology) Journal*, 6(2), 89–98. <https://doi.org/10.31598/sintechjournal.v6i2.1372>
- Rina Noviana, & Isram Rasal. (2023). Penerapan Algoritma Naive Bayes Dan Svm Untuk Analisis Sentimen Boy Band Bts Pada Media Sosial Twitter. *Jurnal Teknik Dan Science*, 2(2), 51–60. <https://doi.org/10.56127/jts.v2i2.791>
- Salam, R. R., Jamil, M. F., Ibrahim, Y., Rahmadden, R., Soni, S., & Herianto, H. (2023). Analisis Sentimen Terhadap Bantuan Langsung Tunai (BLT) Bahan Bakar Minyak (BBM) Menggunakan Support Vector Machine. *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, 3(1), 27–35. <https://doi.org/10.57152/malcom.v3i1.590>
- Sandi, D., Utami, E., & Kusnawi, K. (2023). Analisis Sentimen Publik Terhadap Elektabilitas Ganjar Pranowo di Tahun Politik 2024 di Twitter dengan Algoritma KNN dan Naïve Bayes. *Jurnal Media Informatika Budidarma*, 7(3), 1097. <https://doi.org/10.30865/mib.v7i3.6298>

LAMPIRAN

Lampiran A. Hasil perbandingan label manual dengan metode *lexicon based*

Lampiran ini berisi hasil pelabelan manual dari dataset yang didapatkan melalui *website crawling* terkait dengan opini publik di sosial media x terhadap partai politik di Indonesia. Pelabelan ini akan dilakukan dengan 3 kelas yaitu label negatif, positif, dan juga netral. Proses pelabelan ini menggunakan data contoh dari hasil *crawling* dengan jumlah 100 dataset yang dilakukan oleh :

Nama : Ahmad Naufal Hilmy
Program Studi : Pendidikan Bahasa dan Sastra Indonesia
NPM/NIM : 2020124131
Nama Perguruan Tinggi : Universitas Negeri Yogyakarta

Beliau berperan penting dalam membantu pelabelan secara manual guna untuk melakukan pelabelan sentimen yang terkandung dalam dataset yang digunakan. Proses ini bertujuan mengetahui tingkat kepercayaan dari metode *lexicon based* dalam melakukan pelabelan sentimen publik terhadap partai politik di Indonesia. Dari 100 data contoh dibawah, didapatkan sebanyak 90 data yang sama - sama benar baik menggunakan menggunakan metode *lexicon based* maupun manual. 10 data sisanya terjadi kesalahan pelabelan dengan menggunakan metode *lexicon based* dalam melakukan pelabelan yang mayoritas data tersebut memiliki aspek bertingkat yang dimana hal tersebut merupakan kelemahan dari *lexicon based* yang tidak bisa mengenali konteks, sarkas atau ironi pada data.

Yogyakarta, 1 Desember 2024
Mengetahui,
Mahasiswa Pendidikan Bahasa dan Sastra Indonesia

Ahmad Naufal Hilmy
20201241031

No.	Dataset	Lexicon Based	Manual
1.	@lauklaut @detikcom Dagelan anjir parpol di Indonesia. Aturan aja bisa dibokongin suka2 petingginya wkwkwkwkwk sampis emang Fi	negatif	negatif
2.	@ferrykoto @DPR_RI Tidak ada parpol yg baik di Indonesia	positif	positif
3.	@RfgL88 dlm kasus DPR/D aturan memang membolehkan krn yg berkuasa parpol. kita memilih parpol bukan orang. mau gimanalagi. soal dinasti.? ah.. itu cm kl terjadi sm orang lain kl sm saya ya bukan	positif	positif
4.	Salah satu yang harus di reformasi adalah parpol. Ketika pilihan kita menang namun bisa jadi tak menjabat karena di depak aturan parpol. Inilah liciknya parpol dan rakyat harus sadar.	negatif	negatif

5.	@roarrgh @03__nakula Nah itu dia makanya negara perlu Oposisi. Hari ini kan Hampir gak ada parpol yang menempatkan diri sebagai oposisi Pemerintah. Saat rakyat langsung yang jadi Oposisi malah diperlakukan kurang baik. Lalu dimana letak Demokrasinya kalo Oposisi hendak dibungkam? #GWSIndonesia	positif	positif
6.	@tjitrosoenarjo1 Susah karna pakai parpol Jadi harus seperti itu apa bedanya sama outsourcing parpol itu bukan kemauan pilihan rakyat sendiri. Intinya bangsa ini maju hrs ada pemimpin yg jujur dan amanah semua teknologi dari industry pertambangan pertanian dll. hrs di kuasai sndr no asing	netral	netral
7.	@peppertipexx Kampus dengan aktivis dagelan yang mahasiswanya lupa belajar dan lupa akar rumput? Lagian kampus jelek kok dibanggain Pantas aja lebih memilih sekolah ke luar negeri. Kampus lokal cuma menelorkan aktivis senam jempol dan tukang teriak-teriak biar dilirik parpol.	negatif	negatif
8.	10 tahun terakhir memang rezim paling busuk. Rezim parpol maksudnya. Berteriak paling kencang soal MK ditambah adegan menangis ala drama Korea saat anggota partai sendiri di-zalimi demi mencarikan kerja utk cucu ketum malah diem aja. Ini baru namanya politik rezim dinasti.	negatif	negatif
9.	@tjitrosoenarjo1 Cuma orang tolol yang percaya sama parpol di Indonesia. Selama parpol diisi oleh lonte2 politik selama itu pula parpol hanyalah alat legitimasi untuk berkuasa. No more no less	negatif	negatif
10.	Preman Politik yg begal partai pecah belah partai nyandera para Ketum Parpol dg kasus korupsi dan yg gergaji pohon beringin itu LEBIH JAHAT dari Preman Bloon Pembubar Acara Diskusi.	negatif	negatif
11.	Apabila kita melihat ke belakang sebagai guru untuk melangkah di masa depan kasus2 korupsi ini apa tidak ada kaitannya dengan parpol tertentu? Secara jabatan2 koruptor ini sarat dengan rekomendasi elit politik. Tp kenapa mahasiswa maupun tokoh enggan https://t.co/Xu50sITRyy	negatif	negatif
12.	Wahai rakyat kalian sadar gak kalau kalian ditipu mentah2 sama parpol. Sosok yg kalian pilih mewakili kalian dlm kehidupan berdemokrasi dengan seenaknya diganti oleh parpol atas nama kepentingan parpol. Silakan kalian bersikap menghadapi pileg berikutnya.	positif	positif
13.	@NenkMonica salah satu kberhasilan..dpt dukungan parpol..salah stu syrt mju kontestasi politik... klo yg nga dpt dukungan dn merajuk itu	negatif	negatif

	nga brhasil..urusan nnt dipilih nga nya second matter...		
14.	@naceyawade Pake motif logo parpol apa mau kak?	netral	netral
15.	Pakar Hukum Sebut Pemecatan Anggota DPR Terpilih Cerminkan Internal Parpol Belum Demokratis https://t.co/eSh1NSIMoe #TempoNasional	positif	positif
16.	@PKSejahtera Nahi munkar? Di Indonesia zaman now Baru FPI yg melaksanakan Dgn segala kekurangannya FPI sbg ormas bisa PKS sbg parpol harusnya lebih bisa	negatif	negatif
17.	Parpol perlu ambil bagian cegah hoaks ciptakan Pilkada 2024 damai https://t.co/328icWOXbi	positif	positif
18.	@AriefOpg @ArekNdeso79 @03__nakula @lompong82 Adanya dekrit krn kasus bulog dan Brunei tdk dpt dbuktikan alias fitnah dan taufiq kemas turut serta melobi parpol jika GD tdk mau dilengserkan ada deal politik agar mentri diganti ssuai keinginan parpol bkn ssuai GD tp GD menolak dgn tegas dr pd hrs komprosi dgn poltisi busuk	negatif	negatif
19.	@kozirama Dari dlu kenapa susah bgt ya dapet Menpora yg bener. Nasib jabatan politik jarang diisi yg bener2 ahli cuman jadi jatah parpol.	negatif	negatif
20.	@PngAdilnR4kyt IBLIS-IBLIS SESAT PD KEPINGIN VIRAL MENJADIKAN SEMUA KARYA KERJA JOKOWI DAN IKN SBG SASARAN PEMBUATAN KONTEN YG BISA MENDONGKRAK POPULARITAS MENJANJI DAN MENDAPAT BAYARAN KEPENTINGAN POLITIK PARPOL PECUNDANG DAN ASING-ASENG DPR-RI CUMA KOREA-2 PENUNGGU PERINTAH KETUM ..	negatif	negatif
21.	Harusnya di baris keempat masukin coblos semua (biar adil). Karena kalau saja Warga Jakarta yang bermukim di LN seperti saya memiliki hak ikut Pilkada saya akan coblos semua. Atau jika saya pas di Jakarta saya akan datang ke TPS dan coblos semua heuheu. Parpol alih-alih	positif	positif
22.	KILAS BALIK KPU TDK LAYAK UTK PEMILU. Pd pilpres 2024 parpol pengusung 01 dan 03 tahu jk KPU berpihak pd 02 sampai gugat ke MK Mengapa parpol yg di curangi KPU tsb tdk ada upaya utk mengganti KPU dlm bentuk lembaga penyelenggara pemilu yg lbh INDEFENDEN? https://t.co/v2bMePuArE	netral	netral
23.	@abu_waras @prabowo GIBRAN DIGANTI .. BERARTI RAKYAT NKRI DIBAWAH JAJAHAN KEDAULATAN PARPOL TERBUKTI SUARA RAKYAT DAN PENGHAMBURAN RATUSAN TRILYUN	negatif	negatif

	DUIT PEMILU DIANGGAP PENGGUGUR KEWAJIBAN DEMOKRASI 5 TH-AN MENUTUPI TIPUAN PEMBENARAN KOMPROMI SEPIHAK PARA KETUM PARTAI POLITIK KOALISI. BIADAB !!!		
24.	@PartaiSocmed Mungkin aneh kalo saya moment di sini karna saya tak tahu harus kemana Apakah bapak sekalian bisa membatu saya saya tiba terdaftar sebagai anggota parpol saya tidak pernah mendaftar yang meyebabkan saya gagal jadi anggota kpps	negatif	negatif
25.	Bocoran Ikrar dari Petinggi Parpol: IKN Pencitraan Jokowi Sama Seperti Esemka https://t.co/oleRqrQxxp	negatif	negatif
26.	@bbtjkiy @gvffjtj @prabowo @gibrantweet @PDemokrat @AgusYudhoyono @puanmaharani_ri @RWati65 @kurawa @abby6mei Kabinet Zaken nggak berarti anti-parpol. Profesionalisme bisa datang dari mana saja.	netral	netral
27.	@gvffjtj @prabowo @gibrantweet @PDemokrat @AgusYudhoyono @puanmaharani_ri @RWati65 @kurawa @abby6mei Zaken kok masih bawa-bawa nama-nama parpol?	netral	netral
28.	@jokoanwar Ormas piaraan negara piaraan parpol. Untuk melakukan pekerjaan kotor dan jadi lumbung suara. Rakyatnya diperas dan diancam sama ormas2 ini. Anggotanya dijadikan bodoh dan pemalas supaya bisa jadi anjing2 peliharaan. Nggak ada satupun tokoh politik yang mau bubarkan mereka.	negatif	negatif
29.	Coblos Semua Paslon di Pilgub Jakarta Tokoh Aceh Utara: Itu Ekspresi Politik yang Tak Boleh Dikriminalisasi. Gerakan coblos merupakan protes voting yg kritis sehrsnya disikapi dgn bijaksana oleh parpol elite politik penyelenggara pemilu dan pemerintah https://t.co/wPV5spLTSy https://t.co/PRwDzWdcLl	positif	positif
30.	@yaniarsim Ormas piaraan negara piaraan parpol. Untuk melakukan pekerjaan kotor dan jadi lumbung suara. Rakyatnya diperas dan diancam sama ormas2 ini. Anggotanya dijadikan bodoh dan pemalas supaya bisa jadi anjing2 peliharaan. Nggak ada satupun tokoh politik yang mau bubarkan mereka.	negatif	negatif
31.	@PaltiWest2024 ini tuh jadi catatan or at least perhatian si @ridwankamil atau tim dia yg di dukung begitu bnyk Parpol baik partai yg ada di parlemen maupun parpol non parlemen. ini sih kalo gue liat @pramonoanung dan bang Doel akan menang dgn mudah	positif	positif
32.	@CNNIndonesia Parpol lebih bangsat dr preman	negatif	negatif

33.	Pastiin orang kompeten dan berintegritas harusnya dilakukan partai politik. Kalau parpolnya pragmatis ya susah. Soal #FufufafaWapresPsikopat dia juga parpol yg usung kan? Yg penting gimana bisa menang halalin segala cara terabas etika itu parpol kacrut! Catatan 1 Oktober	negatif	negatif
34.	1. Apakah dasar opini Anda terkait adanya kandidat bhw dia eks antek mulyono bukannya masih? 2. Bagaimana mekanisme prinsip demokrasi bisa terpenuhi sedangkan semua kandidat adl figur yg disodorkan oleh parpol tnp melalui proses usulan pendapat rakyat? 3. HANCURKAN KOTAK SUARA	negatif	negatif
35.	@aniesbaswedan Kalah di pilpres gagal nyagub dijauhi parpol masih mengandalkan kadrun jualan agama..??	negatif	negatif
36.	@choymarkochoy @HMTata3 @Yurissa_Samosir @aniesbaswedan @nomerorae @ay_prakasa2225 @MuhiqR @Panitia68285950 @yaniarsim @JamalBoegis @mattdilaga @EkoPutr86520727 @Putri96960977 Ya aneiesnyanjedog bae. Klo mw maju dan tw dijegal sana sini hrsnya ia masuk parpol. Kan kagak. Mwnya enak dewek. Pt 7.5% turun setelah pks deklarasi. Ga ns mundur krn ada kadernya disitu.	positif	negatif
37.	@JhonSitorus_18 woi kader @PDI_Perjuangan harun masiku ke mana? bajingan koruptor terhebat ada di parpol mana lae ?	negatif	negatif
38.	@PartaiSocmed Masih ada yang percaya PDIP aja udah aneh. Harusnya tuh parpol jangan dikasih bangkit lagi.	positif	negatif
39.	Survei Paslon Pilkada Jatim 2024 Persaingan Luluk Khofifah Risma Parpol dan Sebaran Pemilihnya https://t.co/veQ26NnLUK lewat @tribunkaltim	netral	netral
40.	@NenkMonica @ridwankamil @PKSejahtera @aniesbaswedan @officialMKRI sudahlah kalo lw pndykung @aniesbaswedan sgera declare parpol baru ... gak usah sok asik dan sok usik urus parpol lain lw mestinya tanya ke internal pkb - nasdem - pks knapa Anies gak jd dipilih Jangan cuma bikin opini sepihak itu namanya otak sungsang	negatif	negatif
41.	@ferrykoto Pendukung anis kalo mau dia buat parpol ya urunan donk. Jngan mau enak nya aja kasihan junjungan mu montang manting sendiri. Bacot doang uang gak turun	positif	positif
42.	@ShadowBone_ @RaffiAhmad Kampusnya university's yang kasih gelar HC ke Raffi Ahmad cukup dalam hati. Cc @RaffiAhmadLagi #FufufafaWapresPsikopat #FufufafaWapresPsikopat	netral	netral

43.	@Hilmi28 Presiden Jokowi Dodo saja tidak betah diam di IKN sangat DUNGU MENGELOLA NEGARA INI HMBUR UANG HAMBUR PIKIRAN MINIM PENGETAHUAN KOK BISA JADI PEMIMPIN ; IYALAH ITU KAN DIAJUKAN OLEH PARPOL-PARPOL	negatif	negatif
44.	@Piyusaja2 Maling dtahan dhukum tdk ada parpol yang tdk ada malingny apakah semua parpol hars dpersepsikan sarang koruptor ?	negatif	negatif
45.	Setwan DPRD Riau Belum Terima Surat dari Dua Parpol untuk Calon Pimpinan https://t.co/3F0hiZPGEk	positif	positif
46.	@Catatan_ali7 @ZUL_Hasan Permendag 8/2024 duitnya gede Cuk. Dari importir yang ucapkan terima kasih koceknnya jos. Walaupun membunuh anak2 bangsa yang lain tapi dia dapet duit segede gaban. Makanya @ZUL_Hasan tega bikin keputusan jahat buat Rakyat #FufufafaWapresPsikopat #FufufafaWapresPsikopat	negatif	negatif
47.	Dari parpol yang proporsinya tidak terlalu lebih besar daripada yang profesional katanya.	netral	netral
48.	Begitu juga dengan nama-nama terutama dari parpol yang proporsinya tidak terlalu lebih besar daripada yang profesional. Nah ini masih kemudian ada yang masuk ada yang tarik sambungnya.	netral	netral
49.	@tempodotco DPR tidak selalu mewakili rakyat biasanya mewakili Istana dan ParpolParliament itu tempat nya untuk berpolitik bukan tempat nya untuk rakyatrakyat hanya kambing conge sedangkan IKN di putuskan oleh President di setuju oleh Parpol dan di ikuti rakyat	positif	positif
50.	Kisah-kisah menarik apa saja yang terjadi pada kampanye parpol di masa itu? Bagaimana PKI berhasil meraup suara dengan cara yang efektif dan efisien berbiaya rendah tetapi bisa mendapatkan hasil yang maksimal? #G30SPKI https://t.co/4FaIjUJ3zC	negatif	negatif
51.	Bu Risma di pilih bukan dari hasil kompromi elite parpol bukan hanya cuman sekedar balas budi tapi di pilih karena keberhasilannya memimpin surabaya saat menjadi walikota dan saat menjadi Mensos telah di anggap sukses Fakta....	positif	positif
52.	Banyak Akademisi Siap Gabung ke Wadah Gerakan Anies Sebaiknya Bentuk Ormas Sebelum Parpol. Banyak sivitas akademika yg mendukung gagasan Anies Baswedan. Jika ormas msh memungkinkan mrk bisa bergabung dan masuk dalam struktur kepengurusan. https://t.co/7IJ6UX9bK4 @JeblukAkun https://t.co/piA5y2hdK3	positif	netral

53.	Praktik Demokrasi Internal Parpol Dikritik Parpol Mirip PT atau Milik Keluarga https://t.co/1ldBgx1KoJ via @suaraislamID_	negatif	negatif
54.	@Mantanmontir22 @RagilSemar Karena di Era Pak Harto Partai Politik itu tidak bisa masuk Pemerintahan hanya bisa di DPR saja sbg Wakil Rakyat.. Tentu para Politikus tidak suka dg hal ini mereka membuat Reformasi agar Parpol bisa masuk ke dalam Pemerintahan siapapun bisa jadi meski tanpa keahlian..	positif	positif
55.	@AbdiNata035 @realfedinuril @jokowi @DPR_RI Parpol.ny lbh brengsek dri presiden ny...semua ketua parpol tersangkut masalah korupsi mka ny jkwi gampg jerat mereka	negatif	negatif
56.	Pasca Pemilu Parpol Cenderung Bentuk Koalisi Besar Emoh Jadi Oposisi https://t.co/msp6nyRsqr #InvestorDaily #investasi #Saham	negatif	negatif
57.	@ferrykoto Capek bgt jelasin. Kami yg kau sebut Anak Abah itu utamanya bukan mau memenangkan Anies jd presiden lah atau bikin parpol dan menang pileg lah. Kami cuman mengkritik kebijakan pemerintah kok. Gak pandang bulu. Misal presidennya skrg Anies jg kami jg tetep kritik kalau ada salah.	negatif	negatif
58.	@yourbae ditolak parpol kah kawand?	netral	negatif
59.	Teori konspirasi otak gue mereka cmn alat buat nejutuhin parpol yg banyak itu. WKWKWK	negatif	negatif
60.	@Mdy_Asmara1701 2 hari jadi anggota parpol.. langsung bisa jadi ketum. Kira2 siapa yg nggak waras ya @kaesangp ? Sok asik.. semua bentuk bully/kritik lu iyain dijadikan candaan. NDESO..	netral	netral
61.	Ada yang diusulkan parpol pendukung ada juga yang diusulkan kelompok lain seperti organisasi masyarakat kelompok profesi dan lain sebagainya.	positif	netral
62.	Agak lain memang Anak2 Abah ini. Sebagai buih khayalan mereka tinggi sekali. Sampai masuk jebakan @PartaiSocmed saking merasa mimpinya akan wujud. Lagipula Anies itu sopo? Memangnya Anies punya Parpol. Punya kekuatan di Parlemen? Parpol yg usung dia saja dituding tersandera https://t.co/Tz8RBGQjKb	positif	negatif
63.	@lucywiryono Itu sudah jadi fenomena Bu... Lihat saja... Di sekolah ada bully.. Di jalanan ada geng motor.. Di parpol ada pembegalan.. Pilkada ada penghalangan kandidat.. Di dpr ada persengkokolan.. Di tambang ada yg ilegal.. Di diskusi aktivis dibubarin preman .. Mengenaskan memang Bu..	negatif	negatif
64.	Susunan kabinet Prabowo-Gibran dikatakan masih dinamis lantaran mempertimbangkan porsi menteri profesional akan lebih banyak dari kalangan partai politik (parpol). Hal ini diungkap	positif	positif

	oleh Ketua Harian DPP Partai Gerindra Sufmi Dasco Ahmad Senin (30/9). #kabinetprabowogibran https://t.co/epPDmuJEWr		
65.	Ada juga caleg gagal yg artinya tdk dipilih tapi bisa dilantik koq.. Yg salah bukan hanya parpol doang...penyelenggara jg bnyk yg bangcat	negatif	negatif
66.	KILAS BALIK KPU TDK LAYAK UTK PEMILU. Pd pilpres 2024 parpol pengusung 01 dan 03 tahu jk KPU berpihak pd 02 sampai gugat ke MK Mengapa parpol yg di curangi KPU tsb tdk ada upaya utk mengganti KPU dlm bentuk lembaga penyelenggara pemilu yg lbh INDEFENDEN? https://t.co/orwzdsKCqX	netral	netral
67.	KILAS BALIK KPU TDK LAYAK UTK PEMILU. Pd pilpres 2024 parpol pengusung 01 dan 03 tahu jk KPU berpihak pd 02 sampai gugat ke MK Mengapa parpol yg di curangi KPU tsb tdk ada upaya utk mengganti KPU dlm bentuk lembaga penyelenggara pemilu yg lbh INDEFENDEN? https://t.co/sYmO92tfex	netral	netral
68.	@Boediantar4 Persetujuan siapa?DPR wakil parpol bukan wakil rakyat kalo mau tahu https://t.co/RnLJXULOuZ	positif	negatif
69.	@PaltiWest2024 Pemimpin opo cuma wakil. Yg jadi pemimpinnya ya putra asli parpol.	positif	positif
70.	@kumparan Melayani dinasti @adearmando61 @Dennysiregar7 @DokterTifa @mohmahfudmd @msaid_didu harus tahu ini... Jgn bicara internal parpol krn negara dimulai dijalankan oleh parpol sbg jembatannya.	netral	netral
71.	@kompascom Memang ada yang berani ngelawan anaknya ketum parpol ??	positif	positif
72.	@geloraco Ternyata Siapa Yg sebenarnya Pemecah belah Bangsa ini...sudah kelihatan otoriternya...makanya Semua Parpol dibujuk masuk ke kelompok mereka biar tdk ada Yg mengkritik Pemerintahan Jokowi dan Pak Wowo.. Apakah kita akan kembali ke Zamannya Soeharto?	negatif	negatif
73.	@realfedinuril @jokowi Kl gembel yg menganalisis malah jd menarik nih mas jg tak ada keinginan & aspirasi dr rakyat utk kepikiran ibukota dipindah mimpi pun tidak! Jk p' jkw klaim disetujui @DPR_RI jelaslah bkn krn diijinkan rakyat sebab politis skr itu artinya suka2 parpol pendukung rezim sj	positif	positif
74.	UU MD3 tak direvisi aturan ketua DPR tetap pada parpol pemenang Pemilu 2024. https://t.co/WrDLLwAnsg	positif	positif
75.	@yaniarsim Bisa bisanya preman tolol bisa dekat dengan elit2 parpol? Apa sengaja dipelihara? Makin rusak saja Demokrasi di Indonesia. @jokowi @prabowo @SBYudhoyono	negatif	negatif

	@PartaiGolkar @NasDem @PDemokrat @PKSejahtera @pakmul63 @MataNajwa @ernestprakasa @sudjiwotedjo @susipudjiastuti @iwanfals		
76.	Jumlah calon menteri kabinet Prabowo dari parpol diperkirakan tak akan lebih banyak dibandingkan dari profesional. https://t.co/6IeB47H3hh #Polhuk #AdadiKompas	netral	netral
77.	Kalau ada ketum parpol bilang: Tingali aing mah bisa korupsi tapi teu ditewak. Terus disambut tepuk tangan dan sorak sorai pendukungnya. Ceuk aing mah gering kabeh sih.	negatif	negatif
78.	@mas_veel Menteri banyak jatah parpol pendukung.. Menteri hak prerogatif presiden hanya teori presidensiil. Presiden hanya kebagian menkeu (bu sri).dan pupr (pak bas)	positif	positif
79.	Kader kader Parpol sebaiknya duduk dikursi legeslatif sebagai wakil Rakyat yang ada di Parleman bukan duduk di kursi eksekutif di Pemerintahan sebagai Menteri.. Tradisi dan budaya yang ada di lingkungan Parpol kita sudah kebiasaan keenakan duduk di kursi Menteri barang kali ya	positif	positif
80.	@fjsdvmps kktawa tlga tandem nitong dalawang to	netral	netral
81.	Dengar2 Anies dijegal mendirikan Parpol. Katanya semua takut jk Anies berhasil mendirikan Parpol nanti Parpolnya akan besar dan menang Pileg 2029 raih suara terbanyak. Benarkah? Naga2nya benar ya. Soalnya anak Abah jumlahnya banyak. Pasti menang Pileg. Di X.	positif	negatif
82.	Dalam acara yang mengambil tempat di bawah patung garuda Alun-alun kabupaten Trenggalek tersebut dihadiri oleh jajaran Forkopimda penyelenggara pemilu pasangan calon Parpol pengusung serta stakeholder terkait lainnya. Selasa (25/9).	netral	netral
83.	@punkjabat @toketriot Dia mah ternakan orang parpol	netral	negatif
84.	@mas_veel Sistem pemerintahan kita adalah presidensial artinya presiden berhak menga angkat dan menghentikan mentri tanpa terikat parpol .kecuali presiden tersandra oleh parpol	negatif	negatif
85.	@JlvJolavi3j @StefanAntonio__ Gibran terpilih melalui konstitusi .Pilpres bisa menang dan bisa kalah krn dipilih rakyat..kalo tdk suka Gibran ya jgn dipilih kok repot.Buktinya rakyat mayoritas pilih Gibran..dan Mayoritas parpol pilih Gibran..	positif	positif
86.	@musksigit Saya juga baca berkali-kali juga ga ngerti maksudnya sender ini apa. Kayaknya dia kira om Sigit ini orang yg sama di kasus makan tiga bayar satu yg katanya petinggi parpol onoh	netral	netral

87.	@Adiprayitno_20 NEGARA LAIN MERASA BANGGA KALAU NILAI MATA UANGNYA NAIK ITU TANDANYA MAJU? INDONESIA TAK PEDULI NILAI MATA UANGNYA MEROSOT YANG PENTING PARPOL MENANG?	positif	Negatif
88.	@RyaWiedy udah basi budos... udah nggak trending lagi... yg lagi laku ya isu dinasti walau hampir semua parpol melestarikan budaya dinasti... kata yg lagi trending skrg antara mulyono dan putra mulyono ... rada kurang kreatif ya...? nggak kayak tag 2019GantiPresiden itu bagus tuh	positif	positif
89.	@VIVAcoid Ketua parpol karbitan kualitas ya begini.	netral	netral
90.	@yourbae setelah jadi pemain film tujuan selanjutnya jadi anggota parpol bg?	netral	netral
91.	Begitu juga dengan nama-nama terutama dari parpol yang proporsinya tidak terlalu lebih besar daripada yang profesional kata dia.	netral	netral
92.	Di sisi lain menurut dia tokoh utusan parpol yang ditunjuk untuk terlibat di kabinet tidak akan terlalu banyak ketimbang kalangan profesional.	netral	netral
93.	@alisyarief pindah ibukota emang sgampang balikan telapak tangan.president selanjutnya utk mngeksekusi lanjutannya.begini lh krn jokowi bermanuver keluar dr parpol.barisan 10thn sakit hati selalu berniat utk menjatuhkn.sabar pak Jokowi pendukung garis kerasmu diam belum tentu tidak membelamu.	negatif	negatif
94.	@NenkMonica Item yg sangat berharga dr salah satu parpol yg dipersembahkan buat rakyat Indonesia he he	negatif	negatif
95.	@BudiBukanIntel Apakah akan jadi SA (Sturmabteilung)? Apakah mereka akan mengusulkan agar Jokowi membentuk parpol baru dan mereka akan memaksa masyarakat untuk memilih parpol tersebut?	negatif	negatif
96.	@jokoanwar Dan kenapa tak pernah ada pemerintahan yang serius melakukan usaha-usaha untuk menghilangkan budaya buruk yang satu ini? Bos sadar gak sih kalau pemerintah itu ya parpol itu bagian dari premanisme itu sendiri? tidak mungkin mereka menghentikan kegiatan kawan mereka sendiri	negatif	negatif
97.	Salut sama Mas Pram beliau tdk punya beban politik meskipun di usung oleh parpol tapi beliau akan senantiasa bkrj keras demi warga Jakarta meskipun kelak beliau tdk terpilih lagi. Emang jarang sih ada pemimpin yg pny pola pikir sprti ini. #JakartaMenyala #PrakarUntukJakarta https://t.co/4JpISvSvxU	negatif	positif
98.	@aingriwehuy Bayik karbitan parpol dijadikan menteri ya mentalnya tetep spt anak anak	netral	netral

99.	Saya jamin dan pastikan ini postingan orang pekok kalo tdk sakit jiwa. Pejabat daerah yg usung itu parpol atau independen dg syarat tertentu. Lalu di pilih rakyat. Lewat pemilu. Woy orang gila negara kita negara demokrasi buka negara warisan. Jangan krn kalah terus jadi gila. https://t.co/yUtf6wEKvz	netral	negatif
100.	Perumus Tatib Delegasi Fraksi ABN dari Dua Parpol https://t.co/Y5R1BESy4L	negatif	negatif