BAB 2
LANDASAN TEORI

2.1 Kampus Merdeka

Berdasarkan pernyataan oleh bapak Prof. drh. Aris Junaidi, Ph.D. Merdeka Belajar-Kampus Merdeka, merupakan kebijakan menteri pendidikan dan kebudayaan, yang bertujuan mendorong mahasiswa untuk menguasai berbagai keilmuan yang berguna untuk memasuki dunia kerja. Kampus Merdeka memberikan kesempatan bagi mahasiswa untuk memilih mata kuliah yang akan mereka ambil. Kebijakan Kampus merdeka ini sesuai dengan permendikbud Nomor 3 tahun 2020 tentang Standar Nasional Pendidikan Tinggi, pada Pasal 18 disebutkan bahwa pemenuhan masa dan beban belajar bagi mahasiswa program sarjana atau sarjana terapan dapat dilaksanakan: 1) mengikuti seluruh proses pembelajaran dalam program studi pada perguruan tinggi sesuai masa dan beban belajar; dan 2) mengikuti proses pembelajaran di dalam program studi untuk memenuhi sebagian masa dan beban belajar dan sisanya mengikuti proses pembelajaran di luar program studi. Tujuan Kebijakan Merdeka Belajar-Kampus Merdeka adalah untuk meningkatkan kompetensi lulusan, baik *soft skills* maupun *hard skills*, agar lebih siap dan relevan dengan kebutuhan zaman, menyiapkan lulusan sebagai pemimpin masa depan bangsa yang unggul dan berkepribadian [9].

2.2 Text Mining

Text Mining merupakan teknik yang digunakan untuk memenuhi kebutuhan dalam proses klasifikasi dokumen dengan koten apapun. Cara kerjanya yaitu upaya dalam memunculkan variasi dari kumpulan data yang tersedia dalam jumlah besar, untuk kemudian berusaha menemukan pola yang sesuai dengan apa yang diharapkan dari kumpulan teks yang ada [10]. *Text Mining* juga merupakan teknik yang digunakan untuk menangani klasifikas, *clustering, information extraction,* dan *information retrival*. Perbedaan *text mining* dan *data mining* adalah pola yang digunakan *text mining* diambil dari sekumpulan bahasa alami yang tidak terstruktur, sedangkan *data mining* pola yang diambil dari *database* yang terstruktur [11].

2.3 Analisis Sentimen

Analisis sentimen adalah atau opinion mining adalah salah satu pembelajaran yang memahami, mengekstrak dan mengolah data teks untuk mendapatkan informasi sentimen yang terkandung dalam opini orang yang tertuang dalam sebuah kalimat opini. Analisis sentimen dilakukan untuk melihat pendapat atau opini terhadap sebuah masalah atau objek oleh seseorang, apakah cenderung berpoini positif atau negatif [12].

Analisis sentimen merupakan sebuah proses untuk menganalisis atau mengidentifikasi sebuah opini seseorang yang menunjukan sikap terhadap suatu topik atau produk tertentu masuk ke dalam kategori positif, negatif, atau netral. Opini sentimen memiliki karakteristik yang menunjukan bahwa hal tersebut bersifat subjektif [7]. Pemeriksaan opini diperlukan agar dapat melakukan peringkasan terhadap suatu pendapat.

2.4 Analisis Sentimen Berdasarkan Aspek

Analisis sentimen berdasarkan aspek atau *Aspect Based Sentiment Analysis (ABSA)* adalah salah satu perkembangan dari analisis sentimen yang mengacu pada sebuah kalimat. Ada dua tugas utama didalam analisis sentimen berbasis aspek yaitu aspek ekstraksi, dan klasifikasi aspek. Dalam ekstraksi aspek, sentimen ditentukan oleh jenis aspek yang dibahas. Sedangkan pada aspek sentimen klasifikasi, sentimen diklasifikasikan sebagai positif, negatif, atau netral untuk aspek itu [13]. ABSA melakukan analisis sentimen yang lebih dalam terhadap suatu kalimat opini atau teks ulasan, analisis sentimen dilakukan pada tingkat aspek untuk mengelompokkan hal-hal yang dikeluhkan oleh pelanggan ke dalam aspek-aspek tertentu kemudian divisualisasikan melalui *dashboard* sehingga informasi yang dibutuhkan oleh pihak manajemen dalam menentukan solusi yang tepat menjadi lebih rinci dan efektif [14]. Untuk mencapai tujuan analisis sentimen berbasis aspek, kita perlu melakukan enam tugas dasar yang dibutuhkan kemampuan NLP yang dalam. Di antaranya, dua tugas telah menerima perhatian yang lebih oleh peneliti yaitu:

2.4.1 Aspect Extraction

Tugas ini adalah untuk mengekstraksi aspek dan entitas yang telah dievaluasi. Sebagai contoh, dalam kalimat “Kualitas suara dari handphone ini sangat bagus”, kita dapat mengekstrak kualitas suara pada kalimat tersebut menjadi entitas aspek yang diwakilkan oleh handphone itu. Untuk memudahkan dalam penyajian sering kali entitas dihilangkan dan berfokus hanya pada aspek tetapi dalam sebuah aspek tetaplah memiliki entitas yang jika tidak masa aspek itu tidak memiliki makna, seperti pada contoh kalimat tadi yang menunjukan bahwa handphone tidak menunjukan aspek umum karena lebih berfokus kepada suaranya.

2.4.2 Aspect sentiment classification

Tugas ini menentukan apakah opini pada aspek yang berbeda bernilai positif, negatif, atau netral. Pada contoh kalimat “Kualitas suara dari handphone ini sangat bagus” aspek suara adalah positif sedangkan untuk pendapat untuk aspek umum atau keseluruhan entitas juga bersifat positif [15].

2.5 Twitter

Twitter adalah layanan bagi teman, keluarga, dan teman sekerja untuk berkomunikasi dan tetap terhubung melakui pertukaran pesan yang cepat dan sering. Pengguna dapat memposting tweet berisikan teks, foto, video dan tautan. Pesan diposting ke profil pengguna, terkirim ke pengikut penggunam dan dapat dicari di pencarian Twitter [16]. Twitter dapat mengirim dan membaca pesan berbasis teks yang awalnya 140 karakter dan telah ditambah hingga 280 karakter pada tnaggal 07 November 20017 yang dikenal dengan tweet atau kicauan. Twitter API (*Application Programming Interface*) adalah akses programatik kepada perusahaan, pengembang, dan pengguna ke data Twitter [11].

2.6 Crawling

Untuk mengunduh sejumlah halaman *web* secara otomatis, maka perlu dilakukan *crawling*. *Crawling* adalah proses menjelajahi *web* dan mengunduh halaman *web* secara otomatis untuk mengumpulkan informasi. Program yang khusus bertugas melakukan *crawling* disebut *Crawler* [17].

2.7 Preprocessing

Data yang telah dikumpulkan dalam tahap pengumpulan data, kemudian dilanjutkan ke tahapan berikutnya yaitu *preprocessing* yang digunakan untuk transofrmasi tata data menjadi lebih terstruktur [18]. Dalam penelitian ini *text preprocessing* terbagi atas *cleansing, case folding, tokenization, normalization, stopwords removal,* dan *stemming*.

2.7.1 Cleaning

Proses *cleaning* bertujuan untuk membersihkan data dari angka, hashtag, alamat website, email, *username,* dan simbol dalam suatu kalimat [19].

2.7.2 Case Folding

Case Folding merupakan tahapan mengubah kata/term menjadi bentuk yang sama, seperti huruf kecil atau huruf besar [2].

2.7.3 Tokenization

Tokenization merupakan proses pemotongan kalimat pada teks menjadi sebuah kata serta menentukan struktur dari setiap kata tersebut. Secara umum, setiap kata diidentifikasikan atau dipisahkan dengan kata lain oleh karaktek spasi, karakter petik tunggal (‘), titik (.), koma (,), dan titik dua (:) [2].

2.7.4 Normalization

Normalization adalah proses perbaikan kata-kata yang salah eja atau disingkat dalam bentuk tertentu. Tahap ini bertujuan untuk memperkecil dimensi kata yang memiliki arti yang sama tetapi memiliki ejaan yang salah atau disingkat dalam bentuk tertentu.

2.7.5 Stopwords removal

Stopword removal merupakan tahapan penyaringan kata-kata penting dari data yang didapat. Kata yang ada di dalam *list stopwords* akan dihapus karena dianggap tidak memeranguhi hasil analisis sentimen [18].

2.7.6 Stemming

Stemming merupakan proses pemetaan berbagai variasi dari morfologi kata yang dikembalikan ke bentuk dasarnya [18]. Mengidentifikasi akar ataupun kata

dasar dari sebuah kata tiap dokumen guna menghapus berbagai *suffix*, mengurangi jumlah kata, dan menghemat waktu dan ruang memori [2].

2.8 Pembobotan TF-IDF

TF-IDF (*Term Frequency-Inverse Document Frequency*) merupakan salah satu metode untuk melakukan pembobotan kata dari proses ekstraksi kata dengan menerapkan perhitungan kata umum di *information retrival*. Metode pembobotan ini merupakan salah satu metode untuk melakukan pembobotan kata dari proses ekstraksi kata dengan menerapkan perhitungan kata umum di *information retrival*. Metode pembobotan ini merupakan penggabungan antara *term frequency* dan *inverse document frequency* [20].

Term frequency menyatakan nilai frekuensi *term* yang sering muncul pada sebuah dokumen. Semakin besar jumlah kemunculan suatu *term* dalam dokumen, semakin besar pula bobotnya atau akan memberikan nilai kesesuaian yang semakin besar. $f(t_k, d_j)$ mendefinisikan jumlah kemunculan term *k* pada sebuah *j*. Rumus TF dapat didefinisikan sebagai berikut [8]:

$$TF(t_k d_j) = f(t_k d_j) \quad (2.1)$$

Inverse document frequency adalah proses untuk mengukur seberapa penting kata dalam suatu dokumen [20]. Semakin sering suatu *term* muncul di banyak dokumen maka nilai IDFnya akan kecil. Jumlah dokumen pada data akan dibagi dengan jumlah dokumen yang mengandung *term* d*f*(t). IDF dapat dihitung dengan rumus sebagai berikut [8]:

$$IDF(t_k) = log \frac{D}{df(t)} \quad (2.2)$$

TF-IDF dapat dirumuskan sebagai berikut:

$$TF\ IDF(t_k, d_j) = TF(t_k, d_j) * IDF(t_k) \quad (2.3)$$

Keterangan:

$f(t_k, d_j)$ = Jumlah kemunculan *term* k pada sebuah dokumen j

D = Jumlah dokumen pada data

$df(t)$ = Jumlah dokumen yang mengandung *term*

2.9 Topic Modeling

Dalam pembelajaran mesin dan pemrosesan bahasa alami atau NLP, model topik adalah model generatif, yang menyediakan kerangka kerja probabilistik. Metode pemodelan topik umumnya digunakan untuk mengatur, memahami, mencari, dan meringkas arsip elektronik besar secara otomatis. "Topik" menandakan hubungan variabel yang tersembunyi, diperkerikan, yang menghubungkan kata-kata dalam kosakata dan kemunculannya dalam dokumen [5]. Konsep *topic modelling* menurut Blei terdiri dari entitasentitas yaitu "kata", "dokumen", dan "corpora". Kata dianggap sebagai unit dasar dari data diskrit dalam dokumen, didefinisikan sebagai item dari kosa kata yang diberi indeks untuk setiap kata unik pada dokumen. Dokumen adalah susunan dari kata. Corpora merupakan bentuk jamak dari corpus. Ide dasar dari *Topic Modelling* adalah bahwa sebuah topik terdiri dari kata-kata tertentu yang menyusun topik tersebut, dan dalam satu dokumen memiliki kemungkinan terdiri dari beberapa topik dengan peluang masing-masing. Namun secara pemahaman manusia, dokumen-dokumen merupakan objek yang dapat diamati, sedangkan topik, distribusi topik per-dokumen, dan penggolongan setiap kata pada topik perdokumen merupakan struktur tersembunyi, maka dari itu topic modelling bertujuan untuk menemukan topik dan katakata yang terdapat pada topik tersebut [21].

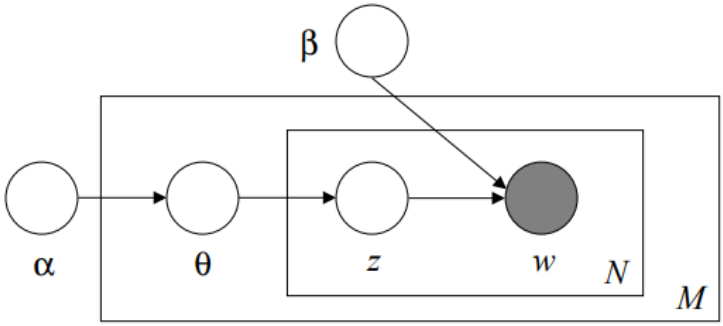
2.10 Latent Dirichlet Allocation (LDA)

Latent Dirichlet Allocation (LDA) adalah teknik pemodelan topik yang secara otomatis menemukan topik dalam dokumen teks. LDA menganggap dokumen sebagai campuran dari berbagai topik dokumen. Algoritma ini pertama kali disajikan sebagai model grafis untuk penemuan topik. Tujuan dari LDA adalah memetakan semua dokumen ke topik sedemikian rupa, sehingga kata-kata dalam setiap dokumen sebagian besar terkait dengan topik tersebut [5]. *Latent Dirichlet Allocation* (LDA) merupakan *to*[5]*pic modeling* dan topik analisis yang paling populer saat ini dan banyak digunakan dalam melakukan analisis pada dokumen

yang berukuran sangat besar. *Latent Dirichlet Allocation* (LDA) saat ini sering digunakan karena dapat melakukan klasterisasi, melakukan peringkasan, menghubungkan, dan dapat memproses data dengan memberikan bobbot pada masing-masing dokumen yang nantinya menghasilkan daftar topik. Ide dasar darinya adalah menganggap bahwa dokumen yang diujikan dapat direpresentasikan sebagai sebuah model yang dicampur dari berbagai topik yang dibutuhkan, oleh karena itu disebut sebagai laten [22].

Ide dasar dari LDA adalah dokumen dipresentasikan sebagai campuran dari topik acak yang tersembunyi, dimana setiap topik dicirikan sebagai distribusi dari kata-kata [23]. Secara formal, didefinisikan notasi berikut:

a. Kata adalah bentuk dasar dari data diskrit
b. Sebuah dokumen adalah barisan kata-kata *N* yang dinotasikan dengan **w** = (w1, w2, …, w_N), dimana w_N adalah baris kata ke-*n*
c. Sebuah *corpus* adalah koleksi dari *M* dokumen dinotasikan dengan *D* = (**w1**, **w2**, …, $\mathbf{w}_N$).



Gambar 2. 1 Graphical Model LDA

Berdasarkan *graphical model* pada gambar 2.1 mempresentasikan model grafis dari LDA. Kotak-kotak itu adalah "pelat" yang mewakili replika. Pelat luar mewakili dokumen, sedangkan pelat dalam mewakili pilihan perulangan topik dan kata-kata dalam dokumen. Parameter α dan β diberikan untuk corpus, paramater α adalah parameter untuk distribusi topik dari dokumen sedangkan β adalah

parameter untuk distribusi kata dari topik. Semakin besar nilai α, maka setiap dokumen mengandung sebagian besar topik, artinya tidak hanya ada satu topik spesifik. Sedangkan semakin besar nilai β maka setiap topik mengandung sebagian besar kata, tidak hanya untuk beberapa kata spesifik yang membedakan topik satu dengan lainnya [23]. LDA adalah *soft clustering*, maka setiap dokumen bisa terdiri dari beberapa topik yang berbeda. Kemudian bisa ditentukan topik dari setiap kata pada setiap dokumen yang dinotasikan dengan *z*, untuk nantinya akan menjadi cluster-cluster. Hasilnya adalah campuran kata-kata di tiap topik yang sudah ditentukan sebelumnya kemudian diinterpretasi hasil tiap cluster tersebut membahas topik apa saja [24]. Untuk menghitung probabilitas setiap kata menggunakan rumus berikut [25]:

$$P(Z_t = j|z_{-t}, w_t, d_t) = \frac{C_{w.j}^{WT} + \beta}{\Sigma_{w=1}^{W} C_{w,j}^{WT} + W\beta} \; X \; \frac{C_{d,j}^{DT} + \alpha}{\Sigma_{t=1}^{T} C_{d,t}^{DT} + T\alpha} \qquad 2.4$$

Keterangan dari rumus:

β = Distribusi kata per topik (parameter konsentrasi)

W = Panjang kosakata (Jumlah token/kata unik dalam dokumen lengkap)

α = Distribusi topik per dokumen

T = Jumlah topik

$C_{w.j}^{WT}$ = Jumlah kemunculan kata/token pada topik

$\Sigma_{w=1}^{W} C_{w,j}^{WT}$ = Jumlah kemunculan topik dalam matriks

$C_{d,j}^{DT}$ = Kemunculan topik dalam setiap dokumen

$\Sigma_{t=1}^{T} C_{d,t}^{DT}$ = Total jumlah kali setiap dokumen muncul sebagai topik 1 dan topik 2

$\frac{C_{w.j}^{WT} + \beta}{\Sigma_{w=1}^{W} C_{w,j}^{WT} + W\beta}$ = Distribusi probabilitas akta pada suatu topik

$\frac{C_{d,j}^{DT} + \alpha}{\Sigma_{t=1}^{T} C_{d,t}^{DT} + T\alpha}$ = Distribusi probabilitas topik pada suatu dokumen

Untuk penentuan topik akhir dimana topik yang akan dipilih berdasarkan probabilitas tertinggi dari dua topik untuk sebuah kata.

2.11 Coherence dan Perplexity

Coherence adalah sekumpulan pernyataan atau fakta yang saling mendukung, dalam filsafat ilmiah, pendekatan yang digunakan adalah dengan menggunakan gabungan fungsi dan probabilitas marginal yang terasosiasi dengan faktanya. *Topic Coherence* digunakan pada model topik karena belum adanya jaminan interpretabilitas pada hasil topik model tersebut, cara dalam melakukan koherensi otomatis adalah dengan memperlakukan kata-kata pada topik sebagai fakta lalu membatasi koherensi yang digunakan didasarkan pada perbandingan sepasang kata [26]. Berdasarkan kata-kata yang digunakan yang terdapat dalam dokumen, penggalian relasi topik dilakukan dengan asumsi bahwa satu dokumen mencakup suatu set kecil dari topik yang ringkas. *Topic Coherence* mengukur skor satu topik dengan mengukur tingkat kesamaan semantik antara kata-kata dengan skor tinggi dalam topik [27]. *Topic Coherence* dilakukan untuk memperbaiki masalah bahwa model topik tidak memberikan jaminan pada interpretasi output. Pengujian koherensi pada Latent Dirichlet Allocation merupakan tahapan proses uji clustering aspek untuk mengetahui nilai aspek atau nilai dari K terbaik [28].

Perplexity adalah ukuran seberapa baik model probabilitas dalam menerima set data yang baru. Penggunakan *perplexity* dari set uji untuk mengevaluasi model dan nilai *perplexity* yang didapat akan secara monoton menurun dari kemungkinan uji data. Semakin rendah nilai akan menunjukan kinerja generalisasi yang lebih baik [26].

2.12 Naïve Bayes

Naïve Bayes adalah metode *machine learning* untuk probabilitas atau peluang. *Naïve Bayes* merupakan metode untuk klasifikasi *text* dengan kecepatan pemrosesan yang tinggi jika dalam data besar [29]. *Naïve Bayes* merupakan metode

supervised learning yang sederhana dan banyak digunakan. *Naïve Bayes* merupakan salah satu algoritma *machine learning* tercepat dan dapat menangani *feature* atau *class* terlepas dari model yang sederhana.

Naïve Bayes merupakan pengklasifikasian dengan metode probabilitas dan statistik yang dikemukan oleh ilmuan Inggris *Thomas Bayes*, yaitu memprediksi peluang di masa depan berdasarkan pengalaman dimasa sebelumnya sehingga dikenal sebagai *Teorema Bayes.* Klasifikasi *Naïve Bayes* diasumsikan bahwa ada atau tidak ciri tertentu dari sebuah kelas tidak ada hubungannya dengan ciri kelas lainnya [30]. Ciri dari algoritma *Naïve Bayes* ini adalah independensi yang sangat kuat (*Naïve*) dari masing-masing kondisi atau kejadian [31].

Persamaan dari teorema *Bayes* adalah [30]:

$$P(H|X) = \frac{P(X|H).P(H)}{P(X)} \qquad 2.5$$

Keterangan:

X : Data dengan class yang belum diketahui

H : Hipotesis data *X* merupakan suatu class spesifik

P(H|X) : Probabilitas hipotesis *H* berdasar kondisiX (posteriori probability)

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

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

P(X) : Probabilitas X

Berdasarkan rumus tersebut, maka peluang masuknya sampel tertentu ke dalam kelas H adalah peluang munculnya kelas H (sebelum masuknya sampel tersebut), dikalikan dengan peluang kemunculan karakteristik sampel pada kelas H, dibagi dengan peluang kemunculan karakteristik sampel keseluruhan [31].

Ide dasar dari aturan Bayes adalah bahwa hasil hipotesis atau peristiwa (H) dapat diperkirakan berdasarkan pada beberapa nilai bukti € yang diamati. Ada beberapa hal penting dari aturan Bayes tersebut, yaitu:

1. Sebuah probabilitas awal/prior A atau P(A) adalah probabilitas dari suatu hipotesis sebelum bukti diamati.
2. Sebuah probabilitas akhir H atau P(H|E) adalah probabilitas dari suatu hipotesis setelah bukti diamati

2.13 Naïve Bayes untuk Klasifikasi

Naïve Bayes Classifier (NBC) atau *Naïve Bayes* untuk klasifikasi adalah sebuah metode pengklasifikasian yang dapat diterapkan pada data teks, dengan menggunakan probabilitas sederhana yang berakar pada *Teorema Bayes* dan memiliki asumsi ketidak tergantungan (independent) yang tinggi dari masing-masing kondisi atau kejadian [32]. Kaitan antara *Naïve Bayes* dengan klasifikasi, korelasi hipotesis, dan bukti dengan klasifikasi adalah bahwa hipotesis dalam teorema *Bayes* merupakan label kelas yang menjadi target pemetaan dalam klasifikasi, sedangkan bukti merupakan fitur-fitur yang menjadi masukan dalam model klasifikasi [33].

Jika X adalah vektor masukan yang berisi fitur dan Y adalah label kelas, *Naïve Bayes* dituliskan dengan P(Y|X). Notasi tersebut berarti probabilitas label kelas Y didapatkan setelah fitur-fitur X diamati. Notasi ini disebut juga probabilitas akhir (posterior probability) untuk Y, sedangkan P(Y) disebut probabilitas awal (prior probability) Y.

Formulasi *Naïve Bayes* untuk klasifikasi adalah:

$$P(Y|X) = \frac{P(Y)\prod_{i=1}^{q} P(X_i|Y)}{P(X)} \qquad 2.6$$

Keterangan:

P(X|Y) = Probabilitas data dengan vector X pada kelas Y.

P(Y) = Probabilitas kelas awal Y.

$\prod_{i=1}^{q} P(X_i|Y)$ = Probabilitas independen kelas Y dari semua fitur dalam vektor X.

Nilai P(X) selalu tetap sehingga dalam perhitungan prediksi nantinya tinggal menghitung $P(Y)\prod_{i=1}^{q} P(X_i|Y)$ dengan memilih yang terbesar sebagai kelas yang dipilih sebagai hasil prediksi. Sementara probabilitas independen $\prod_{i=1}^{q} P(X_i|Y)$ merupakan pengaruh semua fitur dari data terhadap setiap kelas Y, yang dinotasikan dengan:

$$P(Y|X = y) = \prod_{i=1}^{q} P(X_i|Y) = y \qquad 2.7$$

Setiap set fitur $X = \{X_1, X_2, X_3, \ldots, X_q\}$ terdiri atas q atribut (q dimensi).

Pada umumnya *Bayes* mudah dihitung untuk fitur bertipe kategoris seperti pada kasus ketersediaan pupuk bernilai tidak tersedia dan tersedia [34]. Klasifikasi dengan *Naïve Bayes* bekerja berdasarkan teori probabilitas yang memandang semua fitur dari data sebagai bukti dalam probabilitas. Hal ini memberikan karakteristik Naive Bayes sebagai berikut [35]:

a. Metode *Naïve Bayes* bekerja teguh (robust) terhadap data-data yang terisolasi yang biasanya merupakan data dengan karakteristik berbeda (outliner *Naïve Bayes* juga bisa menangani nilai atribut yang salah dengan mengabaikan data latih selama proses pembangunan model dan prediksi.

b. Tangguh menghadapi atribut yang tidak relevan.

c. Atribut yang mempunyai korelasi bisa mendegradasi kinerja klasifikasi Naïve Bayes karena asumsi independensi atribut tersebut sudah tidak ada

2.14 Confusion Matrix

Confusion Matrix merupakan suatu instrumen yang digunakan untuk mengevaluasi performa dari model klasifikasi yang telah dihasilkan. Pada *confusion matrix*, hasil kelas prediksi akan dibandingkan dengan kelas data yang sebenarnya. Hasil tersebut kemudian akan digunakan untuk menghitung nilai

accuracy, *precision*, *recall*, dan *f-score*. Pengukuran evaluasi pada *confusion matrix* dapat dilihat pada tabel 2.1 berikut [20]:

Tabel 2. 1 Confusion Matrix

Data Aktual	Data Prediksi		
	True	False	Total
TRUE	TP	FN	P
FALSE	FP	TN	N
TOTAL	P’	N’	P+N

Keterangan:

TP (*True Positive*) = Data positif yang terklasifikasi secara benar.

TN (*True Negative*) = Data negatif yang terklasifikasi secara benar.

FP (*False Positive*) = Data negatif yang terklasifikasi menjadi positif.

FN (*False Negative*) = Data positif yang terklasifikasi menjadi negatif.

Keempat parameter diatas digunakan untuk menghitung metode evaluasi yakni [36]:

1. *Recall*, yaitu perbandingan jumlah dokumen yang relevan terkenali dengan jumlah seluruh dokumen relevan. *Recall* memiliki rumusan sebagai berikut:

$$Recall = \frac{TP}{TP + FP} \; X \; 100\% \qquad 2.8$$

2. *Precision*, yaitu perbandingan jumlah dokumen yang relevan terkenali dengan jumlah dokumen yang terkenali. *Precision* memiliki persamaan sebagai berikut:

$$Precision = \frac{TP}{TP + FN} \; X \; 100\% \qquad 2.9$$

1. *F-Measure*, merupakan nilai yang mewakili seluruh kinerja sistem yang merupakan penggabungan nilai *Recall* dan *Presicion F-Measure* memiliki persamaan sebagai berikut:

$$F - Measure = 2\,X\,\frac{Precision\ x\ Recall}{Precision + Recall}\,X\,100\% \qquad 2.10$$

2. *Confusion Matrix* menunjukan tingkat akurasi dari proses klasifikasi yang telah dilakukan. Tingkat akurasi menunjukan proporsi jumlah prediksi yang benar.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}\,X\,100\% \qquad 2.11$$