BAB 2
TINJAUAN PUSTAKA DAN DAFTAR TEORI

2.1 Tinjauan Pustaka

Algoritma *Random Forest* merupakan salah satu metode *machine learning* yang dimanfaatkan untuk klasifikasi data yang memiliki jumlah banyak (Basar, et al., 2022). *Random Forest* dapat dibangun menggunakan bagging dengan pemilihan atribut acak. Metode CART (*Clasification and Regresion Tree*) sendiri dapat digunakan untuk menumbuhkan pohon keputusan, pohon keputusan tersebut tumbuh hingga ukuran maksimum dan tidak akan dipangkas sehingga dihasilkan kumpulan pohon yang kemudian disebut *forest*.

Algoritma *Naïve Bayes* merupakan teknik klasifikasi dengan bentuk model probabilistik dan statistik yang disederhanakan dengan berdasar pada teorema *Bayes* dengan asumsi bahwa setiap atribut bersifat bebas (*independence*). Dengan kata lain, algoritma ini mengasumsikan bahwa ada atau tidak ciri tertentu dari sebuah kelas tidak ada hubungannya dengan ciri dari kelas lainnya.

Terdapat beberapa penelitian berkaitan dengan analisis sentimen yang menggunakan berbagai algoritma. Penelitian pertama dilakukan oleh (Aji, et al., 2020). Penelitian ini membahas mengenai analisis sentimen terhadap *review fintech* dengan metode *Naïve Bayes Classifier* dan *K-Nearest Neighbor.* Penelitian ini dilakukan atas dasar banyaknya *review* yang ditampilkan pada kolom komentar yang telah disediakan oleh Google Play Store di aplikasi Dana. Sehingga dibutuhkan analisa untuk mengklasifikasi ulasan yang diberikan termasuk positif dan negatif. Penelitian ini juga bertujuan untuk menentukan hasil akurasi analisa sentimen yang dihasilkan algoritma *Naïve Bayes* dan *K-Nearest Neighbor*. Akurasi yang dihasilkan dengan algoritma *Naïve Bayes* menghasilkan nilai akurasi sebesar 84,76%. Berdasarkan hasil akurasi dapat disimpulkan bahwa algoritma *Naïve Bayes* memiliki kinerja yang lebih baik.

Lalu penelitian yang kedua dilakukan oleh (Agustian, et al., 2022). Penelitian ini membahas mengenai penerapan analisis sentimen dan *Naïve Bayes* terhadap opini penggunaan kendaraan listrik di *Twitter*. Berdasarkan hasil klasifikasi respon masyarakat terhadap kendaraan listrik menunjukan lebih mengarah positif. Akurasi yang didapat melalui metode *confusion matrix* yaitu sebesar 80%.

Penelitian yang ketiga dilakukan oleh (Fitri, et al., 2020). Penelitian ini membahas mengenai analisis sentimen pada aplikasi Ruangguru dengan algoritma *Naïve Bayes, Random Forest,* dan *Support Vector Machine*. Data yang digunakan merupakan hasil *review* aplikasi Ruangguru di Google Play Store dengan data yang diambil sebanyak 1.629 data komentar. Berdasarkan hasil pengujian didapatkan

akurasi tertinggi pada metode *Random Forest* dengan jumlah akurasi sebesar 97.16%. Sehingga dapat disimpulkan bahwa hasil performa menunjukkan *Random Forest* memiliki nilai akurasi tertinggi dibanding dengan metode lainnya.

Penelitian yang keempat dilakukan oleh (Karthika, et al., 2019). Penelitian ini membahas mengenai analisis sentimen pada produk dari situs belanja *online* yaitu flipkart.com berdasarkan aspek produk yang diklasifikasikan menjadi positif, negative dan netral. Penelitian tersebut dilakukan untuk membandingkan akurasi di antara metode *Random Forest* dan *Support Vector Machine.* Berdasarkan hasil pengujian didapatkan bahwa *Random Forest* memiliki akurasi terbaik sebesar 97% dibanding kan dengan *Support Vector Machine.*

Penetitian yang kelima dilakukan oleh (Basar, et al., 2022). Penelitian tersebut membahas mengenai penerapan algoritma *Random Forest* untuk melakukan klasifikasi terhadap opini pengguna *Twitter* terhadap *platform* ShopeePay. Data yang digunakan yaitu dengan mengumpulkan *tweet* pada media sosial *Twitter* yang didapatkan menggunakan *API Twitter* dengan teknik *web crawling*. Hasil penelitian tersebut menunjukkan tingkat akurasi sebesar 95%.

Dari kelima penelitian terdahulu, berikut pada Tabel 2.1 merupakan ringkasan dari penelitian terdahulu yang digunakan:

Tabel 2.1 Ringkasan Hasil Referensi Penelitian

No.	Nama Penelitian	Judul	Variabel	Metode	Hasil/Akurasi
1.	Sopian Aji, Surohman, Rousyati, Fanny Fatma Wati (Aji, et al., 2020)	Analisa Sentimen Terhadap *Review Fintech* Dengan Metode *Naïve Bayes Classifier* Dan *K- Nearest Neighbor*	232 Data Ulasan Pada Toko Aplikasi *Google Play Store*	*Naive Bayes Classifier* Dan *K-Nearest Neighbor*	*Naïve Bayes*: 84.76% **(Lebih Akurat)** *K-Nearest Neighbor*: 82.92%
2.	Adittia Agustian, Tukino, Fitria Nurapriani (Agustian, et al., 2022)	Penerapan Analisis Sentimen Dan *Naïve Bayes* Terhadap Opini Penggunaan Kendaraan Listrik Di *Twitter*	Data Melalui *Twitter API*	*Confusion Matrix*	*Naïve Bayes*: 80%
3.	Evita Fitri, Yuri Yuliani, Susy Rosyida,	Analisis Sentimen Terhadap Aplikasi Ruangguru	1.629 Data Ulasan Pada Toko Aplikasi	*Naïve Bayes*, *Random Forest* Dan	*Naïve Bayes*: 94.16% *Random Forest*: 97.16%

No.	Nama Penelitian	Judul	Variabel	Metode	Hasil/Akurasi
	Windu Gata (Fitri, et al., 2020)	Menggunakan Algoritma *Naïve Bayes*, *Random Forest* Dan *Support Vector Machine*	*Google Play Store*	*Support Vector Machine*	**(Lebih Akurat)** *SVM*: 96.01%
4.	P. Karthika, R. Murugeswari, R. Manoranjithem (Karthika, et al., 2019)	*Sentiment Analysis of Social Media Network*	Ulasan Produk Dari Situs flipkart.com	*Random Forest* Dan *Support Vector Machine*	*Random Forest*: 97% **(Lebih Baik)** *SVM*: 92%
5.	Thifal Fadiyah Basar, Dian Eka Ratnawati, Issa Arwani (Basar, et al., 2022)	Analisis Sentimen Pengguna *Twitter* terhadap Pembayaran *Cashless* menggunakan Shopeepay dengan Algoritma *Random Forest*	Data Melalui *Twitter API*	*Random Forest*	*Random Forest*: 95%

2.2 Dasar Teori

2.2.1 *Childfree*

Childfree adalah sebuah keputusan dalam memilih kehidupan tanpa harus memiliki keturunan. Fenomena pasangan suami-istri untuk tidak memiliki keturunan dalam rumah tangga baik itu yang dilahirkan dari rahim sang wanita maupun mengadopsi anak (Nursyamsiah Mingkase & , 2022).

Alasan *childfree* dijadikan sebagai sebuah pilihan adalah anggapan bahwa memiliki anak atau keturunan bukanlah hal yang dapat dipaksakan karena merupakan bagian dari hak asasi manusia. Selain itu, alasan lainnya adalah untuk menekan ledakan populasi dan mencegah peningkatan anak-anak terlantar. Adanya *politic of body* juga menekankan bahwa tubuh perempuan adalah milik dirinya sendiri sehingga siapapun tidak berhak memaksa mereka untuk mengandung dan melahirkan anak. Fenomena ini juga didukung dengan keberadaan feminisme, yaitu gerakan memperjuangkan hak-hak perempuan agar tidak dipandang rendah dan memiliki

posisi yang setara dengan laki-laki. Dengan pemikiran hak asasi manusia dan kepentingan bersama, serta ditambah adanya isu kesetaraan gender membuat *childfree* menjadi salah satu pilihan masyarakat modern pada abad ke-21 (Cornellia, et al., 2022).

2.2.2 *Twitter*

Twitter merupakan salah satu media sosial yang populer digunakan oleh pengguna saat ini. Pengguna *Twitter* akan memberikan kabar terbaru atau komentar tentang hal yang sedang menjadi topik utama di dunia. Hal yang sedang menjadi topik utama dan banyak dan sering dikomentari oleh pengguna akan menimbulkan *trending topic* di media sosial *Twitter* (Darwis, et al., 2021). Data *Twitter* menjadi salah satu sumber data penelitian yang lama sering digunakan karena data didapatkan dari berbagai sumber, melampirkan konten, dan komunikasi real time. Komponen *tweet* yang dapat digunakan untuk mengekstrak informasi yaitu (Fauziyyah & Gautama, 2020):

Username : identifikasi pengguna
Time stamp : waktu saat *tweet* itu dikirim
Tweet text : isi dari teks *tweet*
Hashtags : simbol #, tagar dikaitkan dengan topik tertentu
Replies : balasan pesan teks dari *tweet*
Retweet : ketika pengguna membagikan *tweet* dengan pengikut mereka

Kelebihan dari media sosial *Twitter* yaitu salah satunya menyediakan API (Application Programming Interface) yang sangat baik, sehingga memudahkan setiap orang untuk mengambil data dari *Twitter*. Pengumpulan data dari *Twitter* dapat digunakan untuk berbagai kebutuhan seperti, mengetahui popularitas kandidat pilkada atau pemilu, mendapat informasi mengenai popularitas suatu produk atau untuk yang sederhana dapat digunakan untuk melihat semua mention, *retweet* atas suatu akun *Twitter* tertentu (Pintoko & L, 2018).

2.2.3 Analisis Sentimen

Analisis sentimen atau yang biasa dikenal dengan istilah opinion mining merupakan salah satu cabang penelitian dari *text* mining yang bertujuan untuk menentukan persepsi atau subjektivitas publik (khalayak) terhadap suatu topik pembahasan, kejadian, ataupun permasalahan (Rachman & Pramana, 2020). Analisis sentimen adalah proses memahami, mengekstrak, dan mengolah data tekstual secara otomatis untuk mendapatkan informasi sentimen yang terkandung dalam suatu kalimat opini (Rahutomo, et al., 2018). Tugas dasar analisis sentimen adalah dengan mengumpulkan data atau teks yang ada menjadi sebuah kalimat atau dokumen, yang kemudian mengkategorikan kalimat atau dokumen tadi menjadi bentuk positif atau negatif (Pratama, et al., 2023).

Secara umum analisis sentimen dikelompokan menjadi dua tipe yaitu, analisis sentimen berbasis polaritas dan analisis sentimen berbasis emosi. Analisis sentimen berbasis polaritas melakukan pengambilan informasi pada review berdasarkan kelasnya. Yakni, positif negatif dan netral. Jika Analisis sentimen berbasis emosi pengambilan informasi pada review didasarkan pada emosi dasar manusia (Pratiwi, et al., 2021).

2.2.4 *Website*

Website berasal dari kata *World Wide Web*, yakni layanan yang didapati oleh pemakai komputer yang terhubung dengan jaringan internet. Website merupakan aplikasi tertentu yang berjalan di atas platform atau operation system browser. Dengan demikian website yang dimaksud dalam penelitian ini berarti sebuah halaman informasi yang tersedia secara online dan dapat diakses di seluruh dunia selama tersambung dengan jaringan internet (Surentu, 2020).

Website adalah kumpulan halaman web yang dijalankan menggunakan browser dan internet. *Website* berada dalam domain atau subdomain yang sering disebut WWW atau *World Wide Web*. Sebuah website dibuat dengan bahasa pemrograman HTML (*Hyper Text Markup Language*) yang diakses melalui protokol di internet. Selain menggunakan bahasa pemrograman HTML, *website* dapat dikembangkan dengan bahasa pemrograman dinamis, salah satunya adalah bahasa pemrograman PHP (*Hypertext Preprocessor*) yang merupakan bahasa pemrograman *open-source server side* (Endra, et al., 2020).

Web adalah salah satu aplikasi yang berisikan dokumen–dokumen multimedia (teks, gambar, suara, animasi, video) di dalamnya yang menggunakan protokol HTTP (Hyper*text* Transfer Protokol) dan untuk mengakses menggunakan perangkat lunak yang disebut browser. Fungsi website diantaranya (Hasugian, 2018):

1. Media Promosi
2. Media Pemasaran
3. Media Informasi
4. Media Pendidikan
5. Media Komunikasi

2.2.5 *Preprocessing* Data

Data pre-processing merupakan teknik data mining yang melibatkan transformasi data mentah menjadi format yang mudah dimengerti. Langkah data pre-processing diperlukan untuk menyelesaikan beberapa jenis masalah termasuk noisy data, data redudansi, nilai data yang hilang, dan lain-lain. Pre-processing merupakan tahapan data yang diperoleh dikumpulkan menjadi satu dokumen yang kemudian dilakukan analisis (Pratama, et al., 2023).

Adapun tujuan utama dalam preprocessing data ini ialah sebagai berikut: pembersihan data, yaitu mengisi nilai yang hilang, menghaluskan noise data, mengidentifikasi dan menghapus outlier serta menyelesaikan inkonsistensi. Selanjutnya, integrasi data, integrasi beberapa database, kubus data atau file. Selain itu transformasi data, normalisasi dan agregasi. Selain itu juga ada pengurangan data memperoleh penurunan representasi dalam volume tetapi menghasilkan hasil analitis yang sama atau serupadan yang terakhir diskretisasi data, pengurangan data namun sangat penting, terutama untuk data numerik (Khakim, 2022). Adapun langkah-langkah data pre-processing adalah sebagai berikut (Pratama, et al., 2023) :

1. *Case Folding*

   Case folding adalah proses mengubah semua huruf yang ada menjadi huruf kecil. Contoh dari *case folding* yaitu:

Tabel 2.2 Hasil Dari *Case Folding*

Sebelum	**Sesudah**
RT @barikade_98: Dukung Penuh Penggunaan Kendaraan Dinas Listrik	rt @barikade_98 dukung penuh penggunaan kendaraan dinas listrik

2. *Tokenizing*

   Tokenizing adalah proses untuk memecah kata menjadi beberapa bagian. Hasil kata yang sudah dipecah ini yang disebut token. *Tokenizing* juga bisa digunakan untuk menghapus tanda baca yang tidak akan digunakan dalam *preprocessing*. Berikut merupakan contoh dari *tokenizing*.

Tabel 2.3 Hasil Dari *Tokenizing*

Sebelum	**Sesudah**
dukung penuh penggunaan kendaraan dinas listrik	‘dukung’, ‘penuh’, ‘penggunaan’, ‘kendaraan’, ‘dinas’, ‘listrik’

3. *Stopwords Removal*

   Stopwords removal adalah penghapusan kata yang tidak penting atau tidak dibutuhkan berupa kata keterangan dan kata sambung. Berikut merupakan contoh dari *stopwords removal*:

Tabel 2.4 Hasil Dari *Stopwords removal*

Sebelum	Sesudah
dukung penuh penggunaan kendaraan dinas listrik	‘dukung’, ‘penuh’, ‘kendaraan’, ‘listrik’

4. *Stemming*

   Stemming adalah proses mengubah suatu kata menjadi ke bentuk aslinya atau kata dasar. Berikut merupakan contoh dari *stemming*:

Tabel 2.5 Hasil Dari *Stemming*

Sebelum	Sesudah
dukung penuh penggunaan kendaraan dinas listrik	‘dukung’, ‘penuh’, ‘guna’, ‘kendara’, ‘dinas’, ‘listrik’

2.2.6 *Part-of-Speech Tagging*

Part-of-Speech (*PoS) Tagging* merupakan proses dasar dalam mengolah Bahasa pada *Natural Language Processing* (NLP) yang dilakukan secara otomatis oleh teknologi. Contoh penggunaannya pada proses dasar aplikasi question answering (QA), analisis sentimen, dan named entity recognition (NER). *PoS Tagging* berfungsi untuk menghilangkan kata ambigu pada suatu kalimat, terutama pada kata dengan lafal yang sama (*homofon*). Contoh dari kata bang dan bank yang mempunyai kelas kata yang berbeda tetapi dengan lafal yang sama (Firmansyah, et al., 2021).

PoS Tagging memiliki beberapa kelas kata tertentu yang menjadi posisi penting dalam deskripsi suatu kalimat pada bahasa tertentu, kelas kata tersebut biasanya dikenal dengan istilah tagset. *PoS Tagging* yang telah dilakukan pada beberapa bahasa, setiap bahasa memiliki tagset sendiri untuk digunakan pada *PoS Tagging* bahasa tersebut. Setiap tagset dirancang untuk menyesuaikan sintaksis dalam suatu kalimat, karena setiap bahasa memiliki sintaksis yang berbeda. Bahasa Indonesia memiliki tagset sendiri yang telah banyak dirancang, agar tagset sesuai dengan sintaksis Bahasa Indonesia (Firmansyah, et al., 2021).

2.2.7 *Labelling*

Proses labelling dilakukan menggunakan sebuah *corpus* yang sudah tersedia. Metode ini sering disebut dengan Lexicon Based. Lexicon-Based adalah salah satu metode pada permasalahan bahasa alami (natural language processing) yang menggunakan pendekatan berdasarkan kamus yang berisi daftar kata yang mengandung opini dengan skor polaritasnya. Rentang skor polaritas berada pada rentang -1 hingga 1 dimana rentang dengan nilai minus (-) akan masuk kedalam kategori negatif, nilai plus (+) akan masuk kedalam kategori positif dan jika nilai bukan di keduanya maka masuk kedalam kategori netral (Nafan & Amalia, 2019). Artinya cara lexicon-based bekerja nantinya akan membandingkan kata-kata yang telah dilakukan feature extraction denga kamus (dictionary) yang ada di lexicon-based (Hernandez, et al., 2023). Terdapat 1600 kata yang terdapat pada *corpus* positif dan 2573 kata yang terdapat pada *corpus* negatif.

Tabel 2.6 *Corpus* Positif

No.	***Corpus* Positif**
1	a+
2	acungan jempol
3	adaptif
4	adil
5	afinitas
6	afirmasi
7	agilely
8	agung
9	ahli
10	ahlinya
1600	zuki moba

Tabel 2.7 *Corpus* Negatif

No.	***Corpus* Negatif**
1	abnormal
2	absurd
3	acak
4	acak-acakan

No.	*Corpus* Negatif
5	acuh
6	acuh tak acuh
7	adiktif
8	adil
9	agak lama
10	agak ngeri
2573	Yahudi

Selanjutnya tinggal menghitung kata setiap kalimat yang nantinya akan diselesaikan menggunakan formula 2.1.

$$P_d(T) = \sum s(m) \tag{2.1}$$

Dengan:

$P_d(T)$ = Kata pada dokumen

$s(m)$ = Polaritas pada kata dokumen

2.2.8 Delta TF-IDF

Pembobotan *Term Frequency-Inverse Document Frequency* (TF-IDF) adalah salah satu proses dari teknik ekstraksi fitur dengan proses memberikan nilai pada masing-masing kata yang ada pada tweets latih (data latih). Untuk mengetahui seberapa penting sebuah kata mewakili sebuah kalimat, akan dilakukan pembobotan atau perhitungan. Pemberian skor dalam TF-IDF berdasarkan frekuensi munculnya kata dalam dokumen (Pravina, et al., 2019). Inverse document frequency (IDF) adalah jumlah dokumen yang mengandung sebuah term didasarkan pada seluruh dokumen yang ada pada dataset (Amalia & Sibaroni, 2020). Berikut rumus yang digunakan pada *inverse document frequency* (IDF) pada 2.2.

$$tf - idf = tf * \log_2\left(\frac{N_p * C + 0.5}{A * N_n + 0.5}\right) \tag{2.2}$$

Dengan:

N_p = Total dokumen kelas positif

N_n = Total dokumen kelas negative

A = Total kemunculan kata dalam dokumen positif

C = Total kemunculan kata dalam dokumen negatif

Kemudian untuk proses pembobotan dari *term* yang ada menggunakan rumus sebagai berikut :

$$w_{t.d} = tf_{t.d} \times \log\left(\frac{n}{df_i}\right) \quad (2.3)$$

Dengan:

$w_{t.d}$ = Nilai delta TF_IDF untuk term t pada dokumen d

$tf_{t.d}$ = Jumlah term t yang muncul pada dokumen d

$\log\left(\frac{n}{df_i}\right)$ = Invers Document Frequency (IDF)

n = Banyaknya Dokumen

df_i = Banyaknya Dokumen yang memiliki term t

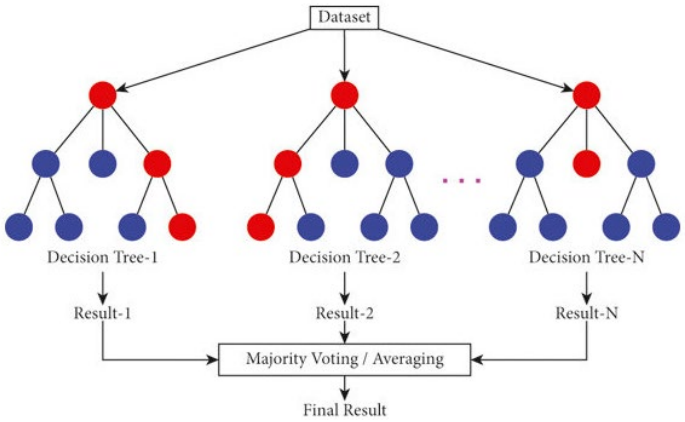
Sedangkan metode pembobotan Delta TF-IDF merupakan keterbaruan dari metode TF-IDF. Delta TF-IDF meningkatkan pentingnya kata-kata yang tidak merata antara kelas positif dan negatif. Pada metode ini di pembobotan memberikan nilai yang berbeda pada dataset kelas positif dan negated sehingga hasil klasifikasi menjadi lebih relevan. Setelah dilakukan pembobotan maka akan terlihat perbedaan nilai antara term yang positif maupun negatif. Rumus Delta TF-IDF dapat dilihat pada persamaan 2.4.

$$\delta tf - idf(t_i) = tf(t_i, d_j) \times log_2\left(\frac{N_p \times C + 0.5}{A \times N_p + 0.5}\right) \quad (2.4)$$

Dimana Np adalah jumlah dokumen dengan class positif; Nn adalah jumlah dokumen dengan class negatif; A adalah adalah jumlah dokumen dengan class label adalah positif dimana term ti ditemukan paling tidak sekali; C adalah adalah jumlah dokumen dengan class label adalah negatif dimana term ti ditemukan paling tidak sekali (Sari, et al., 2023).

2.2.9 *Random Forest*

Random Forest merupakan salah satu metode machine learning yang dimanfaatkan untuk klasifikasi data yang memiliki jumlah banyak (Basar, et al., 2022). *Random Forest* dapat dibangun menggunakan bagging dengan pemilihan atribut acak. Metode CART (*Clasification and Regresion Tree*) sendiri dapat digunakan untuk menumbuhkan pohon keputusan, pohon keputusan tersebut tumbuh hingga ukuran maksimum dan tidak akan dipangkas sehingga dihasilkan kumpulan pohon yang kemudian disebut *forest*. Gambaran algoritma sederhana *Random Forest* dapat dilihat sebagai berikut (Fitri, et al., 2020). Hasil akhir dari metode *Random Forest* diambil dari major voting dari pohon yang terbentuk (Basar, et al., 2022).



Gambar 2.1 Algoritma Sederhana *Random Forest (sumber:.....)*

Metode *CART* (*Clasification* and *Regresion Tree*) menggunakan *information gain* untuk mengukur pemilihan atribut yang akan digunakan pada setiap *node* sebuah *tree*. Rumus menghitung *entropy* total pada persamaan 2.5.

$$Entropy(S) = \sum_{i=1}^{n} - p_i * log_2\ pi \qquad (2.5)$$

Dengan:

S = Jumlah Kasus atau Total Data

p = probabilitas suatu label terhadap jumlah kasus

n = Jumlah partisi

Gain atau *information gain* merupakan kriteria yang paling populer untuk pemilihan atribut, information gain dapat dihitung dari output data atau variabel dependen y yang dikelompokkan berdasarkan atribut A, dinotasikan dengan gain (y,A). Gain (y,A) dari atribut A relatif terhadap output data y. Formula dari information gain pada persamaan 2.6.

$$Information\ Gain\ = Entropy\ (S) - \sum \frac{Si}{S} * Entropy\ (Si) \quad (2.6)$$

Dengan:

S = Himpunan

Si = Probabilitas kriteria tiap fitur terhadap label

S = Jumlah kasus kriteria terkait pada fitur

Setelah mendapatkan nilai entropy setiap kriteria pada setiap atribut, maka selanjutnya dapat menghitung information gain fitur terkait dengan formula di atas. Sebelum itu harus menghitung entropy dua atribut, maksud dua atribut tersebut adalah entropy total dari dari kedua kriteria dengan formula 2.7.

$$Entropy\ (T,x) = \sum_{c \in X} P(c) * E(c) = \sum \frac{Si}{S} * Entropy\ (Si) \quad (2.7)$$

Dengan:

T, X = Atribut T dan Atribut X

P(C) = Probabilitas kelas atribut

E(C) = Nilai Entropy Kelas Atribut

S = Himpunan

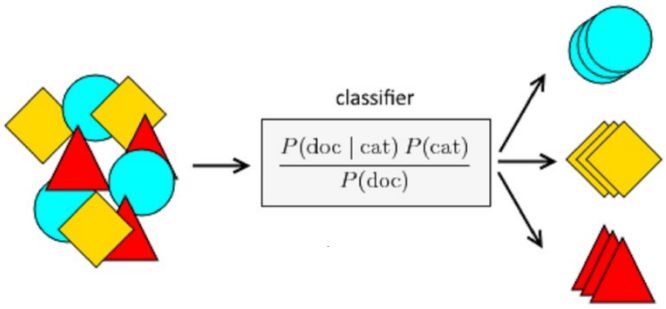
Si = Probabilitas kriteria tiap fitur terhadap label

S = Jumlah kasus kriteria terkait pada fitur

2.2.10 *Naïve Bayes*

Algoritma *naïve bayes* merupakan teknik klasifikasi dengan bentuk model probabilistik dan statistik yang disederhanakan dengan berdasar pada teorema *Bayes* dengan asumsi bahwa setiap atribut bersifat bebas (*independence*). Dengan kata lain, algoritma ini mengasumsikan bahwa ada atau tidak ciri tertentu dari sebuah kelas tidak

ada hubungannya dengan ciri dari kelas lainnya. Berikut pada Gambar 2.2 merupakan skema pada metode naïve bayes (Buttercup, 2021).



Gambar 2.2 Skema Metode *Naive Bayes*

Pada naïve bayes terdapat perhitungan *prior probability,* perhitungan *conditional probability*, dan perhitungan *posterior probability.* Berikut merupakan persamaan 2.8 untuk perhitungan *prior probability,* yang digunakan untuk menghitung probabilitas tiap kelas dari seluruh dokumen.

$$P(C) = \frac{dc}{d} \tag{2.8}$$

Dengan:
dc = Dokumen yang memiliki kelas dimaksud
d = Total dokumen

Setelah menghitung *prior probability* dilakukan perhitungan *conditional probability* dengan model multinomial yang digunakan untuk menghitung *conditional probability* setiap kata yang ada dengan menggunakan persamaan 2.9.

$$P(term|C) = \frac{W_{ct} + 1}{Total\ TF - IDF\ Setiap\ Kelas + Total\ IDF} \tag{2.9}$$

Dengan:
W_{ct} = Nilai pembobotan tf-idf term di dikategori C

Setelah menghitung *conditional probability* selanjutnya dilakukan perhitungan *posterior probability*, dimana perhitungan tersebut untuk menghitung nilai yang akan menentukan data tersebut akan masuk ke salah satu kelas. Berikut merupakan persamaan rumus *posterior probability* pada 2.10.

$$P(D|C) = P(C) * P_1(term \mid C) * .. * P_i(term \mid C) \quad (2.10)$$

Dengan:
$P(c)$ = Nilai Prior Probability Kelas Terkait
$P(term|C)$ = Probabilitas kata terhadap kelas terkait

2.2.11 *Confusion Matrix*

Evaluasi terhadap metode dilakukan untuk mengertahui performa pendekatan yang diusulkan. Pendekatan yang diusulkan dievaluasi menggunakan perhitungan *accuracy* dan *error rate*. Perhitungan *accuracy* dan *error rate* memanfaatkan metode *confusion matrix* (Astari, et al., 2020). *Confusion Matrix* merupakan sebuah tabel yang memberikan informasi perbandingan hasil klasifikasi yang dilakukan oleh sistem (prediksi) dengan hasil klasifikasi yang sebenarnya (Fikri, et al., 2020). Pengukuran kinerja menggunakan *confusion matix,* terdapat 4 istilah diantaranya yaitu sebagai berikut (Nuswantoro, 2021).

1. *True Positive* (TP) yaitu jumlah data positif yang terklasifikasi dengan benar oleh system.
2. *True Negative* (TN) yaitu jumlah data negatif yang terklasifikasi dengan benar oleh system.
3. *False Positive* (FP) yaitu jumlah data positif namun terklasifikasi salah oleh system.
4. *False Negative* (TN) yaitu jumlah data negatif namun terklasifikasi salah oleh system.

Tabel 2.8 *Confusion Matrix*

Kelas	**Terklasifikasi Positif**	**Terklasifikasi Negatif**
Positif	TP (*True Positive)*	FN *(False Negative)*
Negatif	FP (*False Positive)*	TN *(True Negative)*

Berikut merupakan rumus dari *Confusion Matrix*:

1. Akurasi
   Akurasi digunakan untuk mengukur jumlah total prediksi yang benar dibandingkan dengan total data. Dalam menghitung nilai akurasi dapat menggunakan persamaan (2.11).

$$Akurasi = \frac{TP + TN}{TP + TN + FP + FN} * 100\% \quad (2.11)$$

2. Presisi
   Presisi adalah rasio *item* yang relevan dipilih untuk semua *item* dipilih. Dalam menghitung nilai presisi dapat menggunakan persamaan (2.12).

$$Presisi = \frac{TP}{FP + TP} * 100\% \quad (2.12)$$

3. *Recall*
   Recall didefinisikan sebagai rasio *item* yang relevan dipilih dengan jumlah total *item* yang relevan. Dalam menghitung nilai presisi dapat menggunakan persamaan (2.13).

$$Recall = \frac{TP}{FN + TP} * 100\% \quad (2.13)$$

4. *F1-Score*
   F1-Score adalah kombinasi rata-rata *harmonic precision* dan *recall* yang berbanding lurus dengan nilai keduanya. Dalam menghitung nilai F1-*Score* dapat menggunakan persamaan (2.14).

$$\mathrm{F1} - Score = 2 \times \frac{precission * recall}{precission + recall} * 100\% \quad (2.14)$$