Building of Informatics, Technology and Science (BITS)
Volume 6, No 4, March 2025 Page: 2559−2572
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i4.6949

Perbandingan Algoritma Support Vector Machine, Decision Tree, Naïve Bayes, dan Neural Network dalam Klasifikasi Email

Dika Wicaksono[*], I Made Artha Agastya

Ilmu Komputer, Program Studi Informatika, Universitas Amikom Yogyakarta, Yogyakarta, Indonesia
Email: [1,*]dikawicaksono@amikom.ac.id, [2]artha.agastya@amikom.ac.id
Email Penulis Korespondensi: dikawicaksono@amikom.ac.id
Submitted: **08/02/2025**; Accepted: **12/03/2025**; Published**: 13/03/2025**

Abstrak−Penelitian ini bertujuan untuk membandingkan efektivitas empat model machine learning dalam klasifikasi email, yaitu Support Vector Machine (SVM), Decision Tree, Naive Bayes, dan Neural Network. Penelitian ini menggunakan dataset yang diperoleh dari website *Kaggle*. Dataset pertama berisi 18.650 email phishing (7.328 phishing dan 11.322 non-phishing). Dataset kedua merupakan hasil penggabungan dua dataset berbeda yang berisi email spam berbahasa Indonesia, sehingga menghasilkan total 4.681 email (2.670 spam dan 2.011 non-spam). Penggabungan dilakukan untuk memperoleh jumlah data yang lebih representatif dalam evaluasi model. Hasil penelitian dari kedua dataset di atas menunjukkan bahwa Neural Network mencapai akurasi tertinggi dengan rata-rata 96.60%. Kemudian, diikuti SVM dengan akurasi rata-rata 96.43%. Sementara itu, Decision Tree memiliki akurasi yang cukup tinggi dengan rata-rata 92.38%. Sebaliknya, Naive Bayes mencatatkan kinerja terendah dengan akurasi rata-rata 90.22%. Meskipun Neural Network memiliki akurasi tertinggi, model lain mungkin lebih sesuai tergantung pada kebutuhan sistem. Model dengan akurasi lebih rendah, seperti Naive Bayes, bisa lebih berguna dalam sistem dengan keterbatasan komputasi karena efisiensinya. SVM menawarkan keseimbangan antara akurasi tinggi dan efisiensi komputasi, menjadikannya pilihan ideal untuk sistem yang membutuhkan performa optimal tanpa beban komputasi terlalu besar. Decision Tree lebih unggul dalam interpretasi hasil, sehingga cocok untuk aplikasi yang memerlukan transparansi dalam pengambilan keputusan.

Kata Kunci: Klasifikasi Email; *Neural Network*; *Support Vector Machine* (SVM); *Naive Bayes*; *Decision Tree*

Abstract−This study aims to compare the effectiveness of four machine learning models in email classification, namely Support Vector Machine (SVM), Decision Tree, Naive Bayes, and Neural Network. This research uses datasets obtained from the Kaggle website. The first dataset contains 18,650 phishing emails (7,328 phishing and 11,322 non-phishing). The second dataset is the result of merging two different datasets containing Indonesian spam emails, resulting in a total of 4,681 emails (2,670 spam and 2,011 non-spam). The merging was done to obtain a more representative amount of data for model evaluation. The results of the study of the two datasets above showed that the Neural Network achieved the highest accuracy with an average of 96.60%. Then, followed by SVM with an average accuracy of 96.43%. Meanwhile, Decision Tree has a fairly high accuracy with an average of 92.38%. In contrast, Naive Bayes recorded the lowest performance with an average accuracy of 90.22%. Although Neural Network has the highest accuracy, other models may be more suitable depending on the needs of the system. Models with lower accuracy, such as Naive Bayes, can be more useful in systems with computational limitations due to their efficiency. SVM offers a balance between high accuracy and computational efficiency, making it an ideal choice for systems that require optimal performance without too much computational burden. Decision Tree is superior in result interpretation, making it suitable for applications that require transparency in decision making.

Keywords: Email Classification; Neural Network; Support Vector Machine (SVM); Naive Bayes; Decision Tree

1. PENDAHULUAN

Sektor komunikasi telah mengalami perubahan yang signifikan dalam beberapa dekade terakhir karena perkembangan teknologi. Salah satu perkembangan yang paling signifikan dalam industri komunikasi adalah transisi dari komunikasi analog ke komunikasi digital. Munculnya internet secara signifikan mempermudah komunikasi jarak jauh dengan mengurangi durasi, biaya, dan kerumitannya. Teknologi telah mengubah cara orang berkomunikasi. Email, media sosial, dan pesan daring adalah beberapa teknologi baru yang muncul dengan perkembangan pada sektor komunikasi.

Namun, perkembangan ini juga mengakibatkan beberapa dampak negatif, seperti maraknya email yang tidak diinginkan, yang biasanya dikenal sebagai email spam. Email spam [1] merupakan email yang tidak diinginkan oleh penerima karena biasanya berisi iklan atau pesan penipuan. Selain itu, terdapat jenis spam yang lebih merugikan, yaitu *phishing*. *Phishing* [2] adalah teknik penipuan di mana penyerang mencoba untuk memperoleh informasi sensitif seperti kata sandi, informasi keuangan, atau data pribadi dengan menyamar sebagai entitas terpercaya. Oleh karena itu, setelah data tersebut diperoleh, pelaku dapat menggunakannya untuk menjadi ancaman kepada korban agar korban dapat membayar uang tebusan atau semacamnya. Berbagai penelitian telah dilakukan dan menunjukkan bahwa pendekatan *machine learning* memilliki keunggulan yang signifikan dibanding dengan metode berbasis aturan tradisional dalam hal klasifikasi email. Penelitian ini berfokus pada empat algoritma yang menonjol: *Support Vector Machine* (SVM), *Decision Tree*, *Naive Bayes*, dan *Neural Network*. Penelitian ini bertujuan untuk membandingkan keunggulan dan kelemahan model-model tersebut dalam mengklasifikasikan email berdasarkan tingkat akurasi yang dicapai.

Menurut penelitian yang dilakukan oleh Dina Anggraini dan Tata Sutabri pada tahun 2024 [3], akurasi yang didapat SVM dalam mengklasifikasikan email spam sebesar 95% dengan hanya 5% kemungkinan kesalahan klasifikasi. SVM dapat memberikan performa yang baik dalam memisahkan kelas-kelas yang linier terpisah. Di sini dijelaskan juga beberapa keunggulan SVM, seperti kemampuannya untuk menangani data dengan fitur berdimensi tinggi, yang umum dalam analisis teks email.

Building of Informatics, Technology and Science (BITS)
Volume 6, No 4, March 2025 Page: 2559−2572
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i4.6949

Sedangkan, dalam penelitian yang dilakukan oleh Abhishek Kumar, Jyotir Moy Chatterjee, dan Vicente García Díaz pada tahun 2020 [4], dibuktikan juga bahwa SVM sebagai pengklasifikasi biner yang efektif memiliki akurasi 87%, sensitivitas 88,5%, dan spesifikasi 91%.

Kemudian, pada penelitian lain oleh Ryan Putra Ramadhan dan Teti Desyani pada tahun 2023 [5], digunakan metode *machine learning* yang berbeda yaitu *Decision Tree*. Dalam penelitian ini ditemukan bahwa *Decision Tree*, khususnya algoritma J48, menunjukkan hasil yang cukup tinggi dalam klasifikasi deteksi web *phishing*. Di sini juga dijelaskan bahwa metode ini merupakan suatu prosedur pemecahan yang digunakan dalam pengolahan data dan *machine learning*, yang dapat mengklasifikasikan data ke dalam kelas-kelas dan memprediksi kelas dari suatu data.

Dalam penelitian yang dilakukan oleh Qianhe Ouyang, Jiahe Tian, dan Jiale Wei di tahun 2023 [6], *Naive Bayes* dibandingkan dengan algoritma lain seperti KNN menunjukkan bahwa *Naive Bayes* dapat mencapai akurasi yang cukup tinggi , antara 98% hingga 99% dalam klasifikasi email spam.

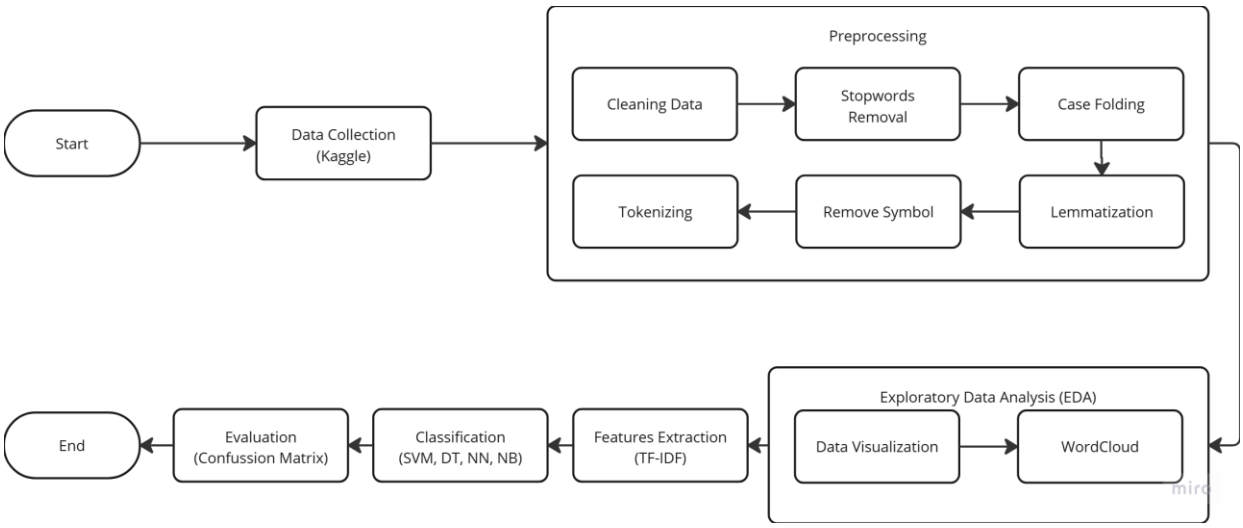
Dalam penelitian lainnya oleh Niken Larasati Octaviani, Eko Hari Rachmawanto, Christy Atika Sari, dan Ignatius Moses Setiadi De Rosal [7], dibandingkan tiga metode *machine learning* untuk mengklasifikasikan email spam. Dari penelitian ini, hasil yang didapat adalah metode SVM mencapai akurasi tertinggi dari ketiga metode *machine learning* yang digunakan. SVM mencapai akurasi hingga 96%, sedangkan *Naive Bayes* mencapai akurasi sebesar 93%. *Neural Network* di sini dengan menggunakan model *baseline Recurrent Neural Network* hanya mencapai akurasi 74% saja.

Penelitian terdahulu sebagian besar hanya berfokus pada penerapan salah satu metode *machine learning* untuk klasifikasi email, tanpa membandingkan efektivitas berbagai metode secara menyeluruh. Misalnya, penelitian menggunakan SVM sering menyoroti akurasi tinggi pada dataset teks, sementara penelitian lain menunjukkan kekuatan *Naive Bayes* dalam efisiensi waktu komputasi. Namun, evaluasi yang mempertimbangkan berbagai metode dalam kondisi yang sama masih terbatas. Oleh karena itu, penelitian ini bertujuan untuk mengisi kesenjangan tersebut dengan melakukan analisis komparatif terhadap metode SVM, *Neural Networks*, *Decision Tree*, dan *Naive Bayes* pada dataset yang mencakup email spam dan *phishing*. Penelitian ini bertujuan untuk melakukan perbandingan yang lebih menyeluruh dan memberikan analisis komparatif yang mendalam terhadap empat algoritma terkemuka dalam klasifikasi email. Dengan menggunakan metrik evaluasi standar seperti akurasi, *recall*, *precision*, dan *F1-score*, penelitian ini memberikan gambaran yang lebih komprehensif tentang kinerja masing-masing model dalam mengklasifikasikan email spam dan *phishing*.

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian

Penelitian ini dilakukan melalui langkah-langkah yang dirancang secara terstruktur untuk memastikan setiap tahap menghasilkan data dan temuan yang valid. Setiap proses disusun dengan tujuan menjawab masalah penelitian, dimulai dari pengumpulan data, pemrosesan awal, hingga analisis hasil. Fokus utama penelitian ini adalah untuk mengevaluasi performa algoritma *Support Vector Machine* (SVM), *Decision Tree*, *Naive Bayes*, dan *Neural Network* dalam mengklasifikasikan email, sehingga memberikan wawasan yang lebih mendalam mengenai kekuatan dan kelemahan masing-masing metode. Alur metodologi penelitian dapat dilihat pada Gambar 1.



Gambar 1. Metodologi Penelitian

Building of Informatics, Technology and Science (BITS)
Volume 6, No 4, March 2025 Page: 2559−2572
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i4.6949

2.2 *Data Collection*

Tahap awal penelitian ini adalah pengumpulan data. Data diperoleh dari dua dataset yang tersedia di website *Kaggle*. Dataset email phising dapat diakses di https://www.kaggle.com/datasets/subhajournal/phishingemails, sedangkan untuk dataset email spam dapat di akses di https://www.kaggle.com/datasets/gevabriel/indonesian-email-spam, denngan keterangan sebagai berikut:

a. Dataset Pertama: Dataset terdiri dari 18.650 email, dengan 11.322 email dilabeli sebagai safe email dan 7.312 email sebagai phishing email.
b. Dataset Kedua: Dataset ini merupakan hasil penggabungan dua dataset berisi email berbahasa Indonesia dengan label spam dan ham, sehingga menghasilkan total 4.681 email (2.670 spam dan 2.011 ham).

Dari masing-masing dataset, hanya kolom *text* dan label yang diambil untuk proses analisis. Pemilihan dataset ini didasarkan pada keberagaman bahasa dan kategori email, sehingga dapat menguji keefektifan metode *machine learning* pada data multibahasa.

2.3 *Preprocessing Data*

Data *preprocessing* dilakukan dengan tujuan untuk mengubah data mentah menjadi data yang berkualitas sehingga data layak untuk diolah pada tahapan selanjutnya. Tahapan ini [8] dilakukan pada data mentah untuk menghilangkan data yang bermasalah atau inkonsisten. Berikut merupakan tahapan dalam *preprocessing data:*

a. *Cleaning Data*
   Hal pertama yang dilakukan pada tahap ini adalah melakukan *cleaning data*. *Cleaning data* [9] merupakan proses menghapus atau menangani elemen-elemen non-standar dari sebuah dataset seperti *outliers*, duplikasi, data null, atau elemen serupa lainnya.
b. *Stopwords Removal*
   Stopwords removal [9] adalah proses dalam analisis bahasa alami yang melibatkan penghapusan kata-kata yang dianggap tidak penting untuk analisis, seperti kata hubung atau kata umum lainnya yang tidak menambah makna signifikan pada teks. Proses ini bertujuan untuk meningkatkan efisiensi analisis dengan menghilangkan elemen yang tidak relevan.
c. *Case Folding*
   Case folding [10] merupakan proses untuk mengubah kata menjadi bentuk yang sama. Tujuan dari *case folding* adalah mengembalikan semua kata ke dalam bentuk huruf kecil semua supaya data teks yang diproses semuanya dalam kondisi bentuk yang sama.
d. *Lemmatization*
   Lemmatization [11] adalah proses yang mengubah kata-kata menjadi bentuk dasarnya dengan menghilangkan akhiran infleksional dan menciptakan kata-kata yang valid.
e. *Remove symbol*
   Remove symbol dilakukan untuk menghapus karakter khusus yang terdapat dalam dataset. Langkah ini penting untuk memastikan bahwa teks yang dianalisis bersih dari simbol atau tanda baca yang tidak diperlukan, sehingga analisis data dapat berjalan dengan lebih akurat dan optimal.
f. *Tokenizing*
   Tokenizing [10] merupakan proses analisis bahasa alami yang melibatkan pemisahan teks menjadi unit-unit yang lebih kecil, seperti kata atau frasa. *Tokenizing* dapat dilakukan dengan menggunakan fungsi seperti *word_tokenize* dari NLTK (*Natural Language Toolkit*) [12], untuk memisahkan teks menjadi token berdasarkan spasi dan tanda baca.

2.4 *Exploratory Data Analysis* (EDA)

Exploratory Data Analysis (EDA) [13] adalah metode analisis data yang digunakan untuk memahami distribusi data dan hubungan antar variabel. EDA melibatkan berbagai langkah, termasuk memaksimalkan wawasan ke dalam data, mengungkap struktur data, mengekstrak variabel penting, mendeteksi *outlier* dan anomali, melakukan uji asumsi, mengembangkan model, dan menentukan faktor yang optimal. Kebanyakan teknik EDA berbentuk grafis dengan beberapa teknik kuantitatif. Tujuan utama EDA [14] adalah untuk mengeksplorasi data secara terbuka, dengan grafik yang bertujuan memperkuat analisis yang dilakukan. Pada penelitian ini, dilakukan visualisasi untuk mengungkap distribusi label email dalam dataset yang dianalisis. Selain itu, pembuatan *Wordcloud* juga dilakukan untuk mengidentifikasi kata-kata yang sering muncul pada masing-masing kategori dalam dataset. *Wordcloud* [15] merupakan bentuk visual dari keberadaan kata dalam dokumen. Semakin banyak istilah yang sering muncul dalam dokumen yang dianalisis, maka kata tersebut akan muncul semakin besar dalam gambar *Wordcloud*. Langkah ini bertujuan untuk memperoleh pemahaman mendalam mengenai kata-kata kunci yang dapat membedakan dalam proses klasifikasi email.

Building of Informatics, Technology and Science (BITS)
Volume 6, No 4, March 2025 Page: 2559−2572
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i4.6949

2.5 *Feature Extraction* (TF-IDF)

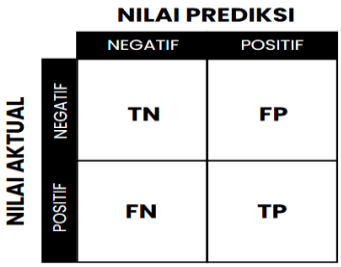
Feature extraction adalah fase penting dalam identifikasi karena setiap huruf mempunyai keunikan tersendiri sehingga membedakan dirinya dengan huruf yang lain. Di sini [16] dijelaskan bahwa *feature extraction* bertujuan untuk mendapatkan karakteristik suatu karakter yang membedakannya dari karakter yang lain yang disebut *feature*. Pada penelitian ini, digunakan metode TF-IDF (*Term Frequency-Inverse Document Frequency*). TF-IDF [17] sendiri merupakan metode yang digunakan untuk mengukur pentingnya kata dalam sebuah dokumen teks dalam suatu koleksi dokumen. Dijelaskan di sini [18], perhitungannya cukup sederhana, yaitu dengan menghitung berapa frekuensi kemunculan suatu kata tertentu dalam suatu berkas/dokumen teks yang termuat dalam suatu *corpus* dan kemudian menggunakannya sebagai dasar untuk perhitungan rumus/formula TF-IDF.

2.6 Klasifikasi

Pada tahap ini dilakukan pembuatan model klasifikasi dari keempat metode *machine learning* yang digunakan, yaitu *Support Vector Machine (*SVM), *Decision Tree*, *Naive Bayes*, dan *Neural Network.* Pada penelitian ini, digunakan implementasi khusus untuk klasifikasi dari SVM yaitu SVC (*Support Vector Classification).* SVC [19] sendiri merupakan algoritma untuk pengenalan pola. SVC memetakan data sampel ke dalam ruang berdimensi tinggi menggunakan fungsi kernel, dan klasifikasi linear dilakukan dalam ruang ini. Kemudian, pada metode *Decision Tree*, digunakan *Decision Tree Classifier* (DTC). DTC [20] merupakan salah satu metode yang cukup terkenal untuk klasifikasi data. Fitur paling signifikan dari DTC adalah kemampuannya untuk mengubah masalah pengambilan keputusan yang rumit menjadi proses yang sederhana, sehingga menemukan solusi yang dapat dipahami dan lebih mudah diinterpretasikan. DTC membangun model klasifikasi dalam bentuk struktur pohon, di mana pohon keputusan yang dihasilkan dapat digunakan untuk pengambilan keputusan. Selanjutnya, untuk *Naive Bayes* digunakan *Gaussian Naive Bayes* (*Gaussian* NB). Di sini [21], dijelaskan bahwa algoritma ini dikenal karena efisiensi komputasinya dan kemampuannya untuk melakukan pembelajaran inkremental, di mana estimasi probabilitas dapat diperbarui dengan data pelatihan baru. Pemilihan *Gaussian* NB didasarkan pada data yang sudah dilakukan proses transformasi TF-IDF, menghasilkan nilai numerik sehingga tidak bisa diselesaikan dengan *Naive Bayes* biasa. Terakhir, digunakan *Multi-Layer Perceptron Classifier* (MLPC). MLPC [22] adalah model klasifikasi berbasis *Neural Network* (NN) yang terdiri dari beberapa lapisan *perceptron* dan digunakan dalam *Deep Learning.* Model ini terdiri dari arsitektur jaringan saraf tiruan *feedforward*. MLP *Classifier* pada dasarnya memiliki tiga lapisan yang umumnya dikenal sebagai lapisan input, lapisan tersembunyi, dan lapisan output.

2.7 Evaluasi

Tahap yang terakhir adalah tahap evaluasi. Tahap ini dilakukan untuk menilai kinerja dan efektivitas dari metode klasifikasi yang telah diterapkan dalam penelitian ini. Evaluasi dilakukan menggunakan *confusion matrix*. *Confusion matrix* [23] merupakan kesimpulan dari hasil klasifikasi, yaitu jumlah data yang telah diklasifikasikan dengan benar dan yang belum diklasifikasikan dengan benar. Gambar 2. Merupakan visualisasi dari *confusion matrix.* yang terdiri dari 4 komponen utama, yaitu:



Gambar 2. Visualisasi *Confusion Matrix*

Dalam evaluasi performa model klasifikasi, terdapat emapt metrik utama yang digunakan untuk mengukur akurasi prediksi, yaitu *True Negative (TN), True Positive (TP), False Negative (FN), dan False Positive (FP)*. *True Negative* (TN) adalah jumlah data di mana model memprediksi kelas negatif dan hasilnya memang benar negatif. Sebaliknya, *True Positive* (TP) menunjukkan jumlah data di mana model memprediksi kelas positif dan hasilnya benar sesuai kenyataan. Di sisi lain, terdapat kesalahan prediksi yang diukur melalui *False Negative* (FN) dan *False Positive* (FP). *False Negative* (FN) terjadi ketika model memprediksi data sebagai negatif, padahal seharusnya positif. Sementara itu, *False Positive* (FP) terjadi saat model memprediksi data sebagai positif, tetapi kenyataannya negatif. Dari tahap dan nilai *confusion matrix* ini, kita dapat menghitung kinerja dan efektivitas metode klasifikasi yang telah diterapkan dalam penelitian ini. Evaluasi ini dilakukan dengan membandingkan hasil klasifikasi dari masing-masing metode berdasarkan metrik evaluasi yang sudah ditentukan, seperti akurasi, presisi, *recall*, dan F1-*score.* Metrik-metrik ini memberikan gambaran tentang seberapa baik model dalam mengklasifikasikan data dengan benar dan akurat, baik untuk kelas positif maupun negatif. Perhitungan metrik tersebut dapat dilakukan menggunakan persamaan (1), (2), (3), dan (4) berikut:

Building of Informatics, Technology and Science (BITS)
Volume 6, No 4, March 2025 Page: 2559−2572
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i4.6949

$$Akurasi = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

$$Presisi = \frac{TP}{TP + FP} \quad (2)$$

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

$$F1\ Score = 2 \times \frac{Presisi\ \times\ Recall}{Presisi\ +\ Recall} \quad (4)$$

Akurasi dihitung menggunakan persamaan pada Rumus 1 dengan membandingkan jumlah prediksi yang benar dengan total jumlah sampel yang diuji. Kemudian, pada Rumus 2 digunakan untuk menghitung presisi dengan membandingkan jumlah prediksi positif yang benar (*True Positive*) dengan total prediksi positif yang dibuat oleh model, sehingga mengukur tingkat ketepatan model dalam mengklasifikasikan data positif. Selanjutnya, perhitungan *recall* terdapat pada Rumus 3, dengan menghitung proporsi data positif yang berhasil diidentifikasi dengan benar oleh model dibandingkan dengan seluruh data positif yang sebenarnya. Untuk menyeimbangkan *presisi* dan *recall*, digunakan F1 *Score*, seperti yang dijelaskan dalam Rumus 4, yang merupakan rata-rata harmonik dari kedua metrik tersebut. F1 *Score* sangat berguna dalam kasus di mana terdapat ketidakseimbangan kelas, karena memberikan gambaran lebih adil mengenai performa model dengan mempertimbangkan baik ketepatan maupun kemampuan model dalam mengenali kelas positif.

3. HASIL DAN PEMBAHASAN

Bagian ini menjelaskan tentang hasil yang diperoleh dari penerapan model serta analisis terhadap performa yang dihasilkan oleh model tersebut. Proses evaluasi sendiri dengan menggunakan metrik yang relevan, seperti akurasi, presisi, *recall*, dan F1-*score*. Hasil-hasil tersebut kemudian dianalisis untuk memahami kekuatan dan kelemahan model dalam mengklasifikasikan data yang diuji. Analisis ini diharapkan dapat memberikan pemahaman yang lebih mendalam mengenai efektivitas setiap model dalam menyelesaikan permasalahan yang ada dalam penelitian ini.

3.1 Data Collection

Dalam penelitian ini, digunakan dua dataset email yang didapat dari website *Kaggle*. Dataset yang dipakai adalah 18.650 email dengan 11.322 email dilabeli sebagai *safe email* dan 7.312 email sebagai *phishing email* yang dapat diakses di https://www.kaggle.com/datasets/subhajournal/phishingemails. Dataset kedua merupakan hasil penggabungan dua dataset yang berisi email spam berbahasa Indonesia, sehingga menghasilkan total 4.681 email dengan 2.670 dilabeli sebagai spam dan 2.011 dilabeli sebagi ham. Dataset ini dapat diakses di https://www.kaggle.com/datasets/gevabriel/indonesian-email-spam. Contoh dari masing-masing dataset yang digunakan dalam penelitian ini disajikan pada Tabel 1 dan Tabel 2.

Tabel 1. Dataset *phishing email*

Email Text	Email Type
12 sep 2002 gary lawrence murphy write much carnivore yep w plan alng typos one consistently fight evls da vore	Safe Email
webcam date hot please	Phishing Email
enron hpl actuals june 6 2000 teco tap 90 000 hpl iferc texoma 20 000 hpl iferc l hpl lsk ic 40 000 enron	Safe Email

Tabel 2. Dataset *spam email*

Kategori	Pesan
spam	Secara alami tak tertahankan identitas perusahaan Anda sangat sulit...
ham	Re: Lisensi Situs untuk Dunia Daya yang saya setujui...
ham	Kuisioner Benchmarking David, saya mengirimkan pertanyaan yang diajukan oleh Petronas untuk pertemuan kami pada 8 Februari...

3.2 Preprocessing Data

Setelah dataset didapatkan, maka proses selanjutnya adalah tahap *preprocessing data*. Proses pertama yang dilakukan yaitu *cleaning* dataset, proses ini terdiri dari beberapa pengecekan seperti cek data duplikat, cek nilai *null*, dan penghapusan kolom yang tidak diperlukan pada dataset email *phishing*. Setelah itu, proses dilanjutkan dengan

Building of Informatics, Technology and Science (BITS)
Volume 6, No 4, March 2025 Page: 2559−2572
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i4.6949

stopwords removal, *case folding*, *lemmatization*, *remove symbol*, dan *tokenization*. Pada proses *remove symbol*, dilakukan beberapa langkah seperti penghapusan karakter khusus, penghapusan angka, dan penghapusan tanda baca yang tidak diperlukan. Semua proses dari *stopwords removal* hingga *tokenization* akan divisualisasikan dalam tabel Tabel 3 sampai Tabel 8.

Tabel 3. Proses *stopwords removal*

Sebelum	Sesudah
URGENT: Anda memenangkan undian! Hubungi kami segera untuk klaim hadiah Anda.	URGENT : memenangkan undian ! Hubungi klaim hadiah .

Tabel 4. Proses *case folding*

Sebelum	Sesudah
URGENT: Anda memenangkan undian! Hubungi kami segera untuk klaim hadiah Anda.	urgent: anda memenangkan undian! hubungi kami segera untuk klaim hadiah anda.

Tabel 5. Proses *lemmatization*

Sebelum	Sesudah
URGENT: Anda memenangkan undian! Hubungi kami segera untuk klaim hadiah Anda.	URGENT : Anda memenangkan undian ! Hubungi kami segera untuk klaim hadiah Anda .

Tabel 6. Proses *remove symbol*

Sebelum	Sesudah
URGENT: Anda memenangkan undian! Hubungi kami segera untuk klaim hadiah Anda.	URGENT Anda memenangkan undian Hubungi kami segera untuk klaim hadiah Anda

Tabel 7. Proses *remove number*

Sebelum	Sesudah
URGENT: Anda memenangkan undian! Hubungi kami segera untuk klaim hadiah Anda.	URGENT: Anda memenangkan undian! Hubungi kami segera untuk klaim hadiah Anda.

Tabel 8. Proses *remove punctuation*

Sebelum	Sesudah
URGENT: Anda memenangkan undian! Hubungi kami segera untuk klaim hadiah Anda.	URGENT Anda memenangkan undian Hubungi kami segera untuk klaim hadiah Anda

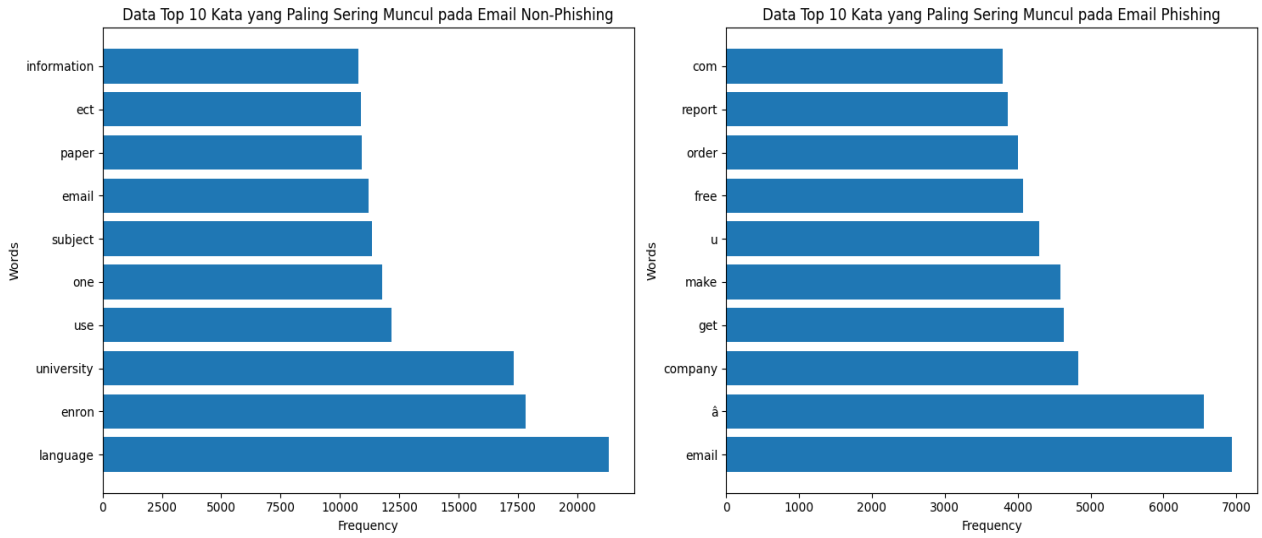
Tabel 9. Proses *tokenization*

Sebelum	Sesudah
URGENT: Anda memenangkan undian! Hubungi kami segera untuk klaim hadiah Anda.	['URGENT', ':', 'Anda', 'memenangkan', 'undian', '!', 'Hubungi', 'kami', 'segera', 'untuk', 'klaim', 'hadiah', 'Anda', '.']

3.3 *Exploratory Data Analysis* (EDA)

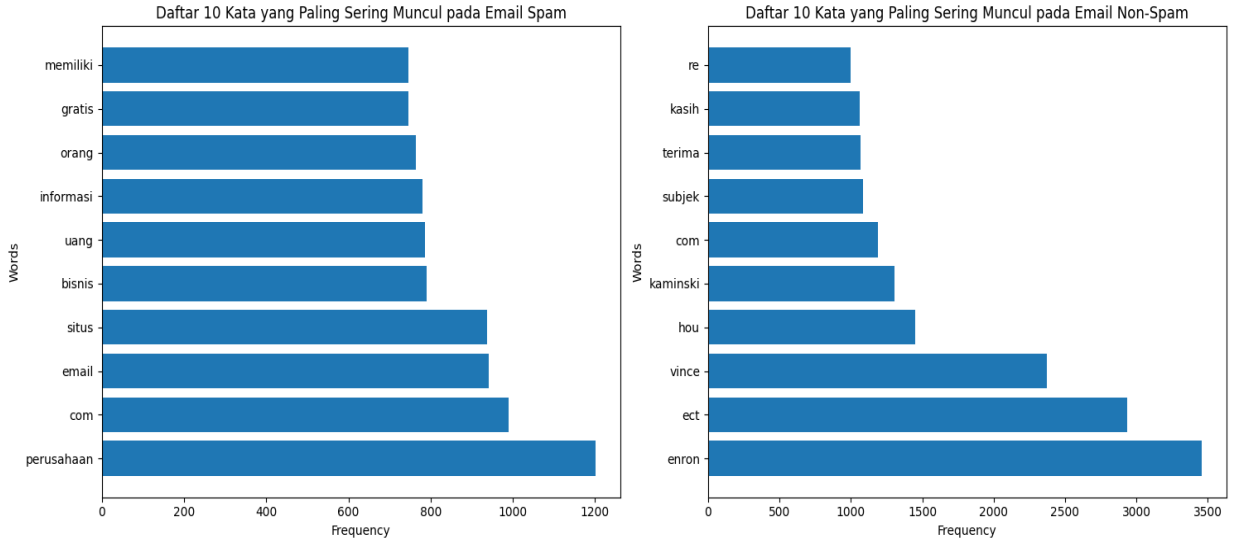
Tahap selanjutnya yang dilakukan adalah EDA. EDA dilakukan untuk memahami isi dari dataset yang digunakan pada penelitian ini. Pada tahap ini terdapat visualisasi *word frequency* atau kata yang sering muncul pada masing-masing dataset serta visualisasi *Wordcloud. Wordcloud* merupakan bentuk visual dari keberadaan kata dalam dokumen. Semakin banyak istilah yang sering muncul dalam dokumen yang dianalisis, maka kata tersebut akan muncul semakin besar dalam gambar *Wordcloud*. Hal ini dilakukan agar diperoleh pemahaman mendalam mengenai kata-kata kunci yang dapat membedakan dalam proses klasifikasi.

Building of Informatics, Technology and Science (BITS)
Volume 6, No 4, March 2025 Page: 2559–2572
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i4.6949



Gambar 3. Visualisasi *word frequency* pada dataset *phishing email*

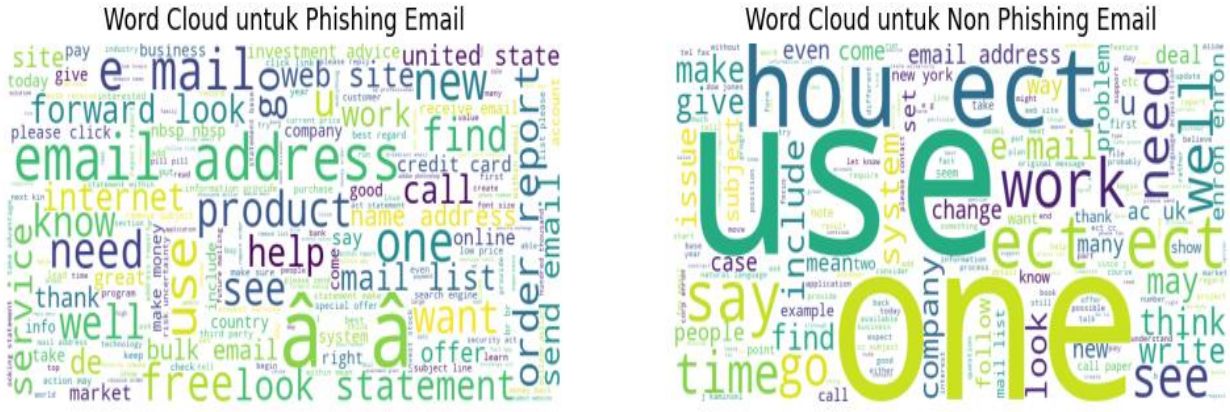
Pada Gambar 3 bisa dilihat bahwa kata "*language*" merupakan kata yang paling sering muncul pada *non-phishing* email, diikuti dengan kata "*enron*", *"university", "use",* dan seterusnya. Hal ini menunjukkan bahwa dari pola kata-kata yang sering muncul email *non-phishing* lebih fokus pada komunikasi yang berbasis informasi sah dan profesional. Sebaliknya, pada email *phishing* kata "*email*" merupakan kata yang paling sering muncul diikuti dengan kata "*company*", "*get*", "*free*", dan seterusnya. Dapat dilihat dari kata yang sering muncul mencerminkan taktik yang digunakan dalam *phishing email* untuk menciptakan rasa urgensi dan menarik perhatian penerima dengan tawaran yang tampaknya menguntungkan.



Gambar 4. Visualisasi *word frequency* pada dataset *spam email*

Pada Gambar 4 menunjukkan bahwa kata "perusahaan" merupakan kata yang paling sering muncul pada *spam email*, diikuti dengan kata "*com*", "*email*", "situs", dan seterusnya. Dari pola kata-kata yang sering muncul email spam sering kali mencoba meniru komunikasi yang tampaknya sah, sering kali dengan mengklaim berasal dari suatu perusahaan atau organisasi. Hal ini bertujuan untuk membangun kepercayaan penerima agar lebih mudah tertipu oleh isi email tersebut, seperti penipuan *phishing* atau promosi palsu. Penggunaan kata di atas menunjukkan adanya upaya manipulatif untuk menarik perhatian penerima dan mendorong mereka untuk mengambil tindakan tertentu, seperti mengklik tautan berbahaya atau mengunduh lampiran yang mencurigakan. Sebaliknya, pada email *non-spam* kata "*enron*" merupakan kata yang paling sering muncul diikuti dengan kata "*ect*", "*vince*", "*kaminski*", dan seterusnya. Dapat dilihat dari kata yang sering muncul mencerminkan komunikasi bisnis atau profesional yang lebih formal dan terorganisir. Penggunaan nama individu serta istilah yang berkaitan dengan pekerjaan menunjukkan bahwa email tersebut lebih banyak digunakan untuk keperluan internal perusahaan. Hal ini memperjelas bahwa email non-spam cenderung memiliki pola bahasa yang lebih spesifik dan berhubungan dengan komunikasi antarpegawai, diskusi proyek, atau koordinasi pekerjaan.

Building of Informatics, Technology and Science (BITS)
Volume 6, No 4, March 2025 Page: 2559−2572
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i4.6949



Gambar 5. Visualisasi *wordcloud* pada dataset *phishing email*

Pada Gambar 5, kata-kata seperti "*e-mail*", "*email address*", "*product*", "*investment*", "*send email*" mendominasi data dari *phishing email*. Hal ini menunjukkan bahwa phishing email sering kali berfokus pada teknik-teknik yang digunakan untuk memanipulasi penerima agar memberikan informasi pribadi atau mengakses situs yang berbahaya. Penggunaan kata-kata seperti "*email address*" dan "*send email*" menunjukkan bahwa penyerang mencoba untuk meniru komunikasi yang sah dan mencoba meyakinkan korban bahwa mereka menerima email dari sumber yang tepercaya. Kata-kata seperti "*product*" dan "*investment*" juga digunakan untuk menarik perhatian penerima dengan menawarkan produk atau investasi yang tampak menguntungkan, yang sebenarnya merupakan bagian dari skema penipuan. Sementara pada email *non-phishing* didominasi kata-kata seperti "*use*", "*work*", "*company*", "*need*", "*system*", dan "*see*". Dapat dilihat dari sini bahwa email *non-phishing* cenderung berfokus pada komunikasi profesional atau internal yang lebih berstruktur dan relevan dengan konteks pekerjaan atau organisasi. Penggunaan kata seperti "*company*" dan "*work*" mencerminkan komunikasi yang sering kali terkait dengan diskusi tentang pekerjaan atau kegiatan dalam sebuah perusahaan.

Gambar 6. Visualisasi *wordcloud* pada dataset email spam

Pada Gambar 6, spam email didominasi oleh kata-kata seperti "situs", "*web*", "perangkat lunak", "gratis", dan "program". Temuan ini menunjukkan bahwa spam email sering kali menggunakan istilah-istilah yang berhubungan dengan tawaran atau promosi layanan yang menggiurkan. Kata-kata seperti "gratis" dan "program" sering dikaitkan dengan penawaran perangkat lunak atau layanan yang tampaknya menarik, namun sering kali berisiko atau ilegal. Selain itu, kata-kata seperti "situs" dan "*web*" menunjukkan bahwa banyak spam email mengarahkan penerima untuk mengunjungi halaman web eksternal yang dapat mengandung ancaman, seperti *phishing* atau *malware*. Di sisi lain, pada ham email didominasi oleh kata-kata seperti "terima kasih", "kontrak", "ect", "*enron*", dan "informasi". Ini menunjukkan kata-kata yang umumnya ditemukan dalam komunikasi yang sah dan professional. Kata-kata seperti "terima kasih", "kontrak", dan "informasi" sering kali terkait dengan percakapan bisnis, menggambarkan interaksi yang sah antara pihak-pihak yang terlibat dalam diskusi resmi. Pola kata-kata ini yang membedakan antara email spam dan ham, dan memberi petunjuk dari ciri-ciri bagaimana komunikasi yang sah.

3.4 *Feature Extraction (*TF-IDF)

Langkah selanjutnya yang dilakukan adalah *Feature Extraction.* Tahap ini dilakukan dengan menggunakan metode TF-IDF (*Term Frequency - Inverse Document Frequency*) yang bertujuan untuk mengubah teks menjadi representasi numerik dengan menyoroti kata-kata yang paling penting berdasarkan frekuensi kemunculannya di dalam dokumen serta seberapa jarang kata tersebut muncul pada dokumen lain. Pada tahap ini, setiap email dalam dataset diubah

Building of Informatics, Technology and Science (BITS)
Volume 6, No 4, March 2025 Page: 2559−2572
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i4.6949

menjadi vektor fitur berdasarkan nilai TF-IDF dari setiap kata yang ada dalam teks. *Term Frequency* (TF) menghitung seberapa sering suatu kata muncul dalam sebuah dokumen, sementara *Inverse Document Frequency* (IDF) mengukur seberapa unik kata tersebut dalam keseluruhan kumpulan dokumen. Perhitungan TF-IDF dapat dilakukan menggunakan persamaan (5), (6), dan (7) berikut :

$$TF(t,d) = \frac{f_{t,d}}{\sum t' \in d\, f_{t',d}} \quad (1)$$

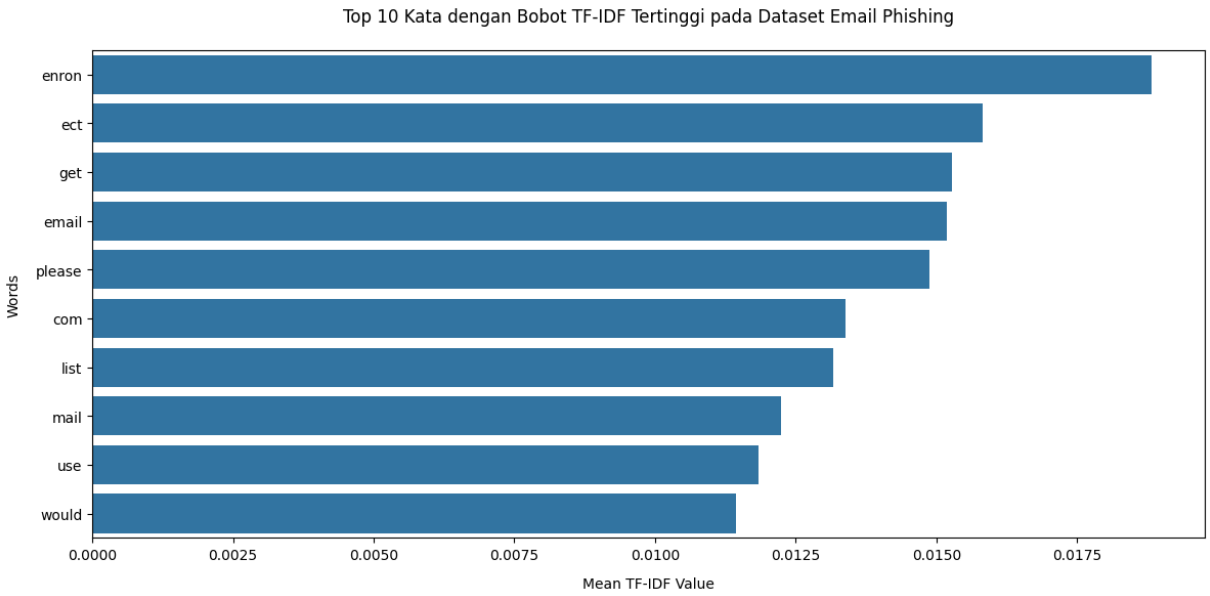
$$IDF(t,D) = log\frac{|D|}{|\{d \in D : t \in d\}|} \quad (2)$$

$$TF - IDF(t,d,D) = TF(t,d) \times IDF(t,D) \quad (3)$$

Keterangan :

a. TF : frekuensi kata dalam dokumen
b. IDF : seberapa jarang kata tersebut muncul di seluruh dokumen
c. TF-IDF : perkalian antara TF dan IDF untuk menentukan bobot dari suatu kata

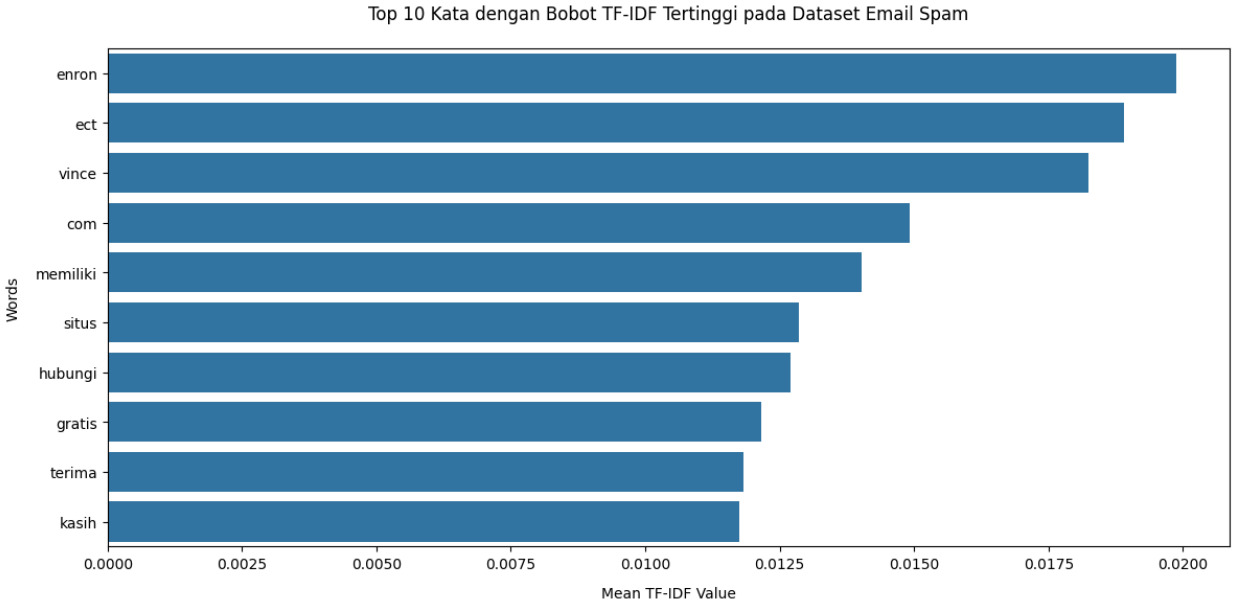
Pada analisis teks, *Term Frequency-Inverse Document Frequency* (TF-IDF) merupakan metode yang digunakan untuk menilai kepentingan suatu kata dalam dokumen relatif terhadap sekumpulan dokumen lainnya. TF-IDF terdiri dari tiga komponen utama: *Term Frequency* (TF), *Inverse Document Frequency* (IDF), dan hasil perkalian keduanya *Term Frequency* (TF), seperti ditunjukkan pada Rumus 5, dihitung menggunakan persamaan: di mana jumlah kemunculan kata dalam dokumen, dan penyebutnya merupakan jumlah total kata dalam dokumen tersebut. TF mengukur seberapa sering suatu kata muncul dalam satu dokumen dibandingkan dengan jumlah total kata yang ada. Semakin tinggi nilai TF, semakin sering kata tersebut muncul dalam dokumen. Namun, frekuensi kemunculan kata saja belum cukup untuk menentukan kepentingannya secara keseluruhan, karena beberapa kata umum dapat muncul berulang kali dalam berbagai dokumen. Untuk mengatasi masalah tersebut, digunakan *Inverse Document Frequency* (IDF), sebagaimana dinyatakan dalam Rumus 6 di mana jumlah total dokumen dalam kumpulan data, dan jumlah dokumen yang mengandung kata. IDF berfungsi untuk mengukur seberapa jarang suatu kata muncul dalam keseluruhan dokumen. Jika suatu kata muncul di banyak dokumen, maka nilai IDF-nya akan kecil, menandakan bahwa kata tersebut kurang informatif (misalnya kata-kata umum seperti "dan", "atau", "adalah"). Sebaliknya, jika suatu kata jarang muncul dalam koleksi dokumen, nilai IDF-nya akan lebih tinggi, menunjukkan bahwa kata tersebut lebih signifikan dalam membedakan dokumen. Akhirnya, TF-IDF diperoleh dengan mengalikan nilai TF dan IDF sesuai Rumus 7. Nilai TF-IDF ini memberikan bobot pada suatu kata dalam dokumen tertentu dengan mempertimbangkan frekuensinya dalam dokumen serta kelangkaannya di seluruh dokumen. Kata yang sering muncul dalam satu dokumen tetapi jarang ditemukan di dokumen lain akan memiliki nilai TF-IDF tinggi.



Gambar 7. Visualisasi 10 kata dengan bobot TF-IDF tertinggi pada dataset email *phishing*

Gambar 7 menunjukkan 10 kata dengan bobot TF-IDF tertinggi di dalam dataset email *phishing*. Kata-kata ini memiliki bobot tinggi karena sering muncul dalam berbagai dokumen dan dianggap sebagai kata yang paling representatif di dalam dataset ini. Kata-kata seperti "*enron*", "*ect*", "*get*", dan "*email*" mendominasi karena sering digunakan dalam email *phishing*. Keberadaan kata-kata ini mencerminkan pola umum dalam email phishing, di mana penyerang sering kali menggunakan istilah-istilah yang umum ditemukan dalam komunikasi email untuk meningkatkan kredibilitas pesan mereka.

Building of Informatics, Technology and Science (BITS)
Volume 6, No 4, March 2025 Page: 2559−2572
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i4.6949



Gambar 8. Visualisasi 10 kata dengan bobot TF-IDF tertinggi pada dataset email spam

Pada Gambar 8 juga ditunjukkan 10 kata dengan bobot TF-IDF tertinggi di dalam dataset email spam. Kata-kata seperti "hubungi", "gratis", "situs", dan "terima kasih" mendominasi karena sering digunakan dalam email spam yang berisi iklan atau penawaran komersial.

3.5 Klasifikasi

Setelah proses *pre-processing* dan *feature extraction* menggunakan TF-IDF, langkah selanjutnya adalah melakukan klasifikasi email menggunakan empat model *machine learning*, yaitu *Support Vector Machine* (SVM), *Decision Tree* (DT), *Naive Bayes* (NB), dan *Neural Network* (NN). Evaluasi dilakukan berdasarkan metrik akurasi, *precision*, *recall*, dan F1-*score* untuk menilai kinerja masing-masing model dalam mengklasifikasikan email sebagai *spam/non-spam* atau *phishing/non-phishing*. Pembuatan data model dibuat menggunakan skenario pembagian data latih dan data uji dalam rasio 80:20.

3.6 Evaluasi

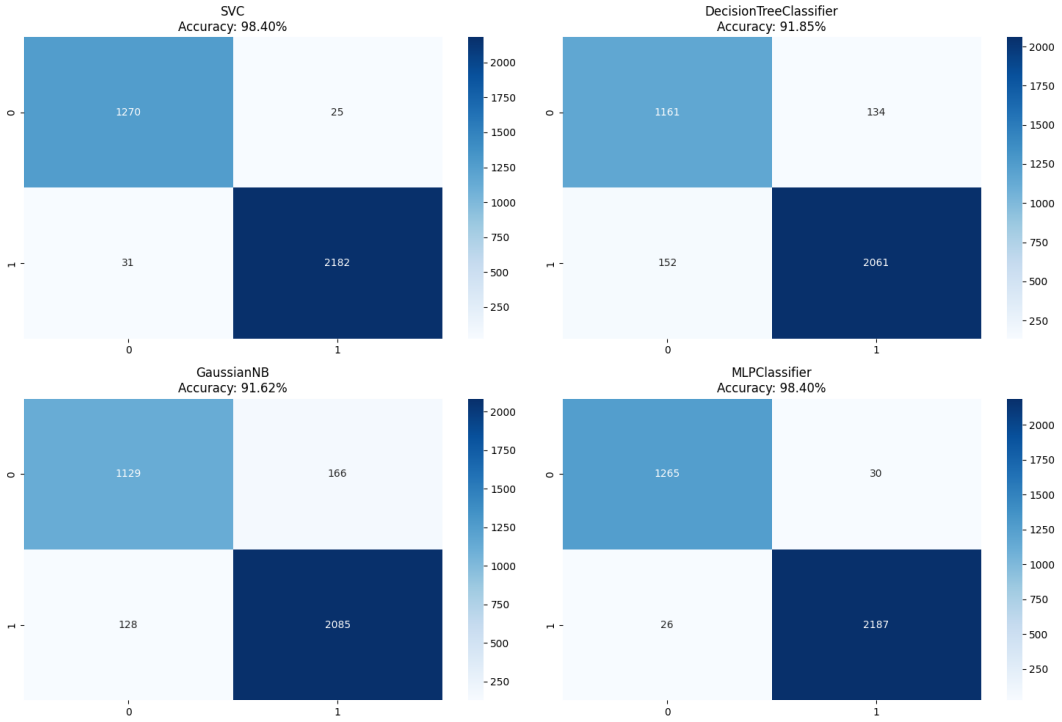
Setelah pembuatan model, langkah selanjutnya adalah evaluasi model. Setelah dilakukan proses pembuatan model didapat metrik berikut dari kedua dataset. Pada Tabel 10, bisa dilihat bahwa SVM dan *Neural Network* unggul dalam akurasi dengan nilai akurasi sebesar 98.40% masing-masing. Kemudian, diikuti dengan *Decision Tree* yang memiliki akurasi sebesar 91.85% dan *Naive Bayes* dengan akurasi 91.62% dalam deteksi email phishing.

Tabel 10. Perbandingan akurasi, presisi, *recall*, dan F1-*score* pada dataset email phishing

	Accuracy	Precision	Recall	F1-score
SVM	98.40%	98.86%	98.59%	98.72%
Decision Tree	91.85%	94.27%	93.04%	93.65%
Naive Bayes	91.62%	92.62%	94.21%	93.41%
Neural Network	98.40%	98.64%	98.82%	98.76%

Agar lebih jelas, Gambar 9 menyajikan perbandingan *confusion matrix* dari masing-masing model yang diuji. Dari visualisasi ini, dapat diamati bagaimana setiap model mengklasifikasikan email, termasuk jumlah *true positives* (TP), *false positives* (FP), *true negatives* (TN), dan *false negatives* (FN). SVM dan *Neural Network* mencatatkan TN dan FN paling sedikit, artinya kedua model ini memiliki kemampuan yang sangat baik dalam mengidentifikasi email phishing. Sementara itu, *Naive Bayes* dan *Decision Tree* disini mencatatkan jumlah TN dan FN lebih dari 100, yang menunjukkan bahwa kedua model ini memiliki tingkat kesalahan yang lebih tinggi dibandingkan SVM dan *Neural Network* dalam mengklasifikasikan email. Namun, di balik kelemahan tersebut, kedua model ini mencatatkan kecepatan komputasi yang cukup cepat daripada SVM dan *Neural Network* yang secara alami memang membutuhkan waktu lebih lama. Dengan demikian, pemilihan model harus disesuaikan dengan kebutuhan pengguna. Jika akurasi menjadi prioritas utama dalam klasifikasi, maka model dengan performa terbaik berdasarkan evaluasi metrik dapat dipilih, meskipun mungkin memerlukan sumber daya komputasi yang lebih tinggi.

Building of Informatics, Technology and Science (BITS)
Volume 6, No 4, March 2025 Page: 2559−2572
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i4.6949



Gambar 9. Visualisasi perbandingan *confusion matrix* dari masing-masing model

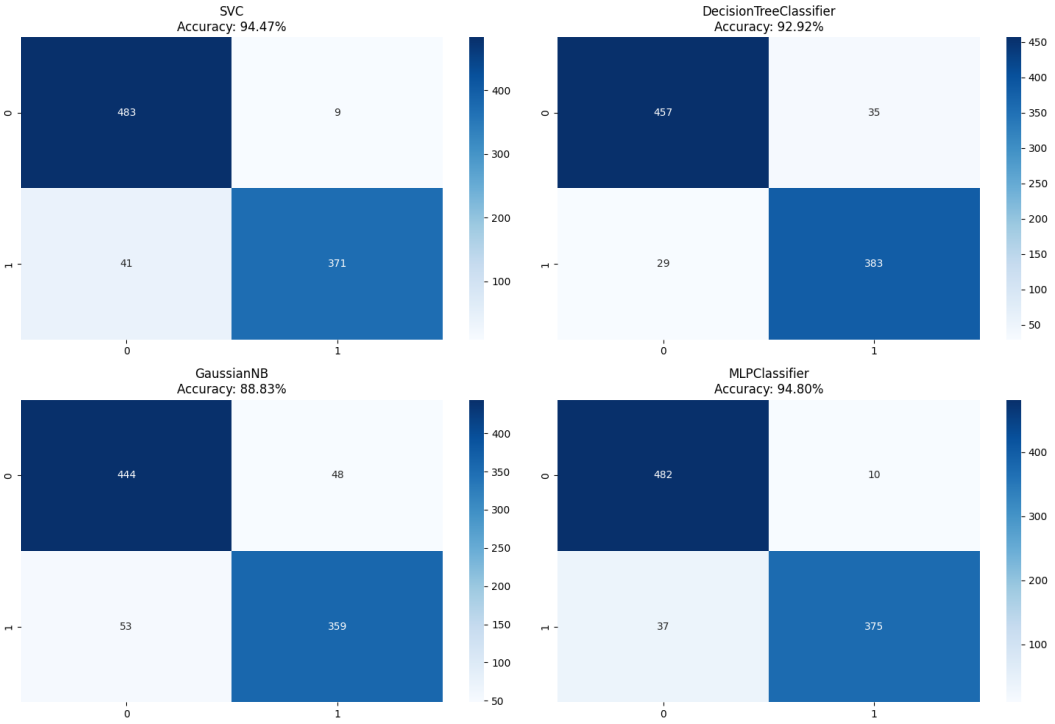
Sementara itu, pada dataset email spam, *Neural Network* masih mencatatkan akurasi tertinggi dibandingkan model lainnya. Berdasarkan Tabel 11, *Neural Network* mencapai akurasi hingga 94.80% diikuti SVM dengan akurasi yang hampir sama dengan nilai 94.47%. Selanjutnya, *Decision Tree* menyusul dengan akurasi sebesar 92.92%, yang masih cukup kompetitif. *Naive Bayes*, meskipun memiliki keunggulan dalam efisiensi komputasi, mencatatkan akurasi terendah sebesar 88.83%. Dari hasil ini, terlihat konsistensi tinggi yang ditunjukkan oleh *Neural Network* dan SVM, di mana kedua model ini selalu mencatatkan akurasi tertinggi pada kedua dataset yang digunakan. Keunggulan konsisten ini mengindikasikan bahwa kedua model tersebut lebih efektif dalam menangkap pola yang relevan dalam data, baik untuk email spam berbahasa Indonesia maupun bahasa Inggris.

Tabel 11. Perbandingan akurasi, presisi, *recall*, dan F1-*score* pada dataset email spam

	Accuracy	**Precision**	**Recall**	**F1-score**
SVM	94.47%	97.63%	90.04%	93.70%
Decision Tree	92.92%	91.62%	92.96%	92.34%
Naive Bayes	88.83%	88.20%	87.13%	87.70%
Neural Network	94.80%	97.40%	91.01%	94.14%

Keunggulan *Neural Network* dan SVM juga tidak hanya pada akurasi yang tinggi. Pada dataset email spam ini, kedua model ini berhasil mengurangi jumlah *false fositive* (FP) dan *false negative* (FN) seperti pada Gambar 10, hal ini menunjukkan adanya peningkatan dalam keakuratan klasifikasi dibandingkan dengan model yang lain. Penurunan FP dan FN ini berarti bahwa kedua model ini lebih mampu menghindari kesalahan klasifikasi dalam deteksi email spam. Dengan berkurangnya *false positive*, email yang sebenarnya aman tidak salah diklasifikasikan sebagai spam, sehingga mengurangi risiko kehilangan informasi penting akibat pemindahan email ke folder spam. Di sisi lain, penurunan *false negative* menunjukkan bahwa lebih sedikit email spam yang lolos ke dalam kotak masuk pengguna, sehingga meningkatkan perlindungan terhadap potensi ancaman seperti *phishing* atau *malware*. Selain itu, keandalan *Neural Network* dalam mengenali pola yang kompleks membantu model ini untuk mengidentifikasi email spam yang menggunakan teknik penyamaran, seperti pengubahan karakter atau penyisipan kata yang tidak relevan. SVM, dengan kemampuannya dalam menemukan hyperplane optimal, memastikan pemisahan yang lebih jelas antara email spam dan non-spam, bahkan dalam kondisi data yang memiliki kemiripan tinggi.

Building of Informatics, Technology and Science (BITS)
Volume 6, No 4, March 2025 Page: 2559−2572
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i4.6949



Gambar 10. Visualisasi perbandingan *confusion matrix* dari masing-masing model

Dari keseluruhan model yang diuji, dapat dilihat pada Tabel 12 bahwa *Neural Network* berhasil mencatatkan akurasi tertinggi, mencapai rata-rata 96.60% dalam mendeteksi email *phishing* dan spam, menjadikannya model yang paling efektif dalam klasifikasi data ini. SVM mengikuti dengan akurasi rata-rata 96.43%, yang menunjukkan performa yang sangat kompetitif, hampir setara dengan *Neural Network*. Selanjutnya, *Decision Tree* menunjukkan hasil yang cukup baik dengan rata-rata akurasi 92.38%. Terakhir, *Naive Bayes* memperoleh rata-rata akurasi 90.22%. *Neural Network* dan SVM unggul karena kemampuannya dalam menangkap pola kompleks dan memberikan akurasi yang sangat baik. *Neural Network* memiliki kemampuan untuk mengidentifikasi pola yang rumit. Model ini, terutama *Multi-Layer Perceptron Classifier* (MLPC), dapat melakukan pembelajaran mendalam dan menghasilkan representasi yang lebih banyak dari data input, yang sangat penting dalam mengklasifikasikan email dengan tingkat kerumitan tertentu, seperti spam dan *phishing*. Sama halnya dengan SVM yang hampir setara dalam hal performa, dengan kelebihannya untuk menangani data yang memiliki banyak fitur, berkat penggunaan fungsi kernel yang memungkinkan SVM untuk melakukan pemisahan data secara fleksibel, meskipun SVM tidak mencapai akurasi setinggi *Neural Network*, performanya tetap sangat kompetitif dan lebih efisien dalam hal komputasi. Disisi lain, *Decision Tree* meskipun memberikan hasil yang cukup baik dengan akurasi 92.38%, memiliki beberapa kelemahan. Salah satunya adalah kecenderungan untuk *overfitting* pada data yang sangat besar atau kompleks. Selain itu, keputusan yang dibuat dalam *Decision Tree* seringkali sangat bergantung pada fitur yang paling dominan, sehingga kurang efektif ketika fitur-fitur tersebut memiliki hubungan non-linier yang rumit. Sementara itu, *Naive Bayes* memiliki akurasi terendah di antara keempat model%. Meskipun demikian, *Naive Bayes* masih berguna dalam skenario yang mengutamakan kecepatan pemrosesan dan efisiensi komputasi. Salah satu kekurangan utama *Naive Bayes* adalah asumsi independensi antar fitur yang mungkin tidak selalu valid dalam kasus email spam atau *phishing*, di mana banyak fitur (kata-kata dalam email) sering kali saling bergantung.

Tabel 12. Perbandingan rata-rata akurasi, presisi, *recall*, dan F1-*score*

	Accuracy	**Precision**	**Recall**	**F1-score**
SVM	96.43%	**98.24%**	94.32%	96.21%
Decision Tree	92.38%	92.95%	93.00%	93.00%
Naive Bayes	90.22%	90.41%	90.67%	90.56%
Neural Network	**96.60%**	98.02%	**94.92%**	**96.45%**

4. KESIMPULAN

Penelitian ini telah dilakukan untuk mendapatkan perbandingan dari empat metode *machine learning*, yaitu *Neural Network*, *Support Vector Machine* (SVM), *Decision Tree*, dan *Naive Bayes* dalam tugas klasifikasi email *phishing* dan spam. Berdasarkan hasil evaluasi di atas, *Neural Network* mencatatkan akurasi tertinggi dengan rata-rata 96.60%, diikuti oleh SVM dengan akurasi 96.43%. Kedua model ini menunjukkan performa yang sangat baik dalam mendeteksi email *phishing* dan spam. Sementara itu, *Decision Tree* dan *Naive Bayes* memiliki akurasi yang lebih

Building of Informatics, Technology and Science (BITS)
Volume 6, No 4, March 2025 Page: 2559−2572
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i4.6949

rendah, masing-masing 92.38% dan 90.22%, namun tetap memberikan hasil yang kompetitif dan relevan dalam aplikasi yang mengutamakan efisiensi komputasi dan kecepatan pemrosesan. Walaupun didapat hasil yang cukup memuaskan, penelitian ini masih memiliki beberapa keterbatasan. Salah satunya adalah penggunaan metrik yang terbatas, yaitu hanya akurasi, *precision*, *recall*, dan F1-*score* saja. Meskipun metrik-metrik ini memberikan gambaran yang baik tentang kemampuan model dalam klasifikasi, mereka belum mempertimbangkan faktor-faktor lain yang juga penting, seperti kecepatan komputasi, *scalability*, dan kemampuan interpretasi model. Oleh karena itu, penelitian selanjutnya diharapkan dapat mempertimbangkan faktor-faktor tambahan ini dalam mengevaluasi model. Di masa depan, akan sangat berguna juga untuk melakukan *hyperparameter tuning* yang lebih mendalam dan menguji model pada dataset yang lebih besar serta lebih beragam, untuk mengevaluasi kemampuan generalisasi model pada data yang lebih kompleks dan beragam.

REFERENCES

[1] R. S. Lutfiyani and N. Retnowati, “Implementasi Pendeteksian Spam Email Menggunakan Metode Text Mining dengan Algoritma Naïve Bayes dan Decision Tree J48,” *Jurnal Komputer dan Informatika*, vol. 9, no. 2, pp. 244–252, Oct. 2021, [Online]. Available: https://doi.org/10.35508/jicon.v9i2.5304

[2] K. M. S. Hidayatullah and T. Sutabri, “Pengembangan Sistem Pengklasifikasi e-mail Berbasis Kecerdasan Buatan untuk Deteksi Spam dan Phishing,” *IJM: Indonesian Journal of Multidisciplinary*, vol. 2, no.2, Apr. 2024. [Online]. Available: https://journal.csspublishing/index.php/ijm/article/view/689

[3] D. Anggraini and T. Sutabri, “Pengembangan Aplikasi Penyaringan Spam e-mail Menggunakan Teknik Machine Learning dengan Metode Support Vector Machines,” *IJM: Indonesian Journal of Multidisciplinary*, vol. 2, no. 3, pp. 106–114, Apr. 2024. [Online]. Available: https://journal.csspublishing/index.php/ijm/article/view/720

[4] A. Kumar, J. M. Chatterjee, and V. G. Díaz, “A Novel Hybrid Approach of SVM Combined with NLP and Probabilistic Neural Network for Email Phishing,” *International Journal of Electrical and Computer Engineering*, vol. 10, no. 1, pp. 486–493, 2020. [Online]. Available: https://doi.org/10.11591/ijece.v10i1.pp486-493

[5] R. P. Ramadhan and T. Desyani, “Implementasi Algoritma J48 Untuk Identifikasi Website Phising,” *BINER: Jurnal Ilmu Komputer, Teknik dan Multimedia*, vol. 1, no. 2, pp. 46–54, Jun. 2023. [Online]. Available: https://journal.mediapublikasi.id/index.php/Biner/article/view/2557

[6] Q. Ouyang, J. Tian, and J. Wei, “E-mail Spam Classification using KNN and Naive Bayes,” *Highlights in Science, Engineering and Technology*, vol. 38, pp. 57–63, Mar. 2023. [Online]. Available: https:// doi.org/10.54097/hset.v38i.5699

[7] N. L. Octaviani, E. H. Rachmawanto, C. A. Sari, and I. M. S. De Rosal, "Comparison of multinomial naïve Bayes classifier, support vector machine, and recurrent neural network to classify email spams," in *Proceedings of the 2020 International Seminar on Application for Technology of Information and Communication (iSemantic)*, Sep. 2020, pp. 17–21. [Online]. Available: https://doi.org/10.1109/iSemantic50169.2020.9234296

[8] F. Alghifari and D. Juardi, “Penerapan Data Mining pada Penjualan Makanan dan Minuman Menggunakan Metode Algoritma Naïve Bayes,” *JURNAL ILMIAH INFORMATIKA*, vol. 9, no. 02, pp. 75–81, Sep. 2021. [Online]. Available: https://doi.org/10.33884/jif.v9i02.3755

[9] D. Chicco, L. Oneto, and E. Tavazzi, “Eleven quick tips for data cleaning and feature engineering,” *PLoS Computational Biology*, vol. 18, no. 12, p. e1010718, Dec. 2022. [Online]. Available: https://doi.org/10.1371/journal.pcbi.1010718

[10] M. U. Albab, Y. Karuniawati P, and M. N. Fawaiq, "Optimization of the stemming technique on text preprocessing President 3 periods topic," *Jurnal Transformatika*, vol. 20, no. 2, pp. 1–12, 2023. [Online]. Available: https://doi.org/10.26623/transformatika.v20i2.5374

[11]. Abidin, A. Junaidi, and Wamiliana, "Text stemming and lemmatization of regional languages in Indonesia: A systematic literature review," *Journal of Information Systems Engineering and Business Intelligence*, vol. 10, no. 2, pp. 217–231, Jun. 2024. [Online]. Available: https://doi.org/10.20473/jisebi.10.2.217-231

[12] M. J. Prasetyo and I. M. A. Agastya, “Sentiment Analysis of Banking Application Reviews on Google Play Store Using Support Vector Machine Algorithm,” *Sistemasi: Jurnal Sistem Informasi*, vol. 13, no. 6, pp. 2386–2400, 2024. [Online]. Available: http://sistemasi.ftik.unisi.ac.id/index.php/stmsi/article/view/4536

[13] R. Ramadhani, R. Ramadhanu, and T. Hidayat, “Exploratory Data Analysis (EDA) untuk Mengetahui Distribusi Data Kualitas Susu Sapi,” *Jurnal SAINTIKOM (Jurnal Sains Manajemen Informatika dan Komputer)*, vol. 23, no. 1, pp. 68-76, Feb. 2024. [Online]. Available: https://doi.org/10.53513/jis.v23i1.9500

[14] M. Radhi, A. Amalia, D. R. H. Sitompul, S. H. Sinurat, and E. Indra, "Analisis Big Data dengan Metode Exploratory Data Analysis (EDA) dan Metode Visualisasi Menggunakan Jupyter Notebook," *Jurnal Sistem Informasi dan Ilmu Komputer Prima*, vol. 4, no. 2, pp. 23–27, 2021. [Online] Available: https://jurnal.unprimdn.ac.id/index.php/JUSIKOM/article/view/2475

[15] S. Sumayah, F. Sembiring, and W. Jatmiko, "Analysis of sentiment of Indonesian community on metaverse using support vector machine algorithm," *Jurnal Teknik Informatika (JUTIF)*, vol. 4, no. 1, pp. 143–150, 2023. [Online]. Available: https://doi.org/10.20884/1.jutif.2023.4.1.417

[16] A. M. R. Armaya, “Pengaruh Feature Selection dan Feature Extraction dalam Peningkatan Akurasi Klasifikasi Kebakaran Hutan,” *JuTI “Jurnal Teknologi Informasi*,” vol. 3, no. 1, p. 13, Aug. 2024. [Online]. Available: http://dx.doi.org/10.26798/juti.v3i1.1039

[17] W. N. I. Al-Obaydy, H. A. Hashim, Y. A. Najm, and A. A. Jalal, “Document classification using term frequency-inverse document frequency and K-means clustering,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 27, no. 3, p. 1517, Sep. 2022. [Online]. Available: https://doi.org/10.11591/ijeecs.v27.i3.pp1517-1524

[18] A. Nugroho, "Text Analysis dan Text Mining," in Data Science Menggunakan Bahasa R, E. S. Mulyanta, Ed. Jogja: Penerbit Andi, 2024, pp. 112–123.

Building of Informatics, Technology and Science (BITS)
Volume 6, No 4, March 2025 Page: 2559−2572
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i4.6949

[19] H. Han, B. Shi, and L. Zhang, “Prediction of landslide sharp increase displacement by SVM with considering hysteresis of groundwater change,” *Engineering Geology*, vol. 280, p. 105876, Jan. 2021. [Online]. Available: https://doi.org/10.1016/j.enggeo.2020.105876

[20] N. A. Priyanka and D. Kumar, “Decision tree classifier: a detailed survey,” *International Journal of Information and Decision Sciences*, vol. 12, no. 3, p. 246, 2020. [Online]. Available: https://doi.org/10.1504/IJIDS.2020.108141

[21] M. V. Anand, B. KiranBala, S. R. Srividhya, K. C., M. Younus, and M. H. Rahman, “Gaussian Naïve Bayes Algorithm: A Reliable Technique Involved in the Assortment of the Segregation in Cancer,” *Mobile Information Systems*, vol. 2022, pp. 1–7, Jun. 2022. [Online]. Available: https://doi.org/10.1155/2022/2436946

[22] D. Singh and N. S. Rajput, "Blockchain Technology for Smart Cities," in *Blockchain Technologies*, D. Singh and N. S. Rajput, Eds. Singapore: Springer Singapore, 2020, pp. 67–68. [Online]. Available: https://doi.org/10.1007/978-981-15-2205-5

[23] N. K. E. Sapitri, U. Sa’adah, and N. Shofianah, “Knowledge Discovery from Confusion Matrix of Pruned CART in Imbalanced Microarray Data Ovarian Cancer Classification,” Scientific Journal of Informatics, vol. 11, no. 1, pp. 227–236, Feb. 2024. [Online]. Available: https://doi.org/10.15294/sji.v11i1.50077