BAB II
LANDASAN TEORI

2.1 Tinjauan Pustaka

Untuk membantu penulis dalam penelitian *Web Crawler* untuk mengumpulkan unggahan bertopik wisata yang berlokasi di pulau sumatera, dibutuhkan sebuah *referensi* atau *literature review* sebagai bahan pembelajaran, yang mana *literature review* ini merupakan mempelajari hasil-hasil penelitian yang dilakukan oleh peneliti–peneliti terdahulu yang berkaitan dengan penelitian yang akan dilakukan. Daftar dari literatur tersebut dapat dilihat pada **Tabel 2.1**.

Tabel 2.1 Tabel Tinjauan Pustaka

1	Judul Penelitian	"*Online News Classification Using Multinomial Naive Bayes*"
	Penulis	Amelia Rahman, Wiranto, Afrizal Doewes
	Tahun	2017
	Tujuan Penelitian	Menganalisis berita dengan cara klasifikasi menggunakan *Naive Bayes* model *Multinomial*
	Permasalahan	Masalah yang umum ditemukan dalam proses klasifikasi maupun *clustering* dokumen adalah tingginya dimensi data, sehingga perlu dilakukan proses seleksi fitur untuk memilih beberapa fitur yang dapat digunakan untuk mewakili dokumen
	Subjek Penelitian	
	Metode Penelitian	*Multinomial Naïve Bayes*
	Hasil Penelitian	Metode *Multinomial Naive Bayes* dapat digunakan untuk menganalisis berita dalam teks Bahasa Indonesia ditunjukan dengan hasil akurasi akhir sebesar 94,29%
2	Judul Penelitian	"Klasifikasi Teks dengan Menggunakan Algoritma *K-Nearest Neighbor* pada Kasus Kinerja Pemerintah di *Twitter*"
	Penulis	Octaryo Sakti Yudha Prakasa dan Kemas Muslim Lhaksamana
	Tahun	2018
	Tujuan Penelitian	Tujuan penelitian ini adalah mengklasifikasikan *tweet* sehingga akan diketahui klasifikasi dari *tweet* tersebut (positif atau negatif) kinerja pemerintah dengan menggunakan berbagai skenario

	Permasalahan	Menentukan *tweet* positif atau negatif dari banyaknya *tweet* pengguna *Twitter* tentang kinerja pemerintah
	Subjek Penelitian	*Tweet* tentang kinerja pemerintah
	Metode Penelitian	Algoritma *K-Nearest Neighbor*
	Hasil Penelitian	Skenario pada pengujian dengan perbandingan data training dan testing 80%:20% memiliki akurasi yang lebih tinggi dari pada yang lain yaitu 90.50% dan setiap skenario pengguna *Twitter* lebih banyak berpendapat bahwa kinerja pemerintah untuk periode kali bagus di segala aspek
3	Judul Penelitian	"Klasifikasi Teks Sosial Media *Twitter* Menggunakan *Support Vector Machine*"
	Penulis	Lalu Mutawalli, Mohammad Taufan Asri Zaen, Wire Bagye
	Tahun	2019
	Tujuan Penelitian	Melakukan sentimen analisis teks postingan dan komentar netizen tentang kasus figure wiranto dalam kasus penusukannya
	Permasalahan	Produksi data oleh khalayak di sosial media tersebut memunculkan sekumpulan data yang sangat besar, kompleks, memiliki waktu kemunculan relatif cepat, sehingga menjadikanya sulit untuk ditangani.
	Subjek Penelitian	*Tweet* tentang penusukan Wiranto
	Metode Penelitian	*Support Vector Machine*
	Hasil Penelitian	Hasil sentimen analisis menujukkan 41% memberikan komentar positif, 29% berkomentar secara netral, dan 29% berkomentar secara negatif terhadap kejadian
4	Judul Penelitian	"Klasifikasi *Tweet* Berdasarkan Keterkaitan *Tweet* Terhadap Topik Tertentu pada *Twitter* Menggunakan *Naïve Bayes*"
	Penulis	Muhamad Baydhowi, Widya Apriliah, Ilham Kurniawan
	Tahun	2019
	Tujuan Penelitian	Mengklasifikasikan *tweet* berdasarkan keterkaitan *tweet* terhadap topik tertentu dengan memanfaatkan account yang dinilai punya keterkaitan terhadap topik yang diteliti dengan harapan hasil penelitian ini bisa digunakan untuk mendapatkan *user* yang terkait dengan topik yang telah ditentukan sebagai sasaran marketing dari produk yang berkaitan dengan topik

	Permasalahan	Memanfaatkan data *tweet* tersebut untuk kepentingan tertentu misalkan untuk mengklasifikasikan *tweet* berdasarkan ketertarikan terhadap topik tertentu dengan kriteria yang telah ditentukan
	Subjek Penelitian	
	Metode Penelitian	*Naïve Bayes*
	Hasil Penelitian	Nilai akurasi yang didapatkan dari semua topik termasuk ke dalam kategori kurang baik dengan tingkat kesalahan diatas 40% dan bahkan ada yang mempunyai akurasi dibawah 50%.
5	Judul Penelitian	"Analisa Perbandingan Metode *Naïve Bayes Classifier* Dan *K-Nearest Neighbor* Terhadap Klasifikasi Data"
	Penulis	Aida Indriani
	Tahun	2020
	Tujuan Penelitian	Melakukan klasifikasi judul topik forum mahasiswa secara otomatis sesuai dengan isi materi dan membandingkan 2 metode klasifikasi *Naïve Bayes Classifier* dan *K-Nearest Neighbor*
	Permasalahan	Judul topik yang sudah terlalu banyak di dalam sebuah forum mahasiswa dapat berakibat salah dalam pemilihan judul
	Subjek Penelitian	
	Metode Penelitian	*Naïve Bayes Classifier* dan *K-Nearest Neighbor*
	Hasil Penelitian	Akurasi sebesar 80% untuk metode k-nn dan sebesar 73% untuk nbc yang dihitung dengan menggunakan metode *Confusion* matrix

Berdasarkan tinjauan pustaka diatas maka perbedaan antara penelitian terdahulu dengan penelitian yang dilakukan adalah sebagai berikut:

1. Perbedaan yang terdapat pada penelitian pertama adalah pada data yang dilakukan klasifikasi yaitu berita online sedangkan pada penelitian ini menggunakan data *tweet* dari media sosial *Twitter*. Kemudian perbedaan lainnya adalah peneliti terdahulu hanya menggunakan satu metode klasifikasi yaitu *Multinomial Naive Bayes*, sedangkan pada penelitian ini menggunakan metode klasifikasi *K-Nearest Neighbor*.

2. Perbedaan yang terdapat pada penelitian kedua adalah pada jenis ektraksi fitur yang digunakan peneliti terdahulu menggunakan *TF binary* sedangkan pada penelitian ini menggunakan *TF-IDF*. Kemudian perbedaan lainnya adalah berfokus pada topik unggahan kasus kinerja pemerintahan, sedangkan pada penelitian ini berfokus pada topik unggahan pariwisata Provinsi Lampung.
3. Perbedaan yang terdapat pada penelitian ketiga adalah pada topik *tweet* yang dilakukan klasifikasi yaitu penusukan wiranto, sedangkan pada penelitian ini topik *tweet* yang dilakukan klasifikasi pariwisata Provinsi Lampung. Kemudian perbedaan lainnya adalah peneliti terdahulu menggunakan satu metode klasifikasi yaitu *Support Vector Machine* sedangkan pada penelitian ini menggunakan metode klasifikasi *K-Nearest Neighbor*.
4. Perbedaan yang terdapat pada penelitian keempat adalah pada pengumpulan *tweet* yang berdasarkan akun, sedangkan pada penelitian ini pengumpulan *tweet* berdasarkan keyword nama pariwisata Provinsi Lampung. Kemudian perbedaan lainnya adalah peneliti terdahulu menggunakan satu metode *Naïve Bayes*, sedangkan pada penelitian ini menggunakan metode klasifikasi *K-Nearest Neighbor*.
5. Perbedaan yang terdapat pada penelitian kelima adalah pada data yang digunakan yaitu judul topik dari forum mahasiswa, sedangkan pada penelitian ini data yang digunakan merupakan *tweet* dari *Twitter* yang memiliki keyword nama pariwisata Provinsi Lampung. kemudian perbedaan lainnya adalah peneliti terdahulu membandingkan 2 metode klasifikasi *Naïve Bayes* dengan *K-Nearest Neighbor*, sedangkan pada penelitian ini hanya menggunakan metode klasifikasi *K-Nearest Neighbor*.

2.2 Analisis Sentimen

Analisis sentimen adalah sebuah pendekatan komputasional yang bertujuan untuk mengidentifikasi, memahami, dan mengevaluasi opini, emosi, atau sikap yang terkandung dalam teks atau data non-numerik lainnya. Tujuan utama analisis sentimen adalah untuk mengklasifikasikan teks menjadi kategori positif, negatif, atau netral, sehingga memberikan wawasan tentang preferensi, tren pasar, atau tanggapan publik terhadap suatu topik tertentu.

Analisis sentimen dapat melibatkan beberapa tahapan, termasuk pemrosesan teks, ekstraksi fitur, dan klasifikasi sentimen. Pemrosesan teks melibatkan langkah-langkah seperti tokenisasi (memecah teks menjadi unit-unit terpisah seperti kata-kata), stemming (mengubah kata-kata menjadi bentuk dasar mereka), dan penghilangan stop word (kata-kata yang umum dan tidak memberikan makna khusus). Setelah pemrosesan teks, fitur-fitur yang relevan diekstraksi, seperti kata-kata kunci atau frasa-frasa yang mengindikasikan sentimen. Akhirnya, model klasifikasi sentimen dilatih menggunakan berbagai algoritma, seperti mesin vektor duktor (SVM), naive Bayes, atau jaringan saraf tiruan (neural networks), untuk mengklasifikasikan teks menjadi kategori sentimen yang tepat.

Berbagai penelitian telah dilakukan dalam bidang analisis sentimen. Sebagai contoh, penelitian oleh Pang et al. (2002) menunjukkan bahwa penggunaan kata-kata emosional dan frase-frase seperti "sangat bagus" atau "sangat buruk" dapat digunakan sebagai fitur penting dalam analisis sentimen. Penelitian lain oleh Turney (2002) menunjukkan bahwa penggunaan metode pengukuran asosiasi seperti PMI-IR (Pointwise Mutual Information with Information Retrieval) dapat membantu dalam penentuan polaritas sentimen.

Selain itu, dengan kemajuan dalam bidang pemrosesan bahasa alami dan pengembangan model bahasa yang berbasis pada jaringan saraf, beberapa penelitian terkini telah menunjukkan bahwa pendekatan berbasis pemodelan bahasa seperti BERT (Bidirectional Encoder Representations from Transformers) atau GPT (Generative Pre-trained Transformer) dapat memberikan kinerja yang lebih baik dalam analisis sentimen (Devlin et al., 2019; Radford et al., 2018).

Penggunaan analisis sentimen telah diterapkan dalam berbagai domain, termasuk media sosial, tinjauan produk, survei pelanggan, dan analisis wacana politik. Dalam setiap konteks ini, analisis sentimen dapat memberikan wawasan berharga tentang opini dan sikap yang tersebar di masyarakat.

2.2.1 Lexicon

Metode Lexicon dalam Analisis Sentimen adalah pendekatan yang digunakan untuk mengidentifikasi sentimen dalam teks menggunakan daftar leksikon atau kamus kata kunci yang telah diberi label sentimen. Metode ini dapat diterapkan pada berbagai jenis teks, seperti ulasan produk, posting media sosial, atau komentar pelanggan. Salah satu contoh kamus kata kunci yang sering digunakan dalam metode lexicon adalah "Sentiment Analysis Lexicon" (SAL) yang dikembangkan oleh Bing Liu (2004). Kamus ini memuat daftar kata-kata yang telah diberi label sentimen positif atau negatif berdasarkan analisis manusia. Metode ini menggunakan pencocokan kata-kata dalam teks dengan kata-kata dalam kamus untuk menentukan sentimen umum dalam teks tersebut.

Selain SAL, terdapat juga kamus-kamus lain yang dapat digunakan dalam metode lexicon, seperti "SentiWordNet" yang menggunakan pendekatan berbasis sinonim dan antonim untuk memberikan nilai sentimen pada kata-kata (Wilson et

al., 2005), dan "VADER" (Valence Aware Dictionary and sEntiment Reasoner) yang merupakan kamus berbasis aturan yang mengkombinasikan kata-kata dengan pola tanda baca dan kata-kata penegas untuk menentukan sentimen (Hutto & Gilbert, 2014).

Metode lexicon dapat diterapkan dengan beberapa teknik, seperti "bag-of-words" atau "bag-of-ngrams", yang melibatkan representasi teks sebagai kumpulan kata atau n-gram (gabungan n kata) tanpa mempertimbangkan urutan kata-kata tersebut. Selain itu, teknik-teknik seperti normalisasi teks, stemming, atau lemmatisasi juga dapat diterapkan sebelum analisis sentimen.

2.3 Media Sosial

Menurut Nasrullah (2015) media sosial adalah medium di internet yang memungkinkan pengguna merepresentasikan dirinya maupun berinteraksi, bekerjasama, berbagi, berkomunikasi dengan pengguna lain membentuk ikatan sosial secara virtual. Media Sosial memberikan ruang bagi kaum muda dan mudi untuk saling berinteraksi satu sama lain. Berbagai *Platform* Media Sosial yang populer bagi kaum muda dan mudi, diantaranya adalah *Facebook*, *Instagram*, *Twitter*, dan *Path* (Rukmiyati & Suastini, 2016).

2.4 *Python*

Python merupakan bahasa pemrograman yang popular pada komunitas ilmiah, terutama pada bidang sains dan teknik, dikembangkan oleh Guido van Rossum pada tahun 1990 di Centrum *Wiskunde & Informatica* (CWI), Amsterdam sebagai kelanjutan bahasa pemrograman ABC. Dikarenakan *Python* bersifat *open source* pengembangan *Python* terus dilakukan oleh sekumpulan *programmer* yang

dikoordinasi oleh *Guido da Python Software Foundation* dan distribusi *Python* sudah mencapai versi 3.5 (Budiharto, 2018)

Berbagai *library* yang terdapat pada *python* sangat mendukung untuk melakukan penelitian ilmiah, seperti *data mining, image processing, text processing, web crawling,* dan lain-lain. *library* yang dapat digunakan untuk melakukan proses web crawling yaitu, *Tweepy*, *Scrapy, Beautiful Soup 4, Requests, Urllib2, LXML, Selenium*, *PySpider* dan lain-lain.

2.5 *Twitter API*

Twitter API adalah *API* (*Application Programming Interface*) yang disediakan oleh *Twitter* untuk memfasilitasi pengguna agar dapat berinteraksi dengan data-data yang ada pada *Twitter* contohnya seperti *tweet*, id pengguna, lokasi, dan waktu pembuatan *tweet*. Bahasa *server side scripting* yang digunakan dalam *Twitter API* diantara seperti *PHP, Python, R* dan lain-lain. Penggunaan bahasa-bahasa tersebut, memungkinkan pengguna untuk melakukan *request* kepada *Twitter API,* dan respon hasilnya dimuat dalam format *JSON*. *Twitter* menerapkan *OAuth* (*Open Authorization*) yang berguna untuk keamanan komunikasi antar pengguna dengan *Twitter API*. *OAuth* adalah protokol terbuka yang memungkinkan pengguna untuk berbagi resource pribadi seperti foto, video, data pengguna dan lain-lain yang tersimpan di suatu situs web, tanpa memberikan nama pengguna dan kata sandi pengguna. *OAuth* mengizinkan pengguna untuk memberikan akses kepada situs pihak ketiga untuk mengakses informasi yang tersimpan di penyedia layanan lain tanpa harus membagi izin akses keseluruhan data (Saputra, 2017).

OAuth bergantung pada tiga set dari token dan *secret* yang didapat dari server dan *Client*, yaitu :

1. *Consumer key* dan *consumer secret*, merupakan unique identifier *Client* yang digunakan untuk melakukan *request* supaya mendapatkan *request* token.
    a. Contoh *consumer key* :

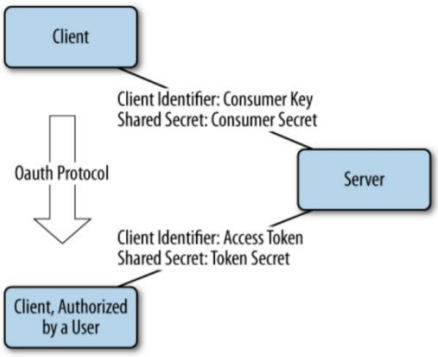
       T1NuLdkR9ACB4GBEOc1Qvw
    b. Contoh *consumer secret* :

       RGwVy9zru6evNMez3il4qLke9UYaW5RCgdx8kfyKKh
2. *Request token* adalah *temporary one-time identifier* yang diberikan oleh server untuk tujuan permintaan pada *user* untuk melakukan *grant permission* pada *Client*. *Token secret* digunakan untuk melakukan *sign request* agar mendapatkan sebuah *access token*.
3. *Access token* adalah *indetifier* yang digunakan oleh *Client* untuk melakukan akses ke *resources* milik *user*. *Client* dapat melakukan akses ke *resources* milik *user* selama token masih valid. Sedangkan server dapat melakukan *revoke* kapanpun karena sudah *expire* atau *user* melakukan *revoke* secara manual. *Secret* digunakan untuk *sign request* ke *resources* yang di proteksi oleh akses *user*.
    c. Contoh *access token* :

       45014365-GUPXF66BQ0t3nLFpT7rkLFPuXBGn4XbH5WyjCwXlu
    b. Contoh *access token secret* :

       xQNMgYntxxj505w50XDHS0m0MU3NrquDOE2VUjUk

OAuth dapat digunakan melalui tahap-tahap berikut, yang bertujuan untuk mendapatkan *access token* dan *secret*. Server bisa mengizinkan *access token* untuk jangka waktu tertentu atau membatasi akses ke *resources* tertentu milik *user* saja, alur dari *OAuth* dapat dilihat pada Gambar 2.2



Gambar 2.1 *OAuth Flow*

1. *Client* melakukan *request* ke server menggunakan sebuah *consumer key*.
2. *Client* menggunakan *consumer key* untuk mendapatkan sebuah *request* token dan *secret*.
3. *Client* melakukan *redirect* pada *user* ke server untuk *grant permission* untuk *Client* melakukan akses ke *resources* milik *user*. Proses ini bisa terjadi jika *request* token telah diautentikasi.
4. *Client* melakukan *request* ke server untuk memberikan *access token* dan *secret*. hanya merepresentasikan sebuah indentifier dan shared *secret* yang *Client* nya bisa gunakan untuk mengakses *resources* atas nama *user*.
5. Ketika membuat sebuah *request* untuk akses ke *resources* terproteksi, *Client* menyatakan *Authorization header* yang berisi *consumer key*. *Access token*, *signature method* dan sebuah *signature*, *timestamp*, sebuah *nonce*, dan untuk opsionalnya adalah versi dari *OAuth* yang digunakan.

2.6 *Preprocessing*

Dalam melakukan *crawling* dibutuhkan suatu proses yang dapat memudahkan dalam pengelolaan informasi dari suatu data *crawling* yang dimana proses tersebut dinamakan *Preprocessing*. *Preprocessing* merupakan tahapan awal

yang dilakukan terhadap dokumen dengan mengubah teks menjadi *term index* yang bertujuan untuk menghasilkan sebuah *set term index* yang dapat digunakan sebagai kata kunci untuk mewakili dokumen tersebut. Menurut Rasywir dan Purwarianti (2016) tahap *Preprocessing* terdiri dari pemrosesan leksikal yang bertujuan untuk memproses token kata. Leksikal sendiri merupakan sebuah makna yang mempunyai arti sebenar-benarnya yang dijelaskan oleh kata tersebut, dalam artian kata yang bermakna leksikal merupakan kata yang memiliki makna langsung (Prasetyo, Rino et al., 2018). Pemrosesan leksikal sendiri terdiri atas tokenisasi, *case folding*, penghapusan *stopword*, dan *stemming.*

2.6.1 Tokenisasi

Tokenisasi merupakan proses pemecahan atau membagi teks yang berupa kalimat menjadi kata atau frase. Pada umumnya pemisah antar kata dalam proses tokenisasi adalah spasi dan tanda baca. Dalam penelitian Rasywir dan Purwarianti (2016) tokenisasi digunakan untuk memisahkan kata pada kalimat dengan menggunakan penanda spasi dan kemudian kata yang sudah dipisah tersebut dijadikan larik (baris).

2.6.2 *Case Folding*

Case Folding merupakan penghilangan karakter selain huruf pada saat pengambilan informasi dan juga melakukan pengubahan huruf menjadi huruf kecil (‘a’ sampai ‘z’). Case folding digunakan dengan asumsi bahwa informasi huruf kapiapital pada dokumen berita tidak mempengaruhi hasil dari teks. Pada case folding, karakter selain huruf dianggap sebagai delimeter (pembatas) sehingga karakter tersebut dihapus dari dokumen. Case folding sendiri bertujuan untuk menghilangkan noise pada saat pengambilan informasi.

2.6.3 Penghapusan *Stopword*

Penghapusan *stopword* merupakan proses pembuangan kata yang sering muncul dan tidak terpakai. Penghapusan ini digunakan untuk menghilangkan fitur kata yang tidak penting yang dapat menganggu proses klasifikasi. Penghapusan *stopword* bertujuan untuk mengurangi volume kata, *stopword* sendiri dapat berupa kata depan, kata *penggabung*, dan kata pengganti. Contoh dari *stopword* bahasa Indonesia adalah “yang”, “ini”, “dari”, “ke”, “di”, “dan”. Berpengaruh tidaknya penghilangan *stopword* tergantung pada jenis klasifikasi dan data yang terkumpul.

2.6.4 *Stemming*

Stemming merupakan proses pemotongan imbuhan atau pengembalian kata berimbuhan menjadi kata dasar, dimana setiap kata yang berimbuhan seperti “di”, “ke”, “mem”, “meng”, dan sebagainya akan diubah menjadi kata dasar. Inti dari proses *stemming* sendiri bertujuan untuk mengurangi variasi kata yang sebenarnya memiliki kata dasar yang sama.

2.7 TF-IDF

TF-IDF (*Term Frequency-Inverse Document Frequency*) merupakan salah satu metode dari pembobotan kata, yang dimana mengubah kata dalam tiap dokumen yang telah melalui proses *Preprocessing* menjadi numerik. TF-IDF (*Term Frequency-Inverse Document Frequency*) digunakan untuk menentukan seberapa berkaitannya kata dalam sebuah dokumen terhadap dokumen dengan cara memberikan bobot pada setiap kata. Metode TF-IDF (*Term Frequency-Inverse Document Frequency*) merupakan perpaduan antara dua konsep yaitu, *Term Frequency* yang menghitung jumlah kemunculnya kata dalam sebuah dokumen dengan *Inverse Document Frequency* yang menghitung invers dari jumlah

kemunculan dokumen. Rumus menghitung TF-IDF terdapat pada persamaan 2.1 dan 2.2(Rasywir & Purwarianti, 2016):

$$tf_{t,d} * idf$$

(2.1)

Keterangan :

tf : jumlah kemunculan kata (*t*)

$tf_{t,d}$**idf* : kemunculan kata (*t*) setiap dokumen (*d*)

idf : kemunculan kata (*t*) pada semua dokumen (pembobotan global)

$$idf = \log \frac{N}{df_t}$$

(2.2)

N : banyaknya dokumen (*d*)

df_t : jumlah dokumen yang mengandung kata (*t*)

2.8 *K-Nearest Neighbor*

KNN (*K-Nearest Neighbor*) merupakan sebuah algoritma klasifikasi dengan berdasarkan pada jarak K antara data uji dengan data latih. KNN(*K-Nearest Neighbor*) merupakan salah satu algoritma yang populer dengan akurasi tinggi dan mudah dipahami (Prasanti et al., 2018). Nilai K pada algoritma ini sangat mempengaruhi tingkat akurasi, semakin tinggi nilai K semakin tinggi juga akurasinya. Algoritma ini mirip dengan teknik *clustering*, dengan mengelompokkan data baru berdasarkan jarak data baru dengan beberapa data dengan nilai terdekat. Untuk menentukan jarak antara dua data yaitu jarak antara data latih dan data uji, dilakukan perhitungan terhadap data tersebut menggunakan rumus *Euclidean Distance*. *Euclidean Distance* sangat sering diandalkan dalam perhitungan jarak. Jarak *Euclidean* digunakan untuk menguji ukuran yang dapat

berfungsi sebagai acuan kedekatan jarak antara data uji dengan data latih, yang dimana perhitungannya dapat dilihat pada persamaan 2.3 (Prakasa & Lhaksmana, 2018)

$$dist = \sum_{i=1}^{p} \sqrt{(x2 - x1)^2}$$

(2.3)

Keterangan :

dist : jarak

x1 : data latih

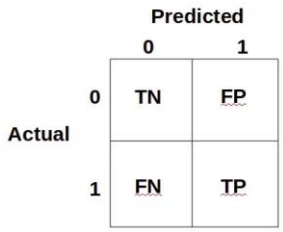
x2 : data uji

i : variable data

p : Jumlah atribut

2.9 *Confusion Matrix*

Confusion Matrix digunakan untuk mengukur performa dari suatu klasifikasi yang dimana keluarannya dapat berupa 2 kelas atau lebih. Bentuk dari *Confusion Matrix* adalah berupa tabel dengan 4 kombinasi berbeda dari nilai klasifikasi dan nilai asli/aktual (Narkhede, 2018). Ada 4 istilah dalam *Confusion* Matrix yaitu *True Positif, True Negatif, False Positif,* dan *False Negatif* yang dapat dilihat pada Gambar 2.3



Gambar 2.2 *Confusion Matrix*

Angka 0 merupakan label untuk negatif, sedangkan Angka 1 untuk label positif, Sedangkan penjelasan 4 istilah dalam *Confusion Matrix* adalah sebagai berikut :

1. *True Negative* (TN): Model memklasifikasi data ada di kelas Negatif dan yang sebenarnya data memang ada di kelas Negatif.
2. *True Postive* (TP): Model memklasifikasi data ada di kelas Positif dan yang sebenarnya data memang ada di kelas Positif.
3. *False Negative* (FN): Model memklasifikasi data ada di kelas Negatif, namun yang sebenarnya data ada di kelas Positif.
4. *False Positive* (FP): Model memklasifikasi data ada di kelas Positif, namun yang sebenarnya data ada di kelas Negatif.

2.10 *Accuracy*

Accuracy adalah metrik evaluasi yang digunakan untuk mengukur sejauh mana model klasifikasi dapat melakukan klasifikasi yang benar. Metrik ini menghitung persentase kesesuaian antara jumlah klasifikasi yang benar (True Positive dan True Negative) dengan total jumlah sampel yang diamati. *Accuracy* dinyatakan dalam persentase atau dalam rentang nilai 0 hingga 1. *Accuracy* dapat dirumuskan seperti pada persamaan 2.4.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad \textbf{(2.4)}$$

Keterangan :

TP = *True Positive*

TN = *True Negatif*

FP = *False Positive*

FN = *False Negatif*

2.11 *Precision* dan *Recall*

Precision dan *Recall* merupakan salah dua pengujian yang dapat dilakukan setelah melalui proses *Confusion Matrix* dengan memanfaatkan nilai dari *True Positif, False Positif,* dan *False Negatif*. *Precision* sendiri merupakan perbandingan antara *True Positive* (TP) dengan banyaknya data yang diklasifikasi positif. *Precision* menggambarkan akurasi antara data asli dengan data hasil klasifikasi yang dihasilkan dari model. *Precision* dapat dirumuskan seperti pada persamaan 2.8(Hariyani et al., 2020).

$$precision = \frac{TP}{TP + FP}$$

(2.5)

Keterangan :

TP = *True Positive*

FP = *False Positive*

Sedangkan, *Recall* merupakan perbandingan antara *True Positive* (TP) dengan banyaknya data yang sebenarnya positif. *Recall* menggambarkan keberhasilan model dalam menemukan kembali sebuah informasi. *Recall* dapat dirumuskan seperti pada persamaan 2.9 (Hariyani et al., 2020).

$$recall = \frac{TP}{TP + FN}$$

(2.6)

Keterangan :

TP = *True Positive*

FN = *False Negatif*

2.12 *F-Measure*

Pengujian *F-Measure* dapat diterapkan setelah mendapatkan nilai *Precision* dan *Recall*, dikarenakan pengujian *F-Measure* digunakan untuk mengukur nilai rata-rata antara *Precision* dan *Recall* yang dibobotkan. *F-Measure* dapat dijadikan acuan jika jumlah data *False Negative* dan *False Positive* pada *Dataset* tidak mendekati. *F-Measure* dapat dirumuskan seperti pada persamaan 2.10 (Hariyani et al., 2020)

$$FMeasure = \frac{2 \times recall \times precision}{recall + precision} \quad \textbf{(2.7)}$$