HOAQ: JURNAL TEKNOLOGI INFORMASI, Volume 17 Nomor 2, Desember 2026, p271-280
p-ISSN: 2337-5280, e-ISSN: 26207427
DOI: https://doi.org/10.52972/hoaq.vol17no2

PENGARUH *LEARNING RATE* TERHADAP KINERJA DAN KONVERGENSI MODEL INDOBERT-BILSTM-CRF PADA *NAMED ENTITY RECOGNITION* BAHASA INDONESIA

Alfred Yulius Arthadi Putra[1], Fredrikus Suarezsaga[1], Kristina[2]

[1] Program Studi Informatika, Universitas Widya Dharma Pontianak
[2] Program Studi Sistem Informasi, Universitas Widya Dharma Pontianak
Jl. HOS Cokroaminoto, Pontianak - Kalimantan Barat, Indonesia
Email: [1]alfred@widyadharma.ac.id, [2]suarezsaga@widyadharma.ac.id, [3]kristina@widyadharma.ac.id

ABSTRAK

Named Entity Recognition (NER) merupakan salah satu tugas penting dalam Natural Language Processing yang digunakan untuk mengidentifikasi dan mengklasifikasikan entitas dalam teks. Kinerja model NER *berbasis pre-trained language* model tidak hanya dipengaruhi oleh arsitektur, tetapi juga oleh konfigurasi hiperparameter, khususnya *learning rate*. Penelitian ini bertujuan menganalisis pengaruh learning rate terhadap kinerja dan konvergensi model IndoBERT-BiLSTM-CRF pada NER Bahasa Indonesia. Empat nilai *learning rate* diuji, yaitu 1×10^{-5}, 2×10^{-5}, 3×10^{-5}, dan 5×10^{-5}, dengan konfigurasi model lainnya dipertahankan tetap. Setiap konfigurasi dijalankan menggunakan tiga random seed untuk mengevaluasi kestabilan hasil. Kinerja model diukur menggunakan Token Macro-F1 dan Entity F1, sedangkan konvergensi dianalisis berdasarkan perubahan *training loss* dan *Validation Entity* F1 pada setiap epoch. Hasil menunjukkan bahwa peningkatan learning rate memberikan peningkatan kinerja hingga rentang tertentu. Entity F1 meningkat dari 0,8951 pada 1×10^{-5} menjadi 0,9392 pada 5×10^{-5}, sedangkan Token Macro-F1 meningkat dari 0,5848 menjadi 0,7902. Dari sisi konvergensi, 5×10^{-5} mencapai Validation Entity F1 sebesar 0,90 pada epoch ke-5, lebih cepat dibandingkan 3×10^{-5} pada epoch ke-6 dan 2×10^{-5}pada epoch ke-9. Meskipun 5×10^{-5} memberikan rata-rata kinerja tertinggi, perbedaan *Entity* F1 dengan 3×10^{-5} sangat kecil. Temuan ini menunjukkan bahwa *learning rate* memberikan pengaruh yang lebih kuat terhadap kecepatan konvergensi, sementara peningkatan kinerja akhir mulai mengalami saturasi pada rentang 3×10^{-5} hingga 5×10^{-5}.

Kata kunci: Named Entity Recognition, IndoBERT, BiLSTM-CRF, learning rate, konvergensi, Bahasa Indonesia

ABSTRACT

Named Entity Recognition (NER) is an important Natural Language Processing task for identifying and classifying entities in text. The performance of NER models based on pre-trained language models is influenced not only by the model architecture but also by hyperparameter configuration, particularly the learning rate. This study investigates the effect of learning rate on the performance and convergence of an IndoBERT-BiLSTM-CRF model for Indonesian NER. Four learning rates were evaluated, namely 1×10^{-5}, 2×10^{-5}, 3×10^{-5}, *and* 5×10^{-5}, *while keeping the remaining model configurations unchanged. Each configuration was evaluated using three random seeds to assess performance stability. Model performance was measured using Token Macro-F1 and Entity F1, while convergence was analyzed based on training loss and Validation Entity F1 across epochs. The results show that increasing the learning rate improves model performance up to a certain range. Entity F1 increased from 0.8951 at* 1×10^{-5} *to 0.9392 at* 5×10^{-5}, *while Token Macro-F1 increased from 0.5848 to 0.7902. In terms of convergence, the* 5×10^{-5} *configuration reached a Validation Entity F1 of 0.90 at epoch 5, earlier than* 3×10^{-5} *at epoch 6 and* 2×10^{-5}, *at epoch 9. Although* 5×10^{-5} *achieved the highest average performance, its Entity F1 was only marginally higher than that of* 3×10^{-5}. *These findings indicate that learning rate has a stronger effect on convergence speed, while final performance begins to saturate within the range of* 3×10^{-5}, *to* 5×10^{-5}.

Keywords: Named Entity Recognition, IndoBERT, BiLSTM-CRF, learning rate, convergence, Indonesian language.

1. PENDAHULUAN

Named Entity Recognition (NER) merupakan salah satu tugas penting dalam *Natural Language Processing* (NLP) yang bertujuan mengidentifikasi dan mengklasifikasikan entitas dalam teks, seperti lokasi, organisasi, individu, serta kategori khusus sesuai domain. NER menjadi komponen dasar dalam berbagai aplikasi, termasuk

Copyright (c) HOAQ (High Education of Organization Archive Quality) : Jurnal Teknologi Informasi

ekstraksi informasi, *information retrieval, question answering*, dan konstruksi *knowledge graph*. Perkembangan NER telah bergeser dari pendekatan berbasis aturan dan machine learning konvensional menuju *deep learning* dan *pre-trained language models*, yang mampu menghasilkan representasi kontekstual lebih baik [1][2]. Arsitektur BiLSTM-CRF juga banyak digunakan karena BiLSTM mampu mempelajari dependensi konteks dua arah, sedangkan CRF mempertimbangkan hubungan antarlabel dalam menghasilkan urutan prediksi yang lebih konsisten [1].

Untuk Bahasa Indonesia, pengembangan model pra-latih seperti IndoBERT memberikan peningkatan penting dalam pemrosesan teks karena model tersebut dilatih menggunakan korpus Bahasa Indonesia. Koto et al. memperkenalkan IndoBERT melalui benchmark IndoLEM dan menunjukkan performa kompetitif pada berbagai tugas NLP Bahasa Indonesia [3]. IndoNLU yang dikembangkan oleh Wilie et al. selanjutnya menyediakan sumber daya dan benchmark yang lebih luas untuk mengevaluasi model Bahasa Indonesia [4]. Dalam konteks NER, penelitian sebelumnya juga menunjukkan bahwa model berbasis IndoBERT dan kombinasi model sekuensial dapat digunakan secara efektif untuk mengekstraksi entitas dalam teks Bahasa Indonesia [5 - 7].

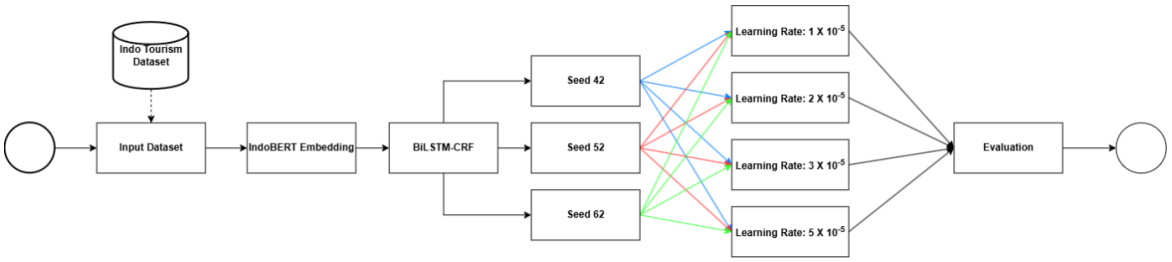
Penggabungan IndoBERT dengan BiLSTM dan CRF membentuk arsitektur yang memanfaatkan tiga kemampuan berbeda. IndoBERT menghasilkan representasi kontekstual dari setiap token, BiLSTM menangkap dependensi sekuensial dalam kedua arah, sedangkan CRF mengoptimalkan prediksi label dengan mempertimbangkan hubungan antarlabel. Pendekatan hybrid BERT-BiLSTM-CRF telah digunakan pada berbagai domain dan menunjukkan kemampuan yang baik dalam tugas NER [8][9]. Meskipun demikian, kinerja model tersebut tidak hanya dipengaruhi oleh arsitektur, tetapi juga oleh konfigurasi hiperparameter yang digunakan selama proses fine-tuning.

Salah satu hiperparameter yang berperan langsung dalam proses optimisasi adalah *learning rate*. Learning rate menentukan besarnya perubahan parameter model pada setiap langkah pembelajaran. Nilai yang terlalu kecil dapat menyebabkan proses pembelajaran berlangsung lambat, sedangkan nilai yang terlalu besar dapat menghasilkan proses optimisasi yang tidak stabil. Kondisi ini menjadi penting pada fine-tuning model berbasis BERT karena parameter model yang telah dipelajari melalui proses pre-training harus disesuaikan dengan tugas spesifik tanpa kehilangan representasi linguistik yang telah terbentuk. Mosbach et al. menunjukkan bahwa proses fine-tuning BERT dapat mengalami ketidakstabilan dan menghasilkan perbedaan performa akibat konfigurasi optimisasi [10]. Penelitian lain juga menunjukkan bahwa variasi learning rate dapat memberikan perbedaan performa pada model berbasis BERT maupun BiLSTM-CRF [11][12].

Penelitian sebelumnya telah mengevaluasi pengaruh learning rate terhadap kinerja model NER dan model berbasis BERT. [11] mengkaji penggunaan fitur kapitalisasi dan learning rate pada model RNN BiLSTM-CRF untuk meningkatkan kinerja NER, sedangkan [12] meneliti proses fine-tuning BERT untuk klasifikasi dan rekomendasi dokumen ilmiah. Penelitian ini mengembangkan fokus evaluasi pada arsitektur IndoBERT-BiLSTM-CRF untuk NER Bahasa Indonesia dalam domain pariwisata, dengan menganalisis hubungan antara variasi *learning rate*, kinerja pengenalan entitas, dinamika pelatihan, dan konsistensi hasil antar-random seed. Analisis tersebut diperlukan untuk memahami apakah konfigurasi yang menghasilkan kinerja akhir yang hampir sama juga menunjukkan perbedaan dalam kecepatan mencapai performa validasi tertentu dan variasi hasil pelatihan.

Berdasarkan permasalahan tersebut, penelitian ini bertujuan menganalisis pengaruh learning rate terhadap kinerja pengenalan entitas, dinamika konvergensi, dan stabilitas pelatihan model IndoBERT-BiLSTM-CRF pada NER Bahasa Indonesia. Empat nilai learning rate, yaitu yaitu 1×10^{-5}, 2×10^{-5}, 3×10^{-5}, 5×10^{-5} , dievaluasi dengan mempertahankan konfigurasi model dan pembagian dataset yang sama untuk mengisolasi pengaruh parameter yang diteliti. Kinerja model diukur menggunakan precision, recall, dan F1-score pada tingkat entitas, sedangkan dinamika konvergensi dianalisis berdasarkan perubahan *training loss, validation loss, validation* F1-score, dan jumlah epoch yang diperlukan untuk mencapai konvergensi. Selain itu, setiap konfigurasi dijalankan menggunakan beberapa random seed untuk mengevaluasi variabilitas dan konsistensi hasil pelatihan. Kontribusi utama penelitian ini adalah menyajikan karakterisasi empiris sensitivitas learning rate pada IndoBERT-BiLSTM-CRF dengan mengintegrasikan evaluasi kinerja, konvergensi, dan stabilitas model. Hasil penelitian diharapkan memberikan dasar empiris bagi pemilihan *learning rate* dengan mempertimbangkan keseimbangan antara ketepatan pengenalan entitas, efisiensi pelatihan, dan konsistensi hasil pada NER Bahasa Indonesia.

2. METODE PENELITIAN



Gambar 1. Kerangka Kerja Metodologi Penelitian

Copyright (c) HOAQ (High Education of Organization Archive Quality) : Jurnal Teknologi Informasi

Penelitian ini dilaksanakan dalam 4 tahap seperti disajikan di Gambar 1. Tahap pertama adalah prapemrosesan data. Dalam tahap ini, proses pembacaan dataset dan prapemrosesan data. Tahap kedua adalah proses pembelajaran, dimulai dari proses Embedding menggunakan IndoBERT [13] dan hasil vektorisasi di Embedding diteruskan ke model BiLSTM-CRF. Tahap ketiga adalah proses pengulangan pengujian. Tahap ini diuji secara berulang sebanyak 12 kali. Tahap keempat adalah evaluasi.

Input *Dataset*

Dataset yang digunakan berasal dari IndoNER-Tourism[1] [14]. *Dataset* ini diolah untuk kepentingan penelitian ini dan dibagi menjadi tiga data, data latih, data validasi, dan data uji. Dataset memiliki format NER, di mana memiliki 3 entitas, WIS, LOC, dan FAS. Skema labeling yang digunakan adalah BIO Tagging. Dataset terdiri dari 116.564 token. Entitas yang ada di dataset sebanyak 8.867. Untuk pembagiannya, data latih memiliki sebanyak 80.696 token, data validasi sebanyak 18.186 token, dan data uji sebanyak 17.682 token. Sedangkan, dari sisi entitas data latih memiliki 6.102 entitas, data validasi memiliki 1.382 entitas, dan data uji memiliki 1.382 entitas. Secara keseluruhan, persentase pembagian dataset adalah data latih sebesar 70%, data validasi 15%, dan data uji 15%. Distribusi label terdiri dari B-WIS sebanyak 5.534, I-WIS sebanyak 2.062, B-FAS sebanyak 1.934, I-FAS sebanyak 263, B-LOC sebanyak 1.399, dan I-LOC sebanyak 117. Sedangkan label O yang menunjukkan non-entitas sebanyak 105.255. Perbandingan label non-entitas dan entitas menunjukkan perbedaan yang jauh, yaitu 90,30% merupakan label non-entitas, sedangkan 9,70% label entitas.

Pembagian dataset dilakukan satu kali menggunakan random seed yang ditetapkan dan dipertahankan sama pada seluruh skenario eksperimen. Pembagian tersebut tidak menerapkan stratifikasi eksplisit berdasarkan label BIO karena satu unit teks dapat mengandung lebih dari satu kategori label. Untuk mengevaluasi potensi ketidakseimbangan kelas, distribusi setiap label pada data latih, validasi, dan uji diperiksa dan disajikan pada Tabel 1. Seluruh konfigurasi learning rate menggunakan pembagian dataset yang identik agar variasi komposisi data tidak menjadi faktor pembeda antarskenario eksperimen.

Tabel 1. Distribusi Label pada Data Latih, Validasi, dan Uji

Label	Training	Validation	Test	Total
B-WIS	3.812	838	884	5.534
I-WIS	1.379	313	370	2.062
B-LOC	926	261	212	1.399
I-LOC	81	24	12	117
B-FAS	1.364	283	286	1.933
I-FAS	186	38	39	263
O	72.947	16.429	15.879	105.255
Total Token	80.696	18.186	17.682	116.564
Total Entitas	6.102	1.382	1.382	8.866

IndoBERT *Embedding*

IndoBERT merupakan model bahasa yang telah dilatih lebih awal, dibangun di atas arsitektur BERT, dan disesuaikan untuk Bahasa Indonesia. Seperti BERT, IndoBERT menggunakan mekanisme Transformer dua arah, di mana representasi sebuah token dibentuk dengan memperhatikan konteks kiri dan kanan secara simultan. Pendekatan ini memungkinkan model untuk menghasilkan representasi kata yang bergantung pada konteks, sehingga satu kata bisa memiliki representasi yang berbeda ketika muncul dalam situasi yang berbeda [2]. IndoBERT dibuat untuk memenuhi tuntutan sumber daya NLP untuk Bahasa Indonesia dan telah memperlihatkan performa kompetitif dalam berbagai tugas pemrosesan Bahasa Indonesia [3], [4] .

IndoBERT digunakan sebagai contextual encoder untuk menghasilkan representasi setiap token sebelum diteruskan ke lapisan BiLSTM dan CRF. Teks terlebih dahulu ditokenisasi menjadi unit subword sesuai tokenizer IndoBERT. Untuk suatu urutan token $X = (x_1, x_2, \dots, x_n)$, IndoBERT menghasilkan representasi kontekstual berupa:

$$H = IndoBERT(X) = [h_1, h_2, \dots, h_3]$$

dengan h_i adalah vektor representasi kontekstual yang digunakan untuk menggambarkan token ke-*i*. Representasi tersebut kemudian digunakan sebagai input BiLSTM untuk belajar dependensi sekuensial, dan CRF digunakan pada tahap akhir untuk menentukan urutan label NER yang paling sesuai. Dengan demikian, IndoBERT menangkap informasi semantik dan kontekstual, sedangkan BiLSTM-CRF memperkuat pemodelan dependensi antar token dan antar label.

BiLSTM-CRF

Bidirectional Long Short-Term Memory (BiLSTM) adalah pengembangan dari LSTM yang dapat memproses urutan data dalam dua arah, yaitu dari awal ke akhir dan dari akhir ke awal. LSTM dirancang untuk mempertahankan informasi jangka panjang melalui mekanisme *memory cell dan gate*, sehingga dapat mengurangi

[1] https://github.com/fathanick/IndoNER-Tourism

Copyright (c) HOAQ (High Education of Organization Archive Quality) : Jurnal Teknologi Informasi

permasalahan *vanishing gradient* pada jaringan *recurrent* [15]. BiLSTM dapat membentuk representasi token dengan dua arah pemrosesan, yaitu berdasarkan konteks sebelum dan sesudah token tersebut. Kemampuan ini sangat penting dalam NER karena identitas suatu entitas sering kali bergantung pada konteks di kedua sisi token yang muncul.

Dalam penelitian ini, keluaran kontekstual dari IndoBERT diteruskan ke BiLSTM. Untuk representasi token h_i, BiLSTM menghasilkan hidden state dari arah maju dan mundur yang kemudian digabungkan sebagai:

$$h_i^{BiLSTM} = [\overrightarrow{h_i}; \overleftarrow{h_i}]$$

Representasi tersebut selanjutnya diproyeksikan ke ruang label menggunakan lapisan linear untuk menghasilkan emission score. Lapisan *Conditional Random Field* (CRF) kemudian digunakan untuk menentukan urutan label terbaik dengan mempertimbangkan tidak hanya skor setiap token, tetapi juga skor transisi antarlabel. Kombinasi BiLSTM dan CRF telah banyak digunakan dalam sequence labeling karena BiLSTM menangkap dependensi kontekstual sedangkan CRF membantu menghasilkan urutan label yang lebih konsisten secara global [16][17].

Secara umum, skor suatu urutan label $Y = (y_1, y_2, \ldots, y_n)$ terhadap masukan X dapat dirumuskan sebagai:

$$S(X,Y) = \sum_{i=1}^{n} P_{i,y_i} + \sum_{i=1}^{n-1} A_{y_i,y_{i+1}}$$

dengan P_{i,y_i} merupakan emission score untuk label y_i pada token ke-i, sedangkan ($A_{y_i,y_{i+1}}$) merupakan skor transisi dari label y_i ke label berikutnya. Pada proses prediksi, CRF memilih urutan label dengan skor terbesar.

Pengujian

Pengujian dilakukan untuk menganalisis pengaruh variasi learning rate terhadap kinerja dan dinamika pelatihan model IndoBERT-BiLSTM-CRF. Empat nilai learning rate yang digunakan 1×10^{-5}, 2×10^{-5} , 3×10^{-5}, 5×10^{-5}. Pemilihan rentang tersebut mempertimbangkan konfigurasi fine-tuning model BERT yang dilaporkan oleh [2] dan penelitian terdahulu mengenai variasi *learning rate* [11], [12]. Nilai 1×10^{-5} turut disertakan untuk mengevaluasi perilaku pelatihan pada pembaruan parameter yang lebih konservatif. Seluruh skenario menggunakan konfigurasi arsitektur, pembagian dataset, dan prosedur pelatihan yang sama, kecuali nilai learning rate yang menjadi variabel independen penelitian.

Data latih digunakan untuk memperbarui parameter model, sedangkan data validasi digunakan untuk memantau perkembangan kinerja dan menentukan checkpoint terbaik. Evaluasi dilakukan pada akhir setiap epoch menggunakan metrik Validation Exact Entity F1. Apabila nilai F1 meningkat dibandingkan nilai tertinggi pada epoch sebelumnya, bobot model disimpan sebagai *checkpoint* terbaik. Setelah proses pelatihan selesai, checkpoint dengan Validation Exact Entity F1 tertinggi dimuat kembali untuk melakukan evaluasi menggunakan data uji. Mekanisme tersebut diterapkan secara terpisah pada setiap kombinasi learning rate dan random seed, sehingga pemilihan model tidak didasarkan pada hasil data uji.

Setiap konfigurasi *learning rate* dijalankan menggunakan tiga random seed, yaitu 42, 52, dan 62, untuk mengevaluasi konsistensi hasil terhadap variasi proses pelatihan. Kinerja dari ketiga pengulangan dirangkum menggunakan nilai rata-rata (*mean*) dan standar deviasi (*standard deviation*). Rata-rata digunakan untuk merepresentasikan kecenderungan kinerja setiap konfigurasi, sedangkan standar deviasi digunakan untuk mengukur variasi hasil antar-seed. Kedua metrik dianalisis secara bersamaan karena konfigurasi dengan variasi hasil yang rendah tidak selalu menghasilkan kinerja prediksi yang tinggi. Konfigurasi pengujian disajikan di Tabel 2.

Tabel 2. Konfigurasi Arsitektur dan Hyperparameter Pelatihan

Hyperparameter	Konfigurasi
Hidden Size	256
BiLSTM *Layer*	2
Dropout	0,1
Maximum Sequence Length	256
Batch Size	16
Optimizer	AdamW
Learning Rate	1e-5, 2e-5, 3e-5, 5e-5
Random Seed	42, 52, 62

Evaluasi

Evaluasi utama dilakukan pada tingkat entitas menggunakan *Exact Matching*, yaitu suatu entitas dianggap benar jika tipe entitas dan posisi awal dan akhirnya sama persis dengan anotasi referensi. Evaluasi berbasis span telah menjadi praktik umum dalam evaluasi NER, termasuk dalam evaluasi CoNLL [18]. Nilai *precision*, *recall*, dan F1-*score* dihitung berdasarkan jumlah *True Positive* (TP), *False Positive* (FP), dan *False Negative* (FN) dirumuskan sebagai:

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

Copyright (c) HOAQ (High Education of Organization Archive Quality) : Jurnal Teknologi Informasi

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

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

Nilai F1 digunakan sebagai metrik utama karena mempertimbangkan keseimbangan antara kemampuan model menemukan entitas yang benar dan menghindari prediksi entitas yang salah.

Stabilitas hasil dievaluasi menggunakan rata-rata dan standar deviasi dari beberapa *random seed*. Untuk suatu metrik *x*, rata-rata dan standar deviasi dihitung sebagai:

$$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i$$

$$SD = \sqrt{\frac{\sum_{i=1}^{N}(x_i - \bar{x})^2}{N-1}}$$

Sementara itu, konvergensi dianalisis berdasarkan pola penurunan training loss, peningkatan Validation F1, dan epoch ketika model mencapai tingkat performa tertentu. Evaluasi tersebut memungkinkan pengaruh *learning rate* dianalisis dari dua sisi sekaligus, yaitu kualitas prediksi akhir dan dinamika proses pelatihan.

3. HASIL DAN PEMBAHASAN

Hasil Pengujian pada Berbagai *Learning Rate*

Pengujian dilakukan menggunakan empat variasi learning rate, yaitu 1×10^{-5}, 2×10^{-5}, 3×10^{-5}, 5×10^{-5}. Setiap konfigurasi dijalankan menggunakan tiga random seed, yaitu 42, 52, dan 62, sedangkan konfigurasi model lainnya dipertahankan tetap. Kinerja model dievaluasi menggunakan Token Macro-F1, *Entity Precision, Entity Recall*, dan *Entity* F1. Nilai akhir disajikan dalam bentuk rata-rata dan standar deviasi dari tiga pengujian untuk memberikan gambaran mengenai performa sekaligus variasi hasil antar-run. Hasil pengujian ditunjukkan di Tabel 3.

Tabel 3. Hasil Pengujian pada Berbagai *Learning Rate*

Learning Rate	Token Macro F1	Entity Precision	Entity Recall	Entity F1
$\mathbf{1 \times 10^{-5}}$	0,5848 ± 0,0012	0,8766 ± 0,0012	0,9144 ± 0,0004	0,8951 ± 0,0005
$\mathbf{2 \times 10^{-5}}$	0,6676 ± 0,0159	0,9114 ± 0,0090	0,9348 ± 0,0053	0,9229 ± 0,0072
$\mathbf{3 \times 10^{-5}}$	0,7833 ± 0,0046	0,9338 ± 0,0031	0,9445 ± 0,0037	0,9391 ± 0,0033
$\mathbf{5 \times 10^{-5}}$	0,7902 ± 0,0030	0,9364 ± 0,0052	0,9421 ± 0,0018	0,9392 ± 0,0035

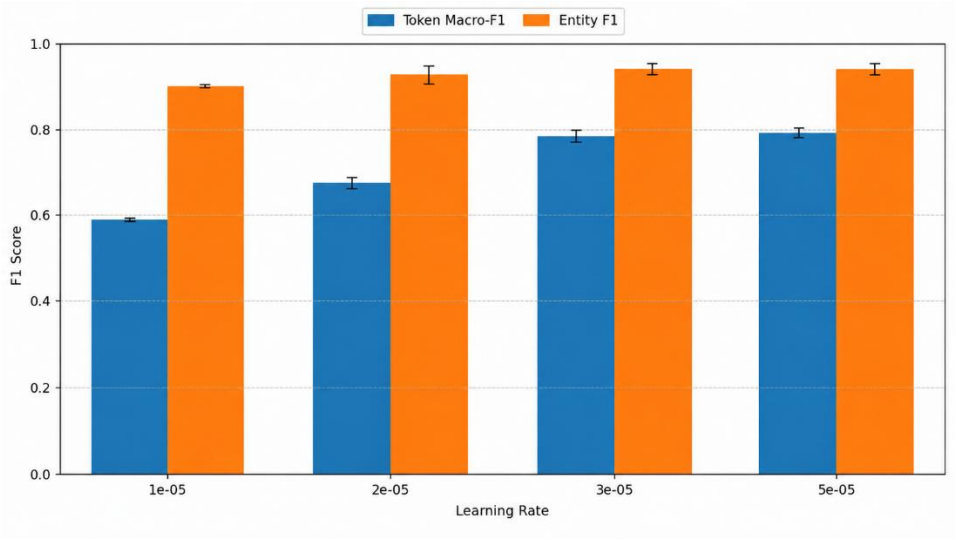
Berdasarkan Tabel 1, peningkatan *learning rate* secara umum diikuti oleh peningkatan kinerja model. *Entity* F1 meningkat dari 0,8951 ± 0,0005 pada 1×10^{-5} menjadi 0,9229 ± 0,0072 pada 2×10^{-5}, kemudian meningkat menjadi 0,9391 ± 0,0033 pada 3×10^{-5}. Nilai tertinggi diperoleh pada 5×10^{-5}, yaitu 0,9392 ± 0,0035. Pola yang sama juga terlihat pada Token Macro-F1 yang meningkat dari 0,5848 menjadi 0,7902.

Pada sisi *precision* dan *recall*, konfigurasi 5×10^{-5} menghasilkan rata-rata *Entity Precision* tertinggi, sekitar 0,9364, sedangkan *Entity Recall* tertinggi diperoleh pada 3×10^{-5}, sekitar 0,9445. Secara keseluruhan, 3×10^{-5} dan 5×10^{-5} merupakan dua konfigurasi dengan performa *entity-level* paling tinggi.

Pengaruh Learning Rate terhadap Kinerja Model

Perbedaan kinerja antar-learning rate menunjukkan bahwa pembaruan yang besar di parameter memberikan pengaruh terhadap kemampuan model dalam menyesuaikan representasi IndoBERT dengan tugas NER. Pada learning rate 1×10^{-5}, proses pembaruan parameter relatif konservatif sehingga kinerja akhir model masih lebih rendah dibandingkan konfigurasi lainnya. Peningkatan learning rate menjadi 2×10^{-5} meningkatkan *Entity* F1 sekitar 2,78 poin persentase, sedangkan penggunaan 3×10^{-5} kembali meningkatkan Entity F1 hingga sekitar 1,62 poin persentase dibandingkan 2×10^{-5}.

Copyright (c) HOAQ (High Education of Organization Archive Quality) : Jurnal Teknologi Informasi



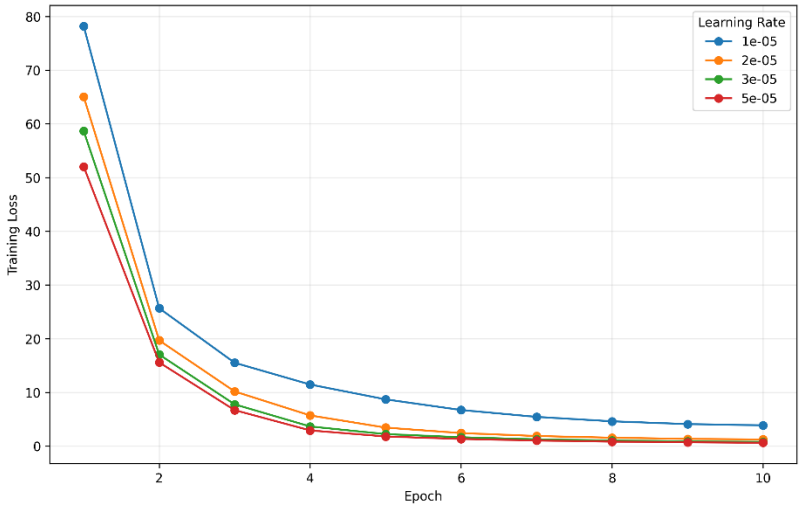
Gambar 2. Perbandingan kinerja IndoBERT-BiLSTM-CRF pada berbagai *Learning Rate*

Peningkatan performa tidak berlangsung secara terus menerus. F1 yang akurat pada 3×10^{-5} adalah 0,9391, sedangkan pada 5×10^{-5} sebesar 0,9392. Perbedaan antara keduanya hanya sekitar 0,0001. Peningkatan learning rate dari 3×10^{-5} menjadi 5×10^{-5} hanya menghasilkan selisih rata-rata Entity F1 sebesar 0,0001. Perbedaan tersebut menunjukkan bahwa peningkatan kinerja akhir secara deskriptif sangat kecil pada kedua konfigurasi. Kondisi ini mengindikasikan adanya kecenderungan saturasi kinerja dalam rentang learning rate yang diuji, meskipun interpretasi mengenai signifikansi statistik masih memerlukan pengujian lebih lanjut.

Meskipun demikian, Token Macro-F1 meningkat dari 0,7833 pada 3×10^{-5} menjadi 0,7902 pada 5×10^{-5}. Hasil tersebut menunjukkan bahwa peningkatan learning rate masih memberikan sedikit keuntungan pada klasifikasi token, tetapi pengaruhnya terhadap Entity F1 telah menjadi sangat kecil. Hasil ini menunjukkan adanya rentang learning rate yang efektif daripada hanya satu nilai learning rate yang paling baik.

Pengaruh *Learning Rate* terhadap Konvergensi Model

Gambar 3 menunjukkan perbandingan konvergensi model di berbagai learning rate. Semua konfigurasi menunjukkan penurunan loss pelatihan selama proses pelatihan, tetapi kecepatan penurunannya berbeda-beda. Berdasarkan rata-rata tiga seed acak, pada epoch pertama nilai loss masing-masing konfigurasi berada pada kisaran 78,18 untuk 1×10^{-5} , 65,04 untuk 2×10^{-5}, 58,66 untuk 3×10^{-5}, dan 52,04 untuk 4×10^{-5}. Pada epoch ke-10, nilai tersebut menurun menjadi sekitar 3,87, 1,24, 0,78, dan 0,62. Data pelatihan menunjukkan tren penurunan yang konsisten pada semua *seed*.

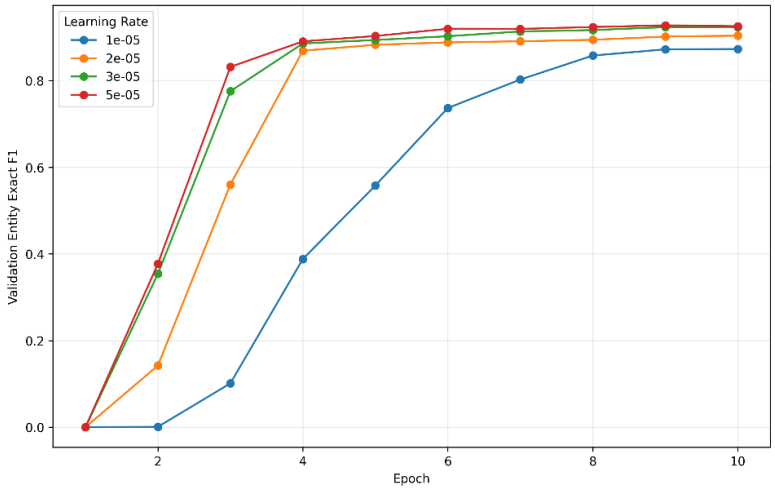


Gambar 3. Konvergensi *Training Loss* pada berbagai *learning rate*

Pola tersebut menunjukkan bahwa learning rate yang lebih tinggi dapat mempercepat proses optimisasi parameter. Konfigurasi 1×10^{-5} masih menunjukkan loss yang relatif tinggi pada beberapa epoch awal, sedangkan konfigurasi 3×10^{-5} dan 5×10^{-5} telah mencapai loss rendah lebih cepat. Nilai training loss yang lebih kecil tidak selalu menyebabkan peningkatan Entity F1 yang sama dalam proporsi. Perbedaan yang jelas dalam loss akhir terlihat pada nilai 3×10^{-5} dan 5×10^{-5}, meskipun menghasilkan hasil Entity F1 test yang sangat mirip.

Copyright (c) HOAQ (High Education of Organization Archive Quality) : Jurnal Teknologi Informasi

Gambar 4 menunjukkan konvergensi Validation Entity F1 pada berbagai *learning rate*. Kecepatan konvergensi dapat dilihat dari epoch pertama ketika rata-rata F1 Validation mencapai 0,90. Pada epoch ke-5, rata-rata F1 mencapai sekitar 0,9032 dengan learning rate 5×10^{-5}. Konfigurasi 3×10^{-5}mencapai nilai F1 sekitar 0,9027 pada epoch ke-6, sedangkan konfigurasi 2×10^{-5} mencapai nilai sekitar 0,9019 pada epoch ke-9. Sementara itu, nilai 1×10^{-5} belum mencapai Entity F1 sebesar 0,90 hingga epoch ke-10. Secara historis, validasi menunjukkan bahwa perbedaan kecepatan tersebut didukung oleh pola per-seed yang konsisten.



Gambar 4. Konvergensi Validation Entity F1 pada Berbagai *Learning Rate*

Hasil di Tabel 4 menampilkan rata-rata Validation F1 dari tiga seed pada setiap epoch. Hasil ini menunjukkan bahwa penggunaan *learning rate* yang lebih tinggi memberikan kontribusi utama pada kecepatan mencapai tingkat performa yang tinggi, bukan hanya pada performa akhir. Dalam 10 epoch yang digunakan pada penelitian ini, 5×10^{-5} memberikan proses konvergensi yang paling cepat. Dengan demikian, learning rate yang lebih kecil memerlukan jumlah iterasi yang lebih banyak untuk mencapai kondisi yang sebanding.

Perbedaan kecepatan pencapaian *Validation Exact Entity* F1 sebesar 0,90 menunjukkan bahwa learning rate memengaruhi dinamika pelatihan dalam anggaran maksimum 10 epoch. Namun, konfigurasi dengan learning rate lebih kecil yang belum mencapai ambang tersebut tidak dapat langsung dinyatakan memiliki kemampuan konvergensi yang lebih rendah secara keseluruhan. Pembatasan jumlah epoch memungkinkan perbandingan perkembangan kinerja seluruh konfigurasi dalam anggaran pelatihan yang sama, tetapi belum dapat memastikan apakah konfigurasi dengan learning rate lebih kecil dapat mencapai performa yang sebanding apabila diberikan waktu pelatihan lebih panjang.

Tabel 4. Kecepatan Pencapaian Target *Validation Exact Entity* F1 ≥ 0,90 pada Berbagai *Learning Rate*

Learning Rate	Epoch Pertama F1 ≥ 0,90	Validation Exact F1 pada Epoch
$\mathbf{1 \times 10^{-5}}$	Tidak Tercapai	0,8733 pada epoch 10
$\mathbf{2 \times 10^{-5}}$	9	0,9019
$\mathbf{3 \times 10^{-5}}$	6	0,9027
$\mathbf{5 \times 10^{-5}}$	5	0,9027

Kecepatan pencapaian performa validasi dianalisis menggunakan jumlah epoch pertama ketika rata-rata *Validation Exact Entity* F1 mencapai atau melampaui ambang 0,90. Ambang tersebut digunakan sebagai kriteria operasional untuk membandingkan kecepatan pencapaian tingkat performa yang sama pada seluruh konfigurasi learning rate, bukan sebagai indikator bahwa model telah mencapai konvergensi optimisasi secara matematis [14].

Stabilitas Kinerja terhadap *Random Seed*

Penggunaan beberapa random seed bertujuan untuk mengevaluasi apakah performa model konsisten terhadap variasi inisialisasi dan urutan proses pelatihan. Hasil *Entity* F1 untuk setiap seed disajikan pada Tabel 5.

Tabel 5. Stabilitas *Entity* F1 Berdasarkan *Random Seed*

Learning Rate	*Seed*			*Mean*	Standar Deviasi
	42	52	62		
$\mathbf{1 \times 10^{-5}}$	0,8956	0,8947	0,8950	0,8951	0,0005
$\mathbf{2 \times 10^{-5}}$	0,9148	0,9260	0,9281	0,9229	0,0072
$\mathbf{3 \times 10^{-5}}$	0,9367	0,9379	0,9429	0,9391	0,0033
$\mathbf{5 \times 10^{-5}}$	0,9387	0,9359	0,9430	0,9392	0,0035

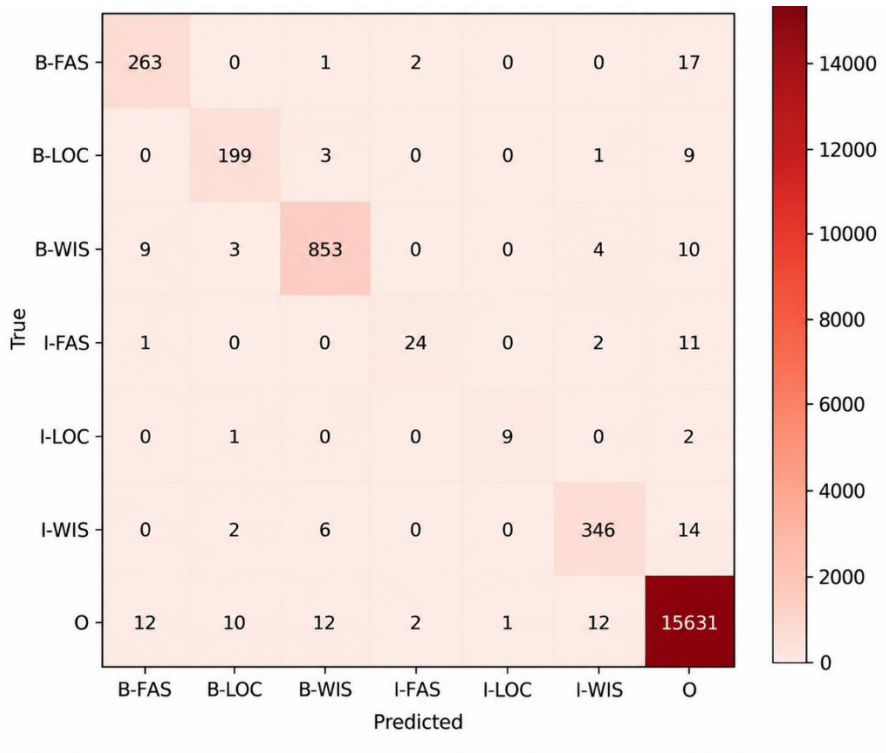
Copyright (c) HOAQ (High Education of Organization Archive Quality) : Jurnal Teknologi Informasi

Pada 1×10^{-5}, *Entity* F1 untuk seed 42, 52, dan 62 masing-masing sebesar 0,8956, 0,8947, dan 0,8950 sehingga menghasilkan standar deviasi yang sangat rendah, yaitu 0,0005. Namun, stabilitas tersebut disertai dengan performa rata-rata paling rendah. Dengan demikian, standar deviasi kecil tidak dapat digunakan secara terpisah sebagai indikator konfigurasi terbaik.

Konfigurasi 3×10^{-5} memperoleh Entity F1 sebesar 0,9367, 0,9379, dan 0,9429, dengan rata-rata 0,9391 $\pm$ 0,0033. Sementara itu, 5×10^{-5} memperoleh nilai 0,9387, 0,9359, dan 0,9430, dengan rata-rata 0,9392 $\pm$ 0,0035. Kedua konfigurasi menunjukkan variasi yang relatif rendah sekaligus performa yang tinggi.

Analisis Kesalahan Model

Analisis kesalahan dilakukan menggunakan confusion matrix pada konfigurasi 5×10^{-5} dengan seed 42. Secara umum, model menunjukkan kemampuan yang tinggi dalam mengenali label B-WIS, B-LOC, B-FAS, I-WIS, serta label O. Pada hasil uji tersebut, B-WIS memperoleh 853 prediksi benar dari 879 token, sedangkan B-FAS dan B-LOC masing-masing memperoleh 263 dari 283 token dan 199 dari 212 token yang diklasifikasikan dengan benar. Hasil ini ditunjukkan di Gambar 5.



Gambar 5. *Confusion Matrix* konfigurasi 5×10^{-5} *seed* 42

Label lanjutan entitas menunjukkan kesalahan yang lebih besar, terutama pada I-FAS dan I-LOC. Dari 38 token I-FAS, 24 dapat diprediksi dengan benar dan 11 diprediksi sebagai O. Pada I-LOC, 9 dari 12 token dikenali dengan benar dan 2 diprediksi sebagai O. I-WIS memiliki performa yang lebih baik dengan 346 prediksi benar dari 368 token. Data tersebut menunjukkan bahwa kesalahan model lebih banyak terjadi pada label lanjutan dengan jumlah contoh yang relatif sedikit.

Temuan ini juga sejalan dengan karakteristik dataset, di mana I-FAS dan I-LOC memiliki jumlah yang jauh lebih kecil daripada B-FAS, B-LOC, dan I-WIS. Rendahnya kemampuan model pada kedua kelas tersebut diduga terkait dengan ketidakseimbangan jumlah contoh, meskipun hubungan kausalnya memerlukan eksperimen khusus untuk dapat dipastikan. Model terkadang berhasil mengenali bagian awal entitas tetapi gagal mempertahankan keseluruhan span entitas multi-token, seperti yang ditunjukkan oleh kesalahan I- yang menjadi O. Dengan demikian, meskipun skor Entity F1 keseluruhan tinggi, penentuan batas untuk kategori entitas yang jarang masih menjadi penyebab kesalahan.

Diskusi dan Limitasi

Hasil penelitian menunjukkan bahwa variasi learning rate menghasilkan perbedaan pada kinerja akhir dan dinamika pelatihan model IndoBERT-BiLSTM-CRF. Peningkatan learning rate dari 1×10^{-5} hingga 5×10^{-5} secara umum diikuti oleh peningkatan Entity F1 dan kecepatan pencapaian target performa validasi. Namun, peningkatan tersebut tidak berlangsung secara proporsional pada seluruh rentang pengujian. Konfigurasi 3×10^{-5} dan 5×10^{-5} menghasilkan rata-rata Entity F1 yang hampir sama, meskipun menunjukkan perbedaan jumlah epoch untuk mencapai Validation Exact Entity F1 sebesar 0,90.

Copyright (c) HOAQ (High Education of Organization Archive Quality) : Jurnal Teknologi Informasi

Temuan tersebut memberikan konteks empiris terhadap penelitian [10], yang membahas ketidakstabilan proses fine-tuning model BERT dan pentingnya konfigurasi optimisasi terhadap hasil pelatihan. Penelitian [11] sebelumnya mengkaji penggunaan fitur kapitalisasi dan variasi learning rate untuk meningkatkan kinerja NER berbasis RNN BiLSTM-CRF, sedangkan [12] meneliti fine-tuning BERT untuk klasifikasi dan rekomendasi dokumen ilmiah. Berbeda dari fokus tugas dan arsitektur tersebut, penelitian ini menyajikan evaluasi sensitivitas learning rate pada IndoBERT-BiLSTM-CRF untuk NER Bahasa Indonesia dengan mempertimbangkan kinerja pada tingkat token dan entitas, perkembangan metrik validasi sepanjang pelatihan, serta variasi hasil antar-*random seed*.

Penelitian ini memiliki beberapa keterbatasan. Pertama, eksperimen dilakukan menggunakan satu dataset NER Bahasa Indonesia pada domain pariwisata, sehingga temuan mengenai pengaruh learning rate belum dapat digeneralisasikan secara langsung pada domain lain. Kedua, pengujian dibatasi pada empat nilai learning rate dalam rentang 1×10^{-5} hingga 5×10^{-5}, sehingga hasil penelitian merepresentasikan sensitivitas model pada rentang parameter tersebut. Ketiga, proses pelatihan dilakukan selama maksimum 10 epoch dengan tujuan membandingkan perkembangan kinerja seluruh konfigurasi menggunakan anggaran pelatihan yang sama. Pembatasan tersebut belum dapat memastikan apakah konfigurasi dengan learning rate lebih kecil dapat mencapai performa yang sebanding apabila diberikan waktu pelatihan lebih panjang. Keempat, setiap konfigurasi dijalankan menggunakan tiga random seed, sehingga evaluasi stabilitas terbatas pada variasi yang teramati dalam ketiga pengulangan tersebut. Selain itu, ketidakseimbangan distribusi label pada dataset berpotensi memengaruhi pengenalan entitas dengan jumlah contoh yang relatif sedikit.

4. SIMPULAN

Penelitian ini menganalisis bagaimana variasi learning rate mempengaruhi kinerja dan konvergensi model IndoBERT-BiLSTM-CRF dalam tugas NER dalam Bahasa Indonesia. Empat nilai *learning rate*, yaitu 1×10^{-5}, 2×10^{-5}, 3×10^{-5}, dan 5×10^{-5}, telah diuji menggunakan konfigurasi model yang sama dan 3 random seed, yaitu 42, 52, dan 62. Eksperimen menunjukkan bahwa meningkatkan *learning rate* memiliki pengaruh yang signifikan pada proses pembelajaran model. F1 meningkat dari sekitar 0,8951 pada 1×10^{-5} menjadi sekitar 0,9392 pada 5×10^{-5}. Pengaruh learning rate lebih dominan dan memberikan peningkatan pada kecepatan konvergensi daripada kinerja akhir.

Learning rate 5×10^{-5} mencapai *Entity* F1 di Validasi sebesar 0,90 lebih cepat daripada konfigurasi lainnya, sementara 1×10^{-5} menunjukkan proses pembelajaran yang lambat dalam 10 epoch. Meskipun 5×10^{-5} memberikan rata-rata kinerja tertinggi, perbedaan Entity F1 terhadap 3×10^{-5} sangat kecil.

Berdasarkan kombinasi kinerja, stabilitas antar-random seed, dan kecepatan konvergensi, 5×10^{-5} menjadi konfigurasi yang paling efektif dalam ruang eksperimen yang diuji. Namun, *learning rate* 3×10^{-5} tetap memberikan performa yang sangat kompetitif dan praktis setara pada level Entity F1. Dengan demikian, ini menunjukkan bahwa pemilihan *learning rate* pada IndoBERT-BiLSTM-CRF sebaiknya tidak hanya didasarkan pada skor akhir, tetapi juga mempertimbangkan kecepatan konvergensi dan kestabilan hasil.

Penelitian selanjutnya dapat meneliti lebih lanjut mengenai pengujian rentang *learning rate* yang lebih rapat. Selain itu, evaluasi dapat dilakukan menggunakan jumlah epoch yang lebih besar untuk mengetahui apakah learning rate kecil seperti 1×10^{-5} mampu mencapai performa yang setara apabila diberikan waktu pelatihan yang lebih panjang. Selain itu, pengujian pada dataset Bahasa Indonesia dari domain berbeda, seperti berita, media sosial, kesehatan, atau layanan publik, dapat dilakukan untuk mengevaluasi generalisasi temuan. Analisis lebih lanjut terhadap kesalahan boundary, ketidakseimbangan kelas, dan entitas multi-token juga penting dilakukan untuk memahami keterbatasan model secara lebih mendalam.

DAFTAR PUSTAKA

[1] Warto, S. Rustad, G. F. Shidik, E. Noersasongko, Purwanto, Muljono, and D. R. I. M. Setiadi, "Systematic Literature Review on Named Entity Recognition: Approach, Method, and Application," *Statistics, Optimization & Information Computing*, vol. 12, no. 4, pp. 907–942, Feb. 2024, doi: 10.19139/soic-2310-5070-1631.

[2] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-Training of Deep Bidirectional Transformers For Language Understanding," in *Proceedings of the 2019 Conference of the North*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2019, pp. 4171–4186. doi: 10.18653/v1/N19-1423.

[3] F. Koto, A. Rahimi, J. H. Lau, and T. Baldwin, "IndoLEM and IndoBERT: A Benchmark Dataset and Pre-trained Language Model for Indonesian NLP," in *Proceedings of the 28th International Conference on Computational Linguistics*, Stroudsburg, PA, USA: International Committee on Computational Linguistics, 2020, pp. 757–770. doi: 10.18653/v1/2020.coling-main.66.

[4] B. Wilie, K. Vincentio, G. I. Winata, S. Cahyawijaya, X. Li, Z. Y. Lim, S. Soleman, R. Mahendra, P. Fung, S. Bahar, and A. Purwarianti, "IndoNLU: Benchmark and Resources for Evaluating Indonesian Natural Language Understanding," in *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the*

Copyright (c) HOAQ (High Education of Organization Archive Quality) : Jurnal Teknologi Informasi

Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing, Stroudsburg, PA, USA: Association for Computational Linguistics, 2020, pp. 843–857. doi: 10.18653/v1/2020.aacl-main.85.

[5] S. O. Khairunnisa, A. Imankulova, and M. Komachi, "Towards a Standardized Dataset on Indonesian Named Entity Recognition," in *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing: Student Research Workshop*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2020, pp. 64–71. doi: 10.18653/v1/2020.aacl-srw.10.

[6] E. Dave and A. Chowanda, "IPerFEX-2023: Indonesian personal financial entity extraction using indoBERT-BiGRU-CRF model," *Journal of Big Data*, vol. 11, no. 1, pp. 1–24, Sep. 2024, doi: 10.1186/s40537-024-00987-6.

[7] Z. Zainuddin, - Mudassir, and Z. Tahir, "Entity Extraction in Indonesian Online News Using Named Entity Recognition (NER) with Hybrid Method Transformer, Word2Vec, Attention and Bi-LSTM," *JOIV : International Journal on Informatics Visualization*, vol. 9, no. 3, pp. 964–973, May 2025, doi: 10.62527/joiv.9.3.2902.

[8] W. Zhang, C. Wang, H. Wu, C. Zhao, G. Teng, S. Huang, and Z. Liu, "Research on the Chinese Named-Entity–Relation-Extraction Method for Crop Diseases Based on BERT," *Agronomy*, vol. 12, no. 9, p. 2130, Sep. 2022, doi: 10.3390/agronomy12092130.

[9] M. B. Shishehgarkhaneh, R. C. Moehler, Y. Fang, A. A. Hijazi, and H. Aboutorab, "Transformer-Based Named Entity Recognition in Construction Supply Chain Risk Management in Australia," *IEEE Access*, vol. 12, pp. 41829–41851, 2024, doi: 10.1109/ACCESS.2024.3377232.

[10] M. Mosbach, M. Andriushchenko, and D. Klakow, "On the Stability of Fine-tuning BERT: Misconceptions, Explanations, and Strong Baselines," Mar. 2021, doi: 10.48550/arXiv.2006.04884.

[11] Warto, Muljono, Purwanto, and E. Noersasongko, "Capitalization Feature and Learning Rate for Improving NER Based on RNN BiLSTM-CRF," in *2022 IEEE International Conference on Cybernetics and Computational Intelligence (CyberneticsCom)*, IEEE, Jun. 2022, pp. 398–403. doi: 10.1109/CyberneticsCom55287.2022.9865660.

[12] M. D. S. Antariksa, A. Sugiharto, and B. Surarso, "BERT Model Fine-tuned for Scientific Document Classification and Recommendation," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 9, no. 4, pp. 754–764, Aug. 2025, doi: 10.29207/resti.v9i4.6789.

[13] K. Kamdan, M. P. Anugrah, M. J. Almutaali, R. Ramdani, and I. L. Kharisma, "Performance Analysis of IndoBERT for Detection of Online Gambling Promotion in YouTube Comments," in *The 7th International Global Conference Series on ICT Integration in Technical Education & Smart Society*, Basel Switzerland: MDPI, Sep. 2025, p. 66. doi: 10.3390/engproc2025107066.

[14] A. F. Hidayatullah, M. F. D. A. Putra, A. P. Wibowo, and K. R. Nastiti, "Named Entity Recognition on Tourist Destinations Reviews in the Indonesian Language," *Jurnal Linguistik Komputasional (JLK)*, vol. 6, no. 1, pp. 30–35, Apr. 2023, doi: 10.26418/jlk.v6i1.89.

[15] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.

[16] X. Ma and E. Hovy, "End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2016, pp. 1064–1074. doi: 10.18653/v1/P16-1101.

[17] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, and C. Dyer, "Neural Architectures for Named Entity Recognition," in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2016, pp. 260–270. doi: 10.18653/v1/N16-1030.

[18] E. F. Tjong Kim Sang and F. De Meulder, "Introduction to the CoNLL-2003 shared task," in *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003* -, Morristown, NJ, USA: Association for Computational Linguistics, 2003, pp. 142–147. doi: 10.3115/1119176.1119195.

Copyright (c) HOAQ (High Education of Organization Archive Quality) : Jurnal Teknologi Informasi