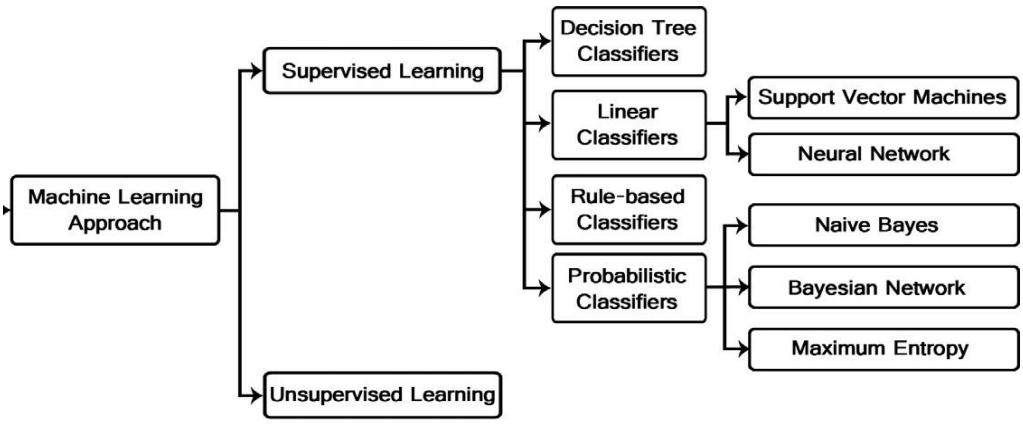
# BAB II
# TINJAUAN PUSTAKA

## 2.1 *Machine Learning*

*Machine learning* dapat didefinisikan sebagai aplikasi komputer dan algoritma matematika yang diadopsi dengan cara pembelajaran yang berasal dari data dan menghasilkan prediksi di masa yang akan dating (Booker dkk., 1989). Adapun proses pembelajaran yang dimaksud adalah suatu usaha dalam memperoleh kecerdasan yang melalui dua tahap antara lain latihan *(training)* dan pengujian *(testing)* (Huang dkk., 2006).

(Somvanshi dkk., 2016) Mengungkapkan bahwa *Machine Learning* terbagi menjadi dua kategori: *Supervised Learning* dan *Unsupervised Learning*. Skema *Machine Learning* dapat dilihat pada gambar 2.1:




Gambar 2.1 Skema Machine *Learning*

*Supervised Learning* adalah metode klasifikasi dimana kumpulan data sepenuhnya diberikan label untuk mengklasifikasikan kelas yang tidak dikenal (Riaddy dkk., 2016). *Supervised learning* dikelompokkan lebih lanjut dalam masalah klasifikasi dan regresi. Masalah klasifikasi adalah ketika variabel output berbentuk kategori, seperti merah atau biru atau penyakit dan tidak ada penyakit. Sedangkan masalah regresi adalah ketika variabel *output* adalah nilai riil, seperti

*dollar* atau berat (Brownlee, 2016). Dalam metode *Supervised Learning* terdapat model klasifikasi yaitu *Probabillistic classifiers*.

Tokoh yang pertama kali kemukakan metode ini adalah seorang ilmuwan dari Inggris bernama *Thomas Bayes*. Metode ini bermodel probabilitas dan statistik yang dikenal dengan penerapan *Teorema Bayes* (Busiarli dkk., 2016). Dalam metode *Naïve Bayes Classifiers* ini akan menghitung probabilitas setiap data kemudian diklasifikasikan nilai probabilitas tertinggi. Dengan struktur *Naïve Bayes Classifier* yang sederhana, metode ini memiliki waktu yang singkat untuk memproses dan tingkat akurasi yang tinggi serta mudah untuk diimplementasikan (Hadna dkk., 2016). Selain itu Teknik *Natural Language Processing* juga diperlukan dalam proses pengklasifikasian data yaitu pada tahap *preprocessing data*.

*Natural Language Processing* adalah sebuah bidang ilmu komputer yang mengembangkan pembelajaran bahasa alami dan linguistik komputasi dengan menggunakan kecerdasan buatan. Pada *NLP* informasi yang akan digunakan berisi data-data yang tidak terstruktur. Sehingga diperlukan sebuah proses pengubahan bentuk menjadi data yang terstruktur untuk kebutuhan penelitian (*sentiment analysis, topic modelling*, dan lain-lain). Dalam *Natural Language Processing* ada beberapa teknik *preprocessing* yaitu *Case Folding, Removal Punctuation, Removal Stopword* dan *Tokenization* (Pustejovsky & Stubbs, 2013).

Pembuatan sistem klasifikasi data dalam penelitian ini tentunya membutuhkan bahasa pemrograman yang cocok dengan algoritma yang digunakan oleh karena itu peneliti menggunakan bahasa pemrograman *Python*. *Python* adalah bahasa pemrograman yang tercipta pada Desember 1989 oleh *Guido Van* memiliki tingkat bahasa yang tinggi. *Python* merupakan bahasa pemrograman yang cocok untuk tujuan *machine learning*. *Python* juga disebut sebagai bahasa yang mudah untuk dipelajari karena memiliki tingkat bahasa yang tinggi, ini juga membantu *programmer* untuk menyingkat waktu untuk mempelajari bahasa pemrogramannya. *Python* sangat cocok untuk mengembangkan *Machine*

*Learning*, ini terbukti dengan banyaknya *library* yang tersedia dalam bahasa pemrograman ini (Ghimire, 2020).

### 2.2 Algoritma *Naïve Bayes*

Algoritma *Naïve Bayes* adalah salah satu algoritma klasifikasi *data mining* yang cukup dikenal dan sering digunakan di bidang kesehatan untuk memprediksi suatu penyakit. Algoritma ini yaitu menggunakan pengelompokkan berdasarkan probabilitas. Menurut pendapat *Bramer, Naïve Bayes* adalah metode tanpa aturan. *Naïve Bayes* menemukan potensi terbesar dari kemungkinan klasifikasi dengan memeriksa frekuensi setiap klasifikasi data pelatihan menggunakan bidang matematika yang dikenal sebagai teori probabilitas. *Naïve Bayes* adalah metode klasifikasi umum dan salah satu dari 10 algoritma teratas untuk *data mining*. Algoritma ini juga dikenal sebagai *Idiot's Bayes*, *Simple Bayes*, dan *Independence Bayes* (Webb, 2016). Klasifikasi *Bayes* didasarkan pada teorema *Bayes*, diambil dari nama seorang ahli matematika yang juga *monitoring Presbyterian Inggris*, *Thomas Bayes* (1702-1761), yaitu:

$$P(H \mid X) = (P(X \mid H) \times P(H))/P(X) \qquad 2.1$$

Keterangan:

*X* : Data dengan kelas yang belum diketahui

*H* : Hipotesis data merupakan suatu kelas spesifik

*P (H|X)* : Probabilitas hipotesis H berdasar kondisi X (posteriori probabilitas)

*P (H)* : Probabilitas hipotesis H (prior probabilitas)

*P (X|H)* : Probabilitas X berdasarkan kondisi pada hipotesis H

P (X) : Probabilitas X

Menurut (Xu dkk., 2017) terdapat tiga model klasifikasi dalam algoritma *Naïve Bayes* yaitu *Multinomial Naïve Bayes*, *Gaussian Naïve Bayes* dan *Bernoulli Naïve Bayes*, ketiga model ini biasanya digunakan dalam klasifikasi teks dan

dokumen, perbedaan antara ketiga model ini dapat dilihat pada penjelasan berikut ini:

### 2.2.1 *Multinomial Naïve Bayes*

*Multinomial Naïve Bayes* adalah sebuah algoritma klasifikasi teks yang didasarkan pada konsep probabilitas. Algoritma ini memproses teks dengan tidak memperhitungkan urutan kata dalam teks dan hanya mempertimbangkan jumlah kata yang muncul pada teks tersebut. Informasi yang terdapat pada dokumen atau kalimat juga dianggap sebagai faktor penentu klasifikasi. Dengan demikian, *Multinomial Naïve Bayes* dapat digunakan untuk mengklasifikasikan teks ke dalam kategori yang sesuai dengan informasi yang terkandung dalam teks (Ashari dkk., 2020). Model *Multinomial Naïve Bayes* menggunakan rumus sebagai berikut:

$$P(c|term\ dok\ d) = P(c) \times P(t1 \mid c) \times P(t2 \mid c) \times P(tn|c) \quad 2.2$$

Keterangan:

P(c|term dok d) : Probabilitas suatu dokumen dalam kelas c

P(c) : Probabilitas prior dari kelas c

P(tn|c) : Probabilitas kata ke-n pada kelas c

tn : kata ke n pada dokumen

### 2.2.2 *Gaussian Naïve Bayes*

Dalam klasifikasi teks, penggunaan fitur-fitur *numeric* kontinu dapat diwakili oleh distribusi *Gauss*. Distribusi *Gauss* dipilih untuk merepresentasikan probabilitas bersyarat dari fitur kontinu pada sebuah kelas, $P(Xi|C)$. Distribusi *Gauss* memiliki dua parameter, yaitu *mean* (rata-rata), $\mu$, dan variansi, $\sigma 2$. Untuk setiap kelas, $cj$, probabilitas bersyarat kelas $yj$. Untuk fitur $Fi$ didefinisikan menggunakan persamaan teorema *Naïve Bayes Gaussian* (Saraswati & Riminarsih,

2021). Dalam penggunaannya, metode ini digunakan untuk mengklasifikasikan teks ke dalam kategori yang sesuai dengan fitur-fitur *numeric* kontinu yang terdapat pada teks tersebut (Saleh, 2015). Berikut adalah rumus yang digunakan untuk model *Gaussian Naïve Bayes*:

$$P(Xi = xi|Y = yj) = \frac{1}{\sqrt{2\pi\sigma ij}} e^{-\frac{(Xi-\mu ij)^2}{2\sigma 2ij}} \quad 2.3$$

Keterangan:

Dimana:

P : Peluang

Xi : Atribut ke i

xi : Nilai atribut ke i

Y : Kelas yang dicari

μ : Mean, menyatakan rata-rata dari seluruh atribut.

σ : Standar deviasi, menyatakan varian dari seluruh atribut

### 2.2.3 *Bernoulli Naïve Bayes*

Algoritma *Bernoulli Naive Bayes* mengimplementasikan klasifikasi untuk data yang *dataset* sesuai dengan distribusi *Bernoulli multivariat*; yaitu, mungkin terdapat beberapa fitur tetapi masing-masing dianggap sebagai variabel bernilai biner (*Bernoulli*, *boolean*). Oleh karena itu, kelas ini membutuhkan sampel untuk direpresentasikan sebagai vektor fitur bernilai biner (Shofiya dkk., 2020). Berikut merupakan rumus yang digunakan untuk model *Bernoulli Naïve Bayes*:

$$P(xi \mid y) = P(i \mid y)\, xi + (1 - P(i \mid y))(1 - xi) \quad 2.4$$

Keterangan:

$(i \mid y)$ : Probabilitas kemunculan fitur $i$ pada kelas target $y$

$x\_i$ : Nilai fitur $i$ pada data yang sedang diproses (1 atau 0)

Dalam rumus ini, dianggap bahwa setiap fitur $xi$ pada sebuah instance data adalah independen secara kondisional terhadap fitur-fitur lainnya, sehingga dapat diterapkan model *Naïve Bayes*.

Rumus ini digunakan untuk menghitung probabilitas suatu instance data tertentu termasuk pada kelas target yang mana. Dalam aplikasinya, rumus ini digunakan untuk memprediksi label kelas dari sebuah *instance* data berdasarkan fitur-fiturnya dengan membandingkan probabilitas kelas yang dihasilkan dari rumus ini untuk setiap kelas target yang ada.

### *2.3 Term Frequency – Inverse Document Frequency (TFIDF)*

*Term frequency-inverse document frequency* adalah statistik *numeric* yang menunjukkan betapa pentingnya sebuah kata bagi sebuah dokumen dalam sebuah koleksi (Christian dkk., 2016). *TFIDF* terdiri dari dua persamaan *term frequency (TF)* dan *inverse document frequency (IDF).* Term Frequency merupakan istilah yang ditentukan oleh jumlah kemunculan sebuah kata dalam dokumen (Albitar dkk., 2014). Yang dapat dihitung dengan persamaan berikut:

$$TF = ft, d \quad (2.5)$$

Dimana $TF$ adalah frekuensi $(f)$ dari (t) di dalam dokumen $(d)$.

Rumus untuk menghitung *IDF* persamaan 2.6 dibawah ini:

$$IDF = log(N/|\{d \in D : t \in d\}|) \quad (2.6)$$

Dengan N jumlah total dokumen di dalam *Corpus* dan $(|\{ d \in D : t \in d\}|)$ menjadi jumlah dokumen $(d)$ dimana $(t)$ muncul.

$$TFIDF = ft, d * log(N/|\{d \in D : t \in d\}|) \quad (2.7)$$

### 2.4 *Confusion Matrix*

*Confusion Matrix* adalah tabel yang berisi rincian klasifikasi, kelas diprediksi ditampilkan di bagian atas *matrix* dan kelas yang diamati ditampilkan di bagian sebelah kiri evaluasi model (Fawcett, 2006).

*Confusion Matrix* menggunakan tabel 2.1 seperti *matrix* di bawah ini:

Tabel 2.1 Klasifikasi Matrix

| Klasifikasi | Kelas = *Yes* | Kelas = *No* |
|---|---|---|
| Kelas = *Yes* | *True Positive* (*TP*) | *Values Negative* (*FN*) |
| Kelas = *No* | *Values Positive* (*FP*) | *True Negative* (*TN*) |

Keterangan:

*TP* : Total kasus *Positive* yang dikategorikan sebagai *Positive*

*FP* : Total kasus *Negative* yang dikategorikan sebagai *Positive*

*TN* : Total kasus *Negative* yang dikategorikan sebagai *Negative*

*FN* : Total kasus *Positive* yang dikategorikan sebagai *Negative*

Berdasarkan nilai *True Negative* (*TN*), *Values Positive* (*FP*), *Values Negative* (*FN*), dan *True Positive* (*TP*) dapat diperoleh nilai akurasi, presisi dan *Recall* (Sokolova & Lapalme, 2009).

1. Nilai Akurasi

   Akurasi adalah ukuran seberapa baik suatu model dapat memprediksi kelas dengan benar. Akurasi didefinisikan sebagai rasio antara jumlah prediksi benar dan jumlah total prediksi. Untuk mendapatkan nilai akurasi dapat dilihat pada persamaan berikut:

$$Akurasi = \frac{(TP + TN)}{(TP + TN + FP + FN)} \quad 2.8$$

2. *Precision*

   *Precision* adalah ukuran seberapa baik model dapat mengidentifikasi kasus positif dengan benar dari semua kasus yang diprediksi positif. Secara ilmiah, *Precision* didefinisikan sebagai rasio antara jumlah kasus positif yang diprediksi dengan benar dan jumlah total kasus yang diprediksi positif. Nilai *Precision* dapat dihitung menggunakan persamaan berikut:

$$Precision = \frac{TP}{(TP + FP)} \quad 2.9$$

3. *Recall*

   *Recall* adalah ukuran seberapa baik model dapat mengidentifikasi semua kasus positif. Secara ilmiah, *Recall* didefinisikan sebagai rasio antara jumlah kasus positif yang diprediksi dengan benar dan jumlah total kasus positif. Nilai *Recall* dapat dihitung dengan menggunakan persamaan berikut:

$$Recall = \frac{TP}{(TP + FN)} \quad 2.10$$

Ketiga ukuran ini penting untuk mengevaluasi kinerja model klasifikasi dan dapat membantu dalam mengidentifikasi jenis kesalahan yang dilakukan oleh model. Kinerja model yang baik diukur dengan akurasi yang tinggi, *Precision* yang tinggi, dan *Recall* yang tinggi.

### 2.5 Kurva *ROC*

Kurva *ROC* adalah grafik digunakan untuk mengevaluasi hasil prediksi, kurva *ROC* adalah teknik untuk memvisualisasikan, mengatur, dan memilih pengklasifikasian berdasarkan kinerja mereka (Fawcett, 2006).

Kurva *ROC* adalah alat dua dimensi yang digunakan untuk mengevaluasi kinerja klasifikasi menggunakan dua kelas keputusan, di mana setiap objek ditugaskan ke salah satu anggota himpunan pasangan positif atau negatif. Pada kurva *ROC*, laju *TP* diplot pada sumbu Y dan laju *FP* diplot pada sumbu X. Untuk klasifikasi data mining, nilai *AUC* dapat dibagi menjadi beberapa kelompok seperti pada tabel 2.2 berikut:

Tabel 2.2 Kelompok *AUC* Score

| No | Nilai *AUC* | Kategori |
|---|---|---|
| 1 | 0.90-1.00 | *Excellent Classification* |
| 2 | 0.80-0.90 | *Good Classification* |
| 3 | 0.70-0.80 | *Fair Classification* |
| 4 | 0.60-0.70 | *Poor Classification* |
| 5 | 0.50-0.60 | *Failure* |

### 2.6 *Cross Validation*

*Cross Validation* adalah teknik data mining yang bertujuan untuk membagi data latih dan data uji dengan kelas yang seimbang untuk memperoleh hasil akurasi yang maksimal. Metode ini sering juga disebut dengan *10-fold cross validation* dimana percobaan sebanyak k kali untuk satu model dengan parameter yang sama (Berrar, 2018). Fungsinya dari penggunaan *cross validation* adalah:

1. Mengetahui performa dari suatu model algoritma dengan melakukan percobaan sebanyak k kali.
2. Meningkatkan tingkat performansi dari model tersebut.
3. Mengolah *dataset* menjadi kelas yang seimbang.

## 2.7 *Software* Yang Digunakan untuk Pembuatan Model Klasifikasi

### 2.7.1 *Anaconda Navigator*

*Anaconda Navigator* adalah sebuah aplikasi *desktop* yang digunakan untuk mengelola lingkungan pengembangan dan analisis data yang dibangun dengan menggunakan platform *Anaconda*. *Anaconda Navigator* menyediakan antarmuka grafis untuk menginstal, mengelola, dan menjalankan aplikasi dan lingkungan *Anaconda* yang berisi paket-paket perangkat lunak populer seperti *Python*, *R*, *Jupyter Notebook*, dan lainnya.

Dalam *Anaconda Navigator*, pengguna dapat dengan mudah membuat dan mengelola lingkungan *virtual* yang berbeda untuk mengembangkan dan menjalankan proyek-proyek yang berbeda dengan konfigurasi yang berbeda-beda. Selain itu, *Anaconda Navigator* juga menyediakan akses ke berbagai alat dan fitur analisis data yang populer seperti *Spyder*, *JupyterLab*, dan lainnya.

### 2.7.2 *Jupyter Notebook*

*Jupyter Notebook* adalah sebuah aplikasi *web open-source* yang digunakan untuk membuat dan membagikan dokumen yang berisi kode interaktif, visualisasi, dan teks naratif. *Jupyter Notebook* dapat digunakan dalam berbagai bahasa pemrograman seperti *Python*, *R*, *Julia*, dan lainnya. *Jupyter Notebook* memungkinkan pengguna untuk menulis kode dan menjalankannya secara interaktif, serta menambahkan catatan, teks, dan visualisasi untuk menjelaskan dan menggambarkan proses pemrograman yang dilakukan. *Jupyter Notebook* sangat populer digunakan oleh para ilmuwan data dan peneliti dalam melakukan eksplorasi data, analisis, dan pemodelan.

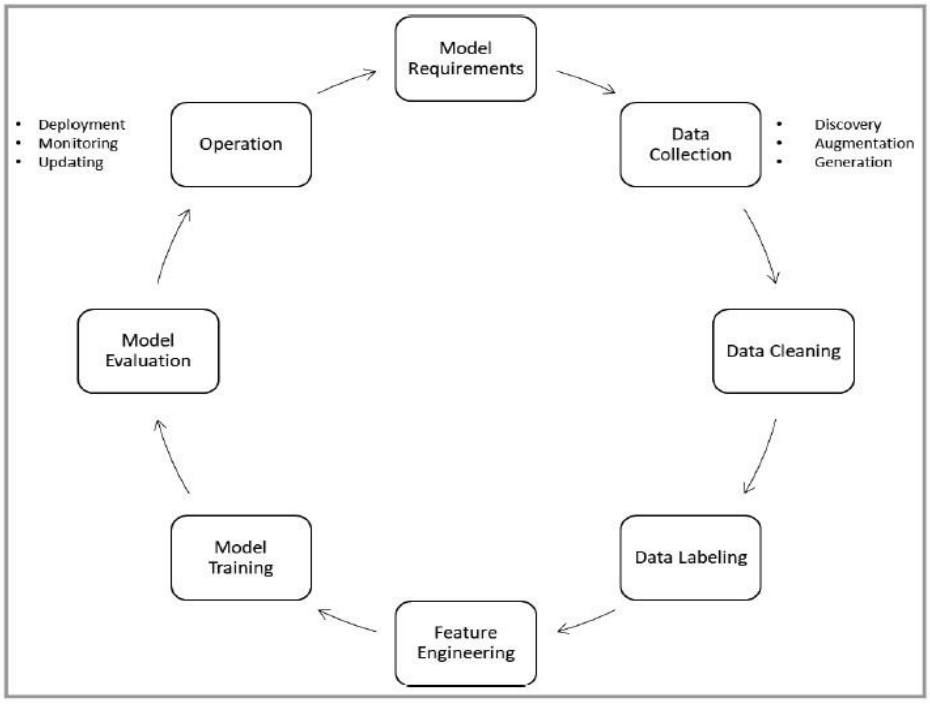
### 2.7.3 *Web Scraper*

*Web Scraper* adalah sebuah fitur yang dapat ditambahkan pada browser web seperti *Google Chrome* atau *Mozilla Firefox*, yang memungkinkan pengguna untuk mengambil data dari halaman *web* dan menyimpannya dalam format yang dapat digunakan dalam analisis data. *Web Scraper* bekerja dengan cara mengekstrak data dari halaman *web* berdasarkan elemen *HTML* dan *CSS* yang ditentukan oleh pengguna, kemudian menyimpan hasil ekstraksi data tersebut dalam *format CSV* atau *Excel*. *Web Scraper* ini berfungsi untuk mengambil dan mengumpulkan data-data *website* dan menjadikanya sebagai *dataset* untuk proses pemodelan *Machine Learning*.

## 2.8 Metode Pengembangan Model Klasifikasi

### 2.8.1 *Machine Learning Life Cycle*

Berikut adalah metode pembuatan model klasifikasi *Machine Learning* dengan menggunakan *Machine Learning life cycle*:




Gambar 2.2 *Machine Learning Life Cycle*

Pada tahapan pembuatan model klasifikasi *Machine Learning* menggunakan *Machine Learning Life Cycle* memiliki beberapa tahapan seperti yang disebutkan oleh (Gärtler dkk., 2021), tahapan-tahapan tersebut diantaranya adalah sebagai berikut:

1. *Model Requirements*

   Pada fase *model requirements* dilakukan pemilihan jenis data, model algoritma, pengukuran kinerja dan teknologi atau platform yang akan digunakan kemudian akan diimplementasikan ke dalam *Machine Learning*.
2. *Data Collection*

   Pada fase *data collection* ini yang dilakukan adalah mengumpulkan *dataset* yang akan digunakan untuk melakukan pengklasifikasian, dalam fase *data collection* ini terdapat tiga teknik dalam pengumpulan *data collection* yaitu, *data discovery* (data yang sudah ada di *repository platform online*), *data augmentation* (pengumpulan data tambahan yang dilakukan oleh peneliti ) dan *data generation* (pengolahan data dari *data discovery* dan *data augmentation* agar sampel data yang telah dikumpulkan lebih berkualitas dan menghindari terjadinya kehilangan data secara acak).
3. *Data Cleaning*

   *Data Cleaning* digunakan untuk memastikan kualitas dari sampel seperti akurasi kelengkapan, konsistensi, keunikan dan integritas. Pada *text processing*, *data cleaning* sangat diperlukan untuk standarisasi ukuran dari suatu *text* yang ada dalam dokumen *dataset*.
4. *Data Labeling*

   *Data labeling* adalah proses pelabelan data apakah data-data tersebut termasuk dalam kategori pornografi atau non-pornografi.

5. *Feature Engineering*

   *Feature engineering* adalah proses pemeriksaan *dataset* untuk meningkatkan fitur dari sistem yang digunakan yang nantinya hasil dari proses *feature engineering* ini akan direpresentasikan dalam bentuk *numeric* data atau data *matrix*.

6. *Model Training*

   *Model training* mencakup aktivitas untuk melatih dan memilih model *Machine Learning* kemudian menyesuaikan *hyperparameter* pada data yang sudah dikumpulkan, dibersihkan dan dilabelkan.

7. *Model Evaluation*

   *Model evaluation* Pada tahap ini akan dilakukan analisa untuk mendapatkan dan mengkonfirmasi apakah model *Machine Learning* yang telah dibuat sudah sesuai dengan tujuan ataukah belum berdasarkan hasil analisa dari pendapat ahli bahasa Indonesia.

8. *Operation*

   *Operation* merupakan tahap di mana model *machine learning* yang telah dikembangkan diimplementasikan ke dalam produksi dan mulai digunakan untuk memproses data secara *real-time*.

## 2.9 Penelitian Terkait

Tabel 2.3 Penelitian Terkait

| **Judul, Penulis dan Tahun** | **Jumlah dan Atribut *Dataset*** | **Algoritma** | **Preprocessing** | **Feature Selection** | **Validasi** | **Open Source *Dataset*** | **Akurasi** |
|---|---|---|---|---|---|---|---|
| Comparison between Multinomial and Bernoulli Naïve Bayes for Text Classification (Singh dkk., 2019) | 300 *dataset* berita dan dua atribut | Multinomial Naïve Bayes dan Bernoulli Naïve Bayes | Lower casing, Tokenization, Tokenization dan Stopwords Removal | Term Frequency | Confusion Matrix dan ROC | Tidak Disebutkan | Multinomial Naïve Bayes 73.4%<br>Bernoulli Naïve Bayes 69.15% |
| Perbandingan Kinerja Algoritma Multinomial | 200 *dataset* berita dan dua atribut | Bernoulli Naïve Bayes | Text Normalization, Case Folding, | Pembobotan TF-IDF | K-Fold Cross Validation | *Dataset* tidak bersifat | Multinomial Naïve Bayes sebesar 74%, |

| **Judul, Penulis dan Tahun** | **Jumlah dan Atribut *Dataset*** | **Algoritma** | **Preprocessing** | **Feature Selection** | **Validasi** | **Open Source *Dataset*** | **Akurasi** |
|---|---|---|---|---|---|---|---|
| Naïve Bayes (MNB), Multivariate Bernoulli Dan Rocchio Algorithm Dalam Klasifikasi Konten Berita Hoax Berbahasa Indonesia Pada Media Sosial (Ashari dkk., 2020) | | Dan Rocchio Algorithm | Tokenization, Filtering dan Stemming | | | open source (*Dataset* diambil melalui situs web turnbackhoax.id) | Multivariate Bernoulli sebesar 70% dan Rocchio sebesar 76% |

| **Judul, Penulis dan Tahun** | **Jumlah dan Atribut *Dataset*** | **Algoritma** | **Preprocessing** | **Feature Selection** | **Validasi** | **Open Source *Dataset*** | **Akurasi** |
|---|---|---|---|---|---|---|---|
| Penerapan Algoritma Naïve Bayes Untuk Klasifikasi Konten Berita Olahraga (Wibowo & Darmawan, 2020) | 100 *dataset* dan empat atribut | Naïve Bayes Classifier | Normalize Case, Tokenizing, Filtering dan Stemming | Tidak Disebutkan | Secara umum tidak disebutkan namun validasi dalam penelitian ini mengacu pada nilai akurasi | *Dataset* tidak bersifat open source (*Dataset* diambil melalui situs web https://www.bola.net/ ) | Rata-rata akurasi 69,27% |

| **Judul, Penulis dan Tahun** | **Jumlah dan Atribut *Dataset*** | **Algoritma** | **Preprocessing** | **Feature Selection** | **Validasi** | **Open Source *Dataset*** | **Akurasi** |
|---|---|---|---|---|---|---|---|
| Analisis Komparasi Algoritma Klasifikasi Data Mining Dalam Klasifikasi *Website* Phising (Putri & Wijayanto, 2022) | 1.353 *dataset* dan sepuluh atribut | Naïve Bayes, Decision Tree, Random Forest dan Support Vector Machine (SVM) | Tidak Disebutkan | Tidak Disebutkan | Confusion Matrix | *Dataset* tidak bersifat open (*dataset* diambil dari berbagai *website* dan untuk *dataset website* phising diambil | Naïve Bayes sebesar 82,31%, Decision Tree sebesar 85,77%, Random Forest sebesar 90,77%, Support Vector Machine sebasar 86,25%. |

| Judul, Penulis dan Tahun | Jumlah dan Atribut *Dataset* | Algoritma | Preprocessing | Feature Selection | Validasi | Open Source *Dataset* | Akurasi |
|---|---|---|---|---|---|---|---|
| | | | | | | dari *website* www.phishtank.com ) | |
| Klasifikasi Emosi Ulasan Aplikasi Traveloka Pada Google Play Menggunakan Naïve Bayes (Janah dkk., 2020) | 2.662 *dataset* ulasan dan empat atribut | Multinomial Naïve Bayes, Bernoulli Naïve Bayes, dan Gaussian Naïve Bayes | Emoticon Handling, Case folding, Data Cleansing, Tokenizing, stopword removal dan Stemming | Pembobotan TF – IDF | Confusion Matrix | *Dataset* tidak bersifat open source (*Dataset* diambil menggunakan library BeautifulS | Multinomial Naïve Bayes sebesar 86%, Bernoulli Naïve Bayes Sebesar 69%%, Gaussian Naïve Bayes Sebesar 44%. |

| Judul, Penulis dan Tahun | Jumlah dan Atribut *Dataset* | Algoritma | Preprocessing | Feature Selection | Validasi | Open Source *Dataset* | Akurasi |
|---|---|---|---|---|---|---|---|
| | | | | | | oup, requests dan chromedriver pada ulasan aplikasi "Traveloka" pada *website* Google Play) | |
| Klasifikasi Berita Hoax | 360 *dataset* konten | Naïve Bayes | Parsing, Tokenisasi, | Pembobotan TF – IDF | K-Fold Cross | *Dataset* tidak | Akurasi sebesar 85.28% |

| **Judul, Penulis dan Tahun** | **Jumlah dan Atribut *Dataset*** | **Algoritma** | **Preprocessing** | **Feature Selection** | **Validasi** | **Open Source *Dataset*** | **Akurasi** |
|---|---|---|---|---|---|---|---|
| Dengan Menggunakan Metode Naïve Bayes (Mustofa & Mahfudh, 2019) | berita dan dua atribut | | Filtering dan Stopword Removal dan Stemming | dan Cosine Similarity | Validation dan Confusion Matrix | bersifat open source (*Dataset* diambil melalui situs web turnbackhoax.id) | |
| Klasifikasi Sentimen Sara, Hoaks Dan Radikal Pada Postingan Media | 260 *dataset* tweet dan empat atribut | Multinomial Naïve Bayes | Data Integration, Case Folding, | Tidak Disebutkan | Confusion Matrix | *Dataset* tidak bersifat | Akurasi sebesar 99.62%. |

| **Judul, Penulis dan Tahun** | **Jumlah dan Atribut *Dataset*** | **Algoritma** | **Preprocessing** | **Feature Selection** | **Validasi** | **Open Source *Dataset*** | **Akurasi** |
|---|---|---|---|---|---|---|---|
| Sosial Menggunakan Algoritma Naive Bayes Multinomial Text (Purwiantono & Aditya, 2020) | | | Filtering dan Tokenizing | | | open source (*Dataset* diambil menggunakan web crawler API (Application Programming Interface) berbasis PHP pada | |

| Judul, Penulis dan Tahun | Jumlah dan Atribut *Dataset* | Algoritma | Preprocessing | Feature Selection | Validasi | Open Source *Dataset* | Akurasi |
|---|---|---|---|---|---|---|---|
| | | | | | | media sosial twitter) | |
| Classification of Public Complaint Data in SMS Complaint Using Naive Bayes Multinomial Method (Yance Nanlohy dkk., 2020) | 1.038 dokumen *dataset* dan enam atribut | Multinomial Naïve Bayes | Case Folding, Stop Removal, Word Removal Frequency, Spelling Correct, Tokenization dan Stemming | Number of Word, Number of Characters, Average word dan Numbers of word | Confusion Matrix | Tidak Disebutkan | Akurasi sebesar 91.39% |

| **Judul, Penulis dan Tahun** | **Jumlah dan Atribut *Dataset*** | **Algoritma** | **Preprocessing** | **Feature Selection** | **Validasi** | **Open Source *Dataset*** | **Akurasi** |
|---|---|---|---|---|---|---|---|
| Klasifikasi Berita Pada Akun Twitter Suara Surabaya Menggunakan Metode Naïve Bayes (Hayaza dkk., 2020) | 670 *dataset* tweet dan empat atribut | Multinomial Naive Bayes, BernoulliNaive Bayes dan Gaussian Naive Bayes | Konversi menjadi huruf kecil, Penghapusan karakter-karakter tertentu, Memisahkan kalimat menjadi kata berdasarkan spasi, Menghilangkan stopword, Menghapus | Pembobotan TF – IDF dan Cosine Similarity | Confusion Matrix | Data diambil pada media social twitter | Akurasi Multinomial Naive Bayes sebesar 89%. |

| Judul, Penulis dan Tahun | Jumlah dan Atribut *Dataset* | Algoritma | Preprocessing | Feature Selection | Validasi | Open Source *Dataset* | Akurasi |
|---|---|---|---|---|---|---|---|
| | | | imbuhan dan menjadikan kata ke bentuk kata dasarnya dan Menggabungkan dua kata yang memiliki satu arti, | | | | |
| Klasifikasi Artikel Berita Bahasa Indonesia Dengan Naive Bayes Classifier | 3.809 *dataset* dan lima atribut | Naïve Bayes Classifier | Stopwords dan Stemming | N-Gram | Tidak Disebutkan | Tidak Disebutkan | Naïve Bayes Classifier 94.7% dan Fitur N-Gram Unigram sebesar 0.947 dan Bigram sebesar 0.519. |

| Judul, Penulis dan Tahun | Jumlah dan Atribut *Dataset* | Algoritma | Preprocessing | Feature Selection | Validasi | Open Source *Dataset* | Akurasi |
|---|---|---|---|---|---|---|---|
| (Setiawan dkk., 2020) | | | | | | | |
| Klasifikasi Berita Kriminal Menggunakan Algoritma Naïve Bayes Berbasis PSO (Dzaffa 'Ulhaq dkk., 2022) | 120 *dataset* dan dua atribut | Naïve Bayes | Case Folding, Tokenizing, Stopword Removal dan Stemming | Particle Swarm Optimization | Secara umum tidak disebutkan namun validasi dalam penelitian ini mengacu pada nilai akurasi | *Dataset* tidak bersifat open source (*Dataset* diambil dari https://jpnn.com/kriminal) | Akurasi metode Naïve Bayes sebesar 81.67%. dan setelah penambahan PSO menjadi 93.33% |

| **Judul, Penulis dan Tahun** | **Jumlah dan Atribut *Dataset*** | **Algoritma** | **Preprocessing** | **Feature Selection** | **Validasi** | **Open Source *Dataset*** | **Akurasi** |
|---|---|---|---|---|---|---|---|
| Klasifikasi Karya Ilmiah (Tugas Akhir) Mahasiswa Menggunakan Metode Naive Bayes Classifier (NBC) (Nurdin dkk., 2021) | 170 *dataset* dan lima atribut | Naïve Bayes | Case Folding, Tokenizing dan Stopword Removal | Tidak Disebutkan | Secara umum tidak disebutkan namun validasi dalam penelitian ini mengacu pada nilai akurasi | *Dataset* tidak bersifat open source | Akurasi Naïve Bayes Classifier sebesar 86.68%. |

| **Judul, Penulis dan Tahun** | **Jumlah dan Atribut *Dataset*** | **Algoritma** | **Preprocessing** | **Feature Selection** | **Validasi** | **Open Source *Dataset*** | **Akurasi** |
|---|---|---|---|---|---|---|---|
| Perbandingan Klasifikasi Berita Hoax Kategori Kesehatan Menggunakan Naïve Bayes dan Multinomial Naïve Bayes (Harahap dkk., 2021) | 200 *dataset* dan empat atribut | Gaussian Naïve Bayes dan Multinomial Naïve Bayes | Case folding, Tokenizing, Stop forward removal dan Stemming | Pembobotan TF – IDF dan MinMax Scaler | K-Fold Cross Validation | *Dataset* tidak bersifat open source (*dataset* hoax diambil dari *website* urnbackhoax.id dan untuk *dataset* | Akurasi Gaussian Naïve Bayes sebesar 83.3% dan akurasi Multinomial Naïve Bayes sebesar 90% |

| Judul, Penulis dan Tahun | Jumlah dan Atribut *Dataset* | Algoritma | Preprocessing | Feature Selection | Validasi | Open Source *Dataset* | Akurasi |
|---|---|---|---|---|---|---|---|
| | | | | | | non-hoax diambil dari CNN, CNBC, Health.detik, dan health.kompas) | |
| Klasifikasi Cerita Pendek Berbahasa Bali Berdasarkan Umur Pembaca dengan Metode | 90 *dataset* dan tiga atribut | Naïve Bayes | Case Folding, Tokenisasi, Filtering Dan Stemming | Pembobotan TF – IDF | K-Fold Cross Validation dan Confusion Matrix | Tidak Disebutkan | Akurasi model sebesar 72%, precision 72%, recall 78% dan F1-score 73% |

| **Judul, Penulis dan Tahun** | **Jumlah dan Atribut *Dataset*** | **Algoritma** | **Preprocessing** | **Feature Selection** | **Validasi** | **Open Source *Dataset*** | **Akurasi** |
|---|---|---|---|---|---|---|---|
| Naive Bayes (Ristiari dkk., 2022) | | | | | | | |
| BBC News Data Classification Using Naïve Bayes Based On Bag Of Word (Salman & Obaida, 2021) | 2.225 *dataset* dan lima atribut | Multinomial Naïve Bayes, Bernoulli Naïve Bayes, Complement Naïve Bayes dan Gaussian Naïve Bayes | Tokenization, Remove unwanted characters, Normalization, Stop-words removal dan Stemming | Pembobotan TF – IDF | Confusion Matrix dan ROC | *Dataset* tidak bersifat open source (*dataset* diambil dari *website* BBC News ) | MultinomialNB sebesar 95.209%, BernoulliNB sebesar 92.365%, ComplementNB sebesar 97.604% dan GaussianNB sebesar 90.718%. |

Berdasarkan tabel di atas maka didapatkan poin-poin utama sebagai berikut:

1. Untuk meningkatkan akurasi model algoritma, diperlukan peningkatan ukuran kumpulan data dengan tujuan untuk memprediksi label kelas dengan benar dan akurasi yang lebih tinggi.
2. Semakin banyak data yang digunakan pada penelitian, semakin optimal hasil yang didapat.
3. Memberikan penambahan pada preprocessing dan mengumpulkan lebih banyak data agar dapat mengurangi ketimpangan dan melakukan pengukuran yang lebih akurat dari berbagai teknik evaluasi.

Berdasarkan poin-poin tersebut, terdapat usulan untuk meningkatkan keterbaruan dan optimalisasi hasil dengan beberapa cara, seperti meningkatkan ukuran kumpulan data, melakukan penambahan preprocessing, mengumpulkan lebih banyak *dataset*, dan menggunakan metode-metode alternatif seperti Stemmer. Dalam penelitian klasifikasi, hasil yang lebih akurat dapat diperoleh melalui pengumpulan *dataset* yang lebih beragam dengan mempertimbangkan konten dokumen yang spesifik. Menggunakan bahasa Indonesia yang benar dengan melakukan stemming atau pencarian kata pada bentuk kata dasar khususnya dapat membantu dalam ekstraksi fitur.

Penelitian yang dilakukan memiliki beberapa kebaruan dibandingkan dengan penelitian-penelitian terdahulu. Pertama, penelitian ini memfokuskan pada klasifikasi konten *website* atau *blog*, yang merupakan topik yang berbeda dengan topik yang sebagian besar yang telah diteliti dalam penelitian sebelumnya. Kedua, penelitian ini mengidentifikasi apakah sebuah halaman *web* mengandung unsur pornografi atau tidak, yang merupakan topik yang cukup sensitif dan penting dalam konteks pengawasan internet di Indonesia. Ketiga, meskipun telah banyak penelitian tentang klasifikasi teks, namun penelitian dalam dalam mengklasifikasikan konten *website* yang mengandung unsur pornografi dan non-pornografi masih terbatas, sehingga penelitian ini dapat memberikan wawasan

baru dan bermanfaat dalam pengembangan metode klasifikasi konten *website* khususnya untuk mengklasifikasikan konten yang mengandung unsur pornografi. Keempat, kontribusi yang diberikan dari penelitian yang dilakukan ini adalah dengan menambahkan beberapa teknik pada proses data cleaning dimana, pada penelitian terkait yang disebutkan dalam tabel 2.3 di atas hanya menggunakan empat sampai lima enam teknik cleaning dan dalam penelitian yang telah dilakukan ini menambah beberapa teknik cleaning sehingga penelitian ini menggunakan sepuluh teknik data cleaning, feature selection atau dalam penelitian ini disebut dengan feature engineering dimana, penelitian ini menambahakan teknik *SMOTE* sehingga *data training* menjadi seimbang dan menambahkan nilai parameter untuk model klasifikasi serta mengevaluasi dan memvalidasi model dengan berbagai teknik seperti kurva *ROC, confusion matrix, learning curve, validation curve, overfitting* dan *underfitting, akurasi, precision, recall, f1-score, cross validation* dan pendapat ahli bahasa. Kelima, metodologi penelitian yang digunakan dalam penelitian ini berbeda dengan penelitian-penelitian yang telah dilakukan seperti pada tabel 2.3 di atas Terakhir, pada penelitian ini ukuran dari *dataset* lebih besar dan beragam, dimana *dataset* yang digunakan dalam penelitian ini diambil dari berbagai situs *website* dan *blog* yang mana hasil akhir dari *dataset* yang digunakan dalam penelitian ini berjumlah 49.702 ribu data *website* dan *blog* yang digunakan sebagai *dataset* dalam penelitian ini.