p-ISSN : 2301-8402, e-ISSN : 2685-368X, available at : https://ejournal.unsrat.ac.id/index.php/elekdankom

Comparison of The Performance of Fasttext and Word2Vec Methods in Detecting Fake News

Perbandingan Kinerja Metode FastText dan Word2Vec dalam Mendeteksi Berita Palsu

Try Setiawan Iksan, Agustinus Jacobus, Fransisca J. Pontoh
Dept. of Electrical Engineering, Sam Ratulangi University Manado, Kampus Bahu St., 95115, Indonesia
e-mails: trysetiawanik@gmail.com, a.jacobus@unsrat.ac.id, fransisca@unsrat.ac.id
Received: 27 Juni 2025; revised: 23 Januari 2026; accepted: 2 February 2026

Abstract* — *The spread of fake news (hoaxes) on social media has a significant negative impact on society, such as a decline in public trust and increased uncertainty about information. This study aims to develop and compare accurate and reliable Indonesian-language fake news detection systems, with the hope of improving media literacy among the public. The methods used include collecting several datasets of fake and authentic news, data preprocessing (cleaning, tokenisation, lemmatisation, stopword removal), and applying two word embedding algorithms, FastText and Word2Vec, with two architectures (CBOW and Skipgram). The classification model used is Bi-LSTM, and evaluation is conducted using accuracy, precision, recall, and F1-score metrics. The results show that both algorithms can produce high-accuracy fake news detection models on large datasets (FastText >85%, Word2Vec >87%), but performance decreases on small datasets due to overfitting. This study provides theoretical and practical contributions to the evaluation of word embedding algorithm performance for detecting Indonesian-language fake news based on NLP. In conclusion, the comparison results show that the evaluated word embedding approach is effective in identifying Indonesian-language fake news and can serve as a reference for algorithm selection in the development of future fake news detection technology.

***Key words* — Bi-LSTM; FastText; hoax; NLP; Word2Vec**

Abstrak* — Penyebaran berita palsu (hoaks) di media sosial menimbulkan dampak negatif yang signifikan bagi masyarakat, seperti menurunnya kepercayaan publik dan meningkatnya ketidakpastian informasi. Penelitian ini bertujuan untuk mengembangkan dan membandingkan sistem deteksi berita palsu berbahasa Indonesia yang akurat dan andal, dengan harapan dapat meningkatkan literasi media masyarakat. Metode yang digunakan meliputi pengumpulan beberapa dataset berita palsu dan asli, praproses data (cleaning, tokenisasi, lemmatisasi, penghapusan stopwords), serta penerapan dua algoritma word embedding FastText dan Word2Vec dengan dua arsitektur (CBOW dan Skipgram). Model klasifikasi yang digunakan adalah Bi-LSTM, dan evaluasi dilakukan menggunakan metrik akurasi, precision, recall, dan F1-score. Hasil penelitian menunjukkan bahwa kedua algoritma mampu menghasilkan model deteksi berita palsu dengan akurasi tinggi pada dataset besar (FastText >85%, Word2Vec >87%), namun performa menurun pada dataset kecil akibat overfitting. Penelitian ini memberikan kontribusi teoretis dan praktis dalam evaluasi performa algoritma word embedding untuk deteksi berita palsu berbahasa Indonesia berbasis NLP. Kesimpulannya, hasil perbandingan menunjukkan bahwa pendekatan word embedding yang dievaluasi efektif dalam mengidentifikasi berita palsu berbahasa Indonesia dan dapat menjadi acuan pemilihan algoritma untuk pengembangan teknologi deteksi berita palsu di masa depan.

***Kata kunci* — Bi-LSTM; FastText; hoaks; NLP; Word2Vec**

I. PENDAHULUAN

Di tengah pesatnya arus informasi di era digital, berita palsu (hoaks) telah menjadi tantangan krusial dalam menjaga kredibilitas informasi, khususnya di Indonesia. Platform media sosial yang terbuka dan mudah diakses memfasilitasi penyebaran informasi secara luas, namun sekaligus membuka peluang besar bagi beredarnya informasi yang menyesatkan [1], [2]. Data dari Masyarakat Anti Fitnah Indonesia (MAFINDO) mencatat lebih dari 6.000 kasus hoaks teridentifikasi selama periode 2018-2022, menunjukkan urgensi pengembangan sistem pendeteksi hoaks yang andal dan efektif.

Pendekatan berbasis Natural Language Processing (NLP) melalui analisis teks telah menjadi metode populer dalam pendeteksian berita palsu [3], [4]. Dalam konteks ini, pemilihan teknik representasi teks menjadi elemen fundamental yang berpengaruh langsung terhadap kinerja model klasifikasi [5], [6]. Word embedding sebagai metode representasi kata telah terbukti meningkatkan performa berbagai tugas NLP [7], [8], dengan FastText dan Word2Vec sebagai dua teknik embedding yang menawarkan keunggulan unik dalam memahami makna semantik kata [9], [10].

Meskipun beberapa penelitian telah mengeksplorasi deteksi hoaks berbahasa Indonesia, studi komparatif yang secara sistematis membandingkan FastText dan Word2Vec dengan kedua arsitektur CBOW dan Skip-gram untuk bahasa Indonesia masih sangat terbatas [10], [11]. Sebagian besar penelitian juga hanya mengevaluasi model pada single dataset, sehingga robustness dan generalizability model belum tervalidasi [12], [13]. Karakteristik bahasa Indonesia yang kaya morfologi dan penggunaan bentuk kata turunan yang kompleks menimbulkan tantangan tersendiri dalam proses representasi kata [14], [15].

Penelitian ini bertujuan melakukan analisis komparatif antara FastText dan Word2Vec dalam mendeteksi berita palsu berbahasa Indonesia menggunakan model klasifikasi berbasis Bidirectional Long Short-Term Memory (Bi-LSTM). Evaluasi dilakukan pada tiga dataset berita palsu berbahasa Indonesia

dengan total lebih dari 6.000 sampel untuk memvalidasi robustness model, sekaligus memberikan kontribusi pada pengembangan teknologi NLP untuk mengatasi tantangan sosial dalam penyebaran informasi yang akurat [6].

A. Penelitian terkait

Berikut adalah beberapa penelitian sebelumnya yang mendasari dan menjadi referensi penelitian ini.

Adipradana et al. membandingkan FastText dan GloVe pada berbagai arsitektur RNN (LSTM, Bi-LSTM, GRU, Bi-GRU) untuk klasifikasi berita Indonesia ke dalam tiga kategori: fake, valid, dan satire, dengan Bi-GRU + FastText mencapai akurasi 94.3% [11]. Namun, penelitian ini tidak mengevaluasi Word2Vec dan hanya menggunakan satu dataset.

Balaji dan Bharathi membandingkan TF-IDF dan FastText untuk deteksi berita palsu berbahasa Urdu dengan berbagai classifier, menunjukkan FastText + MLP mencapai akurasi 78.7% [16]. Penelitian ini belum mengeksplorasi Word2Vec dan arsitektur deep learning seperti Bi-LSTM [17].

Çelik dan Koç melakukan klasifikasi berita Turki menggunakan TF-IDF, Word2Vec, dan FastText, dengan FastText + SVM mencapai akurasi tertinggi 95.75% [18]. Penelitian ini menunjukkan kemampuan FastText menangani OOV lebih baik, namun tidak menggunakan Bi-LSTM [12].

Dar dan Hashmy membandingkan BERT, TF-IDF, dan FastText untuk deteksi artikel palsu COVID-19 pada 18.288 artikel, dengan RoBERTa mencapai akurasi 89.66% namun memerlukan computational resources besar [Dar & Hashmy, 2023]. Gap yang teridentifikasi adalah kurangnya eksplorasi traditional embeddings yang lebih efisien [10].

Pimpalkar et al. membandingkan FastText dan Word2Vec pada Bi-LSTM untuk klasifikasi hoaks, mencapai akurasi 99% pada dataset berbahasa Inggris dengan pembagian train-test 0.85-0.15 [12]. Penelitian ini tidak dapat langsung digeneralisasi untuk bahasa Indonesia yang memiliki karakteristik morfologi berbeda [14], [15].

Ali Ramdhani et al. melakukan kategorisasi berita Indonesia menggunakan CNN mencapai akurasi 90.74%, menunjukkan pentingnya preprocessing khusus untuk bahasa Indonesia [14]. Penelitian ini tidak mengeksplorasi Bi-LSTM dan tidak ada analisis komparatif word embedding [17].

Rollo et al. menganalisa performa FastText dan Word2Vec pada berita Italia menggunakan multiple datasets (DICE: 10.395 berita, RCV2-it: 28.405 berita), menunjukkan pentingnya validasi cross-dataset [19]. Penelitian ini menggunakan classical ML tanpa deep learning dan fokus pada bahasa Italia [15].

Penelitian yang dilakukan Tulu mengevaluasi Word2Vec, GloVe, dan FastText untuk semantic similarity bahasa Turki, dengan FastText unggul dalam coverage kosakata dan minimal OOV [15]. Namun, penelitian ini hanya fokus pada intrinsic evaluation tanpa extrinsic evaluation pada fake news detection [10].

B. Berita Palsu (Hoax)

Hoax adalah informasi palsu yang sengaja disebarkan untuk menipu atau menyesatkan orang lain. Informasi ini seringkali terlihat meyakinkan, tetapi sebenarnya tidak bisa dibuktikan kebenarannya. Ciri-ciri dari berita *hoax* ialah seringkali mengandung fakta yang tidak diverifikasi atau bukti yang mendukung dengan menggunakan statistik palsu, gambar yang diedit dan klaim yang tidak terbukti.

Dampak dari *hoax* ialah ketidakpercayaan terhadap informasi, kebingungan dan kekacauan, ketegangan sosial dan politik, kerugian finansial, peningkatan kepanikan, kerugian reputasi, dan berpotensi mengancam kehidupan [1], [2].

C. Word Embedding

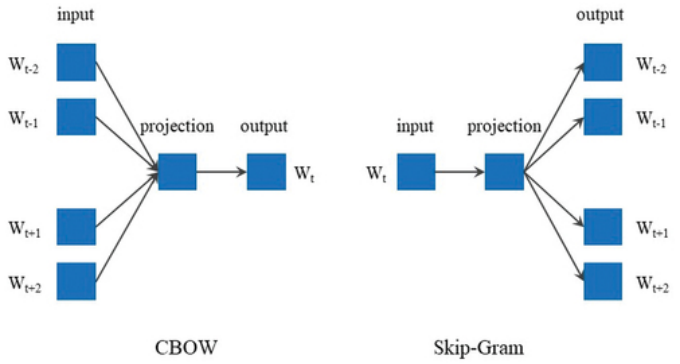
Menurut Selva Birunda & Kanniga Devi (2021) bahwa Penyematan kata (*word embedding*) merupakan representasi kata dalam bentuk vektor numerik yang menghubungkan pemahaman manusia tentang bahasa dengan pemahaman mesin. Representasi ini memungkinkan mesin untuk menangkap makna dan hubungan antar kata secara matematis. Penyematan kata dapat dianggap sebagai pemetaan kata ke dalam ruang berdimensi tinggi, di mana kata-kata dengan makna serupa ditempatkan berdekatan satu sama lain [8].

D. Continuous Bag of Words (CBOW) dan Skip-gram

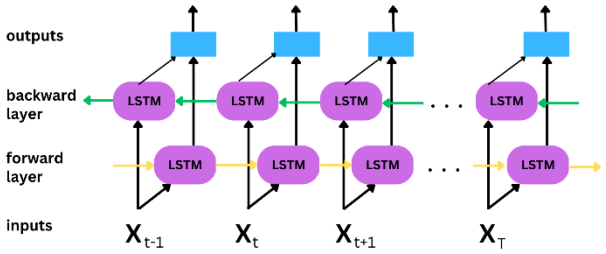
Word embedding seperti FastText dan Word2Vec dapat dilatih menggunakan dua arsitektur pembelajaran yang memiliki pendekatan berbeda, yaitu Continuous Bag of Words (CBOW) dan Skip-gram. Perbedaan fundamental antara keduanya terletak pada arah prediksi, dimana CBOW memprediksi kata target dari konteks sekitarnya, sedangkan Skip-gram memprediksi konteks dari kata target. Pemahaman karakteristik masing-masing arsitektur penting karena dapat mempengaruhi kualitas embedding yang dihasilkan, terutama untuk bahasa dengan morfologi kompleks seperti bahasa Indonesia.

Continuous Bag of Words (CBOW) memprediksi kata target berdasarkan kata-kata konteks di sekitarnya dalam window tertentu, seperti permainan tebak kata dimana model diberi petunjuk dari sisi kiri dan kanan untuk menebak kata di posisi tengah. Arsitektur ini menggunakan pendekatan bag-of-words yang mengabaikan urutan kata dan memperlakukan semua kata konteks dengan bobot sama, dengan mengambil rata-rata vektor embedding kata konteks sebagai input untuk memprediksi kata target. Keunggulan CBOW terletak pada efisiensi komputasi yang tinggi, lebih cepat dalam pelatihan terutama pada dataset besar, serta lebih efektif untuk kata-kata dengan frekuensi tinggi karena menghasilkan representasi yang stabil dengan mengurangi noise dari variasi konteks individual [9], [10].

Skip-gram bekerja dengan prinsip berlawanan, memprediksi kata-kata konteks berdasarkan satu kata target sebagai input. Setiap kata target menghasilkan multiple training examples melalui prediksi berbagai kata konteks, memberikan lebih banyak kesempatan belajar untuk kata-kata langka. Skip-gram terbukti menghasilkan representasi yang lebih akurat untuk hubungan semantik dan sintaktik, karena fokusnya pada prediksi konteks beragam memungkinkan model menangkap nuansa makna yang lebih halus. Penelitian menunjukkan Skip-gram lebih efektif untuk semantic similarity tasks dan analogy tasks dibandingkan CBOW, meskipun membutuhkan waktu



Gambar 2. Perbandingan Arsitektur CBOW dan Skip-gram [32]



Gambar 1. Arsitektur Bi-LSTM [33]

pelatihan lebih lama dan computational resources lebih besar [10], [15].

E. FastText

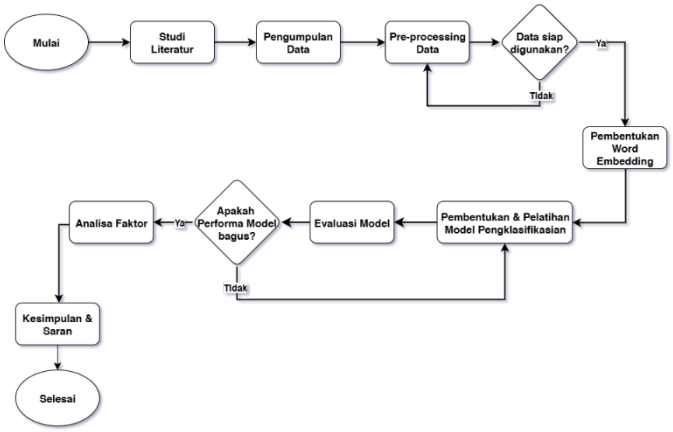
FastText dibuat oleh Facebook pada tahun 2016 dan didasarkan pada model Word2Vec [18]. Algoritma FastText digunakan untuk memilih kata dari seluruh konteks dalam teks atau konteks dari semua kata yang ada di tengah. Pengoperasiannya didasarkan pada prinsip pembelajaran. Proses pembelajaran dapat dikonseptualisasikan sebagai urutan dalam jaringan saraf yang terdiri dari dua lapisan bobot dan tiga lapisan neuron. Hal tersebut dapat dipahami sebagai pembaruan, di mana setiap kata dalam kosakata di dalam masing-masing dua lapisan luar dan lapisan tengah diasosiasikan dengan sejumlah neuron yang sama dengan dimensi bidang penyematan)[11].

F. Word2Vec

Menurut Dharma, bahwa *Word2Vec* merupakan salah satu aplikasi pembelajaran tanpa pengawasan utama (*unsupervised learning*) yang menggunakan jaringan syaraf. Model ini terdiri dari lapisan tersembunyi, yang disebut lapisan proyeksi, dan lapisan yang sepenuhnya terhubung, dilatih menggunakan keturunan gradien stokastik dengan algoritma *backpropagation*[9].

G.Bidirectional Long-short Term Memory (Bi-LSTM)

Menurut Sastrawan, bahwa Bi-LSTM adalah jenis arsitektur Recurrent Neural Network (RNN) tertentu yang terdiri dari dua unit LSTM yang diposisikan berlawanan arah (backward layer & forward layer). Tujuan dari pendekatan arsitektur ini adalah untuk meningkatkan kemampuan memori LSTM dengan memberikan informasi kontekstual dari masa lalu dan masa depan[13].



Gambar 3. Kerangka Pikir Penelitian

TABEL I
INFORMASI DATASET

	DATASET	Sampel
A	tallip_indonesian_fakenews	1960
B	INDONESIAN HOAX NEWS DETECTION DATASET	600
C	Dataset Berita Palsu Bahasa Indonesia dengan Penelusuran Fakta Berbasis LLM	6747
D	indocorpus_mix	95278

II. METODE

A. Eksperimen

Kerangka pikir merupakan logis penelitian atau tahapan yang akan dilakukan selama penelitian. Kerangka pikir dalam penelitian ini dapat dilihat pada Gambar 3.

1) Studi Literatur

Studi literatur dilakukan dengan menelusuri publikasi periode dari 2020-2024, dengan fokus pada penelitian word embedding (FastText, Word2Vec) untuk fake news detection dan arsitektur Bi-LSTM [5], [6]**.** Pencarian menggunakan keyword: "fake news detection", "word embedding", "FastText", "Word2Vec", "Bi-LSTM", "Indonesian NLP". Dari hasil studi literatur, teridentifikasi gap penelitian berupa kurangnya studi komparatif sistematis FastText vs Word2Vec dengan arsitektur CBOW dan Skip-gram pada multiple Indonesian fake news datasets.

2) Dataset

Data yang digunakan berasal dari beberapa sumber, baik berupa korpus teks dan berita berbahasa Indonesia yang tersedia secara publik. Kriteria seleksi dataset mencakup empat aspek penting yaitu dataset harus memiliki label yang jelas antara berita palsu dan berita asli, teks harus dalam bahasa Indonesia yang baik dan benar, dan tersedia secara publik untuk keperluan penelitian. Setelah proses pembersihan yang meliputi penghapusan *missing values* dan duplikasi data, diperoleh dataset valid yang siap untuk eksperimen. Informasi lebih lanjut dari data yang diperoleh ada di Tabel I.

3) Pra-pemrosesan Data

Pra-pemrosesan data merupakan salah satu langkah yang penting sebelum melakukan proses pelatihan model dan evaluasi data. Tujuan utama pra-pemrosesan data adalah untuk

mengurangi atau menghilangkan variabilitas dan efek yang tidak diinginkan, sehingga informasi yang relevan terkait dengan target dapat dimanfaatkan secara optimal untuk pemodelan yang efisien. Proses ini meliputi *feature selection*, pembersihan *missing values,* normalisasi, lemmatisasi, tokenisasi, dan penghapusan *stopwords.*

a) Feature Selection

Berdasarkan tujuan penelitian, tiga fitur dipilih: Content (object), Label (object), dan Label_id (int64), yang dianggap relevan untuk deteksi berita palsu dan meningkatkan efisiensi komputasi.

b) Missing Values

Dari tiga dataset yang digunakan (untuk pengklasifikasian) pada penelitian ini terdapat 292 baris nilai yang hilang dan keputusan yang diambil adalah menghapus nilai yang hilang tersebut.

c) Cleaning

Cleaning dilakukan dengan menggunakan regular expression (regex) untuk menghapus URL, mention, dan hashtag, mengubah seluruh teks menjadi lowercase, menghapus karakter non-alfanumerik kecuali spasi dan menghapus whitespace berlebih. Implementasi menggunakan library Python "re" dan hasil normalisasi disimpan pada kolom baru bernama 'cleaned'.

d) Lemmatisasi dan Tokenisasi

Lemmatisasi adalah proses mengonversi kata-kata ke bentuk dasarnya, sedangkan tokenisasi adalah proses memecah teks menjadi unit-unit yang lebih kecil. Kedua proses ini dilakukan menggunakan library Stanza, yang merupakan *toolkit* NLP yang kuat dan fleksibel. Hasil dari proses lemmatisasi dan tokenisasi akan di timpa di kolom *'cleaned'*.

e) Penghapusan stopwords

Stopwords merupakan kata-kata yang sering muncul dalam teks tetapi tidak memiliki makna substansial, maka dari itu penghapusan *stopwords* dilakukan karena dapat membantu mengurangi dimensi data sehingga dapat mempercepat proses analisis dan model dapat lebih mudah mengenali pola. Hasil dari *stopwords* di timpa lagi di kolom *'cleaned'*.

TABEL III
ARSITEKTUR MODEL

Layer	Parameter
Input Layer	nilai tertinggi disetiap dataset
Embedding Layer	embedding_dim = 50, vocab_size bervariasi per dataset
Bi-LSTM	16 unit, regularizer_l2 = 0.5
Dropout Layer	0.6
Batch Normalization	-
Output Layer	1 unit, activation=sigmoid
Optimizer	Adam(lr=0.0005)
Training	batch_size = 32, epochs = 100, earlystopping_patience = 1, restore_best_weights=True

f) Split Data

Pembagian data pelatihan, pengujian, dan validasi menggunakan train_test_split dari scikit-learn dengan strategi nested split, menghasilkan proporsi akhir 64% data latih, 16% data validasi, dan 20% data uji dengan stratified sampling untuk menjaga distribusi label proporsional. Pembagian ini dirasionalisasi berdasarkan fungsi kritis masing-masing subset: data uji (20%) sebagai hold-out set untuk evaluasi final yang unbiased dan mengukur generalization capability model [20], [21]**,** data validasi (16%) untuk monitoring overfitting dan hyperparameter tuning tanpa "melihat" data test [22], [23], serta data latih (64%) untuk weight update model [24]. Rasio 80:20 dengan nested validation 80:20 ini mengikuti best practice penelitian Bi-LSTM yang memberikan trade-off optimal antara data sufficiency dan evaluation reliability [12], [17].

4) Word Embedding

Penelitian ini melatih model word embedding FastText dan Word2Vec dari scratch menggunakan library Gensim pada Dataset C (indocorpus-mix) yang berisi 80.000+ artikel berbahasa Indonesia untuk memastikan vocabulary coverage yang luas dan meminimalisir Out-of-Vocabulary (OOV) words [25]. Parameter training embedding yang digunakan telah dipaparkan pada Tabel II. Kedua arsitektur dilatih untuk masing-masing teknik, menghasilkan 4 model embedding: FastText-CBOW (FT CBOW), FastText Skip-gram (FT Skip-gram), Word2Vec CBOW (W2V CBOW), Word2Vec Skip-gram (W2V Skip-gram).

5) Arsitektur Model

Arsitektur model yang digunakan adalah *Bidirectional Long-short Term Memory* yang dibangun dari awal dengan detail parameter di Tabel III.

6) Evaluasi

Evaluasi model dilakukan secara menyeluruh melalui dua pendekatan utama, yaitu Ekstrinsik dan Intrinsik, guna memperoleh pemahaman terkait kinerja dan kualitas model yang dibangun. Ekstrinsik difokuskan pada pengukuran performa model dalam tugas klasifikasi berita palsu menggunakan metrik seperti *accuracy, precision, recall, F1-score,* dan *confusion matrix* untuk menilai seberapa baik model dalam membedakan antara berita palsu dan berita *valid* berdasarkan hasil prediksi data uji. Sementara itu, Intrinsik dilakukan menilai kualitas representasi kata yang dihasilkan oleh metode *word embedding* melalui *analogy task* dan *similarity task,* sehingga dapat diketahui sejauh mana *embedding* mampu menangkap hubungan semantik dan analogi antar kata dalam korpus bahasa Indonesia.

TABEL II
PARAMETER TRAINING WORD EMBEDDING

Parameter	Nilai	Keterangan
vector_size	50	Dimensi vektor embedding
window	5	Konteks kata sebelum dan sesudah kata target
min_count	2	Kata frekuensi minimal
epochs	100	Iterasi training
workers	4	Jumlah thread yang digunakan
sg	0, 1	Dua arsitektur embedding, sg=0 untuk CBOW dan sg=1 untuk Skip-gram

III. HASIL DAN PEMBAHASAN

A. Karakteristik Dataset setelah Pra-pemrosesan

Setelah seluruh tahapan pra-pemrosesan selesai, karakteristik final dataset yang digunakan untuk eksperimen adalah sebagai berikut. Dataset A terdiri dari 1.960 artikel dengan panjang sekuens maksimal 194 token, rata-rata 41 token, dan median 32 token, dipilih karena merepresentasikan berita pendek yang umum di media sosial dengan keseimbangan label yang baik. Dataset B berisi 600 artikel dengan panjang maksimal 893 token, rata-rata 224 token, dan median 198 token, digunakan untuk menguji robustness model pada dataset kecil dengan artikel yang lebih panjang. Dataset C terdiri dari 6.747 artikel dengan panjang maksimal 1.243 token, dipilih karena ukurannya yang besar dan variasi panjang artikel yang beragam. Dataset D dengan 95.278 artikel digunakan khusus untuk training word embedding FastText dan Word2Vec, dipilih karena kekayaan vocabulary.

B. Hasil Evaluasi per Dataset

1) Dataset A

Evaluasi performa model pada Dataset A dengan 1960 sampel menunjukkan hasil yang kompetitif untuk keempat varian embedding. Word2Vec CBOW mencapai akurasi 87% dengan precision 0.87, recall 0.89, dan F1-score 0.88 pada kelas fake, menunjukkan keseimbangan baik antara kemampuan deteksi dan akurasi prediksi [21], [26]. Word2Vec Skip-gram menghasilkan performa serupa dengan akurasi 87%, namun dengan precision lebih tinggi (0.88) dan recall lebih rendah (0.83), mengindikasikan pendekatan yang lebih konservatif. FastText CBOW mencapai akurasi 85% dengan precision 0.82 dan recall 0.90, menunjukkan kecenderungan lebih agresif dalam deteksi, sedangkan FastText Skip-gram mencapai 84% dengan performa yang lebih seimbang. Perbedaan ini mengkonfirmasi bahwa CBOW lebih baik menangkap pola umum sedangkan Skip-gram lebih sensitif terhadap nuansa semantik [9], [10].

2) Dataset B

Dataset B ialah sampel terkecil hanya 600 sampel menunjukkan penurunan performa signifikan dengan kisaran akurasi 60-68%. FastText Skip-gram menghasilkan performa terbaik dengan akurasi 68%, precision 0.73, dan recall 0.77, diikuti Word2Vec Skip-gram 66%, FastText CBOW 64%, dan Word2Vec CBOW 63%. Penurunan ini konsisten dengan fenomena overfitting dan grokking pada model deep learning dengan data terbatas [24], [27], [28]. Kompleksitas model Bi-LSTM dengan 16 units menjadi terlalu powerful untuk dataset kecil, menyebabkan model menghafal pola training tanpa mampu generalisasi pada data baru.

3) Dataset C

Dataset C dengan sampel terbanyak yaitu 6747 sampel menunjukkan peningkatan performa signifikan dengan akurasi melewati 90% untuk semua varian. Word2Vec Skip-gram dan FastText Skip-gram keduanya mencapai akurasi tertinggi 91% dengan precision 0.93, diikuti Word2Vec CBOW dan FastText CBOW dengan 90-91%. Peningkatan ini mengkonfirmasi bahwa ekspansi kosakata melalui pemodelan subword pada FastText dan representasi konteks pada Word2Vec mampu menangkap pola semantik kompleks ketika data memadai [9], [29].

TABEL IV
RINGKASAN PERFORMA MODEL PADA SEMUA DATASET

Model	Dataset	Accuracy	Precision (fake)	Recall (fake)	F1-Score (fake)
W2V CBOW	A	0.87	0.87	0.89	0.88
W2V Skip-gram	A	0.87	0.88	0.83	0.85
FT CBOW	A	0.85	0.82	0.90	0.86
FT Skip-gram	A	0.84	0.85	0.84	0.84
W2V CBOW	B	0.63	0.67	0.78	0.72
W2V Skip-gram	B	0.66	0.71	0.74	0.72
FT CBOW	B	0.64	0.67	0.84	0.74
FT Skip-gram	B	0.68	0.73	0.77	0.75
W2V CBOW	C	0.91	0.90	0.84	0.89
W2V Skip-gram	C	0.91	0.93	0.85	0.89
FT CBOW	C	0.90	0.90	0.89	0.89
FT Skip-gram	C	0.91	0.93	0.86	0.89

4) Analogy Task dan Similarity Task

Evaluasi intrinsik melalui analogy task dan similarity task mengkonfirmasi keunggulan Skip-gram dalam menangkap relasi semantik. Pada analogy task, Word2Vec Skip-gram menghasilkan skor kemiripan 0.6806 dibanding CBOW 0.5959, sedangkan pada similarity task dengan kata 'kendaraan', Skip-gram mencapai skor 0.93 dibanding CBOW 0.78, menunjukkan superioritas Skip-gram dalam similarity tasks [7], [15]. Perbandingan FastText dan Word2Vec menunjukkan bahwa FastText unggul dalam menangani morfologi kompleks bahasa Indonesia melalui pendekatan n-gram karakter, menghasilkan performa yang lebih stabil across datasets, sementara Word2Vec mencapai performa puncak lebih tinggi pada dataset besar, mengindikasikan bahwa representasi berbasis kata utuh lebih powerful ketika vocabulary komprehensif [9], [19].

C. Analisis Komparatif dan Faktor yang Mempengaruhi

Hasil evaluasi menunjukkan pola performa yang konsisten berdasarkan karakteristik dataset dan arsitektur embedding. Word2Vec CBOW dan Skip-gram mencapai akurasi stabil 87% pada Dataset A, sementara FastText bervariasi pada 84-85%. Pada Dataset C yang lebih besar, semua varian melampaui 90%, menunjukkan bahwa baik pemodelan subword FastText maupun representasi konteks Word2Vec mampu menangkap pola semantik yang kaya ketika data memadai [9], [10]. Sebaliknya, Dataset B mengalami penurunan signifikan ke 60-68%, fenomena yang konsisten dengan literatur tentang overfitting dan grokking pada model deep learning dengan data terbatas [24], [27], [28].

Perbandingan CBOW dan Skip-gram mengungkapkan trade-off menarik dalam pembelajaran representasi kata. Skip-gram menunjukkan precision lebih tinggi namun recall lebih rendah, mengindikasikan pendekatan yang konservatif dan akurat dalam prediksi. CBOW sebaliknya lebih agresif dengan recall tinggi namun menghasilkan lebih banyak false positive. Perbedaan ini dijelaskan oleh mekanisme pembelajaran konteks yang berbeda, dimana CBOW memprediksi kata target dari konteks sehingga menangkap pola umum, sedangkan Skip-gram memprediksi konteks dari kata target sehingga lebih fokus pada nuansa semantik spesifik [10], [15].

Analisis mendalam mengungkapkan beberapa faktor kritis yang mempengaruhi performa model. Ukuran dataset terbukti menjadi faktor dominan, dimana dataset lebih dari 6.000 sampel menghasilkan performa 90%+, sedangkan dataset kurang dari 1.000 sampel mengalami penurunan drastis hingga 60-68%, mengindikasikan kebutuhan minimal 2.000-3.000 sampel untuk mencapai performa acceptable (>80%) pada bahasa Indonesia dengan vocabulary kaya dan variasi morfologis tinggi [24]. Panjang sekuens teks juga mempengaruhi kemampuan model menangkap konteks, dimana Dataset B dengan rata-rata 224 token menghadapi tantangan lebih besar dibanding Dataset A dengan 41 token karena truncation pada sequence length maksimal 200 token yang menyebabkan hilangnya informasi kontekstual [30], [31]**.** Kualitas preprocessing, hyperparameter embedding (vector dimension 50, window size 5), dan hyperparameter model (dropout 0.6, learning rate 0.0005) yang konservatif terbukti efektif untuk dataset besar namun mungkin terlalu restrictive untuk dataset kecil, mengindikasikan perlunya adaptive tuning berdasarkan karakteristik dataset [10], [22], [23]**.**

IV. KESIMPULAN DAN SARAN

A. Kesimpulan

Penelitian ini berhasil melakukan evaluasi dan perbandingan terhadap performa dua algoritma pemrosesan bahasa alami yaitu FastText dan Word2Vec dengan dua arsitektur CBOW dan Skip-gram dalam mendeteksi berita palsu berbahasa Indonesia menggunakan model Bi-LSTM. Berdasarkan hasil evaluasi komprehensif pada tiga dataset dengan total lebih dari 9.000 sampel, dapat disimpulkan bahwa seluruh tahapan penelitian telah berjalan sesuai dengan tujuan yang ditetapkan. Mengenai perbandingan performa FastText dan Word2Vec dalam mengidentifikasi berita palsu berbahasa Indonesia, hasil evaluasi menunjukkan bahwa Word2Vec sedikit unggul dengan akurasi 87% dibandingkan FastText yang mencapai 85% pada Dataset A, namun FastText menunjukkan keunggulan dalam menangani kata-kata di luar kosakata melalui pendekatan n-gram karakter yang efektif untuk karakteristik morfologi bahasa Indonesia yang bervariasi. Perbandingan arsitektur menunjukkan bahwa Skip-gram menghasilkan performa superior dalam mayoritas tugas semantik dengan skor similarity task mencapai 0.93 untuk Word2Vec Skip-gram dibandingkan CBOW yang mencapai 0.78, meskipun CBOW lebih stabil dan cepat dalam pelatihan, mengkonfirmasi trade-off antara akurasi semantik dan efisiensi komputasi.

Analisis mendalam mengidentifikasi lima faktor kunci yang mempengaruhi performa model deteksi berita palsu. Pertama, ukuran dataset terbukti menjadi faktor dominan dimana Dataset C dengan 6.747 sampel menghasilkan akurasi lebih dari 90%, Dataset A dengan 1.960 sampel mencapai 85-87%, sedangkan Dataset B dengan 600 sampel hanya mencapai 63-68% karena fenomena grokking dan overfitting pada data terbatas. Kedua, panjang sekuens rata-rata mempengaruhi kompleksitas pembelajaran, dimana dataset dengan rata-rata sekuens pendek 41 token lebih mudah dipelajari dibanding sekuens panjang 224 token yang mengalami truncation pada sequence length maksimal 200 token sehingga kehilangan informasi kontekstual penting. Ketiga, vocabulary size yang besar dengan 43.455 kata unik memerlukan parameter embedding lebih banyak namun memberikan representasi yang kaya untuk menangkap variasi semantik bahasa Indonesia. Keempat, tahapan preprocessing meliputi cleaning, lemmatisasi menggunakan Stanza, dan penghapusan stopwords terbukti krusial dalam meningkatkan performa model dengan mengurangi noise dan meningkatkan signal-to-noise ratio. Kelima, arsitektur model dengan regularisasi L2 0.5, dropout 0.6, dan early stopping berhasil mencegah overfitting pada dataset besar namun belum optimal untuk dataset kecil yang memerlukan adaptive hyperparameter tuning. Hasil penelitian ini dapat dijadikan acuan untuk pengembangan model deteksi hoax yang lebih akurat dan efisien di masa depan, baik dengan mengoptimalkan salah satu algoritma maupun menggabungkan keduanya dalam ensemble methods, serta memperkaya literatur NLP di ranah sosial khususnya dalam konteks bahasa Indonesia dan mendorong penelitian lanjutan terkait penerapan NLP untuk isu-isu sosial lainnya.

B. Saran

Berdasarkan temuan dan keterbatasan penelitian ini, beberapa saran dapat diberikan untuk pengembangan penelitian selanjutnya. Pertama, penelitian ini sangat bergantung pada kualitas dan representativitas dataset berita palsu berbahasa Indonesia, sehingga disarankan untuk memperluas cakupan dataset baik dari segi jumlah data maupun variasi topik untuk memastikan model dapat menggeneralisasi dengan baik pada berbagai jenis berita palsu yang beredar di masyarakat. Kedua, tahapan preprocessing data dalam penelitian ini masih dapat dioptimalkan dengan mengimplementasikan preprocessing yang lebih komprehensif seperti named entity recognition, sentiment analysis, atau part-of-speech tagging untuk mendapatkan kualitas representasi teks dan performa model yang maksimal. Ketiga, model yang dikembangkan saat ini hanya efektif untuk berita berbahasa Indonesia sehingga pengembangan model multibahasa sangat penting agar sistem dapat digunakan di berbagai konteks bahasa dan budaya, serta integrasi data multimodal seperti teks, gambar, dan video dapat meningkatkan akurasi deteksi berita palsu mengingat banyak hoaks saat ini disebarkan dalam berbagai format media yang saling mendukung untuk memperkuat narasi yang menyesatkan.

Keempat, penelitian ini hanya membandingkan FastText dan Word2Vec sehingga untuk penelitian selanjutnya disarankan untuk mengeksplorasi dan mengintegrasikan algoritma lain seperti GloVe, contextualized embeddings seperti IndoBERT atau multilingual BERT, atau model berbasis neural network lainnya dengan teknik ensemble learning yang dapat menggabungkan kekuatan berbagai pendekatan untuk meningkatkan robustness dan performa keseluruhan sistem. Kelima, parameter yang digunakan dalam penelitian ini masih dapat dioptimalkan melalui fine-tuning parameter baik pada model classifier maupun word embedding secara lebih sistematis menggunakan metode seperti grid search atau bayesian optimization untuk memberikan keseimbangan yang baik antara kualitas performa dengan efisiensi komputasi, serta mencegah overfitting dan fenomena grokking terutama pada dataset berukuran kecil yang memerlukan strategi regularisasi adaptif. Keenam, implementasi explainable AI techniques seperti attention visualization atau LIME sangat penting untuk

memberikan interpretasi tentang fitur-fitur yang paling berpengaruh dalam keputusan model, sehingga dapat membangun kepercayaan pengguna dan memfasilitasi human-in-the-loop verification yang krusial untuk aplikasi sensitif seperti deteksi berita palsu. Ketujuh, berita palsu terus berkembang dan menjadi semakin canggih dalam taktik penyebarannya, oleh karena itu model deteksi perlu diperbarui secara berkala dengan continuous learning atau online learning approaches agar tetap relevan dengan taktik dan teknik terbaru yang digunakan dalam penyebaran disinformasi. Dengan memperhatikan saran-saran di atas, diharapkan penelitian dan pengembangan sistem deteksi berita palsu ke depan dapat menjadi lebih komprehensif, adaptif, dan memberikan kontribusi yang lebih besar dalam meningkatkan literasi media serta melindungi masyarakat dari dampak negatif disinformasi.

V. KUTIPAN

[1] M. Deddy Satria and others, "The Phenomenon of Fake News (Hoax) in Mass Communication: Causes, Impacts, and Solutions," *Open Access Indonesia Journal of Social Sciences*, vol. 6, no. 3, pp. 980–988, 2023.

[2] D. Susilo Wijayanto *et al.*, "SOCIALIZATION ON PREVENTING THE SPREAD OF HOAX NEWS IN ONLINE NEWS MEDIA AT GROGOL VILLAGE, WERU, SUKOHARJO," *Abdi Masya*, vol. 6, no. 1, pp. 42–47, May 2025, doi: 10.52561/ABDIMASYA.V6I1.452.

[3] M. A. B. Al-Tarawneh, O. Al-irr, K. S. Al-Maaitah, H. Kanj, and W. H. F. Aly, "Enhancing Fake News Detection with Word Embedding: A Machine Learning and Deep Learning Approach," *Computers 2024, Vol. 13, Page 239*, vol. 13, no. 9, p. 239, Sep. 2024, doi: 10.3390/COMPUTERS13090239.

[4] H. Allam, L. Makubvure, B. Gyamfi, K. N. Graham, and K. Akinwolere, "Text Classification: How Machine Learning Is Revolutionizing Text Categorization," *Information 2025, Vol. 16, Page 130*, vol. 16, no. 2, p. 130, Feb. 2025, doi: 10.3390/INFO16020130.

[5] K. Taha, P. D. Yoo, C. Yeun, D. Homouz, and A. Taha, "A comprehensive survey of text classification techniques and their research applications: Observational and experimental insights," *Comput Sci Rev*, vol. 54, p. 100664, Nov. 2024, doi: 10.1016/J.COSREV.2024.100664.

[6] L. Galke *et al.*, "Are We Really Making Much Progress in Text Classification? A Comparative Review," *ACM Comput Surv*, vol. 1, Jan. 2025, Accessed: Jul. 14, 2025. [Online]. Available: https://arxiv.org/pdf/2204.03954v6

[7] T. B. Hashimoto, D. Alvarez-Melis, and T. S. Jaakkola, "Word Embeddings as Metric Recovery in Semantic Spaces," *Trans Assoc Comput Linguist*, vol. 4, pp. 273–286, Dec. 2016, doi: 10.1162/TACL_A_00098.

[8] S. Selva Birunda and R. Kanniga Devi, "A Review on Word Embedding Techniques for Text Classification," 2021, pp. 267–281. doi: 10.1007/978-981-15-9651-3_23.

[9] E. M. Dharma, F. L. Gaol, H. Warnars, and B. Soewito, "The accuracy comparison among word2vec, glove, and fasttext towards convolution neural network (cnn) text classification," *J Theor Appl Inf Technol*, vol. 100, no. 2, p. 31, 2022.

[10] S. Khomsah, R. D. Ramadhani, and S. Wijayanto, "The Accuracy Comparison Between Word2Vec and FastText On Sentiment Analysis of Hotel Reviews," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 6, no. 3, pp. 352–358, Jun. 2022, doi: 10.29207/RESTI.V6I3.3711.

[11] R. Adipradana, B. P. Nayoga, R. Suryadi, and D. Suhartono, "Hoax analyzer for indonesian news using rnns with fasttext and glove embeddings," *Bulletin of Electrical Engineering and Informatics*, vol. 10, no. 4, pp. 2130–2136, Aug. 2021, doi: 10.11591/eei.v10i4.2956.

[12] A. Pimpalkar, M. Singh, S. Sheikh, K. Gedam, and A. Khadgi, "Fake news classification using bi-directional LSTM-recurrent neural network," *Journal of Huazhong University of Science and Technology ISSN*, vol. 1671, p. 4512, 2021.

[13] I. K. Sastrawan, I. P. A. Bayupati, and D. M. S. Arsa, "Detection of fake news using deep learning CNN–RNN based methods," *ICT Express*, vol. 8, no. 3, pp. 396–408, Sep. 2022, doi: 10.1016/j.icte.2021.10.003.

[14] M. Ali Ramdhani, D. S. Maylawati, and T. Mantoro, "Indonesian news classification using convolutional neural network," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 19, no. 2, p. 1000, Aug. 2020, doi: 10.11591/ijeecs.v19.i2.pp1000-1009.

[15] C. N. Tulu, "Experimental Comparison of Pre-Trained Word Embedding Vectors of Word2Vec, Glove, FastText for Word Level Semantic Text Similarity Measurement in Turkish," *Advances in Science and Technology. Research Journal*, vol. 16, no. 4, pp. 147–156, 2022, doi: 10.12913/22998624/152453.

[16] N. N. A. Balaji and B. Bharathi, "SSNCSE_NLP@ Fake news detection in the Urdu language (UrduFake) 2020," *Health (Irvine Calif)*, vol. 100, p. 100, 2020.

[17] H. Padalko, V. Chomko, and D. Chumachenko, "A novel approach to fake news classification using LSTM-based deep learning models," *Front Big Data*, vol. 6, p. 1320800, 2024, doi: 10.3389/FDATA.2023.1320800.

[18] Ö. Çelik and B. C. Koç, "TF-IDF, Word2vec ve Fasttext vektör model yöntemleri ile Türkçe haber metinlerinin snflandrlmas," *Dokuz Eylül Üniversitesi Mühendislik Fakültesi Fen ve Mühendislik Dergisi*, vol. 23, no. 67, pp. 121–127, 2021.

[19] F. Rollo, G. Bonisoli, and L. Po, "A comparative analysis of word embeddings techniques for italian news categorization," *IEEE Access*, vol. 12, pp. 25536–25552, 2024.

[20] J. H. Cabot and E. G. Ross, "Evaluating Prediction Model Performance," *Surgery*, vol. 174, no. 3, p. 723, Sep. 2023, doi: 10.1016/J.SURG.2023.05.023.

[21] O. Rainio, J. Teuho, and R. Klén, "Evaluation metrics and statistical tests for machine learning," *Sci Rep*, vol. 14, no. 1, pp. 1–14, Dec. 2024, doi: 10.1038/S41598-024-56706-X;SUBJMETA=117,531,639,705;KWRD=COMPUTER+SCIENCE,STATISTICS.

[22] M. K. Anam, S. Defit, Haviluddin, L. Efrizoni, and M. B. Firdaus, "Early Stopping on CNN-LSTM Development to Improve Classification Performance," *Journal of Applied Data Sciences*, vol. 5, no. 3, pp. 1175–1188, Aug. 2024, doi: 10.47738/JADS.V5I3.312.

[23] M. Vilares Ferro, Y. Doval Mosquera, F. J. Ribadas Pena, and V. M. Darriba Bilbao, "Early stopping by correlating online indicators in neural networks," *Neural Networks*, vol. 159, pp. 109–124, Feb. 2023, doi: 10.1016/J.NEUNET.2022.11.035.

[24] L. Brigato and L. Iocchi, "A Close Look at Deep Learning with Small Data," 2021.

[25] O. Kwon, D. Kim, S.-R. Lee, J. Choi, and S. Lee, "Handling Out-Of-Vocabulary Problem in Hangeul Word Embeddings," pp. 3213–3221, 2021.

[26] G. M. Foody, "Challenges in the real world use of classification accuracy metrics: From recall and precision to the Matthews correlation coefficient," *PLoS One*, vol. 18, no. 10, p. e0291908, Oct. 2023, doi: 10.1371/JOURNAL.PONE.0291908.

[27] A. Power, Y. Burda, H. Edwards, I. Babuschkin, and V. Misra, "Grokking: Generalization Beyond Overfitting on Small Algorithmic Datasets," Jan. 2022, Accessed: Jun. 13, 2025. [Online]. Available: https://arxiv.org/pdf/2201.02177

[28] A. I. Humayun, R. Balestriero, and R. Baraniuk, "Deep Networks Always Grok and Here is Why," *Proc Mach Learn*

Res, vol. 235, pp. 20722–20745, Feb. 2024, Accessed: Jun. 13, 2025. [Online]. Available: https://arxiv.org/pdf/2402.15555

[29] S. Takase, R. Ri, S. Kiyono, and T. Kato, "Large Vocabulary Size Improves Large Language Models," Jun. 2024, Accessed: Jun. 13, 2025. [Online]. Available: https://arxiv.org/pdf/2406.16508

[30] M. Kunilovskaya and A. Plum, "Text Preprocessing and its Implications in a Digital Humanities Project," 2021. doi: 10.26615/issn.2603-2821.2021_013.

[31] M. Siino, I. Tinnirello, and M. La Cascia, "Is text preprocessing still worth the time? A comparative survey on the influence of popular preprocessing methods on Transformers and traditional classifiers," *Inf Syst*, vol. 121, p. 102342, Mar. 2024, doi: 10.1016/J.IS.2023.102342.

[32] S. Rezaei *et al.*, "An experimental study of sentiment classification using deep-based models with various word embedding techniques," *Journal of Experimental and Theoretical Artificial Intelligence*, Nov. 2024, doi: 10.1080/0952813X.2024.2384568;PAGE:STRING:ARTICLE/CHAPTER.

[33] "Why is BiLSTM better than LSTM ?. Know the underlying functionality | by Sourasish Nath | Medium." Accessed: Nov. 09, 2025. [Online]. Available: https://medium.com/@souro400.nath/why-is-bilstm-better-than-lstm-a7eb0090c1e4

Try Setiawan Iksan merupakan pria kelahiran Gorontalo, 1 April 2003, yang saat ini berdomisili di Kelurahan Komo Luar, Kecamatan Wenang, Manado. Riwayat pendidikannya dimulai dari TK Islamic Centre Manado, kemudian berlanjut di SD Negeri 06, SMP Negeri 01, hingga SMK Negeri 01 Manado. Pada tahun 2021, penulis melanjutkan studi di Program Studi S1 Teknik Informatika, Jurusan Teknik Elektro, Fakultas Teknik, Universitas Sam Ratulangi. Di lingkungan kampus, penulis aktif berkontribusi dalam Unit Kegiatan Mahasiswa Unsrat IT Community (UNITY).