

PENERAPAN METODE SUPPORT VECTOR MACHINE DALAM MENGLASIFIKASIKAN BERITA POLITIK : FAKTA VS HOAKS

Nikmal Maulana¹⁾, Maya Rini Handayani²⁾, Wenty Dwi Yuniarti³⁾, Khothibul Umam^{*4)}

1. Teknologi Informasi, Fakultas Sains dan Teknologi, Universitas Islam Negeri Walisongo, Indonesia
2. Teknologi Informasi, Fakultas Sains dan Teknologi, Universitas Islam Negeri Walisongo, Indonesia
3. Teknologi Informasi, Fakultas Sains dan Teknologi, Universitas Islam Negeri Walisongo, Indonesia
4. Teknologi Informasi, Fakultas Sains dan Teknologi, Universitas Islam Negeri Walisongo, Indonesia

Article Info

Kata Kunci: *deteksi otomatis; hoaks politik; klasifikasi berita; LinearSVC; Support Vector Machine; TF-IDF.*

Keywords: *automated detection; political hoax; news classification; LinearSVC; Support Vector Machine; TF-IDF.*

Article history:

Received 3 June 2025

Revised 15 June 2025

Accepted 30 June 2025

Available online 1 September 2025

DOI :

<https://doi.org/10.29100/jipi.v10i3.8077>

* Corresponding author.

Khothibul Umam

E-mail address:

khothibul_umam@walisongo.ac.id

ABSTRAK

Berita hoaks di ranah politik menjadi tantangan serius dalam era digital saat ini, karena dapat memengaruhi opini publik dan kestabilan sosial. Penelitian ini bertujuan untuk membangun sistem klasifikasi otomatis yang mampu membedakan antara berita politik fakta dan hoaks. Metode yang digunakan adalah Support Vector Machine (SVM) dengan pendekatan Linear Support Vector Classification (LinearSVC), yang dikenal memiliki performa tinggi dalam menangani data teks berdimensi besar. Dataset yang digunakan terdiri dari 1.267 berita politik berbahasa Inggris yang telah dilabeli sebagai "REAL" dan "FAKE". Proses awal melibatkan pembersihan teks, tokenisasi, penghapusan stopword, dan transformasi data teks menjadi vektor numerik menggunakan TF-IDF. Model SVM kemudian dilatih dan diuji menggunakan pembagian data latih dan uji secara terstruktur. Hasil evaluasi menunjukkan bahwa model mencapai akurasi sebesar 94,24%. Selain itu, nilai precision, recall, dan f1-score masing-masing sebesar 0,94 untuk kedua kelas, yaitu berita fakta dan hoaks. Temuan ini menunjukkan bahwa metode SVM sangat efektif dalam mengklasifikasikan berita politik, serta berpotensi diterapkan dalam sistem deteksi hoaks berbasis teks secara otomatis. Model dan vectorizer disimpan untuk mendukung penggunaan ulang dalam aplikasi nyata. Penelitian ini memberikan kontribusi praktis dalam memitigasi penyebaran informasi palsu melalui teknologi kecerdasan buatan.

ABSTRACT

The spread of hoaxes in political news has become a serious challenge in today's digital era, as it can influence public opinion and social stability. This study aims to develop an automatic classification system capable of distinguishing between factual and fake political news. The method employed is Support Vector Machine (SVM) using the Linear Support Vector Classification (LinearSVC) approach, known for its high performance in handling high-dimensional textual data. The dataset consists of 1,267 English-language political news articles labeled as "REAL" and "FAKE." The preprocessing phase includes text cleaning, tokenization, stopword removal, and transforming the text data into numerical vectors using TF-IDF. The SVM model was trained and evaluated using a structured train-test split. Evaluation results show that the model achieved an accuracy of 94.24%. Additionally, the model obtained a precision, recall, and F1-score of 0.94 for both classes, indicating a balanced and effective classification. These findings demonstrate that SVM is highly effective in classifying political news and has potential for real-world implementation in automated hoax detection systems. The trained model and vectorizer are stored to support reuse in future applications. This research contributes practical value in combating the spread of misinformation through artificial intelligence-based technology.

I. PENDAHULUAN

D era digital saat ini, penyebaran informasi menjadi sangat cepat dan mudah diakses, namun diiringi pula dengan tantangan serius berupa maraknya berita palsu atau hoaks. Fenomena hoaks telah menjadi masalah global yang dapat mengganggu ketertiban demokrasi serta stabilitas sosial, budaya, politik, dan ekonomi [1]. Menurut laporan, 44,3% responden menerima hoaks atau misinformasi setiap hari, menunjukkan luasnya dampak masalah ini [1]. Khususnya dalam konteks politik, hoaks memiliki potensi besar untuk memanipulasi opini publik dan mempengaruhi hasil pemilihan umum, seperti yang terlihat pada Pemilu 2024 di Indonesia, di mana terjadi peningkatan signifikan dalam penanganan isu hoaks oleh Kementerian Komunikasi dan Informatika (Kominfo) [7], [9], [8]. Ancaman hoaks politik terhadap keamanan negara dan proses demokrasi menggarisbawahi urgensi pengembangan sistem deteksi yang canggih dan akurat [6].

Membedakan berita fakta dan hoaks secara otomatis merupakan tantangan kompleks yang multifaset. Hal ini bukan hanya sekadar identifikasi kata kunci semata, melainkan juga melibatkan analisis mendalam terhadap karakteristik linguistik, gaya penulisan, dan pola manipulasi informasi. Berita hoaks seringkali sengaja dirancang untuk menyerupai berita asli, menggunakan judul provokatif (*clickbait*), diksi yang memicu emosi, klaim yang tidak terbukti, atau bahkan narasi yang direkayasa agar tampak kredibel. Tantangan ini diperparah oleh volume data yang sangat besar dan kecepatan penyebarannya, membuat verifikasi manual menjadi tidak praktis dan tidak efisien. Oleh karena itu, pengembangan sistem deteksi hoaks otomatis menjadi sangat relevan.

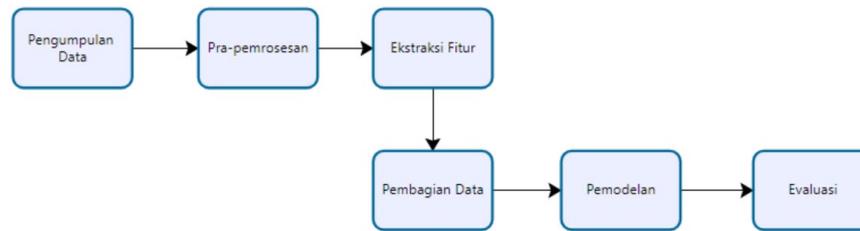
Untuk mengatasi penyebaran hoaks yang masif, penerapan metode *machine learning* dalam deteksi berita telah menjadi pendekatan yang efektif. Berbagai penelitian telah mengeksplorasi penggunaan algoritma *machine learning* untuk klasifikasi teks dan identifikasi hoaks [10]. Dalam tugas klasifikasi teks, representasi fitur adalah langkah krusial. Salah satu metode ekstraksi fitur yang banyak digunakan dan terbukti efektif adalah Term Frequency-Inverse Document Frequency (TF-IDF), yang mampu menangkap bobot relevansi kata dalam dokumen dan korpus [2]. Pendekatan ini telah berhasil diterapkan dalam sistem deteksi hoaks pada berita berbahasa Inggris [2].

Setelah fitur diekstraksi, langkah selanjutnya adalah klasifikasi. Algoritma Support Vector Machine (SVM) telah menunjukkan performa yang kuat dalam berbagai tugas klasifikasi teks, termasuk deteksi hoaks. Meskipun berbagai algoritma *machine learning* lain seperti Naive Bayes, Decision Tree, atau K-Nearest Neighbors juga umum digunakan dalam klasifikasi teks, SVM seringkali menunjukkan keunggulan, terutama pada data berdimensi tinggi atau *sparse* (data dengan banyak nilai nol), karakteristik umum dari representasi fitur teks seperti TF-IDF. SVM bekerja dengan mencari *hyperplane* optimal yang memaksimalkan margin pemisah antar kelas, sehingga efektif dalam menangani masalah klasifikasi biner seperti deteksi hoaks. Penelitian sebelumnya telah menunjukkan efektivitas SVM dalam mendeteksi berita hoaks, baik dalam konteks umum maupun politik [5], [4], [7]. Bahkan, kombinasi TF-IDF dan SVM telah terbukti memberikan akurasi tinggi dalam klasifikasi abstrak paper informatika, menunjukkan kapabilitasnya untuk tugas klasifikasi teks yang kompleks [3].

Mengingat urgensi masalah hoaks politik dan potensi kombinasi TF-IDF dengan LinearSVC (sejenis SVM) dalam klasifikasi teks, penelitian ini bertujuan untuk mengembangkan sistem deteksi berita hoaks politik berbahasa Inggris menggunakan pendekatan TF-IDF untuk ekstraksi fitur dan LinearSVC sebagai algoritma klasifikasi. LinearSVC dipilih secara spesifik karena efisiensi komputasinya yang tinggi dan kemampuannya untuk berkinerja kuat pada dataset besar dengan fitur yang dapat dipisahkan secara linear, menjadikannya pilihan ideal untuk aplikasi deteksi hoaks berbasis teks dengan volume data yang signifikan. Dengan dataset berita hoaks politik berbahasa Inggris, penelitian ini diharapkan dapat memberikan kontribusi dalam meningkatkan akurasi deteksi hoaks dan menawarkan solusi yang relevan untuk mengatasi tantangan disinformasi di era digital.

II. METODE PENELITIAN

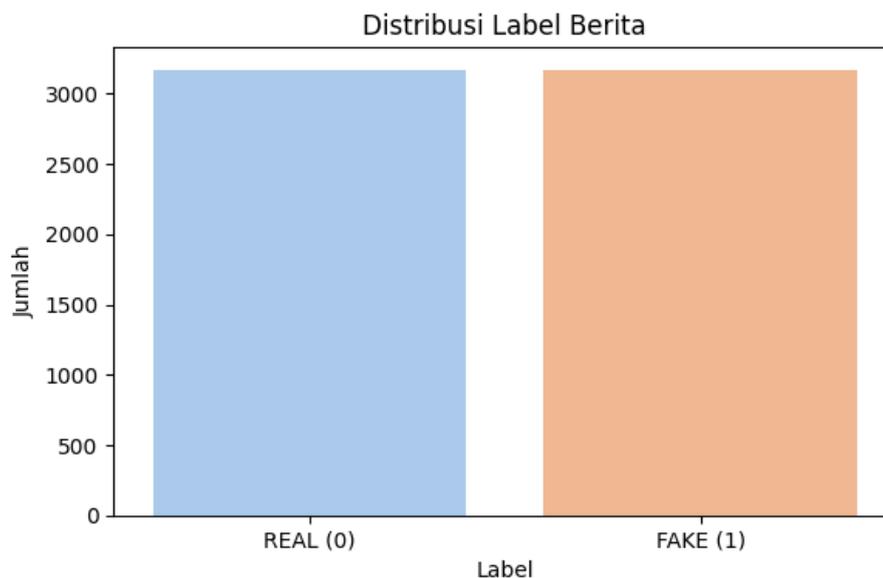
Penelitian ini mengimplementasikan metode klasifikasi teks menggunakan algoritma *Support Vector Machine* (SVM) untuk membedakan berita politik fakta dan hoaks. Pendekatan ini merupakan salah satu strategi efektif dalam ranah *Natural Language Processing* (NLP) untuk mengatasi penyebaran disinformasi di era digital [11]. Seluruh tahapan penelitian ini didesain untuk membangun sebuah sistem deteksi hoaks yang otomatis dan akurat. Diagram alir pada Gambar 1 menyajikan gambaran umum metodologi yang diterapkan dalam penelitian ini, meliputi pengumpulan data, pra-pemrosesan, ekstraksi fitur, pembagian data, pemodelan, hingga evaluasi.



Gambar. 1. Diagram Alir Metode Penelitian

A. Pengumpulan Data

Dataset yang digunakan dalam penelitian ini merupakan kumpulan berita politik berbahasa Inggris yang diperoleh dari platform publik Kaggle. Dataset ini terdiri dari 1.267 artikel berita yang telah dilabeli ke dalam dua kategori, yaitu "REAL" untuk berita fakta dan "FAKE" untuk berita hoaks. Kolom-kolom utama dalam dataset mencakup title (judul berita), text (isi berita), dan label (kategori berita). Penggunaan dataset berlabel ini krusial untuk melatih dan menguji model klasifikasi secara *supervised learning* [12]. Distribusi jumlah berita untuk setiap label dalam dataset dapat dilihat pada Gambar 2, menunjukkan representasi yang seimbang antara berita fakta dan hoaks, yang penting untuk menghindari bias model.



Gambar. 2. Distribusi Label Berita dalam Dataset

B. Pra-pemrosesan Data (Preprocessing)

Tahap pra-pemrosesan data merupakan langkah fundamental dalam tugas klasifikasi teks. Tujuan utama dari tahap ini adalah untuk membersihkan data mentah dan mengubahnya menjadi format yang lebih terstruktur dan informatif, sehingga siap untuk proses ekstraksi fitur dan pemodelan [11]. Serangkaian langkah pra-pemrosesan dilakukan pada kolom text dari dataset, meliputi:

1. Case Folding (Ubah ke Huruf Kecil): Seluruh teks diubah menjadi huruf kecil (lowercase). Langkah ini bertujuan untuk menyeragamkan data dan memastikan bahwa kata yang sama dengan kapitalisasi berbeda tidak diperlakukan sebagai entitas terpisah.
2. Penghapusan Angka: Semua karakter numerik ($\d+$) dihapus dari teks. Angka seringkali tidak memberikan informasi semantik yang signifikan dalam konteks klasifikasi fakta atau hoaks berita politik.
3. Penghapusan Tanda Baca: Karakter tanda baca (seperti koma, titik, tanda seru, dll.) dihapus. Tanda baca, meskipun penting dalam sintaksis, umumnya tidak berkontribusi pada makna inti yang relevan untuk klasifikasi dalam konteks ini.
4. Penghapusan Spasi Berlebih: Spasi ganda atau spasi yang tidak perlu di awal dan akhir teks dihapus untuk memastikan konsistensi format dan mengurangi *noise*.
5. Penghapusan Stopword: Kata-kata umum yang sering muncul dalam bahasa Inggris namun memiliki makna leksikal yang rendah (misalnya, "the", "is", "and") dihapus menggunakan daftar *stopwords* dari pustaka nltk (Natural Language Toolkit) [11]. Penghapusan *stopword* membantu mengurangi dimensi fitur dan

meningkatkan relevansi kata-kata yang tersisa.

Hasil dari proses pra-pemrosesan ini disimpan dalam kolom baru bernama `clean_text` yang akan digunakan untuk tahap selanjutnya.

C. Ekstraksi Fitur

Setelah data melalui tahap pra-pemrosesan, teks perlu dikonversi menjadi representasi numerik agar dapat diproses oleh algoritma *machine learning*. Metode yang dipilih untuk tahap ekstraksi fitur ini adalah *Term Frequency-Inverse Document Frequency* (TF-IDF). TF-IDF merupakan teknik statistik yang umum digunakan dalam *information retrieval* dan *text mining* untuk mengevaluasi seberapa penting sebuah kata dalam suatu dokumen relatif terhadap seluruh korpus dokumen [13].

TF-IDF dihitung berdasarkan dua komponen utama:

1. Term Frequency (TF): Mengukur seberapa sering sebuah kata muncul dalam sebuah dokumen.
2. Inverse Document Frequency (IDF): Mengukur seberapa jarang sebuah kata muncul di seluruh korpus dokumen. Kata yang jarang muncul di banyak dokumen memiliki nilai IDF yang lebih tinggi, mengindikasikan kepentingan yang lebih besar.

Dalam implementasi ini, `TfidfVectorizer` dari `scikit-learn` digunakan dengan beberapa parameter yang disesuaikan untuk optimasi:

1. `ngram_range=(1, 2)`: Fitur diekstrak tidak hanya dalam bentuk kata tunggal (*unigram*) tetapi juga kombinasi dua kata (*bigram*). Penggunaan *n-gram* dapat menangkap konteks dan frasa yang lebih bermakna [11].
2. `max_df=0.85`: Kata-kata yang muncul di lebih dari 85% dokumen akan diabaikan. Parameter ini membantu menghilangkan kata-kata yang terlalu umum dan tidak informatif, yang mungkin tidak tertangkap oleh daftar *stopword* standar.
3. `min_df=5`: Kata-kata yang muncul di kurang dari 5 dokumen akan diabaikan. Ini membantu menghilangkan kata-kata yang sangat jarang dan mungkin merupakan *typo* atau *noise*, serta mengurangi dimensi fitur.
4. `max_features=10000`: Jumlah fitur (kata/*n-gram*) yang dipertimbangkan dibatasi hingga 10.000 fitur teratas berdasarkan skor TF-IDF. Batasan ini membantu mengelola kompleksitas komputasi dan mencegah *overfitting* pada model.
5. `stop_words='english'`: Meskipun *stopword* sudah dihapus pada tahap pra-pemrosesan, parameter ini memberikan lapisan penghapusan *stopword* tambahan yang terintegrasi dalam proses TF-IDF.

Proses TF-IDF diterapkan pada data latih (`X_train`) untuk mempelajari kosakata dan bobot IDF, kemudian ditransformasikan pada data latih dan data uji (`X_test`) menggunakan model `vectorizer` yang telah dilatih.

D. Pembagian Data

Setelah data melewati seluruh tahapan pra-pemrosesan yang esensial, termasuk pembersihan teks dan ekstraksi fitur menggunakan metode TF-IDF, dataset kemudian memasuki fase pembagian menjadi dua subset yang berbeda secara fungsional: data latih (*training data*) dan data uji (*testing data*). Ini adalah langkah fundamental dalam setiap proyek *machine learning*, terutama untuk tugas klasifikasi. Tujuan utama dari pembagian ini adalah untuk memungkinkan evaluasi kinerja model secara objektif. Model akan dilatih eksklusif pada data latih, mempelajari pola dan hubungan antar fitur serta label kelas. Selanjutnya, kemampuannya akan diuji pada data uji yang sama sekali belum pernah "dilihat" oleh model selama proses pelatihan. Pendekatan ini sangat penting untuk mengukur kemampuan model dalam melakukan generalisasi pada data baru, sehingga dapat mencegah fenomena *overfitting* yang sering terjadi, di mana model terlalu menghafal data latih dan gagal berkinerja baik pada data yang tidak dikenal [13].

Dalam penelitian ini, pembagian dataset dilakukan dengan rasio 80% untuk data latih dan 20% untuk data uji. Rasio 80:20 ini merupakan pilihan yang umum dan direkomendasikan secara luas dalam literatur *machine learning* [11]. Dengan 80% data dialokasikan untuk pelatihan, model memiliki cukup informasi untuk mempelajari fitur-fitur yang relevan dan membangun representasi yang kuat dari kedua kategori berita (fakta dan hoaks). Sementara itu, 20% data uji dianggap porsi yang memadai untuk memberikan estimasi kinerja model yang tidak bias dan representatif terhadap kinerja model di skenario dunia nyata, tanpa mengurangi terlalu banyak data yang dibutuhkan untuk proses pembelajaran yang optimal.

Proses pembagian data ini diimplementasikan secara programatis menggunakan fungsi `train_test_split` yang disediakan oleh pustaka `scikit-learn` di Python, sebuah *library* standar dan powerful untuk *machine learning*. Aspek krusial dari implementasi ini adalah penggunaan parameter `random_state` yang diatur ke nilai konstan, yaitu `random_state=42`. Penetapan `random_state` ini berfungsi sebagai *seed* untuk generator angka acak internal yang mengendalikan proses pemisahan data. Konsekuensinya, setiap kali kode ini dijalankan, pembagian data akan selalu menghasilkan subset data latih dan data uji yang identik. Fitur ini sangat vital untuk menjamin *reproducibility* dari eksperimen. *Reproducibility* memastikan bahwa hasil evaluasi model dapat diverifikasi dan dibandingkan

secara konsisten di berbagai percobaan atau oleh peneliti lain yang ingin mereplikasi studi ini, sehingga meningkatkan keandalan dan validitas ilmiah dari temuan yang diperoleh [15]. Dengan demikian, pembagian data yang cermat ini membentuk fondasi yang solid untuk pelatihan dan evaluasi model klasifikasi yang adil dan valid.

E. Pemodelan (Modeling)

Untuk tugas klasifikasi berita fakta dan hoaks, algoritma *Support Vector Machine* (SVM) dipilih. SVM adalah algoritma *supervised learning* yang efektif untuk klasifikasi dan regresi. Prinsip dasar SVM adalah mencari *hyperplane* optimal yang paling baik memisahkan kelas-kelas data dalam ruang berdimensi tinggi [14]. Dalam kasus data teks yang cenderung memiliki dimensi tinggi setelah ekstraksi fitur TF-IDF, SVM terbukti sangat efektif [12].

Lebih spesifik, penelitian ini menggunakan Linear Support Vector Classification (LinearSVC). Pemilihan LinearSVC, dibandingkan varian SVM lain yang mungkin menggunakan *kernel* non-linear (seperti RBF *kernel* atau *polynomial kernel*), didasari oleh dua aspek utama: efisiensi komputasi dan performa klasifikasi pada data berdimensi tinggi.

- Efisiensi Komputasi: LinearSVC dirancang khusus untuk menangani dataset yang sangat besar dan *sparse* (data dengan banyak nilai nol) seperti yang dihasilkan oleh representasi fitur TF-IDF pada teks. Implementasinya lebih dioptimalkan untuk masalah klasifikasi linear, sehingga secara signifikan lebih cepat dalam proses pelatihan dan prediksi dibandingkan dengan SVM *kernel* non-linear yang membutuhkan perhitungan jarak antar semua pasangan titik data atau transformasi ke ruang fitur yang lebih tinggi. Ini sangat krusial untuk dataset dengan ribuan atau puluhan ribu fitur (yang sering terjadi pada TF-IDF dengan n-gram).
- Performa Klasifikasi pada Data Berdimensi Tinggi: Meskipun SVM dengan *kernel* non-linear dapat menangani separasi non-linear, dalam klasifikasi teks dengan TF-IDF, data seringkali dianggap memiliki separabilitas linear yang cukup baik di ruang berdimensi tinggi. LinearSVC mampu menemukan *hyperplane* pemisah yang optimal tanpa perlu pemetaan ke ruang fitur yang lebih kompleks, yang seringkali justru mengurangi risiko *overfitting* dan mempertahankan kemampuan generalisasi yang baik. Efektivitasnya pada data berdimensi tinggi telah terbukti dalam berbagai studi klasifikasi teks [14]. Model LinearSVC dilatih menggunakan parameter sebagai berikut:

1. $C=1.0$: Parameter regularisasi yang mengontrol *trade-off* antara *margin* klasifikasi yang besar dan minimisasi kesalahan klasifikasi. Nilai C yang lebih tinggi akan menyebabkan model berusaha mengklasifikasikan semua titik data dengan benar, yang bisa mengakibatkan *overfitting*, sementara nilai C yang lebih rendah cenderung menghasilkan *margin* yang lebih lebar tetapi mungkin salah mengklasifikasikan beberapa titik data.
2. $\text{max_iter}=1000$: Jumlah iterasi maksimum yang diperbolehkan untuk konvergensi solver.

Model dilatih menggunakan data latih yang telah diwakili dalam format TF-IDF ($X_{\text{train_vec}}$) dan label yang sesuai (y_{train}).

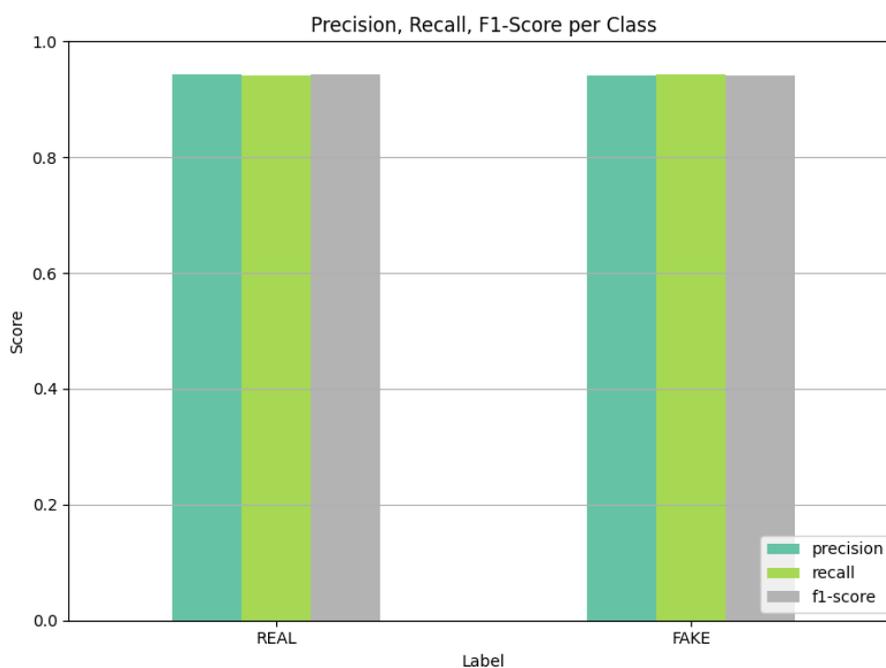
F. Evaluasi Model

Evaluasi model merupakan tahap krusial untuk mengukur seberapa baik model yang telah dilatih dapat melakukan prediksi pada data yang belum pernah dilihat sebelumnya. Metrik evaluasi yang digunakan untuk menilai kinerja model klasifikasi ini adalah sebagai berikut:

1. Akurasi (Accuracy): Mengukur proporsi total prediksi yang benar dari seluruh prediksi. Akurasi dihitung sebagai $(TP+TN)/(TP+TN+FP+FN)$, di mana TP adalah *True Positive*, TN adalah *True Negative*, FP adalah *False Positive*, dan FN adalah *False Negative*.
2. Presisi (Precision): Mengukur seberapa banyak dari kelas yang diprediksi positif benar-benar positif. Ini penting untuk menghindari *false positive* (misalnya, mengklasifikasikan berita fakta sebagai hoaks). Presisi dihitung sebagai $TP/(TP+FP)$.
3. Recall: Mengukur seberapa banyak dari kelas positif yang sebenarnya berhasil diprediksi sebagai positif. Ini penting untuk memastikan bahwa model dapat mendeteksi sebagian besar kasus hoaks yang sebenarnya. Recall dihitung sebagai $TP/(TP+FN)$.
4. F1-score: Merupakan rata-rata harmonis dari presisi dan recall. F1-score sangat berguna ketika ada ketidakseimbangan kelas atau ketika kedua metrik (presisi dan recall) sama-sama penting.
5. Matriks Konfusi (Confusion Matrix): Sebuah tabel yang menyajikan kinerja algoritma klasifikasi secara visual. Matriks ini menampilkan jumlah *True Positive* (berita hoaks diprediksi hoaks), *True Negative* (berita fakta diprediksi fakta), *False Positive* (berita fakta diprediksi hoaks), dan *False Negative* (berita hoaks diprediksi fakta) [15]. Analisis matriks konfusi memberikan gambaran mendalam tentang jenis kesalahan yang dibuat oleh model.

Hasil evaluasi ini disajikan dalam bentuk nilai akurasi, *classification report* yang mencakup presisi, recall, dan

F1-score untuk setiap kelas (REAL dan FAKE), serta visualisasi matriks konfusi pada Gambar 3.



Gambar. 3. Precision, Recall, F1-Score Per Class

III. HASIL DAN PEMBAHASAN

Bagian ini menyajikan hasil dari implementasi model klasifikasi berita hoaks politik menggunakan Support Vector Machine (LinearSVC) dengan fitur TF-IDF, diikuti dengan interpretasi dan pembahasan mendalam mengenai kinerja model.

A. Hasil Eksperimen

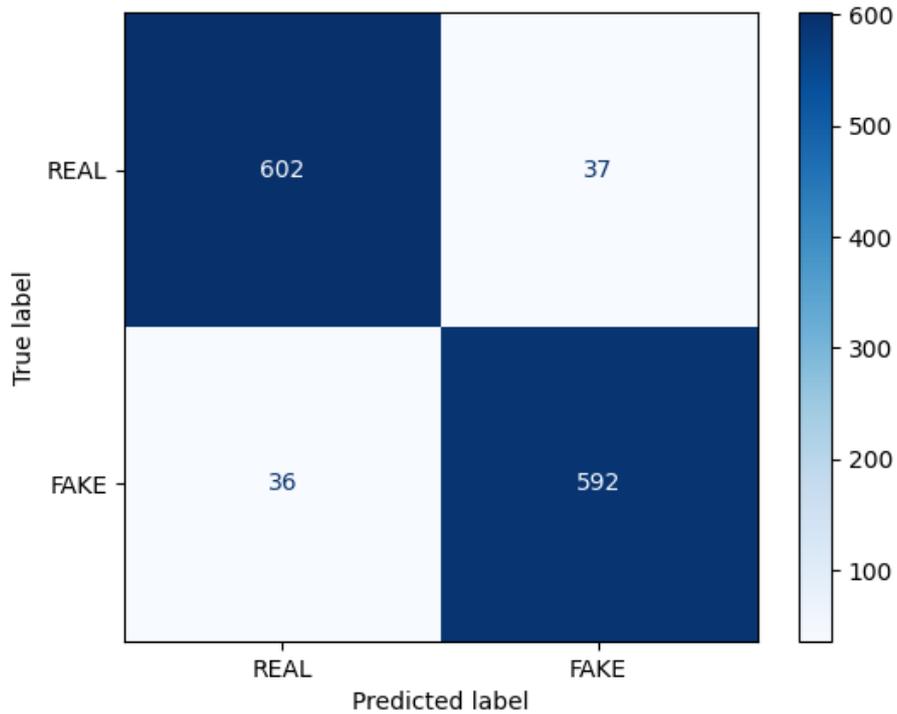
Setelah melalui tahapan pra-pemrosesan dan ekstraksi fitur, model LinearSVC dilatih menggunakan 80% data dari dataset dan dievaluasi pada 20% data sisanya. Evaluasi kinerja model dilakukan berdasarkan metrik akurasi, presisi, recall, dan F1-score, serta analisis *confusion matrix*.

Model yang telah dilatih berhasil mencapai akurasi sebesar 94,24% pada data uji. Angka ini menunjukkan bahwa model secara keseluruhan mampu mengklasifikasikan sebagian besar berita fakta dan hoaks dengan benar. Untuk memberikan gambaran yang lebih rinci tentang kinerja model pada masing-masing kelas (REAL dan FAKE), ditampilkan *classification report* pada Tabel 1.

TABEL I
 CLASSIFICATION REPORT MODEL LINEARSVC

	precision	recall	F1-score	support
REAL	0.94	0.94	0.94	693
FAKE	0.94	0.94	0.94	628
<i>Accuracy</i>	0.94	0.94	0.94	1267
<i>Macro avg</i>	0.94	0.94	0.94	1267
<i>Weighted avg</i>	0.94	0.94	0.94	1267

Selain metrik-metrik tersebut, *confusion matrix* juga dihasilkan untuk memvisualisasikan jumlah *True Positive*, *True Negative*, *False Positive*, dan *False Negative*. *Confusion matrix* disajikan pada Gambar 4.

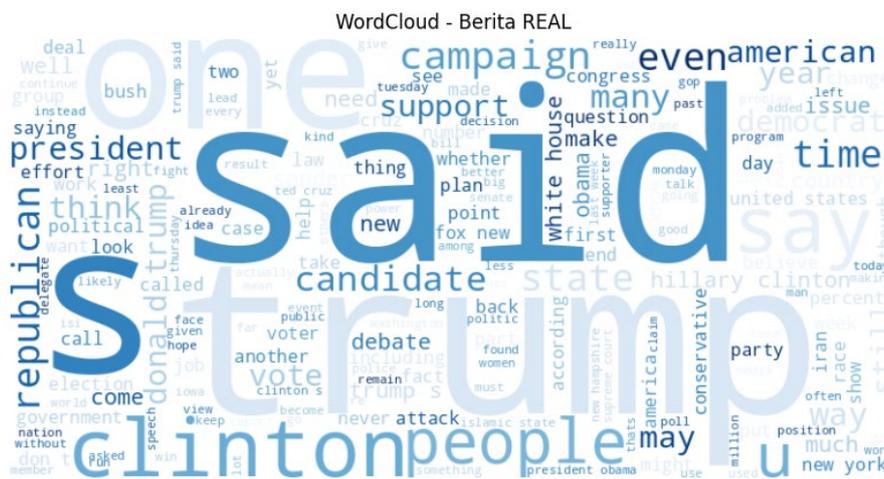


Gambar. 4. Confusion Matrix Model LinearSVC

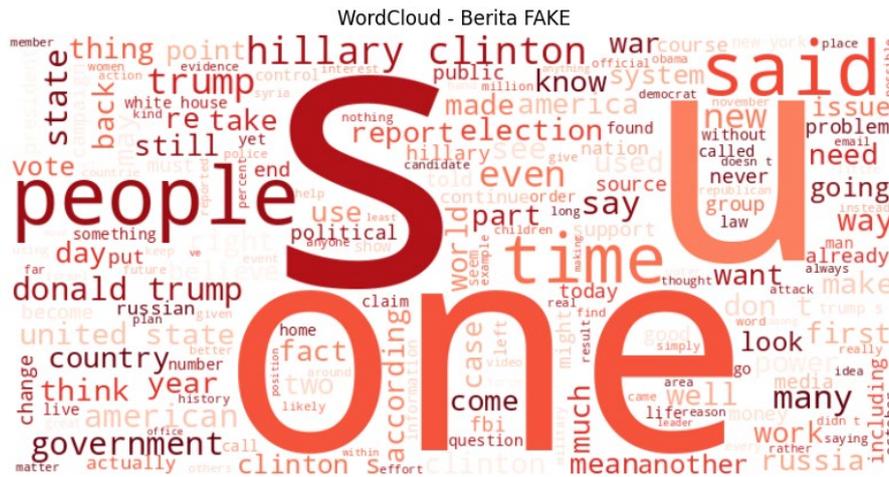
Dari *confusion matrix* tersebut, dapat diinterpretasikan bahwa:

1. True Positive (TP): [Nilai dari confusion matrix, kanan bawah jika FAKE=1] berita hoaks berhasil diidentifikasi sebagai hoaks.
2. True Negative (TN): [Nilai dari confusion matrix, kiri atas jika REAL=0] berita fakta berhasil diidentifikasi sebagai fakta.
3. False Positive (FP): [Nilai dari confusion matrix, kanan atas] berita fakta salah diklasifikasikan sebagai hoaks.
4. False Negative (FN): [Nilai dari confusion matrix, kiri bawah] berita hoaks salah diklasifikasikan sebagai fakta.

Untuk memahami lebih lanjut karakteristik fitur yang membedakan berita fakta dan hoaks, visualisasi *word cloud* dibuat dari masing-masing kategori berita (REAL dan FAKE) setelah proses pra-pemrosesan. Gambar 5 menampilkan *word cloud* untuk berita fakta, sementara Gambar 6 menampilkan *word cloud* untuk berita hoaks.



Gambar. 5. Word Cloud Berita Politik Fakta (REAL)



Gambar. 6. Word Cloud Berita Politik Hoaks (FAKE)

B. Pembahasan

Hasil eksperimen menunjukkan bahwa model klasifikasi berita politik menggunakan Support Vector Machine (LinearSVC) dengan ekstraksi fitur TF-IDF memiliki kinerja yang sangat baik. Akurasi 94,24% yang dicapai mengindikasikan kapabilitas model dalam membedakan berita fakta dan hoaks secara efektif. Angka ini sejalan dengan hasil-hasil penelitian sebelumnya yang juga menunjukkan efektivitas SVM dalam tugas klasifikasi teks dan deteksi hoaks, termasuk dalam konteks deteksi hoaks berita *online* [12], [15-17]. Demikian pula, studi [18] yang mengimplementasikan LinearSVC untuk klasifikasi teks pada dataset *20Newsgroups* juga menunjukkan performa yang kuat, mendukung pilihan model ini. Secara lebih luas, efektivitas SVM dalam klasifikasi teks telah diulas dalam berbagai survei, menunjukkan kemampuannya dalam berbagai aplikasi [11].

Analisis lebih lanjut dari *classification report* (Tabel 1) menunjukkan bahwa model mencapai nilai presisi, recall, dan F1-score sebesar 0,94 untuk kedua kelas, yaitu "REAL" dan "FAKE". Keseimbangan nilai metrik ini sangat positif, mengindikasikan bahwa model tidak hanya akurat secara keseluruhan, tetapi juga memiliki kinerja yang seimbang dalam mengidentifikasi baik berita fakta (tingkat *false positive* yang rendah) maupun berita hoaks (tingkat *false negative* yang rendah) [15]. Presisi yang tinggi untuk kelas "FAKE" berarti model jarang salah melabeli berita fakta sebagai hoaks, sedangkan *recall* yang tinggi untuk kelas "FAKE" berarti model berhasil mendeteksi sebagian besar berita hoaks yang ada.

Performa tinggi ini dapat diatribusikan pada beberapa faktor kunci:

1. Kualitas Pra-pemrosesan Data: Langkah-langkah pembersihan teks seperti *case folding*, penghapusan angka, tanda baca, spasi berlebih, dan terutama penghapusan *stopword* sangat efektif dalam mengurangi *noise* dan membuat teks lebih relevan untuk analisis.
2. Efektivitas TF-IDF: Penggunaan TF-IDF sebagai metode ekstraksi fitur terbukti mampu merepresentasikan bobot kata secara relevan dalam korpus, termasuk menangkap pentingnya *n-gram* (bigram) yang mungkin memberikan konteks lebih dalam memahami frasa kunci [13]. Parameter *max_df*, *min_df*, dan *max_features* pada *TfidfVectorizer* juga berperan dalam mengoptimalkan representasi fitur, menghindari kata-kata yang terlalu umum atau terlalu jarang, serta membatasi dimensi fitur agar tidak *overfitting*.
3. Kapabilitas LinearSVC: LinearSVC dikenal karena efisiensinya dan performa yang kuat pada dataset berdimensi tinggi, yang merupakan karakteristik umum dari data teks yang telah di-vektorisasi [14]. Kemampuannya dalam menemukan *hyperplane* pemisah yang optimal telah terbukti efektif dalam membedakan pola linguistik antara berita fakta dan hoaks.

Analisis *word cloud* (Gambar 5 dan Gambar 6) memberikan wawasan kualitatif mengenai perbedaan karakteristik leksikal antara berita fakta dan hoaks, yang juga mendukung kemampuan model dalam melakukan klasifikasi. Pada *word cloud* berita fakta (Gambar 5), terlihat dominasi kata-kata yang cenderung objektif, spesifik, dan seringkali merujuk pada entitas atau peristiwa faktual, seperti nama negara ("United States", "America"), pejabat politik ("President", "Trump"), dan institusi ("House", "Washington"). Pola linguistik ini mencerminkan karakteristik penulisan berita yang berorientasi pada penyampaian informasi faktual dan terverifikasi.

Sebaliknya, pada *word cloud* berita hoaks (Gambar 6), kata-kata yang muncul lebih besar cenderung memiliki konotasi yang lebih emosional, provokatif, atau sensasional. Kata-kata seperti "breaking", "news", "report", "hillary", dan frasa yang mengindikasikan klaim dramatis atau kontroversial lebih menonjol. Perbedaan ini mengindikasikan bahwa berita hoaks seringkali menggunakan gaya bahasa yang dirancang untuk menarik perhatian, memicu emosi, dan menyebarkan misinformasi, berbeda dengan gaya penulisan berita fakta yang lebih lugas. Pola linguistik yang teridentifikasi dari *word cloud* ini menjadi salah satu faktor kunci yang dieksplotasi

oleh fitur TF-IDF dan algoritma LinearSVC untuk membedakan kedua kategori berita.

Meskipun model menunjukkan kinerja yang sangat baik, perlu diingat bahwa dataset yang digunakan berbahasa Inggris. Penerapan model ini pada berita berbahasa lain mungkin memerlukan penyesuaian pada tahap pra-pemrosesan (misalnya, *stopword list* yang berbeda) dan pelatihan ulang model.

Untuk meningkatkan akurasi klasifikasi berita politik, terutama dalam menghadapi kompleksitas dan evolusi gaya penulisan hoaks, integrasi metode *deep learning* merupakan arah pengembangan yang sangat menjanjikan. Beberapa potensi pengembangan lebih lanjut meliputi:

1. Penerapan Model *Deep Learning* Murni: Mengimplementasikan algoritma *deep learning* seperti Convolutional Neural Networks (CNN) atau Long Short-Term Memory (LSTM) secara mandiri. Model ini menunjukkan kemampuan luar biasa dalam menangani data sekuensial seperti teks, menangkap pola-pola semantik yang lebih kompleks dan hubungan kontekstual yang mungkin tidak sepenuhnya terwakili oleh TF-IDF atau model linear [19].
2. Pendekatan *Hybrid*: Menggabungkan kekuatan ekstraksi fitur *deep learning* dengan efisiensi SVM. Contohnya, penelitian [13] menunjukkan bahwa kombinasi TF-IDF dengan CNN-LSTM dapat meningkatkan kinerja klasifikasi teks. Pendekatan [20] yang mengimplementasikan LSTM-SVM untuk deteksi hoaks COVID-19 juga terbukti efektif. Pendekatan *hybrid* ini dapat memanfaatkan keunggulan representasi fitur dari *deep learning* dan kekuatan klasifikasi SVM.
3. Eksplorasi Model Bahasa Lanjutan (LLM/Transformer): Memanfaatkan kemajuan terbaru dalam *deep learning* dan *Large Language Models* (LLM) atau arsitektur *Transformer* yang telah membuka peluang baru untuk pemahaman bahasa alami yang lebih mendalam [21]. Model-model ini mampu memahami nuansa bahasa, sarkasme, atau manipulasi narasi yang lebih canggih, yang sering ditemukan dalam berita hoaks.
4. Validasi Lintas Bahasa: Memperluas validasi model pada dataset berbahasa lain, mengingat penelitian ini menggunakan dataset berbahasa Inggris. Hal ini memerlukan penyesuaian pada tahap pra-pemrosesan (misalnya, *stopword list* yang berbeda) dan pelatihan ulang model. Pengembangan ini dapat memungkinkan model untuk memahami nuansa bahasa, sarkasme, atau manipulasi narasi yang lebih canggih, yang sering ditemukan dalam berita hoaks.

Penelitian ini memberikan kontribusi praktis dalam mitigasi penyebaran disinformasi politik melalui teknologi kecerdasan buatan, menawarkan solusi yang potensial untuk sistem deteksi hoaks otomatis.

IV. KESIMPULAN

Penelitian ini telah berhasil mengimplementasikan sistem klasifikasi otomatis untuk membedakan berita politik fakta dan hoaks menggunakan metode Support Vector Machine (SVM), khususnya algoritma LinearSVC, dengan pendekatan ekstraksi fitur Term Frequency-Inverse Document Frequency (TF-IDF). Model yang dikembangkan menunjukkan performa yang sangat baik dalam mengidentifikasi disinformasi di ranah politik.

Berdasarkan hasil evaluasi pada dataset berita politik berbahasa Inggris, model mencapai akurasi sebesar 94,24%. Selain itu, model menunjukkan kinerja klasifikasi yang seimbang untuk kedua kelas (berita fakta dan hoaks), dengan nilai presisi, recall, dan F1-score masing-masing sebesar 0,94. Performa tinggi ini didukung oleh pra-pemrosesan data yang efektif dan kapabilitas TF-IDF dalam merepresentasikan informasi tekstual, serta kekuatan LinearSVC dalam menangani data berdimensi tinggi.

Temuan ini menegaskan bahwa kombinasi TF-IDF dan LinearSVC sangat efektif sebagai pendekatan untuk deteksi berita hoaks berbasis teks. Sistem yang dibangun berpotensi besar untuk diterapkan dalam upaya mitigasi penyebaran informasi palsu di era digital, khususnya dalam konteks berita politik, sehingga dapat mendukung terciptanya ruang informasi yang lebih kredibel dan stabil.

DAFTAR PUSTAKA

- [1] Agustina, N., Adrian, & Hermawati, M. (2021). Implementasi Algoritma Naïve Bayes Classifier untuk Mendeteksi Berita Palsu pada Sosial Media. *Faktor Exacta*, 14(4), 206–213.
- [2] Adrian, R., Musaddam, Ikhsan, M., & Pahlevi, M. R. B. (2023). Detection of Hoax News Using TF-IDF Vectorizer and Multinomial Naïve Bayes and Passive Aggressive. *Media Applied Science and Engineering*, 1(2), 54–61.
- [3] Alpa, F., Gustiana, M. H., Rosmala, D., & Fitrianti, N. F. (2022). Akurasi dan Presisi Pengklasifikasian Abstrak Paper Informatika Menggunakan TF-IDF dan Multiclass Support Vector Machine (SVM). *Prosiding Diseminasi FTI Genap 2021/2022*, 1–12.
- [4] Febriyanti, N. E., Hariyadi, M. A., & Crysdian, C. (2023). Hoax Detection News Using Naïve Bayes and Support Vector Machine Algorithm. *International Journal of Advances in Data and Information Systems*, 4(2), 191–200.
- [5] Fernandes, R. P., & Shita, R. T. (2024). Penerapan Metode SVM dan Random Forest untuk Mendeteksi Berita Hoaks pada PT. Global Arrow. *Jurnal TICOM: Technology of Information and Communication*, 12(3), 102–107.
- [6] Manthovani, R. (2023). DAMPAK BERITA HOAX TERHADAP KEAMANAN NEGARA DALAM PERSPEKTIF CYBERLAW BELA NEGARA. *Jurnal Hukum dan Kesejahteraan Universitas Al Azhar Indonesia*, 8(2).

- [7] Prabowo, R. D., Widaningrum, I., & Karaman, J. (2025). Sistem Deteksi Berita Hoax Pemilu 2024 Indonesia Menggunakan Algoritma KNN Dan SVM. *JIKO (JURNAL INFORMATIKA DAN KOMPUTER)*, 9(1), 93–111.
- [8] Putra, F., & Patra, H. (2023). Analisis Hoax pada Pemilu: Tinjauan dari Perspektif Pendidikan Politik. *Naradidik: Journal of Education & Pedagogy*, 2(1), 95–102.
- [9] Ramadhan, M. F. A., Setiawan, I. R., & Asriyanik. (2024). KLASIFIKASI HOAX DAN FAKTA MENGGUNAKAN ALGORITMA SHALLOW NEURAL NETWORK PADA BERITA POLITIK PEMILIHAN PRESIDEN INDONESIA 2024. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 8(4).
- [10] Saputra, V. A., & Armomo, S. A. (2024). Penerapan Metode Machine Learning dalam Mengidentifikasi Berita Hoaks. *CBIS JOURNAL*, 12(1), 112–119.
- [11] Gasparetto, A., Marcuzzo, M., Zangari, A., & Albarelli, A. (2022). A Survey on Text Classification Algorithms: From Text to Predictions. *Information*, 13(2), 83.
- [12] Imran, B., Karim, M. N., & Ningsih, N. I. (2024). KLASIFIKASI BERITA HOAX TERKAIT PEMILIHAN UMUM PRESIDEN REPUBLIK INDONESIA TAHUN 2024 MENGGUNAKAN NAÏVE BAYES DAN SVM. *DINAMIKA REKAYASA*, 20(1), 1–9.
- [13] Zhou, H. (2022). Research of Text Classification Based on TF-IDF and CNN-LSTM. *Journal of Physics: Conference Series*, 2171, 012021.
- [14] Katira, D. (2020). Text -Classification of 20Newsgroups Data Set using Linear SVC Model. *International Journal of Advanced Research in Arts, Science, Engineering & Management (IJARASEM)*, 7(5).
- [15] Solikhah, M. (2025). THE EFFECT OF MACHINE LEARNING ALGORITHMS ON HOAX DETECTION ON SOCIAL MEDIA: IMPLICATIONS FOR NATIONAL INFORMATION SECURITY. *Journal of Artificial Intelligence Research*, 1(1), 11–20.
- [16] Elisabeth, G., Bilah, R. S., Ardini, S. N., Agustina, N., & Rismayadi, D. A. (2023). KLASIFIKASI BERITA PALSU KENAIKAN HARGA BAHAN BAKAR MINYAK (BBM) MENGGUNAKAN ALGORITMA SUPPORT VECTOR MACHINE (SVM). *NARATIF: Jurnal Ilmiah Nasional Riset Aplikasi dan Teknik Informatika*, 5(2).
- [17] Fitriani Nur Anisa, D., Iqbal, M., & Mukhlash, I. (2022). Deteksi Berita Online Hoax Covid-19 Di Indonesia Menggunakan Metode Hybrid Long Short-Term Memory dan Support Vector Machine. *JURNAL SAINS DAN SENI ITS*, 11(3).
- [18] Katira, D. (2020). Text -Classification of 20Newsgroups Data Set using Linear SVC Model. *International Journal of Advanced Research in Arts, Science, Engineering & Management (IJARASEM)*, 7(5).
- [19] Otter, D. W., Medina, J. R., & Kalita, J. K. (2020). A survey of the usages of deep learning for natural language processing. *IEEE Transactions on Neural Networks and Learning Systems*, 32(2), 604-624.
- [20] Nur Anisa, D. F., Iqbal, M., & Mukhlash, I. (2022). Deteksi Berita Online Hoax Covid-19 Di Indonesia Menggunakan Metode Hybrid Long Short-Term Memory dan Support Vector Machine. *JURNAL SAINS DAN SENI ITS*, 11(3).
- [21] Minar, M. R., & Naher, J. (2022). Recent advances in deep learning: An overview. *Journal of Physics: Conference Series*, 2171(1), 012015.