# BAB 2
# LANDASAN TEORI

Telaah literatur merupakan pendukung peneliti untuk mengenal dan memahami teori-teori terkait penelitian lebih dalam. Literatur tersebut antara lain adalah Vaksin COVID-19, *Labelling*, Analisis Sentimen, *Text Pre-processing*, *Term Frequency - Inverse Document Frequency* (TF-IDF), algoritma *Multinomial Naïve Bayes*, *Confusion Matrix*, dan *K-Fold Cross Validation*.

## 2.1. Vaksin COVID-19

COVID-19 adalah sebuah virus *coronavirus* jenis baru (SARS-CoV-2) yang menggemparkan dunia [9]. Pada tanggal 13 Januari 2021, Indonesia sudah menjalankan program vaksinasi COVID-19 gratis dengan Presiden Joko Widodo (Jokowi) sebagai orang pertama yang mendapat suntikan vaksin [10]. Dilansir dari infografis yang dibuat oleh Y.Nurhanisah, dinyatakan bahwa terdapat sepuluh jenis vaksin COVID-19 yang digunakan di Indonesia, yaitu [11]:

1. Sinovac (CoronaVac)

   *Sinovac* atau yang juga dikenal sebagai *CoronaVac* merupakan vaksin yang dikembangkan oleh perusahaan farmasi China. Vaksin jenis ini juga merupakan vaksin yang pertama yang Indonesia dapatkan. Jumlah dosis yang diberikan adalah 2 x (0.5 ml/dosis) dengan jeda pemberian dosis adalah 28 hari. Vaksin ini dibuat dengan memanfaatkan virus penyebab COVID-19 yang sudah dimatikan.

2. Novavax

   *Novavax* merupakan jenis vaksin yang dibuat oleh perusahaan di Amerika Serikat dan berbasis protein. *Novavax* dibuat dengan menggunakan potongan protein yang tidak berbahaya tapi memiliki sifat yang dapat meniru virus COVID-19. Jumlah dosis yang diberikan sama seperti *Sinovac*, yaitu 2 x (0.5 ml/dosis) dengan jeda pemberian dosis adalah 21 hari. Penggunaan vaksin jenis *Novavax* di Indonesia ini tidak sebanyak *Sinovac* maupun *AstraZeneca*.

3. AstraZeneca-Oxford

Vaksin *AstraZeneca-Oxford* merupakan vaksin yang dikembangkan di UK (*United Kingdom*) oleh Universitas *Oxford* dan perusahaan Inggris-Swedia bernamakan *AstraZeneca*. Vaksin ini memanfaatkan virus hasil modifikasi untuk membentuk antibodi. Jumlah dosis vaksin *AstraZeneca-Oxford* ini diberikan dengan jumlah 2 x (0.5 ml/dosis) dengan jeda waktu pemberian dosis yaitu 12 minggu. Jeda waktu pemberian dosis pada vaksin jenis ini merupakan jeda yang paling lama dibandingkan dengan vaksin jenis lainnya.

4. Pfizer-BioNTech

   Vaksin *Pfizer-BioNTech* adalah vaksin COVID-19 yang dikembangkan oleh perusahaan bioteknologi Jerman dengan nama *BioNTech*, dan perusahaan farmasi Amerika dengan nama *Pfizer*. Oleh karena itu, vaksin COVID-19 diberikan nama *Pfizer-BioNTech*. Vaksin jenis ini memiliki efek perlindungan terhadap COVID-19 sebesar 95% dan dibuat dengan memanfaatkan potongan protein *spike* yang ada pada luar permukaan virus COVID-19. Metode ini disebut sebagai mRNA (*messenger RNA*). Jumlah dosis yang diberikan adalah 2 x (0.3 ml/dosis) dengan jeda waktu pemberian dosis adalah 21-28 hari.

5. Sinopharm

   Vaksin *Sinopharm* memiliki tingkat keefektifan hingga 70%. Vaksin jenis ini memiliki ciri yang kurang lebih sama seperti vaksin *Sinovac*. Kedua vaksin sama-sama memanfaatkan virus yang telah dimatikan. Jumlah dosis yang diberikan pun juga sama seperti *Sinovac*, tetapi dengan jeda pemberian dosis adalah 21 hari.

6. Moderna

   Vaksin Moderna telah dikembangkan sejak Januari 2020 oleh *Moderna and Vaccine Research at the National of Allergy and Infectious Disease* (NIAID) di Amerika. Sama seperti vaksin *Pfizer*, vaksin ini juga termasuk vaksin mRNA. Dosis yang diberikan berjumlah 2 x (0.5 ml/dosis) dengan jeda waktu pemberian dosis yaitu 28 hari.

7. Sputnik-V

   Vaksin *Sputnik-V* adalah vaksin yang dikembangkan oleh *Gamaleya Research Institute* di Rusia dengan menggunakan bahan dasar *adenovirus* 26 dan *adenovirus* 5 sebagai vektor protein virus COVID-19. Vaksin ini juga

dikenal sebagai *Gam-COVID-Vac*. Setelah uji coba tahap akhir, vaksin ini memiliki tingkat keefektifan untuk mencegah COVID hingga 91,6%. Dosis yang diberikan adalah 2 x (0.5 ml/dosis) dengan jeda waktu pemberian dosis adalah tiga minggu.

8. Janssen

   Vaksin *Janssen* sama seperti vaksin *Sputnik-V* yang menggunakan bahan dasar adenovirus 26 sebagai vektor protein virus COVID-19. Vaksin ini dikembangkan oleh *Janssen Pharmaceuticals Companies of Johnson & Johnson* sehingga vaksin ini juga dapat dikenal sebagai vaksin J&J. Vaksin jenis Janssen cukup diberikan satu kali dosis saja dengan jumlah 0.5 ml/dosis.

9. Convidencia

   Vaksin *Convidencia* adalah vaksin yang dikembangkan oleh *CanSino Biological Inc* di China. Vaksin ini diberikan Izin Penggunaan Darurat bersamaan dengan vaksin Janssen. Dibuat juga dengan memanfaatkan *adenovirus* 5 sebagai vektornya. Untuk jumlah dosis yang diberikan sama seperti Vaksin Janssen, yaitu cukup satu kali saja dengan jumlah 0.5 ml/dosis.

10. Zifivax

    Yang terakhir adalah vaksin *Zifivax*. Vaksin ini dikembangkan oleh *Anhui Zhifei Longcom Biopharmaceutical* dengan menggunakan rekombinan protein sub-unit. Vaksin ini diberikan dengan jumlah dosis yaitu 3 x (0.5 ml/dosis) dan jeda waktu pemberian dosisnya adalah satu bulan. Vaksin *Zifivax* memiliki tingkat efikasi mencapai 81,71%.

Seluruh vaksin yang didapatkan di Indonesia berasal dari hasil kerja sama atau perjanjian dengan COVAX ataupun donasi dari negara-negara lain. COVAX sendiri ini adalah sebuah organisasi internasional yang dibentuk bersama oleh WHO (*Word Health Organization*), Gavi, dan CEPI (*Coalition for Epidemic Preparedness Innovations*). Organisasi ini memfasilitasi berbagai kepentingan untuk dosis vaksin COVID-19 yang dapat dibagikan ke berbagai negara anggota.

Berdasarkan pada data survei penerimaan vaksin COVID-19 di Indonesia yang dilakukan oleh Kementerian Kesehatan, ITAGI, UNICEF, dan WHO, sebelum vaksin diberlakukan, terdapat permintaan tinggi untuk informasi yang benar dan akurat seputar vaksinasi COVID-19. Informasi mengenai vaksin COVID-19 yang

kurang jelas dan akurat menyebabkan masyarakat Indonesia ragu untuk menerima vaksin ini.

Hingga 22 Mei 2022, jumlah vaksin COVID-19 yang telah diberlakukan di Indonesia sudah mencapai 72,33%, dengan rincian sebanyak 60,37% telah menerima dua kali vaksin (sepenuhnya telah vaksin) dan 11,96% yang masih hanya menerima vaksin sekali (vaksin sebagian) [12].

## 2.2. *Labelling*

*Labelling* adalah proses untuk memberikan label atau *tag* ke data mentah untuk menunjukkan jawaban yang ingin diprediksi pada model *machine learning* dan sebagai representatif. [13]. *Labelling* bekerja dengan cara memahami makna kalimat berdasarkan konteks yang dibicarakan, bukan penilaian kata per kata [14]. Terdapat beberapa cara untuk melakukan *labelling*, antara lain:

1. *Automated Labelling*

   *Automated labelling* dapat dibagi kembali menjadi dua macam, yaitu *Semi-Supervised Learning* dan *Transfer Learning*. *Semi-Supervised Learning* sendiri adalah *labelling* yang dilakukan dengan cara menggabungkan *supervised learning* dan *unsupervised learning*, atau dengan arti lain yaitu menggunakan sejumlah data kecil yang berlabel untuk dipelajari dan digunakan untuk melabeli sejumlah data besar. Sedangkan *Transfer Learning* adalah model *machine learning* yang telah dilatih sebelumnya untuk memberikan label pada sejumlah data.

2. *Manual Labelling*

   *Manual labelling* merupakan *labelling* yang melibatkan manusia secara langsung untuk memberikan label sehingga ini merupakan cara yang paling efektif mengingat manusia lebih baik dalam mengenali pola dalam kumpulan data. *Manual labelling* dibagi menjadi dua macam, yaitu *External Labelling* dan *Internal Labelling*. *External Labelling* adalah *labelling* yang dilakukan oleh tenaga kerja khusus di luar perusahaan, baik pekerja yang teroganisir maupun tidak. Sedangkan *Internal Labelling* adalah *labelling* yang dilakukan oleh seorang ahli dalam perusahaan.

## 2.3. Analisis Sentimen

Analisis sentimen adalah proses pemahaman, ekstrak, dan proses data tekstual secara otomatis untuk mendapatkan sentimen informasi yang terkandung dalam sebuah kalimat opini [15]. Analisis sentimen kerap dikenal dengan *Opinion Mining*. Tetapi kedua hal ini dikatakan memiliki perbedaan terkait apa yang dianalisis. *Opinion Mining* sendiri mengekstrak dan menganalisis opini orang-orang tentang suatu entitas, sedangkan analisis sentimen mengidentifikasi sentimen yang diungkapkan dalam teks lalu menganalisisnya [16]. Analisis sentimen telah menjadi pembelajaran yang makin popular pada beberapa tahun ini. Klasifikasi sentimen melibatkan polaritas sentimen dari kalimat yang telah disaring. Kalimat-kalimat ini dikategorikan sebagai netral, negatif, dan positif tergantung dari kasusnya [17]. Terdapat beberapa jenis analisis sentimen yang diketahui [18], yaitu:

1. *Fine-Grained Sentiment Analysis*

   Jenis analisis sentimen ini merupakan salah satu jenis yang paling sering digunakan, dimana analisis sentimen ini berfokus pada tingkat polaritas pendapat. Contohnya seperti positif, netral, dan negatif. Jenis ini jugalah yang digunakan pada penelitian ini.

2. *Intent Sentiment Analysis*

   *Intent Sentiment Analysis* merupakan jenis analisis sentimen yang memiliki fokus untuk mengidentifikasi dan mencari tahu lebih dalam pesan atau niat yang ada di balik sebuah teks. Misalnya, mengetahui apakah sebuah teks yang ada tersebut merupakan sebuah saran, pendapat, sindiran, atau sebagainya.

3. *Aspect - Based Sentiment Analysis*

   *Aspect - Based Sentiment Analysis* adalah analisis sentimen yang memiliki fokus terhadap elemen yang lebih spesifik dari suatu produk atau layanan.

## 2.4. *Text Pre-processing*

*Text Preprocessing* adalah suatu tahapan untuk mengubah bentuk data teks yang belum terstruktur menjadi data teks yang lebih terstruktur [19]. Tahap ini merupakan tahap penting yang diperlukan sebelum melakukan data akan diproses lebih lanjut ke *training* dan pengklasifikasian. Terdapat beberapa macam tahap yang dilakukan saat *text pre-processing*, antara lain [20]:

1. *Data Cleaning*

   *Data Cleaning* adalah proses pembersihan data teks dengan cara menghilangkan tanda baca, simbol, dan angka. Contohnya yang sering digunakan pada *Twitter* adalah seperti tanda titik (.), koma (,), tagar (#), target (@), dan lain-lainnya. Hal ini dilakukan dengan tujuan untuk mengurangi *noise*.

2. *Case Folding*

   *Case Folding* merupakan proses untuk menyeragamkan seluruh huruf yang ada di data teks, dapat berupa huruf besar (*uppercase*) semua atau huruf kecil (*lowercase*) semua. Misalnya yang paling sering terjadi adalah mengubah seluruh data teks yang ada menjadi dalam bentuk huruf kecil (*lowercase*).

3. *Normalization*

   Tahap *normalization* adalah tahap untuk melakukan normalisasi bahasa terhadap kata yang tidak baku di bahasa Indonesia. Tahap ini digunakan untuk mengembalikan kata tidak baku atau kata yang disingkat menjadi kata yang baku sesuai dengan aturan Kamus Besar Bahasa Indonesia (KBBI). Contohnya, kata ”tkut” akan dinormalisasi menjadi ”takut”.

4. *Stopword Removal*

   *Stopword Removal* merupakan tahap untuk menghapus kata-kata yang tidak memiliki arti penting atau tidak akan terlalu berpengaruh terhadap arti dari sebuah kalimat. Kata-kata tersebut contohnya seperti ”yang”, ”ke”, atau ”dengan”.

5. *Tokenization*

   *Tokenization* adalah tahap untuk memisahkan sebuah kalimat menjadi pecahan yang lebih kecil atau per kata. *Tokenization* akan membantu mempermudah proses *stemming* nantinya untuk mencari kata dasar.

6. *Stemming*

   *Stemming* adalah tahap untuk mencari kata dasar dari suatu kata yang ada tetapi hanya dengan cara memotong kata awalan dan/atau kata akhiran. Misalnya, terdapat kata ”memakan”, kata dasar dari ”memakan” tersebut adalah ”makan”, sehingga hasil dari *stemming* nanti akan berubah menjadi ”makan”. *Stemming* biasanya dilakukan ketika sudah melewati tahap *stopword removal* dan *tokenization*.

7. *Lemmatization*

   *Lemmatization* sama digunakan untuk mencari kata dasar tetapi cara yang digunakan berbeda. *Lemmatization* juga sebenarnya dapat memotong kata awalan dan/atau kata akhiran pada suatu kata, tetapi dari hasil kata yang telah dipotong tersebut juga akan mengembalikan kata dasar dari kata yang ada. Misalnya, jika terdapat kata "*studies*", maka kata dari hasil *stemming* adalah "*studi*". Sedangkan dengan menggunakan *lemmatization*, hasilnya akan kembali ke kata dasar yang seharusnya, yaitu "*study*".

## 2.5. *Term Frequency - Inverse Document Frequency* (TF-IDF)

*Term Frequency - Inverse Document Frequency* (TF-IDF) adalah salah satu metode untuk memberikan bobot nilai pada setiap kata yang ada. TF-IDF dimulai dengan menghitung banyaknya suatu *term* yang muncul dalam suatu dokumen (*term frequency*) kemudian menghitung kebalikannya, yaitu kemunculan *term* di seluruh dokumen [20]. Semakin jarang muncul suatu *term* pada seluruh dokumen, maka akan semakin besar nilai IDF *term* tersebut. Sedangkan, semakin sering muncul suatu *term* pada seluruh dokumen, maka nilai IDF *term* tersebut akan mendekati nilai 0. Hal ini dikarenakan suatu *term* yang sering muncul mengindikasikan bahwa *term* tersebut tidak memiliki pengaruh yang penting untuk membedakan kelas dari dokumen yang ada [19]. Untuk menghitung $tf$ dan $idf$ dapat menggunakan Persamaan 2.1 hingga Persamaan 2.3 [21].

$$tf(t,d) = \frac{n_{i,j}}{\sum_k n_{i,j}} \tag{2.1}$$

$$idf(t) = log(\frac{N}{df_t}) \tag{2.2}$$

$$w_{t,d} = tf(t,d) \times idf(t) \tag{2.3}$$

Dimana:

$tf(t,d)$ = frekuensi *term*
$n_{i,j}$ = jumlah suatu *term* muncul pada suatu dokumen
$\sum_k n_{i,j}$ = jumlah seluruh kata yang ada dalam suatu dokumen
$idf(t)$ = bobot kemunculan *term* t di seluruh dokumen
$N$ = jumlah dokumen secara keseluruhan
$df_t$ = jumlah dokumen yang mengandung *term* t
$w_{t,d}$ = bobot *term* dalam suatu dokumen

## 2.6. Algoritma *Multinomial Naïve Bayes*

*Multinomial Naïve Bayes Classifier* merupakan model pengembangan dari algoritma *bayes* yang cocok dalam pengklasifikasian teks atau dokumen terutama jika diaplikasikan ke dalam data yang besar [22]. Hal ini dikarenakan *Naïve Bayes Classifier* sendiri memiliki akurasi dan kecepatan yang tinggi. Berikut merupakan aturan *bayes* yang didefinisikan di bawah ini [4]:

$$P(c_i|d) = \frac{P(d|c_i).P(c_i)}{P(d)} \tag{2.4}$$

dimana $P(d|c_i)$ merupakan peluang kata $d$ muncul di kelas $c$. $P(c_i)$ merupakan peluang kata dari kelas $c$, $P(d)$ merupakan peluang muncul kata $d$.

*Multinomial Naïve Bayes Classifier* mengasumsikan bahwa semua atribut saling bergantung satu sama lain mengingat konteks kelas, dan mengabaikan semua dependensi antar atribut. Berikut merupakan persamaan dari *Multinomial Naïve Bayes Classifier* [19]:

$$P(c|term\ dokumen\ d) = P(c) \times P(t_1|c) \times P(t_2|c) \times ... \times P(t_n|c) \tag{2.5}$$

Dimana:
$P(t_n|c)$ = probabilitas kata ke-n dengan diketahui kelas c
$P(c|term\ dokumen\ d)$ = probabilitas suatu dokumen yang termasuk kelas c
$P(c)$ = *prior probability* dari kelas c

Untuk menghitung $P(c)$ sendiri dapat menggunakan Persamaan 2.6:

$$P(c) = \frac{N_c}{N} \tag{2.6}$$

$N_c$ adalah jumlah kelas c pada seluruh dokumen dan $N$ adalah jumlah seluruh dokumen.

Selanjutnya terdapat persamaan untuk menghitung $P(w|c_i)$ atau *probability likelihood*. $P(w|c_i)$ diestimasikan dengan menghitung jumlah kemunculan $w$ pada semua dokumen *training* di kelas c [23].

$$P(w_i|c) = \frac{c(w_i,c)+K}{\sum_{w \in V} c(w,c)+|V|} \tag{2.7}$$

Dimana $K$ merupakan nilai parameter yang biasanya diisi dengan 1 dan $|V|$ adalah jumlah atribut pada sampel.

Persamaan 2.7 merupakan metode *Laplacian smoothing*. Metode ini sebenarnya biasanya digunakan untuk mengatasi masalah dimana pengklasifikasi menemukan kata yang tidak pernah ditemukan pada data *training* sehingga probabilitas kedua kelas biasanya bernilai 0 dan tidak ada yang dapat dibandingkan [4].

Sedangkan untuk menghitung *probability likelihood* yang menggunakan metode TF-IDF adalah sebagai berikut [19]:

$$P(w_i|c) = \frac{W_{cw}+1}{(\sum W' \in VW'cw)+B'} \tag{2.8}$$

Keterangan:

$W_{cw}$ = nilai pembobotan TF-IDF dari term $w$ di kelas c

$\sum W' \in VW'cw$ = jumlah total nilai pembobotan TF-IDF untuk seluruh term yang ada di kelas c

$B'$ = jumlah W kata unik dimana nilai IDF tidak dikali dengan TF pada seluruh dokumen

## 2.7. *Confusion Matrix*

*Confusion Matrix* digunakan sebagai cara untuk melakukan evaluasi penelitian yang telah dilakukan. Evaluasi ini akan menghitung akurasi, *precision*, dan *recall*.

Tabel 2.1. Confusion Matrix [24]

| ***Actual*** / ***Prediction*** | ***Positive*** | ***Negative*** |
|---|---|---|
| ***Positive*** | True Positive | False Positive |
| ***Negative*** | False Negative | True Negative |

*Accuracy* merupakan jumlah kasus yang diklasifikasikan dengan benar pada test set dibagi dengan jumlah total kasus dalam *test set*. *Precision* adalah rasio terjadinya secara aktual diklasifikasikan sebagai positif untuk semua ketentuan yang diklasifikasikan sebagai positif. Sedangkan *Recall* adalah rasio terjadinya secara aktual untuk semua ketentuan positif [22]. Selain ketiga perhitungan tersebut, terdapat satu lagi yang digunakan, yaitu F1 - *measure* sebagai rata-rata dari *precision* dan *recall*.

Berikut merupakan persamaan masing-masing perhitungan [22]:

- ***Accuracy***

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.9}$$

- ***Precision***

$$Precision = \frac{TP}{TP + FP} \tag{2.10}$$

- ***Recall***

$$Recall = \frac{TP}{TP + FN} \tag{2.11}$$

- ***F1 - Measure***

$$F1 - Measure = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{2.12}$$

Keterangan:

$TP$ = *True Positive* (nilai aktual positif dan benar diprediksi positif)

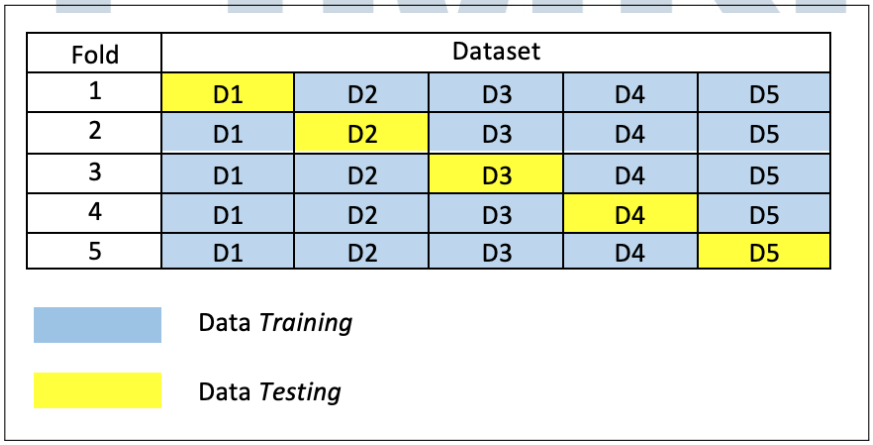
$TN$ = *True Negative* (nilai aktual negatif dan benar diprediksi negatif)

$FP$ = *False Positive* (nilai aktual negatif tetapi diprediksi positif)

$FN$ = *False Negative* (nilai aktual positif tetapi diprediksi negatif)

## 2.8. *K-Fold Cross Validation*

*Cross Validation* adalah sebuah metode statistik yang digunakan untuk mengestimasi validitas dari sebuah model prediktif *machine learning* [25]. Salah satu teknik *cross validation* yang paling sering digunakan adalah *k-fold cross validation*. Evaluasi *k-fold cross validation* bekerja dengan cara memisahkan sebuah *dataset* yang ada menjadi sejumlah *k*-grup atau *fold*, dengan jumlah yang sama pada masing-masing grup sehingga nantinya pengujian juga akan dilakukan sebanyak jumlah *k* yang telah ditentukan [26]. Pengujian validasi dilakukan terhadap data *training* dan data *testing* dimana dari seluruh sub grup hasil dari pembagian *dataset* sebelumnya akan dilakukan secara bergiliran. Misalnya, pada grup atau *fold* pertama, d1 akan menjadi data *testing*, sedangkan sisanya akan menjadi data *training*. Selanjutnya pada grup atau *fold* kedua, maka d2 yang akan menjadi data *testing* dan d1, d3, dan seterusnya yang akan menjadi data *training*. Hal ini dilakukan secara bergiliran setiap *fold*. Untuk gambaran yang lebih jelas, dapat melihat Gambar 2.1.




Gambar 2.1. *K-Fold Cross Validation, k=5*

Gambar 2.1 merupakan contoh untuk nilai $k$ = 5. Dapat terlihat jelas bahwa

untuk kotak yang berwarna biru itulah yang termasuk data *training*, sedangkan kotak berwarna kuning merupakan data *testing*. Setiap *fold* akan dihitung akurasinya dengan menggunakan rumus akurasi seperti pada Persamaan 2.9. Setelah semua pengujian *fold* selesai dilakukan, maka akurasi final akan dihitung dengan cara menghitung rata-rata dari akurasi masing-masing *fold*.

Berdasarkan pada publikasi oleh Daniel Berrar pada tahun 2018, untuk menghindari evaluasi yang bias, *subset* data yang digunakan untuk evaluasi model juga perlu mencerminkan rasio kelasnya. Untuk data di dunia nyata direkomendasikan menggunakan metode *sampling stratified 10-fold cross validation* [27].