

BAB II

TINJAUAN PUSTAKA

2.1 Landasan Teori

2.1.1 Analisis Sentimen

Analisis sentimen merupakan metode otomatis untuk mengungkap emosi dalam teks, mengelompokkannya menjadi *positive*, *neutral*, atau *negative* (Br Sinulingga & Sitorus, 2024)

2.1.2 *Text Mining*

Text mining adalah suatu metode yang memanfaatkan perangkat lunak untuk menggali informasi berharga dari kumpulan data teks yang besar dan tidak terstruktur, seperti konsep, pola, topik, dan kata kunci (Fathonah & Herliana, 2021). (Firdaus & Firdaus, 2021) menjelaskan bahwa *text mining* dapat mengekstrak informasi relevan dari dokumen teks, memberikan hasil kualitatif. Teknik ini mengintegrasikan metode statistik, linguistik, dan pembelajaran mesin untuk membangun model yang mampu belajar dari data dan membuat prediksi pada informasi baru.

2.1.3 *Pre-processing*

Pre-processing terdiri dari serangkaian langkah untuk mengubah data mentah yang berasal dari ekstraksi data menjadi kumpulan data bersih dan rapi. sebelum dilakukan analisa. Tahapan dari *pre-processing* adalah sebagai berikut:

2.1.3.1 *Cleaning*

Data mentah yang diperoleh cenderung tidak lengkap, mengandung banyak noise, dan tidak konsisten. *Cleaning* merupakan proses yang akan membantu pembersihan teks dari karakter, symbol maupun tanda baca yang tidak diperlukan dalam analisis sentimen.

2.1.3.2 *Case Folding*

Case folding merupakan proses transformasi seluruh huruf besar pada teks menjadi huruf kecil.

2.1.3.3 *Tokenizing*

Tokenizing adalah teknik dalam pemrosesan bahasa alami (NLP) yang memecah teks menjadi unit-unit kecil, atau token, untuk analisis lebih lanjut. Token ini bisa berupa kata, angka, tanda baca, atau elemen penting lainnya. Meski sering dikaitkan dengan kata, *tokenizing* juga dapat diterapkan pada tingkat kalimat atau paragraph(Iriananda et al., 2021).

2.1.3.4 *Stopword removal*

Stopword removal adalah teknik *preprocessing* teks yang menghilangkan kata-kata umum dan tidak informatif, seperti kata penghubung ('dan', 'atau') dan kata ganti ('ini', 'itu'). Proses ini bertujuan meningkatkan efisiensi pelatihan model bahasa dengan mengurangi dimensi data dan fokus pada kata-kata yang lebih relevan. Kamus *stopword* khusus bahasa Indonesia digunakan untuk menentukan kata-kata yang akan dihapus(Iriananda et al., 2021).

2.1.3.5 *Stemming*

Stemming merupakan proses pembentukan kata dasar melalui eliminasi prefiks dan sufiks. Contohnya kata 'menggambar', 'gambaran', menjadi 'gambar'.

2.1.4 *Feature Extraction*

Feature Extraction (ekstraksi fitur) adalah proses mentransformasikan data teks (seperti *tweet*) menjadi representasi numerik (vektor fitur) yang dapat dipahami oleh algoritma machine learning(Liu, 2012). Algoritma machine learning umumnya bekerja dengan data numerik, bukan teks mentah.

2.1.5 *Lexicon-based*

Lexicon-based merupakan metode ekstraksi fitur berbasis *lexicon* yang dapat meningkatkan performa sistem dengan memanfaatkan kamus kata untuk mengidentifikasi sentimen dalam teks(Catelli et al., 2022). Kamus ini berisi kata-kata yang diberi label emosi (*positive*, *negative*, atau *neutral*), dibuat secara manual dengan mencantumkan kata-kata kunci kemudian diperkaya dengan menambahkan sinonim dan antonimnya.

2.1.6 TF-IDF

TF – IDF merupakan salah satu teknik ekstraksi fitur dengan ukuran statistik yang dapat mengevaluasi seberapa relevan sebuah kata dalam sebuah dokumen yang bersangkutan. Hal ini dapat diperoleh dengan mengalikan dua metrics yaitu skor TF (*Term Frequency*) dan skor IDF (*Inverse Document Frequency*)(Septiani & Isabela, 2022). Pada proses TF - IDF setiap term dalam sebuah *tweet* akan diberi bobot berdasarkan frekuensi kemunculan *term* tersebut dalam sebuah dokumen yang bersangkutan. Semakin tinggi nilai TF, maka akan semakin tinggi bobot dari term tersebut, dengan kata lain nilai kesesuaiannya akan semakin besar. Sebaliknya, semakin sedikit suatu kata muncul pada dokumen, maka semakin kecil kata tersebut berpengaruh pada dokumen tersebut.

Jika terdapat suatu *term* yang kemunculannya hampir di sebagian besar dokumen, hal ini akan mengakibatkan proses pencarian term unik akan terganggu, maka Dalam hal ini IDF berfungsi mengurangi bobot suatu term jika kemunculannya terlalu banyak tersebar di seluruh dokumen. Rumus pembobotan menggunakan TF-IDF dijelaskan pada persamaan 2.1.

$$\begin{aligned}
 \mathbf{tf} &= 0.5 + 0.5 \times \frac{\mathbf{tf}}{\max(\mathbf{tf})} \\
 \mathbf{idf} &= \log\left(\frac{D}{f_{tD}}\right) \\
 \mathbf{W}_{d,t} &= \mathbf{tf}_{d,t} \times \mathbf{idf}_{d,t}
 \end{aligned}
 \tag{2.1}$$

Keterangan:

- tf adalah frekuensi istilah dalam dokumen. t = term ke- t dari dokumen
- $\max(tf)$ adalah frekuensi maksimum istilah dalam dokumen.
- D adalah total jumlah dokumen.
- dft adalah jumlah dokumen yang mengandung istilah t .
- tfd,t adalah term frequency dari istilah t dalam dokumen d .
- $idfd,t$ adalah inverse document frequency dari istilah t dalam dokumen d .
- Wd,t adalah bobot TF-IDF dari istilah t dalam dokumen d .

2.1.7 XGBoost (*eXtreme Gradient Boosting*)

XGBoost atau *eXtreme Gradient Boosting* adalah algoritma berbasis pohon (Chen dan Guestrin, 2016) dan merupakan bagian dari algoritma pohon seperti *decision tree*, *Random Forest*. XGBoost merupakan algoritma *supervised tree* yang menggunakan prinsip *ensemble* yaitu menggabungkan beberapa set pembelajar (*tree*) yang lemah menjadi sebuah model yang kuat sehingga menghasilkan prediksi yang kuat (Fajar, 2019).

Pembangunan model dilakukan dengan menggunakan metode *gradient boosting* yang merupakan salah satu dari metode *ensemble*, dimana metode ini bekerja dengan cara menggabungkan prediksi dari beberapa model menjadi satu. Hal ini dilakukan dengan mengambil setiap *predictor* secara berurutan dan memodelkannya berdasarkan *residual error* dari model sebelumnya. Proses pertama saat dataset dimasukkan adalah membuat model awal dengan menggunakan data *original*. Kemudian akan peroleh nilai prediksi awal dan skor *error/residual* dari model awal dengan menggunakan persamaan 2.2. dan 2.3:

$$h_0(x) = \text{mean}(Y) \quad (2.2)$$

$$\hat{Y} = Y - h_0(x) \quad (2.3)$$

Dimana $h_0(x)$ merupakan prediksi awal dari model pertama dan \hat{Y} merupakan *residual error* dari model pertama. Kemudian model kedua akan dibangun menggunakan *residual error* dari model pertama dan akan diperoleh hasil prediksi untuk model kedua. Selanjutnya model ketiga akan dibangun dengan menggunakan *residual error* dari model pertama dan kedua dan akan diperoleh prediksi dari model ketiga. Proses iterasi ini akan dilakukan sebanyak n kali sesuai dengan angka $n_estimator$ yang telah ditentukan. Hasil prediksi yang diperoleh merupakan akumulasi dari prediksi setiap model sebelumnya.

Algoritma ini dinamakan *gradient boosting* karena menggunakan *gradient descent* untuk memperkecil error saat membuat model baru (Annisa et al, 2017).

Persamaan yang digunakan pada XGBoost adalah sebagai berikut:

Misalkan sebuah data latih adalah x_i dengan label y_i , algoritma XGBoost akan menggunakan K pembelajar (*tree*) untuk memprediksi output \bar{y}_i seperti persamaan 2.4:

$$\bar{y}_i = \sum K f_k(x_i) \quad (2.4)$$

Dimana f_k menunjukkan struktur pohon yang berisi skor kontinu w_i di dalam setiap i_{th} daunnya. Skor dari setiap pohon dihitung dengan cara meminimalkan fungsi objek seperti persamaan 2.5:

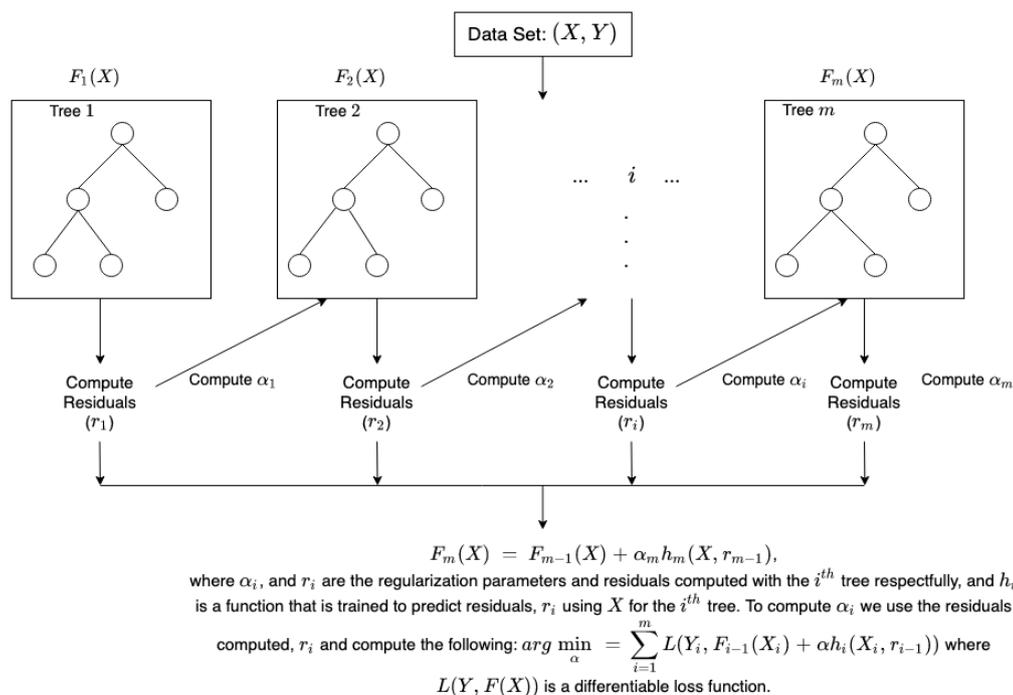
$$L^{(t)} = \sum_{i=1}^n l(y_i \bar{y}_i^{(t-1)} + f_{r(x_i)}) + a(f_j) \quad (2.5)$$

Dimana l merupakan *differentiable convex loss function* untuk mengukur perbedaan antara y_i dan $\bar{y}^{(t)}$. Pada persamaan ini $\bar{y}^{(t)}$ merupakan hasil prediksi dari data ke i_{th} pada iterasi ke t_{th} . Simbol α merupakan *regularization term* yang akan membuat model berusaha menghindari *overfitting*.

Pada gambar 2.1 prediksi akhir diperoleh melalui persamaan 2.6

$$Fm(X) = Fm - 1(X) + \alpha hm(X, rm - 1) \quad (2.6)$$

Dimana r_i merupakan *residual error* dan α_i merupakan *regularization parameter* yang dihitung pada setiap iterasi i^{th} , dan h_i merupakan fungsi yang telah dilatih untuk memprediksi nilai residual r_i menggunakan X untuk setiap pohon pada iterasi i^{th} . Untuk memperoleh nilai α_i digunakan nilai *residual error* yang telah dihitung dari model sebelumnya.



Gambar 2. 1 Ilustrasi cara kerja XGBoost

Sumber : (Farida Eleshin, 2022)

2.1.8 Hyperparameter

Hyperparameter adalah variabel yang ditentukan sebelum melatih model *Machine Learning* (Shah, 2021). *Hyperparameter* berperan penting dalam mengendalikan proses pembelajaran dan memengaruhi performa akhir model. *Grid Search* dan *Randomized Search* adalah dua metode yang digunakan untuk optimasi *hyperparameter* dalam model *machine learning*. Keduanya memiliki tujuan yang sama yaitu menemukan kombinasi *hyperparameter* yang terbaik untuk kinerja model, namun menggunakan pendekatan yang berbeda. *Grid Search* akan mencoba semua kombinasi nilai *hyperparameter* yang telah ditentukan, sehingga dapat menjamin menemukan kombinasi terbaik tetapi memerlukan

waktu komputasi yang lebih lama. *Randomized Search* bekerja secara acak dengan memilih kombinasi nilai *hyperparameter* yang telah ditentukan, sehingga lebih efisien dalam hal waktu komputasi, tetapi mungkin tidak menemukan kombinasi optimal terbaik (Priya, 2023). 3 *hyperparameter* yang paling sering digunakan adalah *learning rate*, *max depth*, dan *n_estimator* karena memiliki pengaruh yang signifikan terhadap performa model dan sensitif terhadap perubahan nilainya. (Blanke, 2022). *Learning rate* menentukan seberapa cepat model belajar. *Max depth* menentukan kedalaman pohon keputusan dalam model. *N_estimator* menentukan jumlah pohon keputusan yang digunakan dalam model.

2.1.9 Confusion matrix

Confusion matrix merupakan alat evaluasi umum dalam pembelajaran mesin yang digunakan untuk mengukur kinerja model klasifikasi dengan membandingkan prediksi model dengan label sebenarnya (Alibrahim & Ludwig, 2021). Matriks ini membandingkan prediksi model dengan label sebenarnya, menghasilkan empat kategori hasil: *true positive*, *true negative*, *false positive*, dan *false negative*. Matriks kinerja yang diturunkan dari matriks ini, termasuk akurasi, presisi, *recall*, dan *F1-score*, memberikan informasi berharga tentang seberapa baik algoritma XGBoost bekerja. Perhitungan untuk matriks ini dapat dilihat pada persamaan 2.7 sampai 2.10.

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (2.7)$$

$$Recall = \frac{TP}{TP+FN} \quad (2.8)$$

$$Precision = \frac{TP}{TP+FP} \quad (2.9)$$

$$F1Score = \frac{2 \times (precision \times recall)}{precision + recall} \quad (2.10)$$

Keterangan:

- a) TP (*True Positive*), yaitu label yang diberikan sistem *positive* dan label aktual bernilai benar
- b) TN (*True Negative*), yaitu label yang diberikan sistem *negative* dan label aktual bernilai benar
- c) FP (*False Positive*), yaitu label yang diberikan sistem *positive* dan label aktual bernilai salah
- d) FN (*False Negative*), yaitu label yang diberikan sistem *negative* dan label aktual bernilai salah

2.2 Penelitian Terdahulu

Penelitian ini memiliki beberapa penelitian terkait yang akan digunakan sebagai bahan acuan dan perbandingan hasil penelitian. Penelitian terdahulu dapat dilihat pada table 2.1.

Tabel 2. 1 Literatur jurnal

1	Judul	ANALISIS SENTIMEN TANGGAPAN PELANGGAN INDIHOME DI PLATFORM SOSIAL MEDIA FACEBOOK DAN TWITTER MENGGUNAKAN <i>SUPPORT VECTOR MESIN</i> DAN PENDEKATAN <i>KLASIFIKASI NAÏVE BAYES</i> (STUDI KASUS: PT. TELKOM INDONESIA)
---	-------	--

	Peneliti & Tahun	(Rahmad Kusuma et al., 2023)
	Penerbit	Jurnal Sains & Teknologi Universitas Darma Persada Vol. 13 No. 1 Maret 2023
	Metode/ Solusi	Membandingkan efektivitas algoritma <i>Support Vector Machine</i> (SVM) dan <i>Naive bayes</i> dalam analisis sentimen dengan studi kasus indihome.
	Hasil Utama	Hasil pengujian pada model <i>Support Vector Machine</i> mencapai akurasi sebesar 91%, sedangkan pada model <i>Naive bayes</i> mencapai akurasi sebesar 85%.
2	Judul	Analisis Media Sosial Penyedia Layanan Internet Menggunakan Algoritma XGBOOST
	Peneliti & Tahun	(Iskandar et al., 2022)
	Penerbit	SEMINAR NASIONAL CORISINDO
	Metode/ Solusi	Pada penelitian ini dilakukan penelitian untuk membuat aplikasi yang dapat membantu pengguna internet di Indonesia untuk mengetahui informasi provider mana yang menyediakan layanan internet yang baik berdasarkan komentar dari pengguna. Penelitian ini dilakukan dengan mengumpulkan data teks melalui API yang disediakan oleh Twitter dan diubah menjadi vektor menggunakan TF-IDF kemudian komentar teks diklasifikasi menggunakan algoritma XGBoost.
	Hasil Utama	Pada penelitian ini metode XGBoost berhasil melakukan klasifikasi 300 data <i>tweet</i> dengan nilai akurasi terbaik sebesar 83.5% dan 263 data <i>tweet</i> dengan nilai akurasi terbaik 89.3%. Metode XGBoost sudah berhasil mengenali kalimat sentimen dalam berbagai kelas kata seperti kata sifat, kata kerja, kata benda yang menggambarkan perasaan pengguna, namun metode ini masih belum dapat

		mengenali sentimen yang memiliki makna kiasan atau makna ganda.
3	Judul	Klasifikasi Sentimen <i>Tweet</i> Pelanggan IndiHome Selama Pandemi <i>Covid-19</i> Menggunakan Algoritma <i>Multinomial Naive bayes</i>
	Peneliti & Tahun	(Pamungkas et al., 2022)
	Penerbit	Seminar Nasional Teknik Elektro, Sistem Informasi, dan Teknik Informatika
	Metode/ Solusi	Banyak pelanggan yang merasa tidak puas dengan pelayanan IndiHome dengan menuliskan <i>tweet</i> di Twitter. Diperlukan sentimen <i>analysis</i> pada komentar <i>tweet</i> untuk mengevaluasi layanan IndiHome berdasarkan opini pelanggan. Klasifikasi menggunakan metode <i>Multinomial Naive bayes</i> diterapkan untuk menghasilkan akurasi yang optimal. Pada klasifikasi <i>Multinomial Naive bayes</i> fitur diasumsikan diambil dari distribusi Multinomial sederhana. Proses analisis sentimen terdiri dari proses labeling data menggunakan <i>VADER Lexicon</i> , lalu <i>pre-processing</i> , selanjutnya akan dihitung jumlah <i>term frequency</i> dan <i>documen frequency</i> . Data akan dibagi menjadi data <i>training</i> dan testing menggunakan <i>k-fold Cross Validation</i> serta klasifikasi teks dengan metode <i>Multinomial Naive bayes</i> dengan mencari nilai probabilitas tertinggi.
	Hasil Utama	Hasil optimal dalam percobaan ini dicapai dengan variasi 11- <i>fold</i> , menunjukkan tingkat akurasi rata-rata 76,7%. Pengukuran presisi mencapai 88,21%, sementara <i>recall</i> mencapai 76,28%. Skor F1, indikator keseimbangan antara presisi dan <i>recall</i> , mencapai 81,78%. Variasi ini juga berhasil mengklasifikasikan

		435 <i>tweet</i> sebagai <i>negative</i> dan 1056 <i>tweet</i> sebagai <i>positive</i> secara akurat.
4	Judul	<i>The Impact of Feature Extraction to Naïve Bayes Based Sentimen Analysis on Review Dataset of Indihome Services</i>
	Peneliti & Tahun	(Permataning Tyas et al., 2022)
	Penerbit	Jurnal Teknologi Informasi dan Komunikasi
	Metode/ Solusi	Fokus penelitian ini adalah untuk membandingkan kinerja ekstraksi fitur TF-IDF dan <i>Word2Vec</i> . <i>Metrics</i> yang digunakan untuk mengukur kinerja kedua teknik seleksi fitur adalah akurasi.
	Hasil Utama	Hasil Experimen dengan <i>k-fold</i> cross validation mengkonfirmasi bahwa TF-IDF lebih baik dibandingkan <i>Word2Vec</i>
5	Judul	Analisis Sentimen terhadap Layanan Indihome di Twitter dengan Metode <i>Machine learning</i>
	Peneliti & Tahun	(Puspita & Widodo, 2021)
	Penerbit	Program Studi Teknik Informatika Universitas Pamulang
	Metode/ Solusi	Indihome sangat banyak mendapatkan kritikan karena koneksi internet yang jarang stabil. Maka dari itu, dilakukanlah analisis sentimen di bidang <i>data mining</i> terhadap <i>customer</i> Indihome pada media sosial Twitter yang terdiri dari 1350 data dan dilakukan <i>filtering</i> menjadi 1309 data sebab banyak data yang terindikasi duplikat. Pada penelitian ini, peneliti menggunakan metode <i>Random Forest</i> dan <i>Gradient Boosted Trees</i> (GBT)
	Hasil Utama	Penelitian menunjukkan hasil bahwa analisis sentimen terhadap layanan Indihome dengan menggunakan

		<p>metode <i>Random Forest</i> mencapai akurasi sebesar 99.54% dengan class precision untuk pred. <i>negative</i> adalah 99.92%, pred <i>positive</i> adalah 25.00%, dan pred. <i>neutral</i> adalah 60.00%. Lalu pada metode GBT tingkat akurasinya mencapai 99.31% dengan class precision untuk pred. <i>negative</i> adalah 99.46%, pred. <i>positive</i> adalah 0.00%, dan pred. <i>neutral</i> adalah 0.00%. Maka dapat disimpulkan bahwa metode <i>Random Forest</i> merupakan metode yang lebih baik jika dibandingkan dengan GBT.</p>
6	Judul	Analisis Sentimen Review Customer Terhadap Produk Indihome Dan First Media Menggunakan Algoritma <i>Convolutional Neural Network</i>
	Peneliti & Tahun	(Hasan Badjrie et al., 2021)
	Penerbit	<i>e-Proceeding of Engineering</i> : Vol.8, No.5 Oktober 2021 Page 9049
	Metode/ Solusi	Algoritma CNN (<i>Convolutional Neural Network</i>) adalah algoritma <i>deep learning</i> yang dapat menggunakan gambar sebagai input, menetapkan kepentingan untuk berbagai aspek dan objek dalam gambar agar dapat membedakan satu dengan yang lain dan memiliki akurasi yang tinggi, sehingga dalam penelitian analisis sentimen review produk IndiHome dan First Media. Penelitian ini bertujuan untuk melakukan penilaian produk terhadap provider menggunakan metode analisis sentimen <i>review customer</i> dari tiap <i>tweets</i> yang pelanggan telah lampirkan dengan algoritma <i>convolutional neural network</i>
Hasil Utama	Hasil akurasi yang didapatkan, memperoleh akurasi tertinggi sebesar 98% untuk provider IndiHome dan	

		91% untuk provider First Media.
7	Judul	Sentimen Analisis Layanan Produk Indihome menggunakan <i>Information Gain</i> dan Metode <i>K-Nearest Neighbor</i>
	Peneliti & Tahun	(Anggraeni & Cholissodin, 2021)
	Penerbit	Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer
	Metode/ Solusi	Pada penelitian ini menggunakan tanggapan dari masyarakat berupa komentar <i>positive</i> dan komentar <i>negative</i> . Untuk melihat komentar tersebut termasuk <i>positive</i> atau <i>negative</i> harus melakukan beberapa tahapan analisis untuk mendapatkan hasil akhir. Tahapan yang dilakukan yaitu <i>pre processing</i> , seleksi fitur <i>Information Gain</i> , pembobotan term dan klasifikasi algoritma <i>K-Nearest Neighbor (KNN)</i> .
	Hasil Utama	Pada penelitian ini menggunakan 480 data latih dan 120 data uji. Penelitian ini mendapatkan nilai akurasi tertinggi sebesar 86,67%, <i>precision</i> sebesar 94,44%, <i>recall</i> sebesar 81,73%, dan <i>f-measure</i> sebesar 87,63%.
8	Judul	<i>Developing an LSTM-based Classification Model of IndiHome Customer Feedbacks</i>
	Peneliti & Tahun	(Arifianto Anditya et al., 2020)
	Penerbit	<i>International Conference on Data Science and Its Applications (ICoDSA)</i>
	Metode/ Solusi	Klasifikasi otomatis umpan balik pelanggan dari media sosial dan layanan chat pada penyedia layanan internet (IndiHome) di Indonesia menggunakan 2 metode, <i>Long Short-Term Memory (LSTM)</i> yang dikombinasikan dengan <i>Word Embedding</i> dan <i>Naive bayes</i> yang dikombinasikan dengan <i>Term Frequency-Inverse</i>

		<i>Document Frequency</i> (TF-IDF).
	Hasil Utama	LSTM dengan <i>Word Embedding</i> mencapai F1-score 87.98%, lebih unggul dari <i>Naive bayes</i> dengan TF-IDF (76.77%). Perbedaan sebesar 11.21% ini signifikan. TF-IDF : Lebih ideal untuk dokumen panjang karena mempertimbangkan keseluruhan dokumen. Word Embedding : Tidak terpengaruh ukuran dokumen, fokus pada representasi kata individual.
9	Judul	Sentimen <i>analysis about product and service evaluation of pt telekomunikasi Indonesia tbk from tweets using TextBlob, naive bayes & K-NN Method</i>
	Peneliti & Tahun	(Hermansyah & Sarno, 2020)
	Penerbit	<i>Proceedings - 2020 International Seminar on Application for Technology of Information and Communication: IT Challenges for Sustainability, Scalability, and Security in the Age of Digital Disruption, iSemantic 2020</i>
	Metode/ Solusi	Menentukan metode klasifikasi sentimen terbaik untuk menganalisis ulasan produk Telkom dari Twitter dengan Perbandingan tiga metode klasifikasi sentimen: <i>TextBlob</i> , <i>Naive bayes</i> , dan <i>K-Nearest Neighbor</i> (K-NN).
	Hasil Utama	K-NN adalah algoritma yang paling akurat dengan skor akurasi 75%. <i>Naive bayes</i> berada di urutan kedua dengan akurasi 69.44%. <i>TextBlob</i> menunjukkan kinerja terendah dengan akurasi 54.67%.
10	Judul	Analisis Sentimen Mengenai Layanan Provider Indihome Berdasarkan Pendapat Pelanggan Melalui Media Sosial Twitter dengan Metode <i>Naïve Bayes Classifier</i>

	Peneliti & Tahun	(Yusuf et al., 2020)
	Penerbit	JIEET: Volume 04 Nomor 02, 2020 (<i>Journal Information Engineering and Educational Technology</i>)
	Metode/ Solusi	Pengujian metode <i>Naïve Bayes</i> pada analisis sentimen Layanan Indihome
	Hasil Utama	Analisis sentimen terhadap data Twitter mengenai layanan Indihome yang dilakukan dengan menggunakan metode <i>naïve bayes</i> dengan nilai akurasi 74.5%. Dari hasil penelitian yang telah dilakukan, metode <i>naïve bayes</i> dapat diaplikasikan untuk analisis sentimen data Twitter berbahasa Indonesia
11	Judul	Analisis Sentimen Terhadap Layanan Indihome Berdasarkan Twitter Dengan Metode Klasifikasi <i>Support Vector Machine</i> (SVM)
	Peneliti & Tahun	(Tineges et al., 2020)
	Penerbit	JURNAL MEDIA INFORMATIKA BUDIDARMA Volume 4, Nomor 3, Juli 2020, Page 650-658
	Metode/ Solusi	Tujuan dari penelitian ini adalah untuk dapat pemodelan klasifikasi sentimen dengan <i>Support Vector Machine</i> (SVM), untuk mengetahui seberapa besar akurasi yang dihasilkan oleh metode SVM, serta untuk mengetahui seberapa puas pengguna layanan Indihome berdasarkan Twitter.
	Hasil Utama	Dengan Metode SVM, akurasi yang didapatkan sebesar 87%, Hasil sentimen layanan Indihome berdasarkan data baru, dengan metode <i>Support Vector Machine</i> mendapatkan hasil nilai <i>positive</i> sebesar 18,4 % dan hasil nilai <i>negative</i> sebesar 81,6%. Berdasarkan hasil tersebut dapat disimpulkan bahwa tingkat kepuasan pengguna layanan Indihome cukup rendah.

		Berdasarkan hasil dan pembahasan menunjukkan bahwa metode SVM dapat diterapkan untuk analisis sentimen data Twitter berbahasa Indonesia
--	--	---

2.3 *Matrix State of the Art*

Beberapa jurnal terkait berhubungan dengan penelitian yang sedang dilakukan dengan penelitian yang akan dilakukan. *Matrix* SOTA dapat dilihat pada tabel 2.2.

Tabel 2. 2 *Matrix* SOTA

No	Peneliti	Model										Ekstraksi fitur / Seleksi Fitur					
		Naive Bayes	Random Forest	Gradient Boosted Tree	Extreme Gradient Boosting	Genetic Algorithm	Convolutional Neural Network	K-Nearest Neighbor	Long short-term memory	TextBlob	Support Vector Machine	TF-IDF	Word2Vec	Automatic	<i>Lexicon</i>	Word Embedding	Information Gain
1	(Rahmad Kusuma et	✓									✓	✓					

No	Peneliti	Model										Ekstraksi fitur / Seleksi Fitur					
		Naïve Bayes	Random Forest	Gradient Boosted Tree	Extreme Gradient Boosting	Genetic Algorithm	Convolutional Neural Network	K-Nearest Neighbor	Long short-term memory	TextBlob	Support Vector Machine	TF-IDF	Word2Vec	Automatic	<i>Lexicon</i>	Word Embedding	Information Gain
	al., 2023)																
2	(Iskandar et al., 2022)				✓												
3	(Pamungkas et al.,	✓												✓			

No	Peneliti	Model										Ekstraksi fitur / Seleksi Fitur					
		Naïve Bayes	Random Forest	Gradient Boosted Tree	Extreme Gradient Boosting	Genetic Algorithm	Convolutional Neural Network	K-Nearest Neighbor	Long short-term memory	TextBlob	Support Vector Machine	TF-IDF	Word2Vec	Automatic	<i>Lexicon</i>	Word Embedding	Information Gain
	2022)																
4	(Permatani ng Tyas et al., 2022)	✓										✓	✓				
5	(Puspita & Widodo,		✓	✓										✓			

No	Peneliti	Model										Ekstraksi fitur / Seleksi Fitur				
		Naïve Bayes	Random Forest	Gradient Boosted Tree	Extreme Gradient Boosting	Genetic Algorithm	Convolutional Neural Network	K-Nearest Neighbor	Long short-term memory	TextBlob	Support Vector Machine	TF-IDF	Word2Vec	Automatic	<i>Lexicon</i>	Word Embedding
	n, 2021)															
8	(Arifianto Anditya et al., 2020)	✓						✓			✓				✓	

No	Peneliti	Model									Ekstraksi fitur / Seleksi Fitur					
		Naïve Bayes	Random Forest	Gradient Boosted Tree	Extreme Gradient Boosting	Genetic Algorithm	Convolutional Neural Network	K-Nearest Neighbor	Long short-term memory	TextBlob	Support Vector Machine	TF-IDF	Word2Vec	Automatic	<i>Lexicon</i>	Word Embedding
11	(Tineges et al., 2020)									✓	✓					
12	(Nabil Ramdhani, 2024)				✓						✓			✓		