BAB 1

PENDAHULUAN

1.1 Latar Belakang

Coronavirus diseases 2019 (*COVID19*) adalah penyakit yang baru dan tidak pernah ditemukan pada generasi yang ada sebelumnya. Gejala yang sering dikeluhkan dari *Covid-19* antara lain gangguan dari pernapasan akut seperti badan panas, batuk, dan sesak nafas [1].

Pada 11 Maret 2020 Organisasi Kesehatan Dunia (*WHO*) telah resmi menyatakan wabah virus corona atau 2019 *coronavirus disease (Covid-19)* sebagai epedemi penyakit secara global. Seseorang mungkin tidak asing dengan gejalah istilah epedemi secara global, tetapi mereka akan merasakan bahwa suatu virus yang besar sedang terjadi disekitar. Seiring berjalannya waktu, ternyata epedemi *Covid-19* menjadi peristiwa yang luar biasa. Hingga 31 Mei 2021, penyakit tersebut telah menyebar dengan cepat di setidaknya 219 negara. Jumlah total infeksi di seluruh dunia telah melampaui 171,5 juta, dan jumlah kematian mencapai 3,7 juta. Pesatnya penyebaran wabah ini telah membawa dampak negatif yang sangat besar bagi semua negara, baik dari segi kesehatan, masyarakat dan kesejahteraan, maupun ekonomi [2].

Pemerintah Indonesia telah memberi laporan kasus *Covid-19* bertambah 20.004 kasus pada Jumat (20/8/2021). Maka dari itu, total kasus menjadi 3.950.304 kasus. Sebanyak 3.499.037 yang dinyatakan sembuh (88,58%) dan

123,981 meninggal (3,14%), sisanya masih menjalani pengobatan. Begitu juga dengan individu dalam pengawasan sebanyak 269.480 individu [2].

Analisis Sentimen merupakan riset komputasional dari opini, sentimen dan emosi yang diekspresikan secara tekstual (Liu, 2010). Penggunaan analisis sentimen pada umumnya digunakan untuk menganalisis tentang suatu produk dalam meningkatkan kualitas produk kedepannya. Dalam hal ini analisis sentimen dapat diterapkan pada ulasan aplikasi PeduliLindungi. Pada ulasan terkadang terdapat kesalahan penulisan, dari hal tersebut perlunya mengartikan ulang untuk mengetahui maksud dari ulasan pengguna. Kata tersebut nantinya akan diklasifikasikan agar dapat diketahui masuk kedalam sentimen yang mana. Dalam hal ini maka dibutuhkan metode klasifikasi untuk menganalisis ulasan [3].

PeduliLindungi adalah aplikasi yang ditujukan untuk mempermudah organisasi penting pemerintah dalam mengikuti pencegahan penyebaran Infeksi *Covid-19.* Aplikasi ini bergantung pada dukungan pengguna untuk berbagi informasi lokasi saat bepergian sehingga riwayat kontak dengan pasien *Coronavirus* dapat dilacak. Jika pengguna aplikasi ini berada di daerah atau di zona merah (misalnya seseorang yang telah mengidap *COVID-19* atau daerah tempat pasien dalam pengawasan), mereka juga akan diberi tahu [4].

Play Store adalah penyedia aplikasi di bawah *Google,* yang menyediakan berbagai aplikasi, contohnya aplikasi permainan, film, musik, dan buku yang mempunyai beberapa kategori. *Google Play Store* bisa diakses dengan Web, Telepon berbasis Android, dan Google TV. Terdapat beberapa fitur di *Google Play Store* contohnya adalah penilaian dan komentar dari pengguna. Komentar

atau ulasan adalah teks atau kalimat yang mempunyai opini. Dari komentar itu sering digunakan dalam melakukan penilaian aplikasi apakah terekomendasi atau tidak bagi pengguna baru [5].

Oleh karena itu Pemerintah Indonesia mengeluarkan aplikasi untuk melacak penyebaran *Covid-19,* PeduliLindungi, sudah rilis dan bisa didapatkan oleh pengguna perangkat Android di *Google Play Store* sejak 30 Maret 2020.

Menurut pantauan dari Tekno Liputan6.com banyak pengguna aplikasi yang masih mengeluhkan *error* pada aplikasi tersebut, hal tersebut dilihat dari Ulasan dan Penilaian pada *Google Play Store* [6]. Dari permasalahan yang ada pada penelitian ini berfokus untuk sentimen analisis dengan memanfaatkan *Rating* dan *Review* dari pengguna aplikasi PeduliLindungi yang didapatkan dari *Platform Google Play Store,* yang bertujuan membantu pengembang dalam menyajikan aplikasi menjadi lebih baik lagi kedepannya.

Pada Penelitian ini data yang digunakan berasal dari *review* pengguna *Google Play Store* yang akan diklasifikasikan menggunakan *Naïve Bayes* dan *Support Mector Machine* (SVM). *Naïve Bayes* dan SVM merupakan bagian dari *Data Mining. Naïve Bayes* adalah salah satu metode *machine learning* yang menggunakan perhitungan probabilitas. Kelebihan dari pengguna *Naïve Bayes* dalam klasifikasi dokumen dapat ditinjau dari prosesnya yang mengambil aksi berdasarkan data-data yang telah ada sebelumnya [7]. Sedangkan *Support Vector Machine* (SVM) adalah suatu teknik yang digunakan untuk melakukan prediksi, baik dalam kasus regresi maupun klasifikasi. Prinsip kerja SVM yaitu mencari *hyperplane* yang optimal dengan margin maksimum yang bertindak sebagai batas

keputusan, untuk memisahkan dua kelas yang berbeda. Kelebihan SVM menawarkan akurasi tinggi dan bekerja dengan baik dengan ruang dimensi tinggi [8].

Penelitian terkait pada analisis sentimen ini akan diklasifikasi dengan menggunakan *Naïve Bayes*, *Support Vector Machine* (SVM) dan *K-Nearest Neighbor* untuk memberi hasil yang lebih akurat, peneliti mengambil tiga algoritma yang sudah disebutkan untuk melakukan perbandingan mana yang lebih akurat. Mengambil dari penelitian yang dilakukan oleh [9] yang meneliti tentang Aplikasi PeduliLindungi di tahun 2020 dengan pengambilan data dari April 2020 sampai Juni 2020 dan mendapatkan 1.364 data menghasilkan perbandingan dari *Naïve Bayes* dan *Support Vector Machine* dengan menghasilkan nilai akurasi dan dilai AUC masing-masing yaitu untuk algoritma *Naïve Bayes* PSO nilai akurasi sebesar 69,00% dan nilai AUC sebesar 0,659, sedangkan untuk algoritma *Support Vector Machine* berbasis PSO mempunyai nilai akurasi sebesar 93,0% dan nilai AUC sebesar 0,997. Untuk itu, *Support Vector Machine Particle Swarm Optimization* penelitian ini menjadi akurasi yang tinggi, hasil ini dibuat untuk memberikan pemecahan masalah analisis sentiment dalam opini pengguna aplikasi PeduliLindungi. Hasil dari penelitian ini akan mendapatkan opini Masyarakat terhadap aplikasi PeduliLindungi yang telah digunakan untuk kepentingan pemantauan perkembangan *Covid-19* di Indonesia, yang akan dibagi dalam dua klasifikasi yaitu Positif dan Negatif.

Dari penjelasan latar belakang diatas maka penulis melakukan penelitain tugas akhir yang berjudul **"Analisis Sentimen Aplikasi PeduliLindungi Pada**

Google Play Store **Menggunakan Metode** ***Naïve Bayes*** **dan** ***Support Vector Machine*** **(SVM)".**

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan, rumusan masalah pada penelitian ini adalah sebagai berikut :

1. Bagaiman implementasi algoritma *Naïve Bayes* dan *Support Vector Machine* dalam analisis sentimen pada aplikasi PeduliLindungi?
2. Bagaimana tingkat akurasi yang diberikan oleh kedua algoritma tersebut pada analisis sentimen?

1.3 Tujuan Penelitian

Tujuan dari penelitian ini untuk mendapatkan hasil pengklasifikasian terbaik dari kedua metode perbandingan *Naïve Bayes* dan SVM. Dan untuk menguji tingkat akurasi dari kedua metode dalam pengklasifikasian sentimen aplikasi PeduliLindungi berdasarkan *Rating* dan *Review* di *Google Play Store.*

1.4 Batasan Masalah

Adapun Batasan dari masalah yang ditemukan peneliti agar pembahasan dalam penelitian adalah :

1. Penelitian ini menggunakan data *Rating* dan *Review* pengguna aplikasi PeduliLindungi pada *Google Play Store.*
2. Jumlah data ulasan yang digunakan sebanyak 2000 ulasan. Ulasan yang diambil berdasarkan ulasan yang terbaru di tahun 2022.
3. Metode yang digunakan dalam perancangan sistem ini adalah *Naïve Bayes* dan SVM.

4. Kelas sentimen terbatas Positif dan Negatif
5. Sistem yang dibuat menggunakan bahasa pemograman *Python*.

1.5 Manfaat Penelitian

Adapun manfaat dari hasil penelitian ini adalah sebagai berikut :

1. Dapat membantu Pengembang sistem mengevaluasi aplikasi PeduliLindungi untuk menyesuaikan dengan keinginan Masyarakat sehingga Pengembang dapat terus meningkatkan aplikasinya dalam masa sekarang maupun kedepannya.
2. Dapat menambah pengetahuan tentang metode *Naïve Bayes* dan *Support Vector Machine* (SVM).
3. Dapat dijadikan referensi untuk pengembangan penelitian selanjutnya.

1.6 Sistematika Penulisan

Sistematika penulisan dari skripsi ini terdiri dari lima bagian utama sebagai berikut:

BAB 1 PENDAHULUAN

Bab ini berisi latar belakang, rumusan masalah tujuan penelitian, batasan masalah, manfaat penelitian dan sistematika penulisan.

BAB 2 LANDASAN TEORI

Bab ini berisi teori-teori yang digunakan pada penelitian ini. Teori-teori yang berhubungan dengan landasan dalam pembuatan aplikasi.

BAB 3 METODOLOGI PENELITIAN

Bab ini berisi tahapan–tahapan dalam pengumpulan data, perancangan sistem perumusan masalah dan analisa.

BAB 4 ANALISA DAN PERANCANGAN

Bab ini berisi analisa dan perancangan Media pembelajaran yang akan dibangun atau dikembangkan.

BAB 5 IMPLEMENTASI DAN PENGUJIAN

Bab ini berisi implementasi dari analisa dan perancangan dan pengujian pada aplikasi yang berhasil dibangun.

BAB 6 PENUTUP

Bab ini berisi rangkuman dari hasil penelitian yang telah dilakukan dan saran–saran untuk pengembangan aplikasi atau penelitian selanjutnya.

BAB 2

LANDASAN TEORI

2.1 PeduliLindungi

PeduliLindungi adalah aplikasi yang dirancang untuk mempermudah pemerintah untuk melakukan Pelacakan dalam hal mencegah penyebaran penyakit Virus *Corona* (*Covid-19).* Aplikasi ini membuat keterlibatan seseorang untuk bertukar informasi lokasi disaat keluar rumah. Sehingga pencarian riwayat kontak dengan pengidap *COVID-19* dapat dilihat. Pengguna dari aplikasi akan mendapatkan pemberitahuan jika berada di dalam keramaian atau berada di kawasan merah, yaitu daerah yang sudah terdokumentasi bahwa ada penderita positif *COVID-19* atau ada penderita Dalam Pengawasan [10].

2.2 *Google Play Store*

Google Play Store adalah penyedia aplikasi dibawah *Google*, yang menyediakan aplikasi dan produk *online*, seperti aplikasi permainan, film, musik, dan buku yang mempunyai beberapa kategori. *Google Play Store* bisa diakses dengan *Web*, Telepon berbasis Android, dan *Google TV*. *Google Play Store* pada berdiri sejak..Maret 2012. Android Market disebut *Google Play Store*, sedangkan buku, musik serta film disebut *Play Books*, *Play Music* dan *Play Movies*. *Google Play* mempunyai fitur yang dimana pengguna bisa memberi ulasan yang berisi teks [11].

2.3 Analisis Sentimen

Analisis Sentimen adalah pengelompokan dari sumber komputer yang berkonsentrasi pada bahasa komputasi, pemrosesan bahasa, dan penambangan data yang bertujuan melihat emosi, penilaian sikap pendapat dari evaluasi seseorang terhadap seorang pembicara atau penulis tentang suatu hal tertentu. Produk, layanan, organisasi, individu, tokoh masyarakat, topik, acara, atau aktivitas. Nilai dari analisis sentimen dapat dibagi menjadi opini Positif, opini Negatif, dan opini yang Netral. Analisis Sentimen mempunyai tujuan untuk menilai perasaan, sikap, pendapat, dan apresiasi seseorang terhadap suatu karya atau tokoh masyarakat [12]

2.4 *Data Mining*

Data Mining adalah suatu istilah yang digunakan untuk menguraikan penemuan pengetahuan di dalam *database. Data Mining* adalah proses yang menggunakan teknik static, matematika, kecerdasan buatan, dan *machine learning* untuk mengekstraksi dan mengidentifikasi informasi yang bermanfaat dan pengetahuan yang terkait dari berbagai *database* besar. Definisi umum dari *data mining* itu sendiri adalah proses pencarian pola-pola yang tersembunyi (*hidden pattren*) berupa pengetahuan (*knowledge*) yang tidak diketahui sebelumnya dari suatu sekumpulan data yang mana data tersebut dapat berada di dalam *database*, data *warehouse*, atau media penyimpanan informasi yang lain [13].

2.5 *Text Mining*

Text Mining merupakan metode digunakan untuk mengklasifikasikan dokumen dimana merupakan varian dari *Data Mining* yang berusaha menemukan

pola yang menarik dari sekumpulan besar data tekstual [14]. Dalam hal ini melihat dari proses penggalian sumber yang berkualitas dalam teks. Informasi berkualitas tinggi sering diperoleh dengan memprediksi pola dan tren melalui cara seperti mempelajari pola dari statistik.

2.6 *Text Preprocessing*

Tahap dari *prepocessing* atau pengolahan data meliputi operasi membangun data dan juga pembersihan data sehingga siap untuk dikelola hingga tahap pemodelan data. Langkah - Langkah pemrosesan data antara lain [15]:

1. Hapus Ekspresi Regular :

   Menghilangkan Ekspresi dari dalam teks.
2. Hapus *URL :*

   Hapus *URL* yang terdapat dalam teks.
3. Hapus Anotasi :

   Menghapus tanda @ (anotasi) dari dalam teks.
4. Hapus Nomor :

   Menghapus Nomor yang ditemukan dalam teks.
5. Tokenisasi :

   Proses tokenisasi pada teks adalah melakukan pemecahan sekumpulan kalimat menjadi banyak karakter yang diperlukan, ini sering disebut token, sehingga menjadi kata-kata dengan nilai tunggal makna tertentu.
6. *Stemming :*

   Menghapus imbuhan dari setiap kata untuk menjadikannya kata dasar, juga untuk membersihkan suatu kata dengan pengejaan yang kurang

tepat. Algoritma *stemming* untuk satu bahasa berbeda dengan algoritma *stemming* untuk bahasa lain.

7. *Transform Case :*

   Mengkonversi teks kapital dalam data menjadikan teks kecil atau sebaliknya. Karena agar ketika memasuki tahap pemodelan klasifikasi, dengan huruf yang seragam sehingga tidak ada kesalahan dalam melakukan *tokenize*, yang umumnya digunakan untuk mengubah menjadi teks yang kecil (*lower case*).

8. *Filter Stopwords :*

   Pemrosesan yang terlibat dengan menghilangkan kata-kata yang diabaikan umumnya adalah kombinasi, kualifikasi, dll. Dalam hasil penguraian arsip, membandingkan ke daftar *stopword* yang berisi kata-kata.

9. Token *Filter* (Berdasarkan Panjang) :

   Penghapusan kata-kata dengan panjang huruf tertentu. Misalnya, minimal 2 karakter dan batas 25 karakter. Ini menyiratkan bahwa kata-kata hanya terdiri dari satu karakter dan lebih dari 25 karakter akan dihapus.

10. Pelabelan :

    Ini adalah langkah dimana hasil dari langkah sebelumnya akan dihitung sesuai dengan perhitungan polaritas komentar yang diambil, sehingga bisa dibuat dua kategori sentiment Positif dan sentiment Negatif, untuk kategori yang (bernilai = 0) tidak untuk diproses.

2.7 *Naïve Bayes Classifier*

Naïve Bayes adalah algoritma induktif terbaik untuk pembelajaran mesin dan data *mining*. Kinerja yang bersaing *naïve bayes* dalam klasifikasi meskipun dengan asumsi atribut yang mandiri (tidak hubungan antar atribut). Asumsi atribut yang mandiri ini dalam data memang jarang terjadi, tetapi meskipun asumsi atribut yang mandiri dilanggar, kinerja pengklasifikasian *naïve bayes* cukup tinggi, yang telah ditunjukan dalam berbagai penelitian eksperimental. *Naïve Bayes* adalah pengelompokan dengan peluang dan metode statistik yang diusulkan oleh para ilmuwan [16].

Salah satu dari pengklasifikasian *Naïve Bayes Classifier* ini adalah asumsi kuat dari setiap kondisi. Metode cocok sebagai pengelompokan sentimen dalam Penelitian ini karena memiliki beberapa keunggulan, di antaranya mudah, cepat, dan sangat akurat. Persamaan pada prinsip dasar *Naïve Bayes :*

$$P(H|X) = \frac{P(X|H).P(H)}{P(H)} \quad \text{(2.1)}$$

Penjelasan Rumus Persamaan *Naïve Bayes* :

X = Data dengan kelas tidak dikenal

Y = Hipotesis data X adalah kelas khusus

P(H|X) = Probabilitas hipotesis H didasarkan pada kondisi X

P(H) = Probabilitas hipotesis H

P(X|H) = Probabilitas hipotesis X didasarkan pada kondisi H

P(X) = Probabilitas X.

Pemrosesan klasifikasi memerlukan beberapa panduan untuk menentukan pengklasifikasi yang sesuai untuk hasil analisis yang diinginkan. Oleh karena itu, metode di atas dapat diberikan sebagai berikut :

$$P(C|F1 \dots Fn) = \frac{P(C)P(F1 \dots Fn|C)}{P(F1 \dots Fn)} \quad \text{(2.2)}$$

C = Mempresentasikan kelas

$F_1 \; dan \; F_n$ = Mepresentasikan Karakteristik untuk klasifikasi

Oleh karena itu, rumus di atas menunjukan bahwa kemungkinan dari kemunculan sampel keistimewaan dalam kelas C (*Posterior*) adalah kemungkinan munculnya kelas C (*Prior*), dikalikan kemungkinan kemunculan sebuah keistimewaan contoh pada kelas C (*likelihood*), kemudian dibagi dengan kemungkinan terjadinya suatu keistimewaan seluruhnya (*evidence*) [17].

$$Posterior = \frac{Prior \; X \; Likelihood}{evidence} \quad \text{(2.3)}$$

Perhitungan keseluruhan selalu konsisten untuk setiap kelas dalam sampel. Hasil Perhitungan dari *posterior* tersebut nantinya akan dibandingkan dengan hasil posterior dari kelas lainnya dalam memutuskan kelas mana sampel akan dikelompokan. Pengembangan lebih lanjut dari rumus *Bayes* dilakukan dengan menghitung ($C\backslash F_1, \dots,$) menggunakan aturan perkalian berikut :

$$
\begin{aligned}
P(C|F_1, \ldots, F_n) &= P(C)\, P(F_1, .., F_n|C) \\
&= P(C)P(F_1|C)P(F_2, \ldots, F_n|C, F_1) \\
&= P(C)P(F_1|C)P(F_2|C, F_1)P(F_3, \ldots, F_n|C, F_1, F_2) \\
&= P(C)P(F_1|C)P(F_2|C, F_1)P(F_3|C, F_1, F_2), P(F_4, \ldots, F_n|C, F_1, F_2, F_3) \\
&= P(C)P(F_1|C)P(F_2|C, F_1)P(F_3|C, F_1, F_2) \ldots P(F_n|C, F_1, F_2, F_3, .., F_{n-1})
\end{aligned}
$$

Gambar 2.1 Contoh Perhitungan Penjabaran

Seperti dapat dilihat, hasil penjabaran tersebut menyebabkan semakin kompleks faktor yang mempengaruhi nilai probabilitas yang hampir tidak mungkin untuk dianalisis satu per satu. Oleh karena itu, perhitungan menjadi sulit untuk dilakukan [17].

2.8 *Support Vector Machine*

Support Vector Machine merupakan perhitungan pembelajaran mesin yang diawasi dan digunakan untuk tantangan pengelompokan atau metode statistik. Terutamanya digunakan dalam kasus pengelompokan. Dalam perhitungan, plot dari setiap data sebagai fokus didalam bagian n-dimensi (dimana *n* adalah banyaknya objek yang dimiliki) dengan hasil perhitungan dari setiap objek menjadi nilai koordinat tertentu. Selanjutnya, dengan melakukan pengelompokan dapat menemukan *hyperplane* yang membedakan antar kedua kelas dengan sangat baik [18].

SVM dikembangkan pada 1990-an berdasarkan pertimbangan teoritis Vladimir Vapnik pada tahun 1990-an. Pengembangan teori statistik pembelajaran: Teori Vapnik-Chervonenkis. SVM dengan cepat diadopsi karena kemampuan

mereka untuk bekerja dengan data besar, sejumlah kecil parameter hiper, jaminan teoritis mereka, dan hasil yang baik dalam praktiknya [19].

Untuk membuat keputusan dengan metode SVM, metode ini menggunakan fungsi kernel K (, x_d). Inti yang digunakan pada penelitian ditunjukkan dalam Persamaan 5 :

$$K\ (x_i\ , x_d) = (X_i^T\ X_j\ + C)^d, \gamma > 0 \quad \text{..... (2.4)}$$

Pemrosesan dilakukan pada data pelatihan menggunakan algoritma pelatihan sekuensial karena merupakan algoritma yang sederhana yang tidak memakan banyak waktu dengan langkah-langkah perhitungan [20]:

1. Memberi nama terhadap parameter, seperti , γ, C, dan ε.

   ai = alfa, untuk mencari *support vector*

   γ = konstanta gamma untuk mengontrol kecepatan

   C = variabel slack

   ε = epsilon digunakan untuk mencari nilai *error.*

2. Hitung matriks Hessian yang didapat dari perkalian antar kernel polynomial dan y yang merupakan *vector* bernilai 1 dan -1. Persamaan dari matriks Hessian adalah :

   $$D_{ij} = y_i y_j \left(K(x_i\ , y_j) + \lambda^2\right) \quad \text{.....(2.5)}$$

3. Lakukan perhitungan berikut hingga interasi data i hingga j :

a. $E_i = \Sigma_j^i \, a_j \, D_{ij}$

b. $\delta a_i = Min \, (Max \, [y(1 - E_i), \; a_i], C - \; a_i$

c. $a_i = \; a_i \; + \; \delta a_i$..... (2.6)

4. Lakukan ketiga langkah diatas secara berulang hingga mencapai batas maksimum interasi

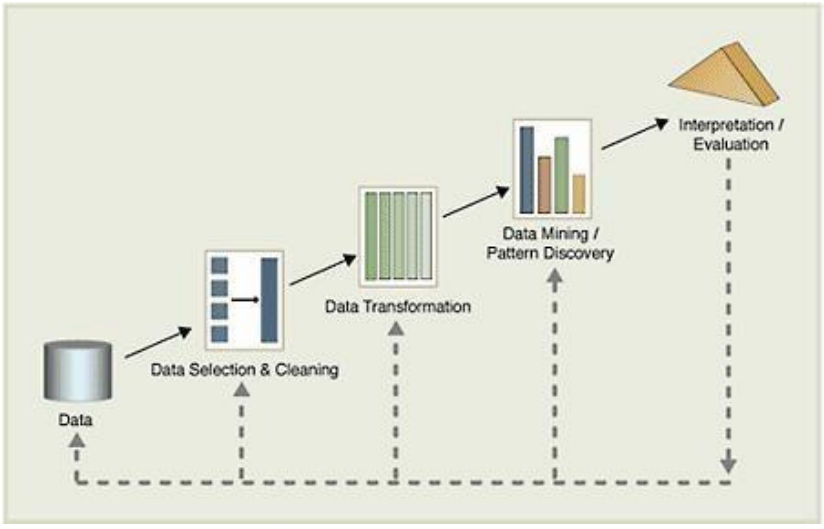
5. Proses *sequential learning* dari tahap 1 hingga 4 akan mendapatkan nilai dari *support vector* (SV), dimana nilai SV = (a_i>thresholdSV) . Setelah itu, perlu dilakukan perhitungan pada nilai bias b yang diperoleh dari Persamaan 10.

$$b = \frac{1}{2} \, (\Sigma_{i=0}^{N} a_i y_i \, K(x_i, x^-) \; + (\Sigma_{i=0}^{N} a_i y_i \, K(x_i, x^+) \quad \text{..... (2.7)}$$

6. Untuk mengetahui hasil klasifikasi teks pada beberapa kelas sentimen, dihitung fungsi *f(x).* Jika hasil dari fungsi negatif, maka dokumen tersebut tergolong sentimen kelas *cyberbullying* negatif. Jika nilai fungsinya positif, maka dokumen tersebut tergolong kelas sentimen *cyberbullying* positif. Fungsi *f(x)* diperoleh pada Persamaan 9.

2.9 *Knowledge Discovery In Database*

Knowledge Discovery In Database (KDD) yaitu seluruh proses non-trivial untuk menentukan pola (*pattern*) yang berada didalam data, di mana *pattern* ditemukan valid dan mudah dipahami. KDD mengacu pada cara untuk mengintegrasikan, secara pengetahuan ilmiah, interpretasi, dan penggambaran model dalam beberapa kumpulan data [21].



Gambar 2.2 Proses KDD

1. *Data Selection*

    a. Membuat kumpulan data target yang baru, pilih fokus pada subset variabel atau sampel data tempat pendalaman akan dilakukan.

    b. Penetapan data dari dataset yang berhubungan harus dijalankan sebelum memulai tahap pendalaman informasi di KDD.

2. *Data Cleaning*

    a. Prapemrosesan adalah proses mendasar seperti penghilangan gangguan dalam data.

    b. Sebelumnya *data mining* dilakukan, harus melakukan pembersihan data sebagai tujuan utama KDD.

    c. Prosedur pembersihan meliputi diantaranya, menghapus data duplikat mengawasi ketidak konsistenan data dan mengawasi kesalahan dalam data.

    d. Dilakukan proses *enrichment* adalah pemrosesan "memperkaya" data sudah ada dengan informasi (eksternal) lainnya.

3. *Data Transformation*

    a. Menemukan karakteristik yang berfungsi menyajikan data tergantung pada tujuan akhir.

    b. Sebuah prosedur memodifikasi data terpilih, sehingga data sudah cocok dalam pemrosesan *data mining*. Prosedur ini merupakan prosedur inovatif dan sangat bergantung dari model informasi yang diinginkan dalam database.

4. *Data Mining*

    a. Pemilihan hasil akhir dari proses KDD seperti klasifikasi, regresi, dll.

    b. Proses *Data Mining* yaitu prosedur dalam menemukan.pola menarik dalam data dipilih dengan memanfaatkan metode tertentu. Metode dalam data *mining* sangat bervariasi. Pemilihan metode atau algoritma yang benar dapat mempengaruhi pada keseluruhan proses dan hasil akhir KDD.

5. Evaluasi

    a. Terjemahan.model dari *data mining.*
    b. Model sumber pengetahuan dari proses *data mining* harus divisualisasi..kebentuk yang mudah dipahami oleh pihak yang bersangkutan.

Langkah ini adalah tahap proses KDD untuk memverifikasi bernakah pola yang dihasilkan berbanding terbalik dengan fakta maupun asumsi yang sudah ada.

2.10 *Machine Learning*

Machine Learning adalah bagian dari kecerdasan buatan manusia yang menyoroti pembangunan dan konsentrasi pada kerangka kerja sehingga dapat memperoleh informasi yang diperolehnya. Tanpa informasi, perhitungan *Machine Learning* tidak dapat bekerja. Informasi yang ada umumnya dikategorikan menjadi dua, yaitu informasi latih dan informasi uji. Informasi latih digunakan untuk mempersiapkan perhitungan, sedangkan informasi pengujian digunakan untuk memutuskan pameran perhitungan yang baru disiapkan ketika mengamati informasi baru yang belum pernah terlihat [22].

2.11 Bahasa *Python*

Python adalah bahasa pemograman interpretative yang bisa dipasang pada berbagai *platform*, khususnya *platform* yang berfokus pada keterbacaan kode. *Data science, internet of things* (IoT), dan *machine learning* merupakan beberapa hal yang berkaitan langsung dengan *Python*. Para *programmer* biasa

menggunakan *Python* untuk membuat *prototype, scripting* guna mengelola infrastruktur, maupun pembuatan *website* dalam skala besar.

Sebuah penelitian yang diterbitkan dalam jurnal *Developer Economics – State of the Developer Nation* mengungkapkan, sudah 69% pengembang *machine learning* dan *data scientist* aktif memakai Python pada tahun 2018. Bahkan, laporan IEEE Spectrum tahun 2019 menyatakan bahasa pemrograman Python menjadi bahasa pemrograman paling populer di dunia [23].

2.12 *Google Colab*

Google Colaboratory atau *Google Colab* adalah *tools* yag dikembangkan oleh *Google. Tools* ini memberikan fasilitas untuk mengolah data menggunakan teknik *machine learning* maupun *deep learning*, namun *tools* ini memiliki keterbatasan perangkat untuk melakukan komputasi. *Google Colab* menyediakan layanan GPU gratis sebagai *backend* komputasi yang dapat digunakan selama 12 jam.

Google Colab dibuat diatas *environment Jupyter* sehingga mirip dengan *Jupyter Notebook.* Penggunaanya pun hampir sama dengan *Jupyter Notebook* hanya saja berbeda dalam hal media penyimpanan. Media penyimpanan pada *Google Colab* menyediakan *runtime Python* 2 dan 3 yang telah dikonfigurasikan sebelumnya dengan berbagai *library*, seperti *TensorFlow, Matplotlib,* dan Keras [24].

2.13 *Term Frequency – Inverse Document Frequency* (TF-IDF)

Tujuan pembobotan kata adalah untuk menurunkan bobot yang layak untuk setiap kata. Mengerjakan bobot ini membutuhkan dua hal, yaitu *Term Recurrence* (TF) dan *Backwards Archive Recurrence* (IDF). *Term Recurrence* adalah jumlah kata atau istilah tertentu dalam suatu record. Sedangkan *Backwards Archive Recurrence* adalah pengulangan kejadian kata atau istilah dalam keseluruhan *record*. Penilaian IDF berbanding terbalik dengan jumlah catatan yang berisi istilah tertentu. Istilah yang muncul secara tidak teratur di semua laporan memiliki nilai IDF yang lebih menonjol daripada nilai IDF dari istilah yang sering muncul. Jika setiap *record* berisi istilah tertentu, maka nilai ekspresi IDF adalah 0 [25].

Rumus *TF-IDF* adalah sebagai berikut :

$$W_{dt} = tf_{dt} x idf_t = tf_{dt}\ x\ \log\left(\frac{N}{dft}\right) \quad \text{.....} \quad (2.8)$$

W_{dj} = Bobot *Term* ke – t dalam dokumen d

tf_d = Banyaknya kemunculan *Term* t didokumen – d

N = Banyaknya dokumen dalam keseluruhan

df_t = Banyaknya dokumen yang berisi *Term* t

2.14 *Confusion Matrix*

Confusion Matrix adalah matriks 2x2 yang merepresentasikan hasil klasifikasi biner dalam suatu kumpulan data. Ada rumus sederhana yang dapat

digunakan untuk menghitung kinerja klasifikasi. Hasil dari akurasi, presisi dan penarikan dapat ditampilkan dalam bentuk persentase [26].

1. Akurasi adalah jumlah prediksi yang benar. Rumus untuk menghitung akurasi terlihat pada persamaan di bawah ini.

$$Accuracy = \frac{TP+TN}{TP+FP+FN} \quad (2.9)$$

2. Presisi adalah jumlah dokumen teks yang dapat diverifikasi dari semua dokumen teks dalam koleksi. Rumus Presisi dapat dilihat dari persamaan di bawah ini.

$$Precision = \frac{TP}{TP+FP} \quad (2.10)$$

3. *Recall* adalah rasio prediksi positif sejati dengan jumlah total prediksi positif. Rumus *Recall* dapat dilihat dari persamaan di bawah ini.

$$Recall = \frac{TP}{TP+FN} \quad (2.11)$$

TP = *True Positive* merupakan data positif yang terdeteksi benar.

FP = *False Positive* merupakan data negatif yang terbaca positif.

TN = *True Negative* merupakan data negatif yang terdeteksi benar.

FN = *False Negative* merupakan data positif yang terbaca negatif.

2.15 *Black Box Testing*

Black Box Testing merupakan pengujian kualitas perangkat lunak yang berfokus pada fungsionalitas perangkat lunak. Pengujian *black box testing* bertujuan untuk menemukan fungsi yang tidak benar, kesalahan anatarmuka,

kesalahan pada struktur data, kesalahan performansi, kesalahan inisialisasi dan terminasi [27].

2.16 Pengujain *User Acceptance Test* (UAT)

Pengujian *User Acceptence Test* (UAT) adalah tahap percobaan terakhir dan yang paling penting pada empat tahapan *testing* perangkat lunak yang pada umumnya dilakukan. Pada tahapan tersebut, sistem pengujian dimanfaatkan agar memilih sebuah sistem yang bisa memenuhi kemauan pengguna dan juga bisa dukungan pada semua konsep bisnis atau penggunanya. UAT ini biasanya dilakukan *costumer* dan pengguna [28].

2.17 Penelitian Terkait

Tabel 2.1 Penelitian Terkait

No	Penulisan dan Tahun	Judul	Metode	Hasil
1.	Fitri dan Evita (2017).	Analisis Sentimen Terhadap Aplikasi Ruang Guru Menggunaka n Algoritma *Naïve Bayes*, *Random Forest* dan *Support*	-*Naïve Bayes* -*Random Forest* -*Support Vector Machine*	Review dari pengguna salah satu aplikasi sangat membantu pengembangan dalam meningkatkan kualitas aplikasi dan dapat menjadi sarana penilaian apakah pengguna merasa puas atau tidak. Penelitian melakukan analisis sentimen pada aplikasi Ruangguru dengan menguji tiga model klasifikasi yaitu NB,

		Vector Machine.		*Random Forest* dan SVM. Penelitian ini memberikan hasil bahwa dari model klasifikasi *Random Forest* 97,16% dengan menggunakan *Cross Validation* dan skor AUC 0,996. Kemudian akurasi dengan model support klasifikasi *Support Vector Machine* menghasilkan tingkat akurasi sebesar 96,01% dengan nilai AUC sebesar 0,543 dan akurasi pada pengujian model klasifikasi *Naive Bayes* sebesar 94,16% dari nilai AUC 0,999. Penelitian ini menunjukkan peningkatan akurasi dari penelitian sebelumnya sebesar 7,16% dengan model klasifikasi *Random Forest's final cut* sebagai model klasifikasi *Random Forest* dengan performansi terbaik.
No	**Penulis dan Tahun**	**Judul**	**Metode**	**Hasil**
2.	Herlianti, Nuraeni, Yuliani, Yuri,	Analisis Sentimen *Zoom Cloud Meetings* di *Play Store*	*-Naïve Bayes -Support Vector*	Dalam ulasan ini, peneliti menggunakan metode *Naive Bayes* dan *Support Vector Machine* dalam sentimen

	Gata, Windu, Samudi, (2014).	Menggunakan *Naïve Bayes* dan *Support Vector Machine.*	*Machine*	analisis ulasan positif atau negatif pada pengguna aplikasi *Zoom* di *Google Play Store.* Penilaian model menggunakan 10 *fold cross validasi* yang didapatkan nilai presisi dan nilai AUC dari setiap perhitungan, khusus untuk NB nilai ketepatan = 74,37% dan nilai AUC = 0,659.
No	**Penulis dan Tahun**	**Judul**	**Metode**	**Hasil**
3.	Ranjan, Sakshi Mishra, Subhankar (2020).	*Comperative Sentiment Analysis of App Reviews* 1-10.	*Machine Learning*	Penelitian ini bertujuan untuk melakukan klasifikasi sentimen ulasan aplikasi dan mengidentifikasi perilaku mahasiswa terhadap pasar aplikasi. Penelitian ini menerapkan algoritma *Machine Learning* menggunakan skema representasi teks TF-IDF dan kinerjanya dievaluasi pada metode pembelajaran. Pasar aplikasi Google menangkap pemikiran pengguna melalui peringkat dan ulasan teks.

				Berdasarkan ulasan Google dan diuji pada ulasan siswa. Analisis sentimen, dengan data latih dan data uji menghasilkan Algoritma SVM dengan akurasi maksimum (93.37%), algoritma LR (87.80%) dan NB (85.5%).
No	**Penulis dan Tahun**	**Judul**	**Metode**	**Hasil**
4.	Mustopa, Ali Hermanto, Anna Pratama, Eri Bayu Hendini, Ade Risdiansyah, Deni (2020).	*Analysis of User Reviews for the PeduliLindungi Application on Google Play Store Using the Support Vector Machine and Naïve Bayes Algorithm bases on Particle Swarm Optimization.*	*-Naïve Bayes -Support Vector Machine*	*Corona Virus 19* (*COVID-19*) merupakan infeksi virus menular yang kini telah menyebar ke berbagai negara, salah satunya Indonesia. Pemantauan penyebaran *COVID-19* di Indonesia ditangani langsung oleh Pemerintah Indonesia khususnya oleh Kementerian Komunikasi dan Informatika (KOMINFO) dengan pembuatan aplikasi *Protected* yang terdapat di *Google Play*. Pengguna memberikan *review* atau komentar tentang aplikasi, tentunya pengguna akan memilih aplikasi yang memiliki *review* bagus. Namun, memantau ulasan dari

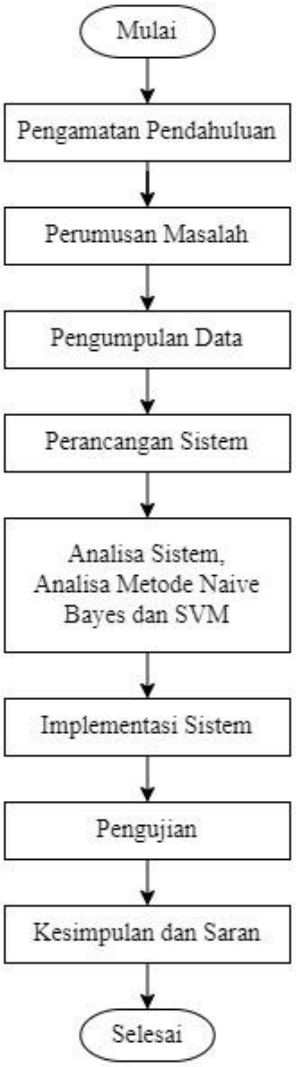
				masyarakat umum tidaklah mudah, karena banyak sekali yang harus diproses. Sehingga peneliti ingin mengetahui sejauh mana analisis *review* pengguna aplikasi PeduliLindungi berdasarkan *review* komentar pengguna dengan menggunakan teknik klasifikasi yaitu Algoritma *Support Vector Machine* (SVM) dan *Naive Bayes Based on Particle Swarm Optimization* (PSO) . Hasil pengujian dengan nilai akurasi dan nilai AUC masing-masing yaitu untuk algoritma *Naive Bayes* berbasis PSO nilai akurasi adalah 69,00%, nilai AUC adalah 0,659, sedangkan untuk algoritma SVM nilai akurasi adalah 93,0% dan nilai AUC adalah 0,977. Untuk itu, aplikasi pada penelitian yang memiliki akurasi lebih tinggi adalah SVM sehingga memberikan pemecahan masalah dalam analisis sentimen *review* komentar pengguna aplikasi PeduliLindungi.

No	Penulis Dan Tahun	Judul	Metode	Hasil
5.	Winda, Putri Anggraini, Manda Syari Utami, Juliafatin Malinda Berlianty, Elvira Sellya Hutagalung, Yogi Juniarto, Rani Nooraeni (2020).	Klasifikasi Sentimen Masyarakat Terhadap Kebijakan Kartu Prakerja di Indonesia.	*Naive Bayes*	Berdasarkan pengujian, didapatkan nilai akurasi sebesar 91.06% dari keseluruhan tweet yang diuji. Selain itu, nilai presisi yang menunjukkan perbandingan jumlah data benar bernilai positif dengan hasil jumlah data benar bernilai positif dan data salah bernilai positif memiliki nilai sebesar 90.35%. Sehingga dapat disimpulkan bahwa metode *Naïve Bayes* sudah cukup baik dalam melakukan klasifikasi.

BAB 3

METODOLOGI PENELITIAN

Penelitian ini dilakukan dengan melaksanakan tahapan demi tahapan yang berhubungan. Tahapan-tahapan tersebut dijabarkan dalam metode penelitian. Metode penelitian diuraikan kedalam bentuk skema yang jelas, teratur, dan sistematis. Berikut tahapan-tahapan penelitian dapat dilihat pada gambar 3.1



Gambar 3.1 Metodologi Penelitian

Penjelasan dari tahapan-tahapan penelitian pada gambar 3.1 dapat dilihat pada penjelasan dibawah ini:

3.1 Pengamatan Pendahuluan

Pengamatan pendahuluan merupakan tahapan awal yang dilakukan dalam penelitian ini, yang menggunakan metode *Naïve Bayes* dan *Support Vector Machine* (SVM) yang dijadikan sebagai penelitian studi pustaka dalam penelitian Tugas Akhir ini. Maka dari itu penulis melakukan penelitian terkait judul tersebut dengan menggunakan metode *Naïve Bayes* dan *Support Vector Machine*.

3.2 Perumusan Masalah

Berdasarkan hasil dari tahapan pengamatan pendahuluan sebelumnya, maka tahapan selanjutnya adalah perumusan masalah. Pada tahapan perumusan masalah akan dirumuskan masalah yang dianggap sebagai penelitian dalam Tugas Akhir ini. Permasalahan-permasalahan yang menjadi rumusan masalah dalam penelitian ini didapatkan dari penelitian-penelitian terkait data pengamatan pendahuluan sebelumnya. Solusi yang didapatkan pada tahapan perumusan masalah ini yang akan menjadi judul penelitian Tugas Akhir ini "Analisis Sentimen Aplikasi PeduliLindungi Pada *Play Store* Menggunakan Metode *Naïve Bayes* dan *Support Vector Machine*".

3.3 Pengumpulan Data

Pengumpulan data adalah tahapan-tahapan yang bertujuan memperoleh data-data informasi yang berhubungan dengan penelitian Tugas Akhir ini. Pada tahapan pengumpulan data ini juga berguna untuk mengumpulkan semua

kebutuhan data yang akan diproses nantinya menggunakan metode *Naive Bayes* dan *Support Vector Machine*, yaitu diantaranya :

1. Pengumpulan informasi mengenai aplikasi PeduliLindungi.
2. Pengumpulan informasi terkait metode *Naïve Bayes* dan *Support Vector Machine.*
3. Pengumpulan data dari jurnal dan buku-buku yang terkait dengan judul penelitian.

3.4 Perancangan Sistem

Pada tahapan implementasi dibutuhkan beberapa komponen pendukung yang memiliki peran yang sangat penting dalam implementasi sistem diantaranya adalah perangkat keras (*hardware*) dan perangkat lunak (*software*) dengan spesifikasi sebagai berikut :

1. Perangkat Keras (*hardware*), antara lain :

   Processor : Intel(R) Core(TM) i3-4005U

   Memory (RAM) : 4.00 GB

   Operating System : Windows 10 Pro 64-bit

   Harddisk : 500 GB

2. Perangkat Lunak (*software*), antara lain :

   Bahasa Pemograman : Python

   Web Browser : *Google Chrome*

   Text Editor : *Google Colab & VS Code*

   Dataset Editor : *Microsoft Excel*

3.5 Pengujian

Pengujian merupakan sebuah tahapan yang memperlihatkan apakah tingkat akurasi dari penelitian sesua dengan yang diinginkan atau tidak. Dalam penelitian ini mempunyai tahapan yaitu :

1. *Data Scrapping* adalah pengumpulan data pada *Google Play Store.*
2. *Data Processing* adalah proses pembersihan data.
3. *Data Transformation* adalah proses perubahan pada data yang dipilih.
4. *Data Mining* adalah langkah untuk menggunakan metode atau algoritma tertentu seperti *Naïve Bayes* dan *Support Vector Machine.*

3.6 Kesimpulan dan Saran

Tahapan terakhir adalah menarik kesimpulan dari hasil penelitian yang didapatkan dalam mengklasifikasikan sentiment ulasan aplikasi PeduliLindungi. Pada tahapan ini juga berisikan saran peneliti bagi pembaca untuk melakukan pengembangan terhadap penelitian ini ke depannya.