BAB I
PENDAHULUAN

1.1. Pendahuluan

Perkembangan teknologi di era yang serba digital ini sudah sangat pesat baik dalam hal laju pengumpulan dan penyebaran informasi saat berselancar di dunia internet. *Youtube* sebagai platform *online* memiliki peran penting dalam distribusi informasi dan hiburan kepada berbagai lapisan masyarakat. Pertumbuhannya yang cepat di Indonesia menjadikan *Youtube* sebagai platform media sosial yang paling populer dan digemari masyarakat (Rustang et al., 2023). Dengan adanya fitur komentar yang tersemat di tiap video *Youtube*, kreator ataupun antar sesama pengguna dapat menilai beberapa aspek dalam lautan opini yang dibuat penonton, seperti; pro dan kontra terhadap konten tersebut, ataupun konteks khusus didalam video tersebut yang dinilai penting sehingga ingin diangkat. Sentimen yang dituangkan penonton didalam komentar tersebut dapat menjadi bahan evaluasi dan menjadi dasar dalam mengambil sebuah kebijakan kedepannya. *Youtube* bukan hanya sekadar wadah untuk berbagi hiburan, tetapi juga menjadi media dalam memberikan edukasi politik, khususnya dalam momen-momen penting seperti pemilihan umum.

Pemilihan Umum (PEMILU) Calon Presiden dan Calon Wakil Presiden (Capres dan Cawapres) adalah sebuah kontestasi yang bertujuan untuk memilih kepala Negara melalui proses demokrasi, masing masing kontestan diusung melalui gabungan partai politik (Gultom et al., 2021). Melalui *Youtube*, saluran-saluran berita berperan besar dalam mengawal pesta 5 tahunan ini untuk mempermudah masyarakat dalam memperoleh informasi masing-masing kandidat. Salah satu acuan masyarakat dalam menilai kandidat yang berkompeten dan pantas dalam memimpin adalah dengan menonton Debat Calon Presiden yang diselenggarakan oleh Komisi Pemilihan Umum (KPU). Hal ini sejalan pada reaksi masyarakat dalam menyikapi debat pemilu yang sudah kali ke 6 terlaksana di Indonesia sejak era reformasi. Setelah debat putaran ke-5 (yang merupakan debat terakhir) telah dilaksanakan. Perdebatan di berbagai lapisan masyarakat seluruh Indonesia pun tak dapat terhindarkan. Debat kelima ini mengangkat tema-tema strategis, seperti pendidikan, kesehatan, ketenagakerjaan, kebudayaan, teknologi, informasi, serta kesejahteraan sosial dan inklusi yang menjadi isu krusial bagi masa depan bangsa. Sebuah metode analisis tentu sangat dibutuhkan untuk memanfaatkan momen ini agar sebaran pendapat terkait arah dukungan ataupun netralitas setelah menonton debat terakhir Capres dan Cawapres pada kolom komentar *Youtube* dapat diekstrak dan dipetakan sehingga dapat diketahui intisarinya dengan mudah. Oleh karena itu, analisis sentimen hadir dalam mengatasi permasalahan ini.

bekerja dengan mengkomputasi opini dan emosi orang terhadap wa atau individu/kelompok. Hal ini bertujuan agar metode dapat o komentar dengan output berupa polaritas dari hasil klasifikasi (Medhat et al., 2014). Analisis ini akhirnya dapat membuat sentimen-sentimen yang dihadirkan oleh masyarakat itu sendiri pandangan baru, salah satunya yaitu arah dukungan berserta

kriteria yang ditetapkan sebagai dasar dalam memilih salah satu kandidat. Penerapan metode pembelejaran mesin yang sering digunakan pada berbagai jurnal penilitian seperti K-NN, *Naïve Bayes*, *Random Forest* dan *Support Vector Machine* (SVM) (Arsi & Waluyo, 2021).

Support Vector Machine (SVM) merupakan salah satu metode yang sudah banyak diterapkan untuk berbagai jenis penelitian dibidang data dan *text mining* karena mampu menunjukkan performa lebih baik. SVM mula-mula diusulkan oleh Vapnik pada tahun 1995. Algoritma SVM didasarkan pada fungsi-fungsi linier dalam sebuah ruang fitur berdimensi tinggi, metode SVM dapat mengklasifikasian data kedalam dua kelas, berupa 0 atau 1 (biner) ataupun multikelas (Alita et al., 2020). SVM yang memiliki dasar perhitungan matematika yang solid, dan generalisasi yang tinggi, dan bersifat praktis jika menemui kasus klasifikasi linier ataupun nonlinier merupakan beberapa keunggulan yang dimiliki SVM.

Salah satu cara untuk meningkatkan model pelatihan SVM yaitu dengan pendekatan metode *ensemble*. Metode tersebut bekerja dengan menggabungkan beberapa model *classifier* dengan tujuan untuk mendapatkan model baru hasil dari model gabungan beberapa *classifier*, model gabungan yang telah terbentuk dinilai lebih akurat jika dibandingkan dengan model awal dalam kinerja klasifikasi (Han et all, 2012). Salah satu algoritma terkenal dalam metode ensemble yaitu *boosting*, dan *Adaboost* adalah salah satu algoritma terpopuler dalam hal *boosting.* Pada *Adaboost*, kumpulan data pelatihan yang digunakan untuk setiap *classifier* dipilih berdasarkan kinerja *classifier* sebelumnya. Sampel yang tidak diprediksi dengan benar oleh *classifier* dalam proses tersebut akan dipilih lebih sering daripada sampel yang diprediksi dengan benar. Sehingga prinsip dasar dari *Adaboost* yaitu memberi perlakuan khusus pada pengamatan yang berstatus kesalahan klasifikasi (Listiana et al., 2023)

Pada penelitian sebelumnya yang dilakukan Arsi dan Waluyo pada tahun 2021, peneliti membahas sentimen data *komentar Youtube* terkait wacana pemindahan ibu kota Indonesia menggunakan metode SVM dan membandingakannya dengan metode BM25, KNN, dan *Naïve Bayes*, SVM lebih baik daripada 3 metode yang lain dengan akurasi sebesar 96,68%. Pada penelitan lain dilakukan oleh Mediana dan Yamasari pada tahun 2024, peneliti membandingkan metode SVM dengan *Decision Tree* untuk menganalisis sentimen dari ulasan aplikasi *Carousell* pada *Google Play*, dengan hasil evaluasi klasifikasi tertinggi diperoleh SVM dengan kernel RBF dengan akurasi sebesar 93,1%. Adapun penelitian terdahulu terkait kinerja Adaboost dalam SVM; Pertama, penelitian dari Sabita dan Trisnawati tahun 2023. Dengan studi kasus memprediksi waktu kelulusan mahasiswa didapatkan hasil metode *Adaboost* mempunyai akurasi yang lebih baik yaitu sekitar 73%. Kedua, penelitian dari Septiana et all tahun 2024 dengan studi kasus data X terhadap penerapan kurikulum merdeka, adapun hasil yang [illegible]de SVM tanpa *Adaboost* yaitu sebesar 86,88%, setelah diberi [illegible]sar 88,85% dan peneliti juga menghitung akurasi metode *Naïve* [illegible]*st* yaitu mula-mula sebesar 79,19%, dan setelah diberi Adaboost [illegible]i sebesar 83,57%. Dari penelitian diatas, disimpulkan bahwa baik [illegible]n NB mempunyai persamaan yaitu terjadi kenaikan akurasi setelah [illegible]*oost*.

Dasar peneliti menggunakan data komentar *Youtube* dibandingkan data *X* (komentar *Youtube)* yang sudah lumrah digunakan pada penelitian analisis sentimen karena peneliti mempertimbangkan beberapa poin; Pertama, data *X* meski terlepas dari opini murni masyarakat, data ini tidak mewakili masyarakat umum. (Blank, 2017) Hal ini berdasarkan tiap tiap cuitan akun mewakili masyarakat di *X* saja dan mereka hanya memutuskan berkontribusi pada utas tagarnya masing-masing (Azionya & Nhedzi, 2021). Pengguna *X* sendiri pun cenderung eksklusif jika dibandingkan dengan *Youtube*, *Instagram*, ataupun *Tiktok* yang masih lebih dianggap mewakili seluruh opini pengguna media sosial di Indonesia. Kedua, *scrapping* data X berdasarkan tagar menjadi tantangan karena sifat data X dan tagar yang terus berkembang. Biasanya pengguna menggunakan tagar berbeda pada waktu berbeda dan menggunakan beberapa tagar dalam sebuah cuitan sehingga mempersulit tugas mengidentifikasi tagar yang relevan untuk topik tertentu. Selain itu, data pengguna X banyak mengandung *noise* dengan mengkombinasikan bahasa gaul, singkatan, emotikon, *URL*, dan kata-kata ambigu. Hal tersebut berdampak pada proses ekstraksi informasi yang harusnya relevan menjadi lebih sulit. (Gupta & Hewett, 2020)

Berdasarkan penjabaran tersebut, peneliti tertarik dalam melakukan penelitian lebih lanjut terkait penerapan metode SVM sebelum dan setelah diberi algoritma *Adaboost* untuk mengetahui netralitas masyarakat mengenai arah dukungan Capres-Cawapres di pemilu 2024 menggunakan data komentar Youtube, serta memperoleh hasil kinerja evaluasi terbaik dari klasifikasi yang dilakukan sebelum dan setelah optimasi dari *Adaboost*.

1.2. Batasan Penelitian

Data komentar Youtube yang digunakan terbatas kepada data komentar masyarakat mengenai debat ke-5 PEMILU Presiden dan Wakil Presiden tahun 2024, data dikategorikan berdasarkan dukungan kepada salah satu Capres ataupun masih bersikap netral. Data hanya diambil pada rentang Februari-Maret 2024. Berikut adalah beberapa parameter yang digunakan dalam analisis.

1. Jumlah maksimal fitur yang digunakan 8.000-12.000 fitur, pada TF-IDF.
2. Nilai k yang ditetapkan pada k-*Cross Validation* sebesar 5.
3. Kernel SVM yang ditetapkan yaitu RBF, untuk nilai parameter C dan nilai parameter γ akan dilakukan *trial* dan *error* pada skala 0.1 hingga 1
4. Iterasi maksimal dibatasi hingga 1-10 iterasi pembobotan pada *Adaboost*.

1.3. Tujuan Penelitian

Tujuan dari penelitian ini untuk mendapatkan hasil klasifikasi beserta evaluasi perbandingan hasil kinerja klasifikasi metode SVM sebelum dan setelah diberi pada data netralitas masyarakat terkait arah dukungan Capres-024.

ian

ian ini untuk mempelajari penggunaan SVM sebagai algoritma dan *Adaboost* sebagai fitur tambahan untuk meningkatkan kinerja data pengujian. Metode ini dapat dijadikan prosedur dalam

menganalisa pandangan atau sentimen masyarakat terhadap suatu kejadian/isu yang sedang menjadi perhatian. Selain itu, penelitian ini dapat terus dikembangkan atau dijadikan rujukan untuk penelitian selanjutnya.

1.5. Teori

1.5.1. *Natural Language Processing*

Natural Language Processing (NLP) merupakan penerapan ilmu komputer yang bertujuan untuk mempelajari bagaimana komputer dapat berinteraksi dengan bahasa alami manusia. NLP berusaha untuk mengatasi tantangan dalam memahami bahasa manusia, termasuk aturan tata bahasa dan semantiknya, dengan tujuan mengonversi bahasa tersebut menjadi representasi formal yang dapat dipahami dan diproses oleh komputer ((Pustejovsky & Stubbs, 2012)).

Gambar 1. Pengaplikasian *Natural Language Processing*.

Menurut (Pustejovsky & Stubbs, 2012), beberapa bidang populer dalam penerapan NLP adalah sebagai berikut:

1. *Information retrieval*, yaitu etode pencarian dokumen yang relevan, pencarian informasi spesifik di dalam dokumen, serta pembuatan metadata
2. *Automatic Summarization*, yaitu pembuatan versi singkat yang berisi poin-poin penting dari suatu dokumen dengan program komputer.
3. *Language Translation*, yaitu sistem penerjemahan secara otomatis menggunakan algoritma sedimikian sehingga suatu bahasa dapat an ke bahasa lain. Contoh pengaplikasian: *Google Translate, Bing* ll.
   awab pertanyaan, mesin secara otomatis menjawab pertanyaan an dengan bahasa dan jawaban yang dijawab dengan bahasa sia. Contoh pengaplikasian: *Chatbot*, dll.

5. *Optical Character Recognition* (OCR), yaitu pengubahan tulisan tangan menjadi tulisan yang telah tersalin ke dalam format teks dokumen pada komputer.
6. Analisis Sentimen, dengan melakukan ekstraksi informasi dari sumber data teks untuk mendeteksi pandangan positif ataupun negative terhadap suatu objek. Biasanya diterapkan untuk mengidentifikasi tren opini public terhadap suatu produk atau perusahaan.

1.5.2. Analisis Sentimen

Analisis sentimen, juga dikenal sebagai *opinion mining*, adalah gabungan antara *data mining* dan *text mining* yang bertujuan untuk mengevaluasi pendapat, sentimen, evaluasi, sikap, penilaian, perasaan, dan emosi seseorang terhadap suatu topik, produk, layanan, organisasi, individu, atau kegiatan tertentu. Melalui analisis sentimen, kita dapat menentukan apakah isi teks tersebut bersifat positif atau negatif. Sentimen ini terkait dengan fokus topik tertentu, pernyataan mengenai topik tertentu dapat memiliki makna yang berbeda tergantung pada subjeknya. Oleh karena itu, dalam beberapa penelitian, langkah awal adalah menentukan elemen-elemen produk yang sedang dibicarakan sebelum melakukan analisis sentimen (Sipayung et al., 2016). Berdasarkan pembahasan pada subbab 1.5.1., analisis sentimen termasuk salah satu dari cabang pengaplikasian *Nature Language Processing* (NLP).

Ada beberapa jenis analisis sentimen, termasuk *Intent Sentiment Analysis* yang bertujuan untuk mengidentifikasi dan memahami motivasi di balik pesan pengguna, seperti keluhan, saran, atau pendapat. Jenis lainnya adalah *Aspect-Based Sentiment Analysis* yang memfokuskan pada elemen-elemen spesifik dari produk dan layanan. Kemudian, terdapat juga *Fine-Grained Sentiment Analysis*, jenis yang paling umum, yang berfokus pada tingkat polaritas pendapat. Prinsip dasar dari teknik analisis sentimen ini adalah untuk mengelompokkan teks, kalimat, atau dokumen ke dalam sentimen atau opini yang positif, negatif, atau netral (Sanjaya & Lhaksmana, 2020).

1.5.3. Pembelajaran Mesin

Pembelajaran mesin adalah jenis sistem kecerdasan buatan yang memungkinkan komputer belajar tanpa diprogram secara eksplisit (Muller dan Guido, 2016). Pembelajaran mesin dapat dibandingkan dengan pembelajaran manusia, ibarat manusia yang belajar melalui banyak pengalaman. Oleh karena itu, pembelajaran mesin dapat digambarkan sebagai sebuah teknologi yang mempelajari cara mengambil keputusan berdasarkan berbagai data yang ada (Suyanto, 2017). Dalam kehidupan sehari-hari, manusia dapat dengan mudah mengenali suatu benda, namun benda tersebut tidak selalu digambarkan secara spesifik. Oleh karena itu, diperlukan untuk mendeteksi, mengidentifikasi, atau memprediksi data pelajari data sebelumnya (Nurhayati & Iswara, 2019).

jelasan dari (Müller & Guido, 2016), ada tiga jenis model metode pembelajaran mesin:

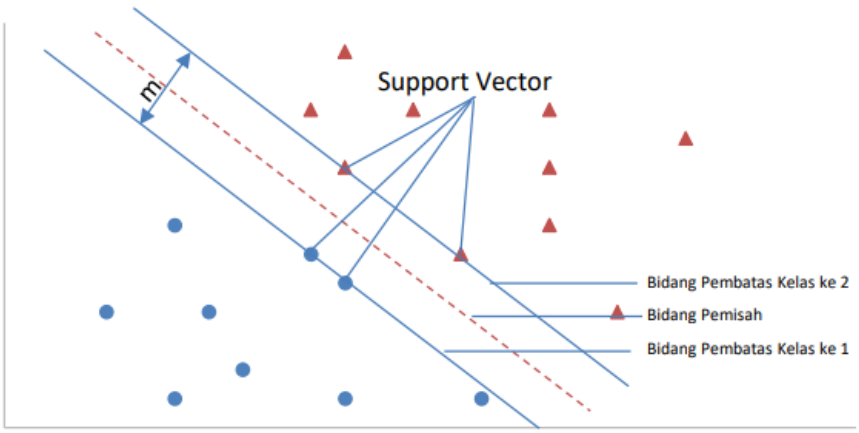
1. *Supervised Learning*, adalah kategori pembelajaran mesin yang memeriksa kumpulan atau kelas data berlabel untuk mencapai klasifikasi data yang ada. Pembelajaran membuat model prediktif dari data yang sudah diberi label.
2. *Unsupervised learning*, adalah kategori pembelajaran mesin yang mempelajari serangkaian kinerja tanpa menggunakan label atau kelas. Oleh karena itu, ia mencari struktur data yang ada dan melakukan pengelompokan berdasarkan informasi yang ada.
3. *Reinforcement learning*, adalah kategori pembelajaran mesin yang mempelajari apa yang harus dilakukan dengan memetakan situasi ke tindakan untuk mendapatkan imbalan *(reward),* atau sebaliknya berupa hukuman *(punishment).*

1.5.4. *Support Vector Machine* (SVM)

Support Vector Machine (SVM) adalah metode yang mempelajari area yang memisahkan antar kategori dalam sebuah observasi. Formulasi optimasi mesin vektor pendukung untuk masalah klasifikasi dibedakan menjadi dua kelas yaitu klasifikasi *linear* dan *non-linear.*

1.5.4.1 Klasifikasi pada *Linearly Separable Data*

SVM pada *linearly separable data* adalah penerapan metode SVM pada data yang dapat dipisahkan secara linier. Misalkan $x_i = \{x_1, x_2, \ldots, x_n\}$ adalah titik data dan $y_i \in \{-1, +1\}$ adalah label kategori atau kelas data untuk *dataset*. Penggambaran *linearly separable data* dapat dilihat pada Gambar 2.



Gambar 2. Model SVM Linear

, kedua kelas data dapat dipisahkan oleh sepasang bidang
ıjar (linier). Data yang berada pada bidang pembatas disebut
or. Persamaan *hyperplane* dapat ditulis seperti pada persamaan

$$f(x) = (\boldsymbol{w}^T \boldsymbol{x}) + b \quad (1)$$

$$f(x) = w_1x_1 + w_2x_2 + \cdots + w_Nx_N + b$$

Dengan $\boldsymbol{w}$ merupakan vektor normal terhadap *hyperplane* yang akan menentukan orientasi dari *hyperplane,* b merupakan suatu konstanta skalar yang menentukan lokasi fungsi *hyperplane* terhadap titik asal (Sari, 2017). Oleh karena itu *hyperplane* pemisah pada kasus ini dapat dituliskan seperti pada model dibawah.

$$(\boldsymbol{w^T x}) + b = \begin{cases} +1, & f(x) = Positif \\ 0, & f(x) = Netral \\ -1, & f(x) = Negatif \end{cases}$$

Pada kasus ini akan dipisahkan 2 *hyperplane* yang sejajar dengan *hyperplane* pertama akan membatasi kelas 1 sedangkan *hyperplane* kedua akan membatasi kelas 2, sehingga dapat dibentuk pertidaksamaan model matematika seperti pada Persamaan (2) dan Persamaan (3).

$$\boldsymbol{w^T x_i} + b \geq 1, \forall \in Kelas\ Positif \tag{2}$$

$$\boldsymbol{w^T x_i} + b \leq -1, \forall \in Kelas\ Negatif \tag{3}$$

Berdasarkan Persamaan (2) dan (3) maka dapat diperoleh nilai margin (jarak) di antara 2 *hyperplane* dengan menggunakan prinsip jarak antara 2 garis sejajar pada Persamaan (4).

$$margin(m) = \frac{|(b-1) - (b+1)|}{\sqrt{\boldsymbol{w^T w}}} \tag{4}$$

karena $\boldsymbol{w^T w} = \|\boldsymbol{w}\|^2$, maka dapat pula dituliskan dengan persamaan (5).

$$margin(m) = \frac{2}{\|\boldsymbol{w}\|} \tag{5}$$

Untuk mendapatkan dua *hyperplane* pemisah dengan jarak sejauh mungkin, maka nilai dari *hyperplane* optimal dapat diperoleh pada persamaan (6) melalui suatu solusi optimasi fungsi tujuan.

$$max \frac{2}{\|\boldsymbol{w}\|} \tag{6}$$

dengan kendala

$$y_i(\boldsymbol{w x_i} + b) \geq 1, \quad i = 1,2,3, \ldots, N \tag{7}$$

(2000) jika $0 \leq a \leq b$ maka $a \leq b \leftrightarrow a^2 \leq b^2 \leftrightarrow \sqrt{a} \leq \sqrt{b}$ sehingga nilai maksimum dari Persamaan (7) nilai $\|\boldsymbol{w}\|$ harus minimum, n fungsi tujuan sebagai berikut:

$$f(w) \coloneqq \frac{1}{2}\|\boldsymbol{w}\|^2 \tag{8}$$

dengan kendala:

$$\mathrm{g(w,b)} \coloneqq \mathrm{y_i}(w\mathbf{x_i} + \mathrm{b}) - 1 = 0, \qquad \mathrm{i} = 1,2,3,\dots,\mathrm{N} \tag{9}$$

Untuk mendapatkan solusi dari permasalahan optimasi bentuk primal pada Persamaan (8) maka akan digunakan metode *Lagrange* dengan kendala pertidaksamaan atau dikenal sebagai *Karush Kuhn Tucker* (KKT). Berdasarkan Persamaan (8) maka diperoleh persamaan *lagrange* pada persamaan (10) untuk kasus ini.

$$L_p(\boldsymbol{w}, b, \alpha) = f(\boldsymbol{w}) + \alpha_i\, g(\boldsymbol{w}, b)$$

$$\mathrm{L_p}(\boldsymbol{w}, b, \alpha) = \left(\frac{1}{2}\|\boldsymbol{w}\|^2\right) - \sum_{\mathrm{i=1}}^{\mathrm{n}} \alpha_\mathrm{i}\,[\mathrm{y_i}(\boldsymbol{w}\mathbf{x_i} + \mathrm{b}) - 1] \tag{10}$$

Peubah α_i merupakan *Lagrange Multiplier* atau pengganda *Lagrange*. Nilai dari *Lagrange Multiplier* ini adalah $\alpha_i > 0$. Syarat perlu agar Persamaan (10) minimum adalah turunan pertama dari Persamaan (10) untuk setiap peubah adalah 0.

$$\frac{\partial L_p}{\partial \boldsymbol{w}} = 0, \text{dan} \quad \frac{\partial L_p}{\partial b} = 0$$

Kondisi 1

$$\begin{aligned}
\frac{\partial \mathrm{L_p}}{\partial \mathbf{w}} &= \left(\frac{1}{2}\|\boldsymbol{w}\|^2\right) - \sum_{i=1}^{n} \alpha_i\,[y_i \boldsymbol{w}\boldsymbol{x_i} + y_i b - 1] \\
&= \left(\frac{1}{2}\|\boldsymbol{w}\|^2\right) - \sum_{i=1}^{n} \alpha_i\,[y_i \boldsymbol{w}\boldsymbol{x_i} + y_i b - 1] \\
&= \left(\frac{1}{2}\|\boldsymbol{w}\|^2\right) - \sum_{i=1}^{n} [\alpha_i y_i \boldsymbol{w}\boldsymbol{x_i} + \alpha_i y_i b - 1] \\
&= \boldsymbol{w} - \sum_{\mathrm{i=1}}^{\mathrm{n}} [\alpha_\mathrm{i} \mathrm{y_i} \mathbf{x_i} + 0 - 0] \\
&= \boldsymbol{w} - \sum_{\mathrm{i=1}}^{\mathrm{n}} \alpha_\mathrm{i}\, \mathrm{y_i} \mathbf{x_i} \\
0 &= \boldsymbol{w} - \sum_{\mathrm{i=1}}^{\mathrm{n}} \alpha_\mathrm{i}\, \mathrm{y_i} \mathbf{x_i}
\end{aligned} \tag{11}$$

$$\frac{\partial \mathrm{L_p}}{\partial b} = \left(\frac{1}{2}\|\boldsymbol{w}\|^2\right) - \sum_{i=1}^{n} \alpha_i \left[y_i \boldsymbol{w}\boldsymbol{x_i} + y_i b - 1\right]$$

$$= \left(\frac{1}{2}\|\boldsymbol{w}\|^2\right) - \sum_{i=1}^{n} \alpha_i \left[y_i \boldsymbol{w}\boldsymbol{x_i} + y_i b - 1\right]$$

$$= \left(\frac{1}{2}\|\boldsymbol{w}\|^2\right) - \sum_{i=1}^{n} \left[\alpha_i y_i \boldsymbol{w}\boldsymbol{x_i} + \alpha_i y_i b - 1\right]$$

$$= 0 - \sum_{\mathrm{i}=1}^{\mathrm{n}} \left[0 + \alpha_i \mathrm{y_i} - 0\right]$$

$$= \sum_{\mathrm{i}=1}^{\mathrm{n}} \alpha_{\mathrm{i}}\, \mathrm{y_i}$$

$$0 = \sum_{\mathrm{i}=1}^{\mathrm{n}} \alpha_{\mathrm{i}}\, \mathrm{y_i}$$

$$b = 0 \qquad (12)$$

Tampak bahwa fungsi tujuan (8) mengandung fungsi kuadrat pada peubah $\boldsymbol{w}$, sehingga hal tersebut akan mengakibatkan cukup sulitnya penyelesaian secara komputasi dan akan memakan waktu yang panjang. Dengan demikian, maka permasalahan tersebut akan lebih mudah dan lebih efisien jika diselesaikan dalam bentuk dual

Menurut teorema dualitas Kecman (2005) jika problem primal memiliki solusi optimal, maka problem dual juga akan mempunyai solusi optimal yang nilainya sama. Untuk memperoleh bentuk dual dari Persamaan (8), maka akan disubtitusikan Persamaan (11) dan (12) ke dalam Persamaan (10) sebagai berikut:

$$= \frac{1}{2}\|\boldsymbol{w}\|^2 - \sum_{i=1}^{n} \alpha_i \left[y_i(w x_i + b) - 1\right]$$

$$= \frac{1}{2}\left\|\sum_{\mathrm{i}=1}^{\mathrm{n}} \alpha_{\mathrm{i}}\, \mathrm{y_i}\mathbf{x_i}\right\|^2 - \sum_{i=1}^{n} \alpha_i \left[y_i \left(\left(\sum_{\mathrm{i}=1}^{\mathrm{n}} \alpha_{\mathrm{i}}\, \mathrm{y_i}\mathbf{x_i}\right) x_i + b\right) - 1\right]$$

$$= \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \boldsymbol{x_i^T}\boldsymbol{x_j} - \sum_{i=1}^{n} \alpha_i y_i \left(\sum_{j=1}^{n} \alpha_j y_j \boldsymbol{x}_j\, \mathbf{x_i}\right) - b\sum_{i=1}^{n} \alpha_i\, y_i + \sum_{i=1}^{n} \alpha_i$$

$$= \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \boldsymbol{x_i^T}\boldsymbol{x_j} - \sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \boldsymbol{x_i^T}\boldsymbol{x_j} - 0 + \sum_{i=1}^{n} \alpha_i$$

$$= \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j (\boldsymbol{x}_i \boldsymbol{x}_j)$$

Menurut teori dualitas meminimumkan ($\boldsymbol{w}$) sama dengan memaksimumkan (α), sehingga masalah pencarian *hyperplane* yang optimal pada kasus *linear separable* dapat dirumuskan:

fungsi tujuan:

$$\max_{\alpha} Lp(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j (\boldsymbol{x}_i \boldsymbol{x}_j) \tag{13}$$

dengan kendala:

$$\sum_{i=1}^{n} \alpha_i y_i = 0, \qquad \forall_i \; dengan \; \alpha_i > 0$$

$$y_i = \begin{cases} 1 \; , & x_i > 0 \\ -1 \; , & x_i < 0 \end{cases}$$

Dengan demikian, nilai α_i dapat ditemukan dengan menyelesaikan kasus optimasi pada (13) dan nilai $\boldsymbol{w}$ akan didapatkan dengan mensubstitusikan α_i pada Persamaan (11). Selanjutnya fungsi *hyperplane* yang optimal pada kasus *linear separable* dapat terbentuk persamaan berikut:

$$f(x_d) = \sum_{i=1}^{n} \alpha_i y_i (x_i x_d) + b \tag{14}$$

dengan x_d merupakan data yang akan diklasifikasikan, α_i merupakan solusi optimasi dari masalah optimasi (13) dan b dicari dengan formula:

$$b = -\frac{1}{2}(\boldsymbol{w}\boldsymbol{x}^{+} + \boldsymbol{w}\boldsymbol{x}^{-}) \tag{15}$$

Dengan demikian, kelas berdasarkan *hyperplane* optimal pada kasus dataset yang *linear separable* terbentuk sebagai berikut:

$$f(x) = (\boldsymbol{w}\boldsymbol{x}) + \boldsymbol{b} \tag{16}$$

1.5.4.2. Klasifikasi pada *Non-Linearly Separable Data* dengan Metode Kernel

Klasifikasi data yang tidak dapat dipisahkan secara linier memerlukan modifikasi pada formula SVM agar dapat menemukan solusinya. Dalam praktiknya, jarang ditemui data

sahkan secara linier. Modifikasi formula tersebut dilakukan pada *lane*. Agar lebih fleksibel maka kedua *hyperplane* diberi tambahan ngan $\xi_i \geq 0, i = 1,2,\ldots,n$, sehingga diperoleh suatu modifikasi ı persamaan berikut:

$$y_i((\boldsymbol{w}\mathbf{x_i}) + b) + \xi_i \geq 1$$
$$y_i((\boldsymbol{w}\mathbf{x_i}) + b) \geq 1 - \xi_i \quad (17)$$

atau dapat dituliskan:

$$\boldsymbol{w}\mathbf{x_i} + b \geq 1 - \xi_i \,, \forall \in \text{Kelas Positif}$$
$$\boldsymbol{w}\mathbf{x_i} + b \leq -1 + \xi_i \,, \forall \in \text{Kelas Negatif} \quad (18)$$

Pencarian *hyperplane* optimal dengan tambahan peubah *slack* ini sering disebut sebagai *soft margin hyperplane.* Selain itu, pada modifikasi formula untuk data yang tidak dapat dipisahkan secara linier juga memerlukan tambahan parameter *penalty* C (Sari, 2017). Sehingga formula untuk mendapatkan *hyperplane* optimal menjadi:

fungsi tujuan:

$$\min \left(\frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^{n} \xi_i \right) \quad (19)$$

kendala:

$$y_i((\boldsymbol{w}\boldsymbol{x_i}) + b) \geq 1 - \xi_i \,, i = 1,2,3 \ldots, n$$

dengan C merupakan parameter yang menetukan besarnya penalti akibat kesalahan dalam klasifikasi data yang ditentukan oleh pengguna. Semakin tinggi nilai C, maka kemungkinan terjadinya kesalahan dalam penentuan solusi akan semakin kecil. Sebaliknya, jika nilai C semakin rendah maka semakin tinggi proporsi kesalahan yang terjadi pada penentuan solusi.

Berdasarkan Persamaan (19), akan dimaksimalkan nilai margin antara 2 kelas dengan meminimalkan $\|\boldsymbol{w}\|^2$. Dalam formula ini, akan dicoba meminimalkan kesalahan yang dinyatakan oleh peubah ξ_i. Penggunaan peubah *slack* bertujuan untuk mengatasi kasus ketidaklayakan (*infeasibility*) dari kendala tersebut. Untuk meminimalkan nilai dari peubah *slack* maka digunakan nilai dari parameter C (Sari, 2017).

Dengan menggunakan teori optimasi, maka didapatkan:

fungsi tujuan:

$$f(\mathbf{w}, \xi) := \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^{n} \xi_i \quad (20)$$

$$) := y_i \, (\boldsymbol{w}\boldsymbol{x_i} + b) \geq 1 - \xi_i \,, \xi_i \geq 0, i = 1,2, \ldots, n$$

Untuk mendapatkan solusi dari permasalahan optimasi bentuk primal pada Persamaan (20), maka seperti pada kasus linear separable akan digunakan KTT. Namun pada kasus *non-linear separable* ini akan memiliki dua pengali *Lagrange* untuk kasus ini menjadi:

$$\begin{aligned} L_p(\boldsymbol{w}, b, \xi, \alpha, \beta) &= f(\boldsymbol{primal}) + \sum_{i=1}^{n} \alpha_i \, [kendala_1] + \sum_{i=1}^{n} \beta_i \, (kendala_2) \\ L_p(\boldsymbol{w}, b, \xi, \alpha, \beta) &= f(\boldsymbol{w}, \xi) + \sum_{i=1}^{n} \alpha_i \, [1 - \xi_i - y_i(\boldsymbol{w}\boldsymbol{x_i} + b)] + \sum_{i=1}^{n} \beta_i \, (-\xi_i) \\ &= \left(\frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n} \xi_i\right) - \sum_{i=1}^{n} \alpha_i \, [y_i(\boldsymbol{w}\boldsymbol{x_i} + b) - 1 + \xi_i] \\ &\quad - \sum_{i=1}^{n} \beta_i \, (\xi_i) \end{aligned} \tag{21}$$

Peubah α_i dan β_i merupakan *Lagrange Multiplier* dengan $\alpha_i \geq 0$ dan $\beta_i \geq 0$. Peubah α_i dan β_i dapat disebut sebagai peubah non-negatif. Fungsi Persamaan (21) akan diminimalkan terhadap peubah $\boldsymbol{w}$, b dan ξ serta harus dimaksimalkan terhadap peubah α dan β (Sari, 2017). Berikut turunan pertama dari fungsi Persamaan (21) yaitu:

Kondisi 1

$$\begin{aligned} \frac{\partial L_p}{\partial w} &= 0 \\ &= f(\boldsymbol{w}, \xi) - \sum_{i=1}^{n} \alpha_i \, [y_i(\boldsymbol{w}\boldsymbol{x_i} + b) - 1 + \xi_i] - \sum_{i=1}^{n} \beta_i(\xi_i) \\ &= \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n} \xi_i - \sum_{i=1}^{n} \alpha_i[y_i(\boldsymbol{w}\boldsymbol{x_i} + b) - 1 + \xi_i] - \sum_{i=1}^{n} \beta_i(\xi_i) \\ &= \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n} \xi_i - \sum_{i=1}^{n} \alpha_i[y_i\boldsymbol{w}\boldsymbol{x_i} + y_i b - 1 + \xi_i] - \sum_{i=1}^{n} \beta_i(\xi_i) \\ &= \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n} \xi_i - \sum_{i=1}^{n} \alpha_i y_i \boldsymbol{w}\boldsymbol{x_i} - b\sum_{i=1}^{n} \alpha_i y_i + \sum_{i=1}^{n} \alpha_i \\ &\quad - \sum_{i=1}^{n} \alpha_i \xi_i - \sum_{i=1}^{n} \beta_i \, (\xi_i) \\ &= \boldsymbol{w} + 0 - \sum_{i=1}^{n} \alpha_i y_i \boldsymbol{x_i} - 0 + 0 - 0 - 0 \\ \boldsymbol{w} &= \sum^{n} \alpha_i y_i \mathbf{x}_i \end{aligned} \tag{22}$$

$$= f(\boldsymbol{w},\xi) - \sum_{i=1}^{n} \alpha_i \left[y_i(\boldsymbol{wx_i} + b) - 1 + \xi_i\right] - \sum_{i=1}^{n} \beta_i(\xi_i)$$

$$= \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n} \xi_i - \sum_{i=1}^{n} \alpha_i[y_i(\boldsymbol{wx_i} + b) - 1 + \xi_i] - \sum_{i=1}^{n} \beta_i(\xi_i)$$

$$= \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n} \xi_i - \sum_{i=1}^{n} \alpha_i[y_i\boldsymbol{wx_i} + y_i b - 1 + \xi_i] - \sum_{i=1}^{n} \beta_i(\xi_i)$$

$$= \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n} \xi_i - \sum_{i=1}^{n} \alpha_i y_i \boldsymbol{wx_i} - b\sum_{i=1}^{n} \alpha_i y_i + \sum_{i=1}^{n} \alpha_i - \sum_{i=1}^{n} \alpha_i \xi_i - \sum_{i=1}^{n} \beta_i(\xi_i)$$

$$= 0 + 0 - 0 - \sum_{i=1}^{n} \alpha_i y_i + 0 - 0 - 0$$

$$0 = \sum_{i=1}^{n} \alpha_i y_i \qquad (23)$$

Kondisi 3

$$\frac{\partial L_p}{\partial \xi} = 0$$

$$= f(\boldsymbol{w},\xi) - \sum_{i=1}^{n} \alpha_i \left[y_i(\boldsymbol{wx_i} + b) - 1 + \xi_i\right] - \sum_{i=1}^{n} \beta_i(\xi_i)$$

$$= \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n} \xi_i - \sum_{i=1}^{n} \alpha_i[y_i(\boldsymbol{wx_i} + b) - 1 + \xi_i] - \sum_{i=1}^{n} \beta_i(\xi_i)$$

$$= \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n} \xi_i - \sum_{i=1}^{n} \alpha_i[y_i\boldsymbol{wx_i} + y_i b - 1 + \xi_i] - \sum_{i=1}^{n} \beta_i(\xi_i)$$

$$= \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n} \xi_i - \sum_{i=1}^{n} \alpha_i y_i \boldsymbol{wx_i} - b\sum_{i=1}^{n} \alpha_i y_i + \sum_{i=1}^{n} \alpha_i - \sum_{i=1}^{n} \alpha_i \xi_i - \sum_{i=1}^{n} \beta_i(\xi_i)$$

$$= 0 + C - 0 - 0 + 0 - \alpha_i - \beta_i$$

$$C = \alpha_i + \beta_i \qquad (24)$$

Untuk memperoleh bentuk dual, maka akan disubtitusikan Persamaan (22), (23) dan naan (21).

$$C\sum_{i=1}^{n} \xi_i - \sum_{i=1}^{n} \alpha_i \left[y_i(\boldsymbol{wx_i} + b) - 1 + \xi_i\right] - \sum_{i=1}^{n} \beta_i(\xi_i)$$

$$= \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\xi_i - \sum_{i=1}^{n}\alpha_i y_i \boldsymbol{w}\boldsymbol{x_i} - b\sum_{i=1}^{n}\alpha_i y_i + \sum_{i=1}^{n}\alpha_i - \sum_{i=1}^{n}\alpha_i\xi_i \sum_{i=1}^{n}\beta_i\,(\xi_i)$$

$$= \frac{1}{2}\|\mathbf{w}\|^2 + \sum_{i=1}^{n}(C-\alpha_i)\xi_i - \sum_{i=1}^{n}\alpha_i y_i \boldsymbol{w}\boldsymbol{x_i} + \sum_{i=1}^{n}\alpha_i - \sum_{i=1}^{n}(C-\alpha_i)\,\xi_i$$

$$= \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{n}\alpha_i y_i \boldsymbol{w}\boldsymbol{x_i} + \sum_{i=1}^{n}\alpha_i$$

$$= \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j \boldsymbol{x_i^T x_j} - \sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j \boldsymbol{x_i^T x_j} + \sum_{i=1}^{n}\alpha_i$$

$$= \sum_{\mathrm{i=1}}^{\mathrm{n}}\alpha_{\mathrm{i}} - \frac{1}{2}\sum_{\mathrm{i=1}}^{\mathrm{n}}\sum_{\mathrm{j=1}}^{\mathrm{n}}\alpha_{\mathrm{i}}\alpha_{\mathrm{j}}\mathrm{y_i y_j}\boldsymbol{x_i^T x_j} \qquad (25)$$

Menurut teori dualitas, meminimumkan $L_p(\boldsymbol{w})$ sama dengan memaksimumkan $L_p(\alpha)$, sehingga masalah pencarian *hyperplane* yang optimal pada kasus *non-linear separable* dapat dirumuskan dengan:

fungsi tujuan:

$$\max_{\alpha} L_p(\alpha) = \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j \boldsymbol{x_i^T x_j}$$

$$\max_{\alpha} L_p(\alpha) = \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j K(x_i x_j) \qquad (26)$$

dengan kendala:

$$\textstyle\sum_{i=1}^{n}\alpha_i y_i = 0 \text{ dan } \beta_i\xi_i = (C-\alpha_i)\xi_i = 0\ \forall_i \text{ dengan } \alpha_i > 0 \qquad (27)$$

Selanjutnya fungsi *hyperplane* yang optimal pada kasus *non-linear* hampir sama dengan kasus *linear separable* yaitu sebagai berikut:

$$\mathrm{f(x_d)} = \sum_{\mathrm{i}\in S}\alpha_{\mathrm{i}}\mathrm{y_i}\mathbf{x_i^T x} + \mathrm{b} \qquad (28)$$

S merupakan data yang akan diklasifikasikan, α_i merupakan solusi optimal dari dan $\boldsymbol{b}$ dicari dengan formula. S adalah himpunan indeks *support* tidak nol untuk *support vector* maka penjumlahan di (28) untuk *support vector*. Untuk α_i tak berhingga, maka persamaan nemuaskan.

$$b = y_i - \boldsymbol{w}\boldsymbol{x_i} \tag{29}$$

Untuk memastikan ketepatan perhitungan, kita menghitung rata-rata bias (b) yang dihitung untuk *support vector* tak berhingga sebagai berikut:

$$b = \frac{1}{|U|} \sum_{i \in U} (y_i - \boldsymbol{w}\boldsymbol{x_i}) \tag{30}$$

U adalah himpunan indeks *support vector* tak berhingga. Dengan demikian, kelas berdasarkan *hyperplane* optimal pada kasus dataset yang *non-linear separable* terbentuk sebagai berikut:

$$f(x) = \sum_{i=1}^{n} \alpha_i y_i \, K(x, x_i) + b \tag{31}$$

Data pelatihan untuk kasus *non-separable*, klasifikasi yang diperoleh mungkin tidak memiliki kemampuan generalisasi yang tinggi meskipun *hyperplane* ditentukan secara optimal. Sehingga diatasi dengan cara *input space* dipetakan ke dalam *dot-product space* berdimensi tinggi yang disebut *feature space*. Fungsi kernel merupakan suatu fungsi yang memetakan data ke ruang dimensi yang lebih tinggi dengan harapan data akan memiliki struktur yang lebih baik sehingga lebih mudah dipisahkan. Menggunakan vektor fungsi *non-linear* sebagai $\phi(x_i): R^n \rightarrow R^{n_k}$ memetakan ruang input ke ruang dimensi yang lebih tinggi. Dalam fungsi *non-linear* pengklasifikasian diperoleh dengan:

$$f(x) = sign\,[\boldsymbol{w}\,\phi(x_i) + b] \tag{32}$$

dengan menggunakan teori optimasi, maka didapatkan:

fungsi tujuan:

$$f(\boldsymbol{w}, \xi) \coloneqq \frac{1}{2} \|\mathbf{w}\|^2 + \frac{1}{2} C \sum_{i=1}^{n} \xi_i^2 \tag{33}$$

dengan kendala:

$$g(\boldsymbol{w}, b, \xi) \coloneqq y_i \, (\boldsymbol{w}\,\phi(x_i) + b) = 1 - \xi_i^2, \qquad \xi i \geq 0, i = 1,2, \ldots, n$$

Untuk menurunkan permasalahan (24), (25) dan (28) digunakan *Lagrange Multiplier* dengan nilai $\alpha_i \geq 0$. Optimal poin akan ada di dalam *saddle point* dari *Lagrange function* menjadi:

$$\xi, \alpha, \beta) = f(\boldsymbol{w}, \xi) + \sum_{i=1}^{n} \alpha_i \, [y_i(\boldsymbol{w}\,\phi(x_i) + b) - 1 + \xi_i] \tag{34}$$

urunan pertama yaitu sebagai berikut:

$$\frac{\partial L_p}{\partial b} = 0$$

$$\frac{\partial L_p}{\partial b} = \frac{\partial L_p(\boldsymbol{w},b,\xi,\alpha,\beta)}{\partial b}$$

$$= \frac{\partial\{f(\boldsymbol{w},\xi) - \sum_{i=1}^{n}\alpha_i[y_i(\boldsymbol{w}\phi(x_i)+b)-1+\xi_i]\}}{\partial b}$$

$$= \frac{\partial\left\{\frac{1}{2}\|\mathbf{w}\|^2+\frac{1}{2}C\sum_{i=1}^{n}\xi_i^2-\sum_{i=1}^{n}\alpha_i[y_i(\boldsymbol{w}\phi(x_i)+b)-1+\xi_i]\right\}}{\partial b}$$

$$= \frac{\partial\left\{\frac{1}{2}\|\mathbf{w}\|^2+\frac{1}{2}C\sum_{i=1}^{n}\xi_i^{\,2}-\sum_{i=1}^{n}\alpha_i y_i \boldsymbol{w}\phi(x_i)-b\sum_{i=1}^{n}\alpha_i y_i+\sum_{i=1}^{n}\alpha_i-\sum_{i=1}^{n}\alpha_i\xi_i\right\}}{\partial b}$$

$$= \sum_{i=1}^{n}\alpha_i y_i$$

$$\sum_{i=1}^{n}\alpha_i y_i = 0 \tag{35}$$

Kondisi 2

$$\frac{\partial L_p}{\partial w} = 0$$

$$\frac{\partial L_p}{\partial w} = \frac{\partial L_p(\boldsymbol{w},b,\xi,\alpha,\beta)}{\partial w}$$

$$= \frac{\partial\{f(\boldsymbol{w},\xi) - \sum_{i=1}^{n}\alpha_i[y_i(\boldsymbol{w}\phi(x_i)+b)-1+\xi_i]\}}{\partial w}$$

$$= \frac{\partial\left\{\frac{1}{2}\|\mathbf{w}\|^2+\frac{1}{2}C\sum_{i=1}^{n}\xi_i^2-\sum_{i=1}^{n}\alpha_i[y_i(\boldsymbol{w}\phi(x_i)+b)-1+\xi_i]\right\}}{\partial w}$$

$$= \frac{\partial\left\{\frac{1}{2}\|\mathbf{w}\|^2+\frac{1}{2}C\sum_{i=1}^{n}\xi_i^2-\sum_{i=1}^{n}\alpha_i y_i \boldsymbol{w}\phi(x_i)-b\sum_{i=1}^{n}\alpha_i y_i+\sum_{i=1}^{n}\alpha_i-\sum_{i=1}^{n}\alpha_i\xi_i\right\}}{\partial w}$$

$$= \boldsymbol{w} - \sum_{i=1}^{n}\alpha_i y_i \phi(x_i)$$

$$\boldsymbol{w} = \sum_{i=1}^{n}\alpha_i\, y_i \phi(x_i) \tag{36}$$

Dengan demikian, kelas berdasarkan *hyperplane* optimal pada kasus dataset yang *non-linear separable* terbentuk sebagai berikut:

$$\mathrm{f(x_d)} = \sum_{\mathrm{i=1}}^{\mathrm{n}} \alpha_\mathrm{i} \mathrm{y_i}\, \mathrm{K(x_i, x_d)} + \mathrm{b} \tag{37}$$

$$b = y_k - \frac{1}{2}\sum_{i=1}^{n}\alpha_i y_i\, K(x_i x_k) \tag{38}$$

K merupakan salah satu fungsi Kernel yang akan digunakan. Fungsi Kernel digunakan untuk memetakan data non linier menjadi linier. Berikut ini adalah beberapa fungsi kernel yang umum digunakan. (Hsu, 2010) dan (Müller & Guido, 2016).

1. Linear

$$K\left(x_i, x_j\right) = x_i x_j \tag{39}$$

2. Polynomial

$$K\left(x_i, x_j\right) = \left(1 + x_i x_j\right)^{\mathrm{d}} \tag{40}$$

3. RBF

$$\mathrm{K}(\mathrm{x_i}, \mathrm{x_j}) = \exp\left(-\gamma \left\| x_{\mathrm{i}} - x_{\mathrm{j}} \right\|^2\right) \text{ dengan } \gamma > 0 \tag{41}$$

Salah satu persamaan (39), (40), (41) diatas akan disubtitusikan pada persamaan Dual Langrange (26) beserta fungsi-fungsi kendalanya pada persamaan (27).

1.5.5. *Adaboost*

Adaboost adalah satu dari sekian algoritma yang meningkatkan performa *boosting*. Algoritma *Adaboost* diperkenalkan oleh Freund dan Schapire (1995) dan memecahkan banyak masalah praktis dari algoritma *boosting* sebelumnya. *Adaboost* mengambil masukan dari sekumpulan data *training* $(x_1, y_1) \ldots (x_m, y_m)$,setiap $x_i \in X$ dan $y_i \in y = \{-1, +1\}$. *Adaboost* berulang kali memanggil *base learner* dengan urutan iterasi t = 1, …, T. Saat memilih sampel *training* yang digunakan oleh *base learner* pada setiap iterasi *boosting* berdasarkan data *training* yang disediakan, *Adaboost* menggunakan distribusi dari sampel *training*. Distribusi yang digunakan pada iterasi ke-t dilambangkan dengan D_t, dan bobot sampel pelatihan ke-i dilambangkan dengan $D_t(i)$.

Secara intuitif, bobot ini adalah ukuran seberapa penting sampel ke-i dapat diklasifikasikan dengan benar dalam suatu iterasi. Selama inisialisasi, semua bobot diatur ke besar yang sama, dan pada setiap iterasi, kumpulan bobot sampel yang salah diklasifikasikan akan ditingkatkan. Sampel yang sulit diklasifikasi diberikan bobot yang lebih tinggi sehingga membuat *base learner* lebih memberi perlakuan khusus pada sampel tersebut. Tugas *base learner* adalah menemukan hipotesis lemah $h_t: X \rightarrow \{-1, +1\}$ menurut distribusi D_t. Kualitas hipotesis lemah diukur menggunakan error.

$$e_t = Pr_{\mathrm{i}\sim \mathrm{D_t}}[h_t(x_i) \neq y_i] = \sum_{i:h_t(X_i)\neq y_i} D_t(i) \tag{42}$$

Perhatikan bahwa error diukur dengan mempertimbangkan distribusi D_t dari hasil *training* suatu weak classifier. Dalam praktiknya, beberapa weak learner mungkin goritma yang bisa menggunakan bobot-bobot D_t pada sampel ernatif, saat hal ini memungkinkan, suatu subset dari sampel sampel berdasarkan D_t, dan sampel-sampel ini bisa digunakan earner.

dengan mempertimbangkan distribusi $D_t$dari hasil *training weak* beberapa *weak classifier* mungkin memiliki algoritma yang dapat $D_t$untuk sampel pelatihan. Alternatifnya, jika hal ini memungkinkan,

Sampel subset sampel pelatihan dapat diambil berdasarkan D_t dan menggunakan sampel ini untuk *weak classifier.*

Secara umum, tahapan awal algoritma *Adaboost* yaitu dengan menginput dataset yang terdiri dari pasangan data (x_1, y_1), (x_2, y_2), … , (x_m, y_m), $x_i \in X$ dan $y_i \in \{-1, +1\}$ merupakan label untuk setiap data. Pada awalnya, bobot $D_1(i)$ untuk setiap contoh data diinisialisasi sama besar, yaitu $D_1(i) = \frac{1}{m}$ untuk setiap i. Artinya, setiap contoh data memiliki kontribusi yang sama dalam proses *training* di awal. Pada setiap iterasi t, sebuah model sederhana yang disebut *weak learner* dilatih menggunakan distribusi bobot D_t yang sudah ada. *Weak learner* ini memprediksi label untuk setiap data dengan tujuan untuk meminimalkan error pada dataset tersebut. Setelah *weak learner* menghasilkan prediksi $h_t(x)$, kesalahan atau *error* e_t dari prediksi tersebut dihitung menggunakan persamaan *error* yang ditunjukkan pada persamaan sebelumnya. Jika error $e_t > 0.5$, proses iterasi dihentikan karena *weak learner* tersebut tidak efektif atau kinerjanya lebih buruk dari sekadar menebak acak.

Setelah *error* dihitung, kekuatan dari model *weak learner* diukur dengan menghitung nilai a_t menggunakan rumus:

$$a_t = \frac{1}{2}\ln\left(\frac{1 - e_t}{e_t}\right) \tag{43}$$

Nilai a_t menunjukkan seberapa kuat *weak learner* tersebut dalam memisahkan kelas. Jika error e_t kecil, a_t akan besar, yang berarti model lebih kuat. Bobot dari setiap contoh data $D_t(i)$ diperbarui agar *weak learner* berikutnya dapat lebih fokus pada contoh-contoh yang sulit diklasifikasikan. Update bobot dilakukan dengan formulasi

$$D_{t+1}(i) = \frac{D_t(i)}{Z_t} \tag{44}$$

$$D_{t+1}(i) = \frac{D_t(i)\exp(-a_t y_i h_t(x_i))}{Z_t} \tag{45}$$

$$\sum_{i=1}^{m} D_{t+1}(i) = 1 \tag{46}$$

Setelah T iterasi, hipotesis akhir $H(x)$dihasilkan dengan menggabungkan semua *weak learner* yang telah dilatih. Hipotesis akhir ini adalah pembobotan dari setiap *weak learner*, di mana kontribusi masing-masing *learner* ditentukan oleh kekuatan a_t.

$$H(x) = sign\left(\sum_{t=1}^{T} a_t h_t(x)\right) \tag{47}$$

$x)$ yang diperoleh ini adalah sebuah *majority vote* yang terboboti
sis, a_t adalah suatu bobot yang ditugaskan untuk h_t. Nilai $a_t \geq$
pun sebaliknya (Freund & Schapire, 1997).

1.5.6. *Text Preprocessing*

Text Preprocessing merupakan sebuah proses untuk membersihkan dan menyiapkan data mentah untuk dianalisis. Proses ini biasanya dilakukan dengan cara menghapus data yang tidak relevan atau mengubah data agar lebih mudah dianalisis. *Preprocessing* sangat penting dalam analisis sentimen, terutama di media sosial, karena teks di media sosial sering kali informal, tidak terstruktur, dan berisi noise (Assyam & Hasan, 2023).

Text Preprocessing berfungsi untuk proses awal sebelum dokumen teks diolah pada tahap selanjutnya dan akan dilakukan proses seleksi data yang akan diproses pada setiap dokumen. Proses ini terdiri atas beberapa proses pembersihan dokumen, yaitu *case folding, cleansing, tokenizing filtering* atau *stopword removal,* dan *stemming*.

Secara umum proses yang dilakukan dalam tahapan *preprocessing* adalah sebagai berikut:

a. ***Case Folding***, merupakan proses penyamaan *case* dalam teks dengan mengkonversi keseluruhan teks dalam dokumen menjadi suatu bentuk yang standar yaitu huruf kecil atau *lowercase*.
b. ***Cleaning***, merupakan langkah pembersihan data teks agar menjadi lebih sederhana dan tidak mengandung banyak noise. Noise yang dimaksud pada suatu teks melingkupi tanda baca, angka, karakter spesial, link (URL), tagar ataupun penggunaan spasi belebihan yang dapat menimbukan proses klasifikasi tidak berjalan secara optimal.
c. ***Tokenizing****,* merupakan tugas untuk memecah kalimat menjadi bagian-bagian yang disebut token dan menghilangkan bagian tertentu seperti tanda baca.
d. ***Normalizing***, merupakan proses mengubah kata-kata yang tidak baku menjadi kata baku
e. ***Stemming***, merupakan proses mengubah kata yang mendapatkan imbuhan menjadi kata dasarnya. Hal ini bertujuan untuk meningkatkan performa dan mengurangi penggunaan *resource* dari sistem dengan mengurangi jumlah
f. ***Stopword Removal,*** merupakan proses menghilangkan kata-kata umum yang sangat sering muncul dalam teks tapi dianggap tidak memberikan kontribusi signifikan terhadap makna atau sentimen teks dikarenakan tidak memuat nilai informasi yang kuat.

1.5.7. *Term Frequency-Inverse Document Frequency* (TF-IDF)

Term Frequency-Inverse Document Frequency (TF-IDF) adalah metode statistik yang digunakan dalam pemrosesan bahasa alami dan sistem temu kembali informasi untuk mengukur pentingnya sebuah term (kata kunci) dalam sebuah dokumen dalam konteks g lebih besar Metode ini bekerja dengan cara menghitung jumlah u dokumen, dan dibandingkan dengan jumlah total kata pada tode ini diterapkan (Ramos, 2003). Dokumen yang memiliki skor ggi untuk suatu kata cenderung dianggap lebih relevan terhadap n pengguna yang mengandung kata tersebut (Septiani & Isabela,

Proses TF-IDF, sesuai namanya, dibagi menjadi dua bagian yaitu proses Term Frequency (TF) dan proses Inverse Document Frequency (IDF) (Hananto et al., 2018). Penjelasan dari kedua bagian tersebut berdasarkan penjabaran dari (Septiani & Isabela, 2023) adalah sebagai berikut:

1. **Term Frequency (TF):** Mengukur seberapa sering suatu kata muncul dalam sebuah dokumen. Pendekatan umum untuk menghitung TF adalah dengan menghitung jumlah kemunculan kata tersebut dibagi dengan jumlah total kata dalam dokumen. Dalam beberapa kasus, TF dapat diubah dengan menerapkan skema penimbangan yang lebih kompleks.
2. **Inverse Document Frequency (IDF):** Mengukur seberapa penting suatu kata dalam konteks koleksi dokumen yang lebih besar. Kata-kata yang muncul lebih jarang di seluruh koleksi dokumen cenderung memiliki IDF yang lebih tinggi. IDF dihitung dengan membagi jumlah total dokumen dalam koleksi dengan jumlah dokumen yang mengandung kata tersebut. Hasilnya kemudian diambil logaritma untuk memperhalus skala.

Selanjutnya nilai TF dan IDF dikalikan bersama-sama untuk menghasilkan bobot kata untuk dalam sebuah dokumen (Septiani & Isabela, 2023). Rumus pembobotan metode TF-IDF berdasarkan persamaan:

$$idf_j = \log\left(\frac{N}{df_t}\right) \tag{48}$$

$$W_{i,j} = tf_{i,j} \times idf_{i,j} \tag{49}$$

Keterangan:

j = *term* ke-j dari dokumen
$W_{i,j}$ = Bobot dokumen ke –i terhadap term ke- j
tf_{ij} = Jumlah kemunculan *term* j ke dalam dokumen i
N = Jumlah dokumen secara keseluruhan
df_t = Jumlah dokumen yang mengandung *term*

1.5.8. *K-Fold Cross Validation*

Cross Validation adalah metode validasi model yang digunakan untuk menilai sejauh mana hasil analisis statistik dapat digeneralisasi pada kumpulan data independen. Teknik ini terutama dimanfaatkan untuk prediksi model dan memperkirakan akurasi model prediktif saat diterapkan di dunia nyata. Salah satu metode dalam cross-validation adalah *K-Fold Cross Validation.*

Secara umum, konsep *k-fold cross-validation* yaitu membagi data secara acak menjadi k bagian yang bersifat mutually exclusive $D_1, D_2, \dots, D_k$ yang masing masing
g sama. Proses *training* dan *testing* dilakukan sebanyak *k* kali.
ta partisi D_i digunakan sebagai data testing, sementara partisi lain
ngkan dan digunakan untuk melatih model. Misalkan pada iterasi
sebagai data *testing*, sedangkan $D_2, D_3, \dots, D_k$ digabungkan untuk
ta *training* untuk menghasilkan model, yang kemudian diuji pada
erasi kedua, data $D_1, D_3, \dots, D_k$ digabungkan dan bertindak sebagai

data *training* dan dilakukan pengujian menggunakan data D_2. Akurasi klasifikasi model diperoleh dengan cara merata-ratakan akurasi dari setiap iterasi (Han et al., 2011).

Nilai *k* yang umum digunakan dalam k-fold adalah 3, 5, 10 dan 20. Pada setiap *fold*, urutan penggunaan data uji harus berbeda antara *fold* yang satu dengan yang lainnya. Data latih di setiap fold terdiri dari bagian data yang tidak digunakan sebagai data uji (Putri et al., 2020).

1.5.9. Evaluasi Kinerja Klasifikasi

Evaluasi kinerja klasifikasi dari pembelajaran mesin, digunakan *confusion matrix. Confusion matrix* adalah suatu struktur tabel yang memberikan perbandingan antara hasil klasifikasi yang diperoleh dari sistem (prediksi) dengan hasil klasifikasi yang sebenarnya. Tabel pada confusion matrix memvisualisasikan jumlah data uji yang terklasifikasi dengan benar dan jumlah data uji yang salah diklasifikasikan (Fikri et al., 2020). Contoh *confusion matrix* untuk klasifikasi biner ditunjukkan pada Tabel 2.3.

Tabel 1. *Confusion Matrix*

Kelas Sebenarnya	Kelas Prediksi	
	Positif	Negatif
Positif	TP	FN
Negatif	FP	TN

Berdasarkan data yang diperoleh pada tabel 1. *Confusion Matrix*, keempat metrik tersebut dapat digunakan untuk menghitung ukuran kinerja klasifikasi antara lain; nilai akurasi, precision, recall dan f1 score.

Akurasi merupakan persentase dari total sentimen yang benar dikenali. Perhitungan akurasi dilakukan dengan cara membagi jumlah data sentimen yang benar dengan total data dan data uji. Untuk menghitung nilai akurasinya dilakukan dengan menggunakan Persamaan (50).

$$Akurasi = \frac{TN + TP}{TN + TP + FN + FP} \times 100\% \quad (50)$$

Pengukuran performa klasifikasi cara berikutnya yang dapat digunakan adalah menghitung *precission, recall* dan *f-measure*. *Precision* merupakan rasio data terprediksi benar positif dibandingkan dengan keseluruhan data yang diprediksi positif. Perhitungan *precision* dilakukan dengan cara membagi jumlah data benar yang bernilai

n jumlah data benar yang bernilai positif dan data salah yang ntuk menghitung nilai precision dapat dilakukan dengan maan (51).

$$Precision = \frac{TP}{TP + FP} \quad (51)$$

Recall merupakan rasio data terprediksi benar positif dibandingkan dengan keseluruhan data yang benar positif. Perhitungan *recall* dilakukan dengan cara membagi data benar bernilai benar positif dengan hasil penjumlahan dari data yang bernilai benar positif dan data yang bernilai negatif namun seharusnya bernilai positif. Perhitungan *recall* dapat menggunakan Persamaan (52).

$$Recall = \frac{TP}{TP + FN} \tag{52}$$

F-measure merupakan rata – rata harmonic dari *precision* dan *recall*. Nilai *f-measure* didapat dari perhitungan hasil perkalian *precision* dan *recall* dibagi dengan hasil penjumlahan *precision* dan *recall* kemudian dikalikan dua dan perhitungan *f-measure* menggunakan persamaan (53) (Sulistiyono et al., 2021).

$$F - measure = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{53}$$

1.5.10. *Youtube*

Sebagai salah satu media sosial, YouTube dipandang sebagai media yang dapat menyediakan informasi secara lengkap dan gratis. Itulah mengapa platform ini sering dianggap sebagai versi terbaru dari televisi yang hanya membutuhkan perangkat lebih sederhana dibandingkan dengan perangkat televisi tradisional. Program televisi yang masih dicari oleh audiens di YouTube biasanya adalah yang berkaitan dengan informasi olahraga atau berita terkini (Kusuma & Prabayanti, 2022).

Terdapat beberapa saluran Youtube yang memberikan informasi kepada masyarakat terkait perkembangan pemilu, seperti tvOne, Kompas TV, Narasi TV dll. Saluran-saluran tersebut memiliki jumlah pelanggan yang sudah mencapai lebih dari jutaan, hal tersebut berimplikasi pada jumlah komentar yang muncul disetiap konten menjadi lebih banyak dan bervariasi sehingga memberi peluang untuk dilakukannya analisis.

Interaksi aktif seperti komentar, *like*, dan *share* pada setiap konten tidak hanya memperkaya informasi yang disajikan, tetapi juga menyediakan data berharga untuk menganalisis tren dan perilaku audiens. Data tersebut memungkinkan peneliti serta praktisi media untuk memahami respons publik secara *real-time*, mengidentifikasi preferensi penonton, dan mengembangkan strategi penyajian konten yang lebih relevan.

BAB II

METODE PENELITIAN

2.1. Sumber Data

Data yang digunakan dalam penelitian ini merupakan data primer. Data yang diperoleh dengan cara *scrapping* yang dilakukan pada platform *Youtube.* Data didapat sebanyak 11.500 data komentar dan format data yang diambil berbentuk teks.

Tabel 2. Jumlah Komentar yang diambil pada Video di Berbagai Saluran *Youtube*

Nama Saluran Youtube	Judul Video	Komentar yang diambil
KPU RI	Debat Kelima Calon Presiden Pemilu Tahun 2024	1000
METRO TV	[FULL] DEBAT PAMUNGKAS CALON PRESIDEN 2024, 04 FEBRUARI 2024	2000
KOMPASTV	[FULL] Debat Terakhir Capres 2024: Anies, Prabowo, Ganjar Bahas Stunting hingga Bansos	600
Official iNews	[FULL] Debat Kelima Capres Pemilu 2024, Minggu, 4 Februari 2024	550
TVRI Nasional	Siaran Nasional | Debat Kelima Calon Presiden PEMILU 2024, 4 Februari 2024	1850
Netmediatama	LIVE : Debat Kelima Calon Presiden Pemilu Tahun 2024	1000
Najwa Shihab	[FULL] Layar Tancap Mata Najwa, Nobar Debat Capres Ronde Kelima | Mata Najwa	4500
tvOneNews	[LIVE] Debat Pamungkas Capres 2024 (4/2/2024) | tvOne	2500

2.2. Tahapan Analisis Data

Tahapan dalam menyelesaikan penelitian ini adalah sebagai berikut:

1. Pengumpulan Data

   Pada tahap ini dilakukan pengumpulan data dari *Youtube* mengenai Debat ke-5 an Wakil presiden, data yang digunakan adalah data dari hasil tar Youtube sejak Februari 2024 hingga Maret 2024 pada 12 ng menyiarkan acara debat. *Scrapping* dilakukan menggunakan rograman *python* dengan menginstal *library* *Downloader* kemudian data di terkumpul dengan format csv *rary pandas.*

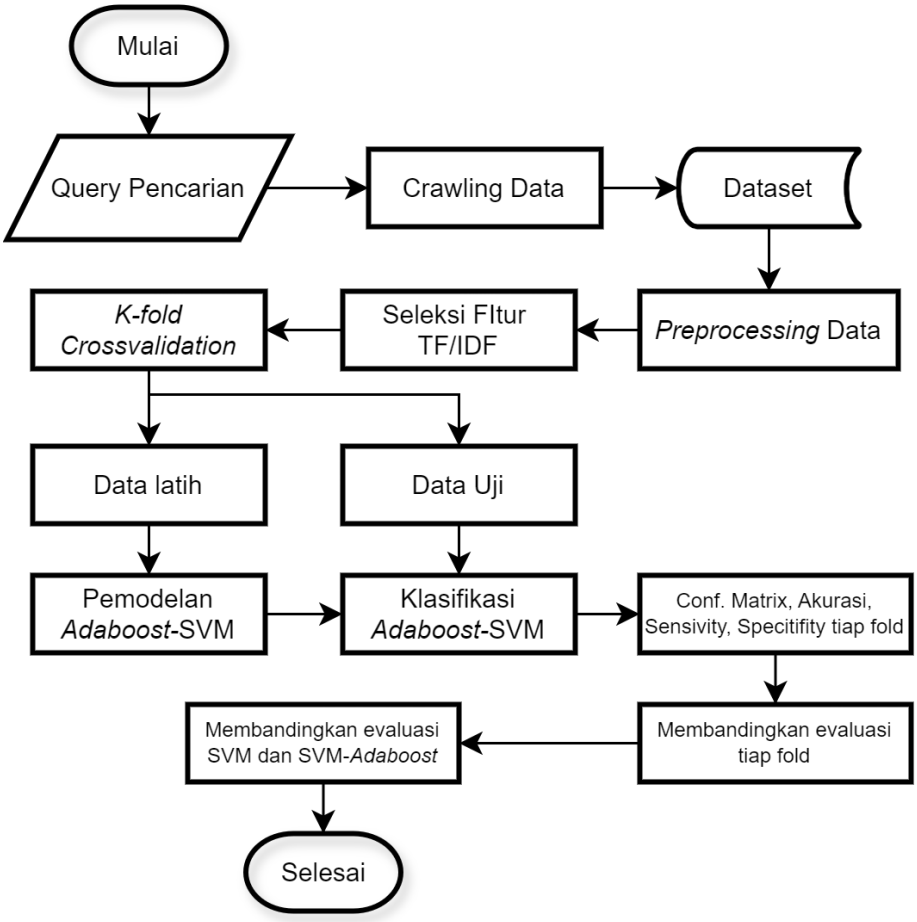
Data yang telah berhasil dikumpulkan harus melewati tahap *preprocessing* agar dapat digunakan pada tahapan selanjutnya. *Preprocessing* yang dilakukan antara lain; *Case Folding, Cleaning, Tokenize, Normalize, Stemming, Stopword Removal*

3. Pembobotan kata
   Pembobotan dilakukan dengan menggunakan metode *Term Frequency-Inverse Document Frequency* (TF-IDF). Hasil dari pembobotan pada tahap ini akan digunakan untuk proses klasifikasi.
4. Membangun Model SVM-*Adaboost*:
   1. Memboboti setiap pengamatan pada data *training* dengan bobot $\mathrm{D}_1(\mathrm{i}) = \frac{1}{\mathrm{m}}$, m adalah banyaknya pengamatan di dalam data *training*.
   2. Menentukan jumlah iterasi maksimal *boosting* yaitu 1-20 iterasi.
   3. Mengambil sampel sebanyak *m* tanpa pengembalian di iterasi pertama dan dengan pengembalian di iterasi selanjutnya menggunakan bobot pengamatan sebagai probabilitas terpilihnya sampel.
   4. Melatih SVM pada data sampel menggunakan Kernel RBF menggunakan sampel-sampel pengamatan yang dibentuk pada tahap (3). Adapun tahapan dalam melatih model SVM sebagai berikut:
      a) Mengoptimasi fungsi tujuan pada persamaan (26) menggunakan kuadratik *programming*, dengan fungsi kendala pada persamaan (27) untuk memperoleh $\boldsymbol{w}$ dan b.
      b) Mengklasifikasikan seluruh pengamatan pada data *training* berdasarkan fungsi pemisah yang diperoleh.
   5. Menghitung *error* dari hasil klasifikasi data *training* menggunakan SVM pada tahap (4). *Error* dihitung menggunakan persamaan (43). Bila diperoleh nilai *error* $e_i = 0$ maka iterasi berhenti.
   6. Menghitung akurasi klasifikasi atau pembobot *boosting* a_i. Perhitungan berdasarkan persamaan (44).
   7. Memperbaharui bobot dari setiap pengamatan pada data latih menggunakan persamaan (45).
   8. Mengulangi tahap 3-4 hingga iterasi maksimal tercapai.
   9. Menormalisasi bobot classifier hingga penjumlahan menjadi 1. Kemudian diperoleh fungsi klasifikasi akhir seperti seperti pada persamaan (47).
5. Klasifikasi Model
   Setelah melakukan seleksi fitur, akan dilakukan klasifikasi pada dokumen yang diperlukan dengan tahap-tahap klasifikasi sebagai berikut:
   a. Membagi data menjadi 2 yaitu data latih (*training*) dan data uji (*testing*) menggunakan prosedur *5-cross validation* dengan rasio pembagian 80:20
   b. Mengklasifikasikan pengamatan-pengamatan pada data *testing* menggunakan si SVM, dan *Adaboost*-SVM yang diperoleh dari model dari data

atriks konfusi dan menghitung performa klasifikasi berdasarkan *·acy, Precision, Sensitivity,* dan *F-measure* menggunakan) sampai (53).
embahasan

Tahap ini adalah tahap akhir, dilakukan analisis yang kemudian hasil dari analisis klasifikasi yang diperoleh sebelumnya akan dibahas untuk melihat kinerja dari metode klafisikasi yang diterapkan pada penelitian ini. Setelah dilakukan pembahasan, maka dapat ditarik kesimpulan.

2.3. Diagram Alir



Gambar 3. Diagram Alir