BAB II

LANDASAN TEORI

2.1 Penelitian Terdahulu

Terdapat beberapa penelitian mengenai analisis sentimen menggunakan algoritma SVM dan LR serta menggunakan SNA untuk mengidentifikasi akun-akun berpengaruh dalam penyebaran informasi. Oleh karena itu, Tabel 2.1 menampilkan beberapa penelitian terdahulu yang dijadikan sebagai acuan dan referensi untuk pengerjaan penelitian ini.

Tabel 2.1 Penelitian Terdahulu

No	Judul, (Vol, No, Tahun)	Penulis	Hasil Penelitian
1	"Analisis Sentimen Masyarakat Indonesia terhadap Pemindahan Ibu Kota Negara Indonesia pada *Twitter*" (Vol.8, No.1, 2022) [18].	Sri Lestari, Mupaat Mupaat, Adhitia Erfina	SVM memiliki akurasi tertinggi yaitu 85,71%, diikuti nilai 76,70% untuk algoritma *NB*. Sementara itu, algoritma KNN memperoleh nilai yang lebih rendah sebesar 52,74%.
2	"Analisis Sentimen Relokasi Ibukota Nusantara Menggunakan Algoritma *Naïve Bayes* dan KNN" (Vol. 10, No.1, 2023) [25].	Syahril Dwi Prasetyo, Shofa Shofiah Hilabi, Fitri Nurapriani	*K-Nearest Neighbors* (KNN) menghasilkan akurasi sebesar 88.12%, lebih unggul dibandingkan dengan *Naïve Bayes* (NB) yang memiliki akurasi 82.27%.
3	Analisis Sentimen Pindah Ibu Kota Negara (IKN) Baru pada *Twitter* Menggunakan Algoritma *Naive Bayes* dan *Support Vector Machine* (SVM) (Vol. 16, No. 3, 2023) [19]	Amril Mutoi Siregar	Penelitian ini menggunakan metode klasifikasi NB dan SVM, hasil akurasi menunjukkan 86.94% untuk NB dan 90.81% untuk SVM.
4	*"Twitter sentiment analysis of the relocationof Indonesia's capital city"* (Vo.9, No.4, 2020) [14]	Edi Sutoyo, Ahmad Almaarif	Hasil penelitian menunjukkan SVM memiliki kinerja terbaik dalam mengklasifikasikan sentimen *tweet* dibandingkan dengan algoritma lainnya. SVM menghasilkan akurasi 97.72%, LR 96.58%, NB 91.65%, dan KNN 90.70%.
5	"Analisis Perbandingan Algoritma *Machine Learning* Terhadap Sentimen Analis Pemindahan Ibu Kota	Arif Rahman Hakim, Windu Gata, Alda Zevana Putri Widodo, Oky Kurniawan,	Data teerdiri dari 489 sentimen negatif dan 297 sentimen positif. Hasil penelitian menunjukkan bahwa penggunaan SMOTE dapat meningkatkan akurasi model, SVM memiliki nilai akurasi paling tinggi sebesar

	Negara" (Vol.7, No.2, 2023) [20]	Arief Rama Syarif	82.82%, NB 81.18% dan *Random Forest* (RF) 79.55%
No	**Judul, (Vol, No, Tahun)**	**Penulis**	**Hasil Penelitian**
6	*"Sentiment Analysis of The Opinion of Moving The Capital City on Twitter with The Support Vector Machine Method"* (Vol.14, No.1, 2021) [15]	Tezza Fazar Tri Hidayat, Garno Garno, Azhari Ali Ridha	Hasil penelitian menunjukkan bahwa akurasi klasifikasi sentimen menggunakan SVM mencapai 77.72%. Ketika dikombinasikan dengan TF-IDF, akurasi model meningkat menjadi 78.33%
7	Sentimen Analisis Twitter Ibu Kota Negara Nusantara Menggunakan Long Short-Term Memory dan Lexicon Based (Vol. 12, No.2, 2022) [26]	Saepul Aripiyanto, Tukino Tukino, Ammar Sufyan, Riandi Nandaputra	Hasil penelitian menunjukkan bahwa sentimen positif tentang IKN Nusantara lebih dominan dibandingkan dengan sentimen negatif dan netral. Dari 5112 data tweet yang dianalisis model menghasilkan nilai akurasi sebesar 79% serta kata-kata yang paling sering muncul dalam tweet adalah "IKN", "IKN Nusantara", "kota", "bangun", dan "story".
8	*"Sentiment Analysis of Indonesian New Capitol (IKN) on Twitter Using Classification Algorithm"* (2023) [16]	Lutfi Aditya, Wibowo, Nur Yunaidah Pratiwi, Martin Suhartana, Emny Hama Yossy	Algoritma SVM dengan *kernel linear* memberikan akurasi tertinggi dalam klasifikasi sentime sebesar 89.77%, *kernel RBF* 88.69%, NB 79.26%, dan RF 66.88%. Secara keseluruhan, distribusi sentimen yang ditemukan adalah 38.67% positif, 12.94% negatif, dan 48.39% netral. Dapat disimpulkan bahwa wacana pembangunan IKN menerima lebih banyak umpan balik positif dibandingkan negatif.
9	*"Mobilizing the Digital Opinion Movement #OraSudiSumbangIKN on Twitter"* (Vol.15, No.1, 2023) [22]	Ratih Anbarini, S. Kunto Adi Wibowo, Nuryah Asri Sjafirah, Aceng Abdullah	Hasil Penelitian menunjukkan bahwa tagar #OraSudiSumbangIKN mendapat perhatian besar di *Twitter* dengan jumlah tweet yang tinggi. Beberapa akun seperti @tjeloup1, @papa_loren, dan @bob_et3k3wer memiliki peran penting dalam penyebaran informasi pada tagar #OraSudiSumbangIKN karena memiliki nilai *degree centrality* yang tinggi.
10	*Social Network Analysis:* Penyebaran Informasi Pembangunan Ibu Kota Negara (IKN) di *Twitter* (Vol 1, No.1, 2022) [23]	Victoria Sundari Handoko, Antonius Budisusila	Penelitian ini menggunakan dua metode, yaitu *Social Network Analysis* (SNA) dengan menghitung nilai *degree centrality* dan *Wordcloud Analysis*. Hasil penelitian menunjukkan @korantempo adalah akun paling berpengaruh dalam diskusi tentang IKN di *Twitter*. Selain itu, kata-kata yang sering muncul dalam pembicaraan tentang IKN adalah kota, negara, dan triliun.

Berdasarkan analisis dari beberapa penelitian terdahulu, yang telah dilakukan mengenai IKN sebagaimana yang diuraikan dalam Tabel 2.1, penelitian ini bertujuan untuk menganalisis opini masyarakat terkait dengan pembangunan IKN.

Berbeda dari penelitian-penelitian sebelumnya yang lebih banyak membahas sentimen terkait wacana pemindahan IKN, fokus penelitian ini adalah memahami reaksi dan pandangan masyarakat terhadap fase pembangunan infrastruktur yang sedang berlangsung, termasuk proyek-proyek besar seperti pembangunan istana kepresidenan, jalan tol, dan perkantoran pemerintah. Penelitian akan menggunakan algoritma *Support Vector Machine* (SVM) dan *Logistic Regression* (LR) untuk menganalisis sentimen, dan selanjutnya membandingkan tingkat akurasi kedua algoritma tersebut. Kedua algoritma ini dipilih karena dapat mengangani berbagai jenis data dan menghasilkan akurasi yang tinggi dalam menganalisis sentimen [18], [19], [20], [15], [16], sedangkan LR dipilih karena menghasilkan *output* dalam bentuk probabilitas sehingga memudahkan interpretasi sentimen serta menghasilkan nilai koefisien yang membantu dalam memahami fitur-fitur yang mempengaruhi sentimen. Selain itu, dengan menggunakan metode *Social Network Analysis* (SNA), penelitian ini akan mengidentifikasi akun-akun yang berperan dalam membentuk dan menyebarkan informasi terkait pembangunan IKN, berdasarkan metrik seperti *degree centrality*, *betweenness centrality*, dan *closeness centrality.* Data yang digunakan pada penelitian ini adalah data terbaru dari periode 1 November 2023 hingga 31 Januari 2024, sehingga memungkinkan analisis yang relevan terhadap sentimen masyarakat saat ini.

2.2 Tinjauan Teori

2.2.1 Ibu Kota Nusantara

Ibu Kota Nusantara (IKN) merupakan kebijakan pemindahan pusat administrasi Indonesia dari Jakarta ke Kalimantan Timur, tepatnya di Kabupaten Penajam Paser Utara dan sebagian Kutai Kartanegara. Keputusan ini merupakan respons terhadap berbagai tantangan yang dihadapi Jakarta, seperti kemacetan lalu lintas, polusi dan krisis air. Pemindahan IKN bertujuan untuk mendorong pemerataan pembangunan ekonomi di seluruh wilayah Indonesia, serta mengurangi ketimpangan di Pulau Jawa, khususnya Jakarta sebagai pusat kegiatan ekonomi dan politik. Pemindahan IKN diharapkan dapat memacu pertumbuhan di wilayah lain dan mengurangi beban Jakarta yang saat ini sudah terlalu padat [27]. Pemilihan Kalimantan Timur didasarkan pada

berbagai faktor, termasuk lokasi geografis yang strategis, yaitu berada di tengah-tengah wilayah Indonesia serta memiliki ancaman bencana alam yang relatif kecil [28], [29]. Selain itu, Kalimantan Timur berdekatan dengan wilayah perkotaan yang telah berkembang seperti Balikpapan dan Samarinda, memungkinkan penekanan biaya pembangunan dengan memanfaatkan infrastruktur yang telah ada. Keberadaan lahan pemerintah dan BUMN seluas 180.000 hektar, juga membantu penghematan biaya untuk akuisisi lahan pembangunan IKN [30].

Gagasan pemindahan ibu kota negara Indonesia telah ada sejak era Presiden Soekarno. Pada tanggal 17 Juli 1957, Presiden Soekarno mengusulkan pemindahan ibu kota ke Palangkaraya, Kalimantan Tengah, karena lokasinya yang strategis di tengah Indonesia dan ideal untuk pengembangan ibu kota yang modern, serta untuk meratakan pembangunan di seluruh wilayah Indonesia [1]. Wacana pemindahan ibu kota kembali muncul pada era Orde Baru sekitar tahun 1990 dengan usulan untuk menjadikan Jonggol sebagai ibu kota baru. Pada masa pemerintahan Presiden Susilo Bambang Yudhoyono, masalah kemacetan dan banjir di Jakarta memicu munculnya kembali diskusi terkait pemindahan ibu kota. Berbagai opsi dipertimbangkan, mulai dari memperbaiki Jakarta sebagai ibu kota, memindahkan ibu kota ke daerah lain, hingga membangun ibu kota baru [27]. Namun, pada masa pemerintahan Presiden Joko Widodo, rencana pemindahan ibu kota negara baru terealisasi.

Pada tanggal 29 April 2019, pemindahan ibu kota negara diumumkan [2]. Presiden Joko Widodo memutuskan untuk memindahkan IKN keluar Pulau Jawa, yang kemudian ditetapkan melalui Rencana Pembangunan Jangka Menengah Nasional (RPJMN) 2020-2024 [31]. Keputusan ini didasarkan pada urgensi untuk pemerataan pembangunan ekonomi dan populasi penduduk yang terlalu terfokus di Pulau Jawa, serta mengatasi berbagai masalah lingkungan dan sosial yang dihadapi Jakarta. Pemindahan IKN diresmikan melalui Undang-Undang Nomor 3 Tahun 2022 tentang Ibu Kota Negara, memberikan dasar hukum untuk proses pemindahan dan pembangunan infrastruktur di Kalimantan Timur. UU ini juga menetapkan tahapan pemindahan akan dilakukan secara

bertahap mulai dari tahun 2024, dimulai dengan pemindahan istana dan beberapa kementerian [32].

Pembangunan IKN Nusantara direncanakan dalam lima tahapan utama, yang mencakup periode dari tahun 2020 hingga 2045 [33], [3].

1. Tahap Pertama (2020-2024)

   Tahap ini merupakan fase awal pemindahan ke kawasan IKN berfokus pada pembangunan infrastruktur seperti jalan, istana kepresidenan dan perkantoran pemerintah. Tahap ini bertujuan untuk mempersiapkan IKN sebagai pusat pemerintahan baru dengan infrastruktur yang memadai untuk mendukung operasional pemerintahan.
2. Tahap Kedua (2025-2029)

   Melanjutkan pembangunan infrastruktur yang telah dimulai pada tahap pertama dan memperluas kawasan perkotaan. Tahap ini akan mencakup pengembangan fasilitas transportasi umum dan perluasan kawasan pemukiman untuk Aparatur Sipil Negara (ASN) serta perkantoran pemerintah pusat.
3. Tahap Ketiga (2030-2034)

   Pengembangan kawasan industri dan sektor ekonomi lainnya yang mendukung pertumbuhan ekonomi IKN. Selain itu, pembangunan infrastruktur pendukung seperti kereta api dan fasilitas lainnya akan diperkuat, serta melakukan pemindahan lanjutan untuk personel TNI/Polri.
4. Tahap Keempat (2035-2039)

   Membangun semua infrastruktur dan ekosistem tiga kota (IKN Nusantara, Balikpapan, Samarinda) yang akan memberi dampak ekonomi signifikan bagi wilayah Kalimantan Timur. Akan ada pengembangan di bidang pendidikan, kesehatan, serta penguatan ketahanan sosial-budaya.
5. Tahap Kelima (2040-2045)

   Berfokus untuk pada mengokohkan reputasi IKN sebagai Kota Dunia Untuk Semua [3]. Target tahapan akhir ini adalah mencapai *net zero carbon emission* dan menggunakan 100% energi terbarukan, serta

menstabilkan pertumbuhan penduduk di IKN. Tahap ini bertujuan untuk memastikan bahwa IKN menjadi kota yang inklusif, hijau, dan berkelanjutan.

2.2.2 *Twitter*

Gambar 2.1 Logo *Twitter* atau *X* [34]

Twitter atau *X* merupakan *platform* media sosial yang dikembangkan pada tahun 2006 oleh Noah Glass, Biz Stone, dan Evan Williams. *Twitter* dikenal sebagai *platform* media sosial dengan format *microblogging*, yang memungkinkan pengguna untuk secara *real-time* mengirim dan membaca pesan singkat yang disebut *tweet* dengan batasan 280 karakter [35]. Di Indonesia, *Twitter* merupakan salah satu *platform* media sosial yang populer digunakan oleh masyarakat [8] dan telah menjadi sumber data yang sering dimanfaatkan untuk berbagai penelitian, terutama yang berkaitan dengan opini publik dan analisis sentimen.

Beberapa fitur utama *Twitter* adalah mengikuti akun lain, yang memungkinkan pengguna mendapatkan pembaruan *tweet.* Pengguna juga dapat berinteraksi dengan *tweet* tersebut melalui fitur *retweet, like,* dan komentar [36]. *Twitter* juga menyediakan fitur *hashtag (#)* untuk mengelompokkan *tweet* berdasarkan topik tertentu atau isu tertentu, sehingga memudahkan pengguna dalam menemukan dan berpartisipasi dalam diskusi.

2.2.3 Analisis Sentimen

Analisis sentimen merupakan salah satu bidang ilmu *Natural Language Program (NLP)* yang diterapkan untuk mengevaluasi pendapat, sikap, dan

emosi seseorang terhadap suatu topik yang diungkapkan dalam bentuk teks [7], [37]. Analisis sentimen bertujuan untuk memahami dan mengklasifikasikan teks ke dalam kategori sentimen tertentu, seperti positif dan negatif. Kategori tersebut akan menjadi parameter dalam menentukan pandangan atau emosi seseorang mengenai topik yang diajukan. Dalam penggunaanya, analisis sentimen memiliki beberapa pendekatan berdasarkan teknik yang digunakan, yaitu:

1. *Machine Learning*

   Pendekatan *Machine Learning* didasarkan pada data yang telah di label dan data yang digunakan perlu melalui proses pelatihan dan pengujian untuk membangun dan memverifikasi model yang akurat [37]. Pendekatan *machine learning* di bagi menjadi dua, yaitu *supervised learning* yang menggunakan data berlabel untuk melatih model, dan *unsupervised learning* yang merupakan data tanpa label, bertujuan untuk mengidentifikasi pola tersembunyi atau struktur dalam data. *Naive Bayes, Support Vector Machine,* dan *Random Forests* merupakan algoritma yang paling sering digunakan pada pendekatan *machine learning* [37]. Keunggulan penggunaan metode *machine learning* terletak pada akurasinya yang lebih tinggi, terutama dalam mengidentifikasi dan menginterpretasikan sentimen dalam teks.

2. *Lexicon-Based*

   Metode *Lexicon-Based* dalam analisis sentimen menggunakan kamus lexicon untuk menilai kata-kata sebagai positif, negatif, atau netral [37]. Penilaian terhadap tiap kata umumnya diberikan nilai (+1) untuk kata dengan sentimen positif, (-1) untuk kata negatif dan (0) untuk netral [38]. Nilai dari setiap kata kemudian dijumlahkan dan dirata-rata untuk menghasilkan skor sentimen suatu teks. Metode *lexicon* dibagi menjadi tiga tahap: *word-level,* menghitung frekuensi kata-kata yang memiliki sentimen positif, negatif, atau netral; *sentence-level,* menghitung sentimen untuk setiap kalimat; dan *document-level,* menghitung sentimen keseluruhan berdasarkan skor sentimen dari setiap kalimat dalam dokumen tersebut [39].

InSet Lexicon, SentiWordNet, Liu Lexicon, AFINN Lexicon, dan Vania Lexicon merupakan beberapa kamus sentimen yang sering digunakan [39].

3. *Hybrid Approach*

Metode *hybrid* merupakan pendekatan yang menggabungkan dua atau lebih metode analisis sentimen, seperti metode *lexicon-based* dan metode *machine learning* [37]. Pendekatan ini bertujuan untuk memanfaatkan kelebihan dari masing-masing metode dan mengatasi keterbatasan yang dimiliki oleh setiap metode secara terpisah, seperti menggabungkan keakuratan analisis sentimen dari metode *machine learning* dengan kecepatan dan kemudahan implementasi dari metode *lexicon-based.*

2.2.4 Data *Preprocessing*

Data preprocessing merupakan proses untuk mengubah dan menyeleksi data menjadi lebih terstruktur. Tujuan utamanya adalah mengoptimalkan hasil perhitungan setiap kata dalam teks, sehingga memungkinkan analisis yang lebih optimal [40]. *Data preprocessing* terdiri dari beberapa tahapan [41], yaitu sebagai berikut:

1. *Case Folding*

Mengubah semua huruf dalam teks ke dalam huruf kecil. *Case folding* bertujuan untuk menyelaraskan penggunaan huruf besar dan kecil, sehingga mengurangi perbedaan pada kata yang sebenarnya sama [41], seperti pada Tabel 2.2 berikut.

Tabel 2.2 Contoh Tahap *Case Folding*

Input	***Output***
Jd IKN ini tempat pembuangan? 🤣 IKN Istana Koruptor Nusantara.	jd ikn ini tempat pembuangan? 🤣 ikn istana koruptor nusantara.

2. *Data Cleaning*

Melakukan penghapusan elemen-elemen yang tidak diperlukan dalam sebuah teks, seperti karakter special atau non-alfabet, penghapusan simbol (!@#$%^&*+_/ dan lainnya), *emoticon, hyperlink,* dan lainnya [41]. *Data Cleaning* bertujuan untuk meningkatkan kualitas data dengan mengurangi

noise yang dapat mempengaruhi hasil analisis. Berikut Tabel 2.3 merupakan contoh *data cleaning.*

Tabel 2.3 Contoh Tahap *Data Cleaning*

Input	***Output***
Jd IKN ini tempat pembuangan? IKN Istana Koruptor Nusantara.	Jd IKN ini tempat pembuangan IKN Istana Koruptor Nusantara

3. *Tokenization*

Memecah kalimat dalam teks ke dalam potongan kata-kata atau disebut token [41]. *Tokenization* bertujuan untuk memudahkan identifikasi dan analisis setiap kata dalam kalimat. Berikut Tabel 2.4 merupakan contoh *tokenization*.

Tabel 2.4 Contoh Tahap *Tokenization*

Input	***Output***
Jd IKN ini tempat pembuangan? IKN Istana Koruptor Nusantara.	"Jd", "IKN", "ini", "tempat", "pembuangan", "IKN", "Istana", "Koruptor", "Nusantara"

4. *Normalization*

Normalisasi memperbaiki kesalahan penulisan dan pengejaan dalam teks serta mengubah singkatan atau *slang* menjadi bentuk standar [41]. Penggunaan normalisasi dapat meningkatkan akurasi klasifikasi sentimen [42], terutama dalam teks yang sering mengandung banyak *noise* atau variasi penulisan. Berikut Tabel 2.5 merupakan contoh penerapan *normalization*.

Tabel 2.5 Contoh Tahap *Normalization*

Input	***Output***
Jd IKN ini tempat pembuangan? IKN Istana Koruptor Nusantara.	"Jadi", "IKN", "ini", "tempat", "pembuangan", "IKN", "Istana", "Koruptor", "Nusantara"

5. *Stopword Removal*

Menghilangkan atau mengurangi kata-kata umum yang sering muncul dalam teks tetapi tidak memiliki makna signifikan [41]. Berikut Tabel 2.6 merupakan contoh penerapan *stopword.*

Tabel 2.6 Contoh Tahap *Stopword Removal*

Input	*Output*
Jd IKN ini tempat pembuangan? ⚡ IKN Istana Koruptor Nusantara.	"IKN", "pembuangan", "IKN", "Istana", "Koruptor", "Nusantara", "setuju", "bangun", "Kalimantan"

6. *Stemming*

Melakukan tranformasi kata dengan mengubah kata berimbuhan menjadi kata dasar yang sesuai dengan struktur bahasa [41]. *Stemming* memudahkan identifikasi dan analisis kata-kata yang memiliki makna dasar yang sama meskipun muncul dalam bentuk yang berbeda-beda karena imbuhan. Berikut Tabel 2.7 merupakan contoh penerapan *stemming*.

Tabel 2.7 Contoh Tahap *Stemming*

Input	*Output*
Jd IKN ini tempat pembuangan? ⚡ IKN Istana Koruptor Nusantara.	"jd", "ikn", "ini", "tempat", "buang", "ikn", "istana", "koruptor", "nusantara"

2.2.5 *Term Frequency-Inverse Document Frequency* (TF-IDF)

TF-IDF merupakan metode pembobotan untuk menemukan nilai-nilai penting dalam dokumen, yang membantu pemrosesan teks dengan lebih akurat. TF-IDF dilakukan dengan menghitung bobot setiap kata dalam sebuah teks [43]. Skor TF-IDF didasarkan pada frekuensi kemunculan kata dalam dokumen, yaitu menghitung jumlah seberapa sering kata muncul dalam dokumen (TF) dan seberapa penting kata tersebut dalam semua dokumen (IDF) [44]. Berikut rumus perhitungan TF-IDF [45].

$$TF = 0{,}5 + 0{,}5 \times \frac{tf}{\max(tf)}$$

Rumus 2.1 Perhitungan Nilai TF

$$IDF_{(t)} = \log\left(\frac{D}{1 + DF_{(t)}}\right)$$

Rumus 2.2 Perhitungan Nilai IDF

$$W_{(t,d)} = TF_{(t,d)} \times IDF_{(t)}$$

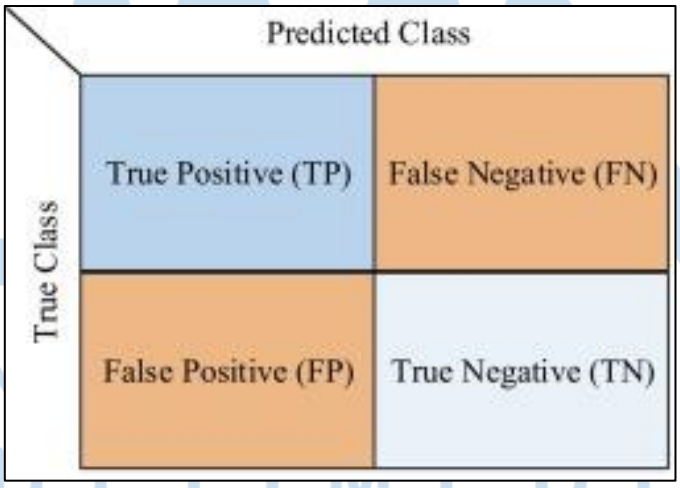
Rumus 2.3 Perhitungan Nilai TF-IDF

Keterangan:

TF	:	Frequensi kemunculan kata
t	:	kata ke-t
d	:	dokumen ke-d
IDF	:	*Inverse Document Frequency*
DF	:	Jumlah dokumen yang mengandung istilah t
D	:	Total dokumen
$TF_{t,d}$	:	*Term Frequency*
W	:	bobot dari dokumen d terhadap kata t

2.2.6 *Confusion Matrix*

Untuk melakukan evaluasi kinerja ataupun performa dari algoritma digunakan *confusion matrix* [46]. Terdapat empat parameter pada confusion matrix yang merepresentasikan hasil klasifikasi, yaitu *True Positive* (TP), *False Positive* (FP), *True Negative* (TN), dan *False Negative* (FN). Nilai *True Positive* (TP) merupakan jumlah data positif yang diprediksikan dengan benar, *False Positive* (FP) merupakan jumlah data negatif yang diprediksi menjadi data positif. Lebih lanjut, nilai *True Negative* (TN) merupakan jumlah data negatif yang diprediksi dengan benar dan nilai *False Negative* (FN) merupakan data positif yang diprediksi menjadi data negatif.



Gambar 2.2 *Confusion Matrix* [51]

2.2.6.1 *Accuracy*

Accuracy merupakan salah satu metrik yang digunakan untuk mengukur kinerja model klasifikasi. *Accuracy* menggambarkan seberapa sering model dapat memprediksi data dengan benar (*true positive* maupun *true negative*) terhadap keseluruhan data [46]. Nilai *accuracy* yang dihasilkan akan berada dalam rentang 0 hingga 1, nilai yang mendekati 1 menunjukkan kinerja model yang lebih baik dalam memprediksi data dengan benar. Berikut merupakan rumus menghitung *accuracy* pada *confusion matrix*.

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

Rumus 2.4 Perhitungan Nilai *Accuracy*

2.2.6.2 *Recall*

Recall atau *sensitivity* menggambarkan seberapa baik model dapat menemukan dan mengklasifikasikan semua data positif yang sebenarnya *(TP)* dari jumlah keseluruhan data positif aktual [46]. Berikut merupakan rumus menghitung *recall* pada *confusion matrix.*

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

Rumus 2.5 Perhitungan Nilai *Recall*

2.2.6.3 *Precision*

Precision menggambarkan seberapa baik model dapat memprediksi data positif yang sebenarnya *(TP)* dari keseluruhan data yang prediksi positif [46]. Berikut merupakan rumus menghitung *precision* pada *confusion matrix.*

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

Rumus 2.6 Perhitungan Nilai *Precision*

2.2.6.4 *F1-score*

F1-score merupakan kombinasi dari nilai *precision* dan *recall* [46]. Tujuan utama *f1-score* adalah untuk mengukur keseimbangan antara kemampuan model dalam mengidentifikasi data positif yang sebenarnya *(recall)* dan kemampuan model dalam menghindari kesalahan mengklasifikasikan data negatif sebagai positif *(precision)*. Berikut ini merupakan rumus menghitung *f1-score* pada *confusion matrix.*

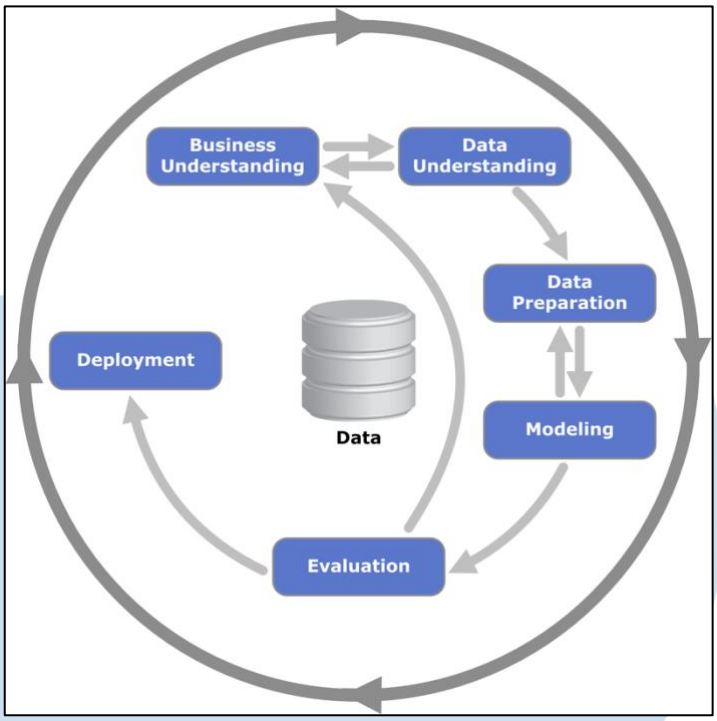
$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precesion + Recall}$$

Rumus 2.7 Perhitungan Nilai *F1-Score*

2.3 *Framework* dan Algoritma

2.3.1 *CRISP-DM Framework*

Cross Industry Standard Process–Data Mining (CRISP-DM) merupakan kerangka kerja sistematis yang digunakan untuk merencanakan, melaksanakan, dan mengevaluasi proyek *data mining* [47]. *CRISP-DM* bertujuan untuk memastikan proyek *data mining* dilakukan terstruktur dan efisien. *CRISP-DM* terdiri dari enam fase yang dimulai dari pemahaman bisnis hingga penerapan *(deployment)* [48], seperti pada Gambar 2.3. Setiap fase pada metode ini memiliki tugas dan output yang mendukung pencapaian tujuan proyek. Berikut ini merupakan setiap fase dari kerangka kerja *CRISP-DM.*



Gambar 2.3 Kerangka Kerja *CRISP-DM* [49]

1. *Business Understanding*

Proses pemahaman mengenai bisnis seperti mengetahui apa yang ingin dicapai melalui proyek *data mining,* memperoleh gambaran umum terhadap situasi dan sumber daya yang tersedia dan yang diperlukan, serta melakukan perumusan masalah untuk mencapai tujuan bisnis yang ditetapkan.

2. *Data Understanding*

Melakukan pengumpulan data, eksplorasi dan analisis seperti statistik dekriptif, mengidentifikasi kualitas data dan memahami struktur serta hubungan antar data yang digunakan.

3. *Data Preperation*

Melakukan transformasi data ke dalam format yang dapat digunakan untuk proses analisis dan permodelan lebih lanjut. Selain itu, dilakukan pemilihan atribut yang relevan, serta pembersihan data untuk memperbaiki inkonsistensi, seperti nilai yang hilang atau duplikat.

4. *Modeling*

Melakukan pemilihan teknik permodelan atau algoritma yang tepat, melakukan pelatihan dengan data *training set*, serta pengujian dengan

testing set. Tujuan utama dari fase ini adalah untuk mengembangkan model yang mampu membuat prediksi atau klasifikasi yang akurat berdasarkan data yang diberikan.

5. *Evaluation*

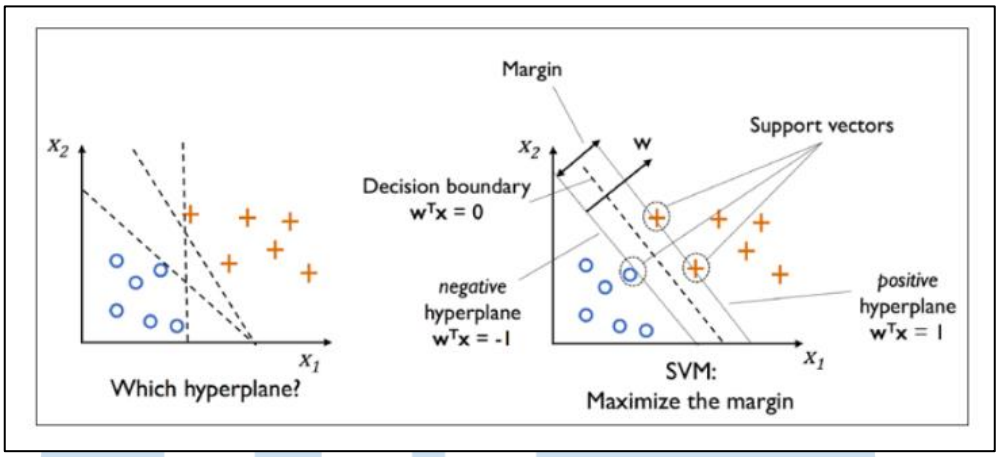
   Melakukan penilaian kinerja atau evaluasi terhadap model untuk melihat sejauh mana model sesuai dengan tujuan awal yang ditetapkan dan mengukur seberapa baik model bekerja untuk memecahkan masalah bisnis *(business understanding)*.

6. *Deployment*

   Mengimplementasikan model untuk digunakan oleh *end-user*. Pada fase ini dilakukan monitoring untuk memastikan bahwa model tetap akurat seiring berjalannya waktu.

2.3.2 *Support Vector Machine*

Support Vector Machine (SVM) pertama kali diperkenalkan oleh Vladimir Vapnik, Benhard Boser, dan Isabelle Guyon pada acara *Annual Workshop on Computational Learning Theory* tahun 1992 [50]. Algoritma ini termasuk dalam kategori *supervised learning* dan digunakan untuk tugas-tugas klasifikasi dan regresi. SVM bertujuan untuk menemukan *hyperplane* optimal yang berfungsi untuk memisahkan kelompok data yang berbeda. *Hyperplane* ini dirancang untuk memaksimalkan *margin*, yaitu jarak antara *hyperplane* dengan titik data terdekat dari masing-masing kelas atau disebut sebagai *support vectors*. Semakin besar *margin* yang digunakan untuk memisahkan kelompok data, dalam hal ini data positif (+1) dari data negatif (-1), maka semakin besar kemungkinan algoritma dapat mengklasifikasikan data ke dalam kelompok yang benar dengan lebih akurat [51], sebagaimana pada ilustrasi Gambar 2.4 berikut.



Gambar 2.4 *Hyperplane* memisahkan dua kelompok data [52]

Dalam kondisi data tidak terpisahkan secara *linear* menggunakan garis lurus atau *hyperplane*, SVM memanfaatkan teknik yang disebut *kernel trick* [53]. *Kernel trick* memungkinkan pemetaan data *non-linear* ke ruang dimensi yang lebih tinggi sehingga data dapat dipisahkan secara linear. Beberapa jenis kernel yang umum digunakan adalah *kernel linear, kernel Radial Basis Function (RBF), kernel polynomial,* dan *kernel sigmoid* [54]. Pemilihan kernel yang tepat sangat kritis karena menentukan ruang fitur di mana fungsi classifier akan beroperasi.

1. *Kernel Linear*

   Kernel Linear efektif untuk data yang sudah terpisah secara *linear*, sehingga tidak memerlukan transformasi ke ruang dimensi lebih tinggi. Kernel ini cocok digunakan untuk data dengan banyak dimensi dan relasi linear antar fitur. Berikut merupakan rumus fungsi *kernel linear.*

   $$K(x_i, x) = x_i^T x$$

   Rumus 2.8 Fungsi *Kernel Linear*

   Dimana, x_i adalah vektor fitur dari data pelatihan dan x adalah vektor fitur dari data pengujian.

2. *Kernel Radial Basis Gaussian (RBF)*

   Kernel RBF digunakan untuk data dengan relasi antar fitur yang kompleks dan non-linear. Fungsi ini menangani efektifitas pada

berbagai jenis data, terutama yang multidimensional. Berikut merupakan rumus fungsi *kernel RBF*.

$$K(x_i, x) = \exp\left(-\gamma |x_i^T x|^2\right)$$

Rumus 2.9 Fungsi *Kernel RBF*

Dimana, x_i adalah vektor fitur dari data pelatihan, x adalah vektor fitur dari data pengujian, dan γ merupakan faktor skalar yang menyesuaikan sensitivitas model terhadap perbedaan dalam ruang fitur.

3. *Kernel Polynomial*

   Kernel Polynomial dirancang untuk memetakan data ke ruang berdimensi tinggi, memungkinkan SVM memodelkan relasi non-linear. Kernel ini sesuai untuk data dengan relasi polinomial. Berikut merupakan rumus fungsi *kernel polynomial.*

$$K(x_i, x) = (\gamma . x_i^T x + r)^p$$

Rumus 2.10 Fungsi *Kernel Polynomial*

Dimana, x_i sebagai vektor fitur dari data pelatihan, x sebagai vektor fitur dari data pengujian, γ sebagai faktor skalar, r sebagai konstanta bias, dan p adalah derajat polinomial.

4. *Kernel Sigmoid*

   Kernel Sigmoid mengadopsi fungsi aktivasi *sigmoid* dari jaringan saraf, yang menghasilkan *output* dalam rentang antara -1 dan 1. Berikut merupakan rumus fungsi *kernel sigmoid*.

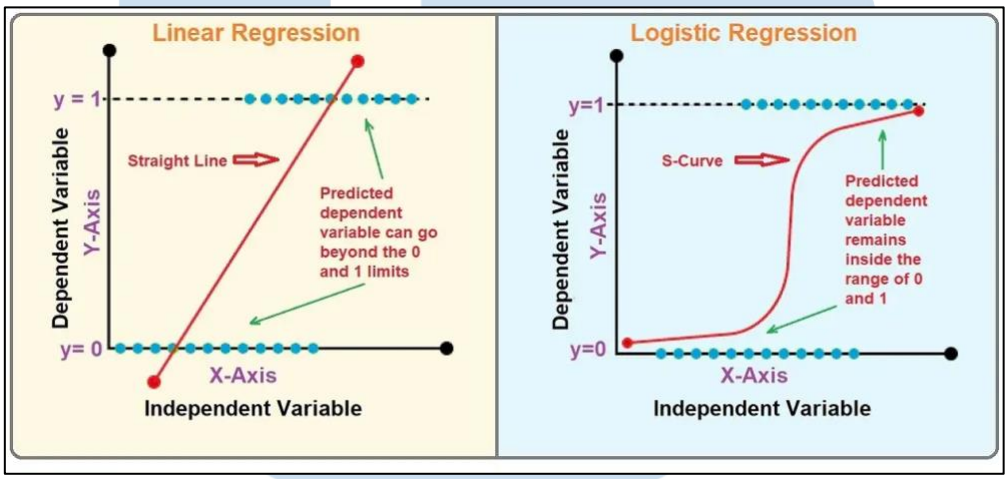
$$K(x_i, x) = \tanh\left(\gamma . x_i^T x + r\right)$$

Rumus 2.11 Fungsi *Kernel Sigmoid*

Dimana, x_i sebagai vektor fitur dari data pelatihan, x sebagai vektor fitur dari data pengujian, tanh merupakan fungsi tangen hiperbolik yang menghasilkan nilai *output* dalam rentang antara -1 dan 1, γ sebagai faktor skalar, r sebagai konstanta bias.

2.3.3 *Logistic Regression*

Logistic Regression (LR), merupakan algoritma klasifikasi dalam kategori *supervised learning* yang digunakan untuk memprediksi probabilitas variabel dependen berdasarkan satu atau lebih variabel independen. LR digunakan untuk klasifikasi biner, di mana variabel dependen memiliki dua kemungkinan nilai, yaitu diskret atau kategorial (misalnya, 0 atau 1, ya atau tidak) [55].



Gambar 2.5 Illustrasi *Linear Regression* dan *Logistic Regression* [56]

LR dikembangkan dari prinsip regresi *linear*, yang mana menghasilkan *output* kontinu, sedangkan LR menghasilkan *output* diskret dengan menggunakan fungsi *sigmoid* atau kurva S *(S-curve)* yang membatasi outputnya antara 0 dan 1. Fungsi *sigmoid* mengubah input *linear* menjadi nilai antara 0 dan 1, memastikan bahwa estimasi hasil selalu berada dalam rentang tersebut [57], sebagaimana yang terlihat pada Gambar 2.5. Dalam konteks analisis sentimen, LR digunakan untuk mengklasifikasikan teks, seperti *tweet* atau ulasan, ke dalam kategori sentimen positif atau negatif. Model ini menghubungkan fitur teks, sebagai variabel independen, dengan kategori sentimen, sebagai variabel dependen, melalui fungsi *sigmoid* [57]. Berikut merupakan rumus dari fungsi *sigmoid* yang mengkonversi *output* ke dalam bentuk probabilitas.

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

Rumus 2.12. Fungsi *Sigmoid Logistic Regression*

Dimana $\sigma(z)$ merupakan fungsi *sigmoid* yang mengubah input *linear* menjadi nilai antara 0 dan 1, z merupakan nilai input yang diterapkan pada fungsi *sigmoid*. Nilai z yang tinggi menghasilkan e^{-z} yang rendah, membuat $\sigma(z)$ mendekati 1, menandakan probabilitas tinggi terhadap variabel dependen. Sebaliknya, nilai z yang rendah menghasilkan e^{-z} yang tinggi, membuat $\sigma(z)$ mendekati 0, menandakan probabilitas rendah terhadap variabel dependen.

2.3.4 *Social Network Analysis* (SNA)

Social Network Analysis (SNA) merupakan metode analisis yang berfokus pada hubungan atau relasi. SNA digunakan untuk menggambarkan informasi berupa pola hubungan antara individu, kelompok, atau entitas dalam jaringan pada berbagai *platform* media sosial [58]. Dalam SNA, interaksi antarpengguna dimodelkan sebagai titik atau *(nodes)* yang mewakili akun seperti individu, kelompok, organisasi, dan lain-lain. Garis *(edges*), merepresentasikan koneksi antara akun yang berupa komunikasi atau jenis interaksi lainnya. Penerapan SNA dapat membantu dalam menemukan akun-akun utama dalam jaringan dan mengukur pengaruh mereka dalam penyebaran informasi, hal ini disebut pengukuran *centrality* [59]. Terdapat beberapa pengukuran *centrality,* yaitu *degree centrality, betweenness centrality,* dan *closeness centrality.*

2.3.4.1 *Degree Centrality*

Degree Centrality digunakan untuk menghitung jumlah hubungan atau interaksi langsung yang dimiliki sebuah *node* atau akun, menggambarkan popularitas atau keterlibatan akun dalam sebuah jaringan [58]. Terdapat dua jenis *degree centrality,* yaitu *in-degree* dan *out-degree. In-degree* merupakan jumlah interaksi yang diterima sebuah akun dari akun lain, berupa *mentions*, *retweets*, *replies* dan lainnya [60]. Nilai *in-degree* yang tinggi menunjukkan sebuah akun sering menjadi fokus atau pusat perhatian dalam jaringan. Sementara itu, *out-degree* merupakan jumlah interaksi yang diinisiasi oleh sebuah akun ke akun lain, dalam bentuk *mentions*, *retweets*, *replies* dan lainnya [60]. Nilai *out-*

degree yang tinggi menunjukkan keaktifan akun dalam berkomunikasi atau berinteraksi. Secara keseluruhan, akun dengan nilai total *degree centrality* yang tinggi dianggap memiliki banyak hubungan, sehingga memiliki pengaruh signifikan dalam jaringan karena mereka terhubung dengan banyak akun lain.

2.3.4.2 *Betweenness Centrality*

Betweenness Centrality digunakan untuk mengukur *node* atau aktor yang berperan sebagai penghubung atau jembatan dalam penyebaran informasi di dalam sebuah jaringan [58]. Akun dengan *betweenness centrality* tinggi berperan sebagai perantara atau penghubung di antara akun-akun lain dalam jaringan, sehingga menunjukkan bahwa akun tersebut memiliki pengaruh besar dalam mengontrol aliran informasi dalam jaringan dan dapat mempengaruhi interaksi antara akun lain.

2.3.4.3 *Closeness Centrality*

Closeness Centrality mengukur seberapa cepat sebuah *node* atau akun dapat mencapai akun-akun lain dalam jaringan, dengan menghitung rata-rata jarak dari satu *node* ke semua *node* lainnya [59]. Akun dengan *closeness centrality* tinggi dapat menyebarluaskan informasi lebih cepat ke semua akun lain dalam jaringan, dikarenakan memiliki jarak yang lebih pendek ke seluruh akun lain sehingga memungkinkan penyebaran informasi lebih cepat dan luas.

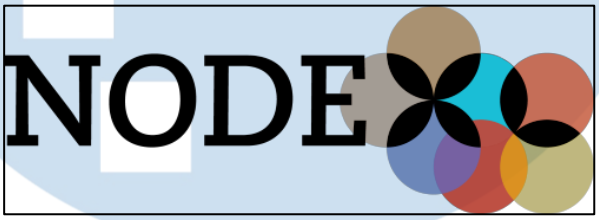
2.4 *Tools* atau *Software*

2.4.1 *Python*

Gambar 2.6 Logo *Python* [61]

Python, yang dikembangkan oleh Guido van Rossum, adalah bahasa pemrograman berorientasi objek. *Python* dapat dijalankan pada berbagai sistem operasi diantaranya bersifat Windows, MacOS, Linux dan bersifat *open source*. *Python* menawarkan beragam *library* yang mendukung penerapan analisis data, visualisasi data, *machine learning, NLP,* dan lainnya. Adapun beberapa *library* yang tersedia dalam *Python* seperti *Pandas* untuk manipulasi data, *NumPy* untuk operasi *array*, *scikit-learn* untuk *machine learning,* dan *Matplotlib* untuk visualisasi data [62]. Selain itu, *Python* juga mendukung integrasi dengan berbagai *API* dan layanan *web*, sehingga memudahkan pengumpulan data dari *Twitter*.

2.4.2 *NodeXL*



Gambar 2.7 Logo *NodeXL* [63]

NodeXL atau *Network Overview, Discovery and Exploration for Excel,* dikembangkan oleh Marc Smith bersama tim *Social Media Research Foundation,* merupakan *add-on* yang dirancang khusus untuk *Microsoft Excel* yang memungkinkan analisis jaringan sosial *(Social Network Analysis, SNA)* dan visualisasi grafis. *NodeXL* membantu dalam mengumpulkan, menganalisis, dan memvisualisasikan jaringan sosial dari berbagai *platform* sosial media seperti *X (Twitter), Reddit, YouTube, Wikipedia, Facebook, Wikipedia,* dan lainnya [64].

NodeXL menawarkan berbagai fitur untuk analisis dan visualisasi jaringan, termasuk pembuatan grafik visual, analisis sentralitas untuk mengidentifikasi aktor kunci persebaran informasi dan lainnya. Selain itu, *NodeXL* mendukung proses pengambilan data dengan memanfaatkan *Application Programming Interface (API)* pada berbagai *platform* sosial media seperti *Twitter, YouTube, Reddit,* dan *Wikipedia* serta mendukung proses impor

dan ekspor data dalam berbagai format seperti *GraphML, Pajek, UCINet,* dan *Matriks Workbook (Excel)* [65].

2.4.3 *Gephi*

Gambar 2.8 Logo *Gephi* [66]

Gephi adalah *platform open source* yang kompatibel dengan berbagai sistem operasi, termasuk Windows, MacOS, dan Linux. *Gephi* memungkinkan pengguna mengeksplorasi, menganalisis, dan memvisualisasikan data jaringan atau grafik. Selain itu, *gephi* memungkinkan impor data dalam berbagai format file seperti *CSV, XLSX, GDF (GUESS), GraphML (NodeXL), GML, NET (Pajek), GEXF,* dan lainnya. *Gephi* menawarkan berbagai fitur yang memungkinkan pengguna untuk melakukan analisis data jaringan seperti *algoritma layout* untuk mengoptimalkan keterbacaan grafik, pengukuran metrik analisis jaringan seperti *degree centrality, betweenness centrality, closeness centrality, diameter, clustering coefficient, pagerank,* dan lainnya [67].