

# Ekstraksi Aspek Aksesibilitas untuk Peningkatan Pengalaman Pengguna Menggunakan NER dengan CNN dan LSTM

Irmma Dwijayanti<sup>1</sup>, Alfirna Rizqi Lahitani<sup>2</sup>, Kartikadyota Kusumaningtyas<sup>3</sup>, Muhammad Habibi<sup>4</sup>

<sup>1</sup>Sistem Informasi, Fakultas Teknik dan Teknologi Informasi, Universitas Jenderal Achmad Yani Yogyakarta

<sup>2</sup>Teknologi Informasi, Fakultas Teknik dan Teknologi Informasi, Universitas Jenderal Achmad Yani Yogyakarta

<sup>3,4</sup>Informatika, Fakultas Teknik dan Teknologi Informasi, Universitas Jenderal Achmad Yani Yogyakarta

[irmmadwijayanti@gmail.com](mailto:irmmadwijayanti@gmail.com)\*, [alfirnalahitani@gmail.com](mailto:alfirnalahitani@gmail.com), [kartikadyota@gmail.com](mailto:kartikadyota@gmail.com), [muhammadhabibi17@gmail.com](mailto:muhammadhabibi17@gmail.com)

## Abstract

*Online transportation has a positive impact on most people, but the use of applications by people with disabilities still faces several challenges. The lack of adequate facilities and good user experience are the main obstacles. This reality shows that special attention is needed to the principle of accessibility to improve user experience and comfort for people with disabilities. Through user reviews, aspects of accessibility can be identified to support improving user experience. This study aims to extract information from user reviews related to accessibility using the NER method with the CNN and LSTM approaches. The data collected through web scraping consisted of 6,255 reviews of the Gojek, Grab, Maxim, and Indriver applications. The evaluation results showed that both models had high accuracy CNN 99.84%, LSTM 99.48%, but needed improvement in detecting entities that rarely appear or have complex contexts. The analysis showed that reviews discussed more application features and complaints related to accessibility. CNN was more effective in capturing specific patterns, while LSTM was stronger in capturing word variations.*

*Keywords: accesibility, user experience, NER, CNN, LSTM*

## Abstrak

Transportasi *online* memberikan dampak positif bagi sebagian besar masyarakat, namun penggunaan aplikasi oleh penyandang disabilitas masih menghadapi sejumlah tantangan. Belum adanya fasilitas yang memadai dan pengalaman pengguna yang baik menjadi kendala utama. Realitas ini menunjukkan bahwa perlunya perhatian khusus terhadap prinsip aksesibilitas untuk meningkatkan pengalaman pengguna dan kenyamanan bagi penyandang disabilitas. Melalui ulasan pengguna dapat diidentifikasi aspek-aspek aksesibilitas untuk mendukung peningkatan pengalaman pengguna. Penelitian ini bertujuan mengekstraksi informasi dari ulasan pengguna terkait aksesibilitas menggunakan metode NER dengan pendekatan CNN dan LSTM. Data yang dikumpulkan melalui *web scraping* terdiri dari 6.255 ulasan aplikasi Gojek, Grab, Maxim, dan Indriver. Hasil evaluasi menunjukkan bahwa kedua model memiliki akurasi tinggi yaitu CNN 99,84%, dan LSTM 99,48%. Namun memerlukan perbaikan dalam mendeteksi entitas yang jarang muncul atau berkonteks kompleks. Hasil analisis menunjukkan bahwa ulasan lebih banyak membahas fitur aplikasi dan keluhan yang berkaitan dengan aksesibilitas. CNN lebih efektif dalam menangkap pola spesifik, sedangkan LSTM lebih kuat dalam menangkap variasi kata.

Kata kunci: aksesibilitas, pengalaman pengguna, NER, CNN, LSTM

©This work is licensed under a Creative Commons Attribution -ShareAlike 4.0 International License

## 1. Pendahuluan

Transportasi *online* menjadi kebutuhan penting bagi masyarakat Indonesia, termasuk penyandang disabilitas. Berdasarkan survei *Institute for Development of Economics and Finance* (INDEF) 2022, Gojek, Grab, Maxim, dan Indriver mendominasi penggunaan aplikasi transportasi *online* di Indonesia [1]. Kehadiran layanan tersebut sangat membantu mobilitas penyandang disabilitas, meskipun masih terdapat tantangan berupa kurangnya fasilitas yang memadai dan pengalaman pengguna yang optimal.

Beberapa aplikasi saat ini mulai mendukung kesetaraan akses, tetapi fitur khusus yang dirancang untuk mempermudah penggunaan bagi penyandang disabilitas belum banyak dikembangkan. Berdasarkan data Kemenko PMK jumlah penyandang disabilitas di Indonesia mencapai 22,97 juta jiwa atau sekitar 8,5% dari total penduduk Indonesia [2]. Dari jumlah tersebut, hanya 9% diantaranya yang memiliki akses digital [3].

Hal tersebut mencerminkan adanya kesenjangan digital yang signifikan bagi pengguna disabilitas dalam memanfaatkan teknologi salah satunya aplikasi transportasi *online*.

Analisis faktor-faktor yang mempengaruhi pengalaman pengguna pada aplikasi transportasi *online*, seperti kualitas sistem dan informasi [4], kepercayaan merek, pelayanan, dan religiusitas [5], serta kualitas layanan, kepuasan, dan loyalitas pelanggan [6], [7] sudah dilakukan, namun penelitian yang secara eksplisit berfokus pada pengalaman aksesibilitas pengguna disabilitas masih terbatas. Realitas ini menunjukkan bahwa perlunya peningkatan perhatian terhadap prinsip aksesibilitas untuk meningkatkan pengalaman pengguna dan kenyamanan bagi pengguna disabilitas.

Aksesibilitas dalam *Human-Computer Interaction* (HCI) merupakan salah satu faktor yang mempengaruhi pengalaman pengguna atau *user experience* (UX) untuk memastikan bahwa suatu

produk atau layanan dapat digunakan oleh semua orang bagaimanapun kondisi penggunanya [8]. Prinsip aksesibilitas bertujuan untuk mewujudkan lingkungan yang inklusif untuk semua individu dapat berinteraksi dengan teknologi dengan mudah dan efektif tanpa memandang kemampuan atau keterbatasannya. Oleh karena itu, diperlukan inovasi dalam pengembangan aplikasi untuk memenuhi kebutuhan khusus individu dengan disabilitas melalui peningkatan aksesibilitas.

Ulasan pengguna di *Play Store* dapat menjadi sumber informasi berharga untuk memahami pengalaman disabilitas dalam menggunakan aplikasi transportasi *online*. Ulasan tersebut dapat memberikan wawasan mengenai kesulitan navigasi, ukuran tombol, kontras warna, atau ketersediaan fitur suara, yang tidak selalu tidak terungkap melalui survei tradisional [9]. Selain itu, analisis ulasan memungkinkan pengumpulan data dalam skala besar dan penggalian informasi yang lebih mendalam. Salah satu tantangan dalam menganalisis data ulasan adalah keberagaman bahasa alami yang digunakan dalam teks ulasan, sehingga menyulitkan untuk memperoleh informasi yang sesuai dengan harapan secara efektif.

Beberapa metode analisis seperti *Support Vector Machine* (SVM) [4]–[6], [10], *Decision Tree* [10], *K-Means* [6], dan *Convolutional Neural Network* (CNN) [7] telah digunakan untuk klasifikasi ulasan pengguna terutama analisis sentimen untuk memahami polaritas positif atau negatif terhadap layanan aplikasi. Namun, metode ini belum banyak diterapkan untuk mengidentifikasi masalah aksesibilitas spesifik.

Penelitian ini mengusulkan pendekatan baru dengan memanfaatkan *Named Entity Recognition* (NER), sebuah metode dalam *Natural Language Processing* (NLP), untuk mengenali dan mengklasifikasikan entitas dari ulasan pengguna yang relevan dengan aksesibilitas [11]. Dalam penelitian ini, tantangan spesifik meliputi identifikasi isu seperti kesesuaian fitur aplikasi dengan *Mobile Application Contents Accessibility Guidelines* (MCAG) 2.0 [12], termasuk pengenalan entitas seperti “suara”, “tombol”, atau “navigasi”.

Beberapa penelitian terdahulu telah menggunakan NER untuk berbagai keperluan, seperti klasifikasi jenis kejahatan dari berita daring menggunakan SVM [13], analisis pasar ponsel pintar menggunakan *Amazon Comprehend Custom NER* [14], dan ekstraksi informasi dari pertanyaan pelanggan menggunakan *Hidden Markov Model* (HMM) dan *Rule Based Template* [15]. Dengan metode NER, diharapkan masalah-masalah tersebut dapat dikenali dan diklasifikasikan secara lebih akurat dibandingkan pendekatan sebelumnya.

Klasifikasi entitas dapat menggunakan berbagai macam algoritma. CNN merupakan teknologi *deep learning* yang efektif dalam pengenalan pola, dan telah menunjukkan performa tinggi untuk meningkatkan performa klasifikasi entitas teks [11]. CNN telah

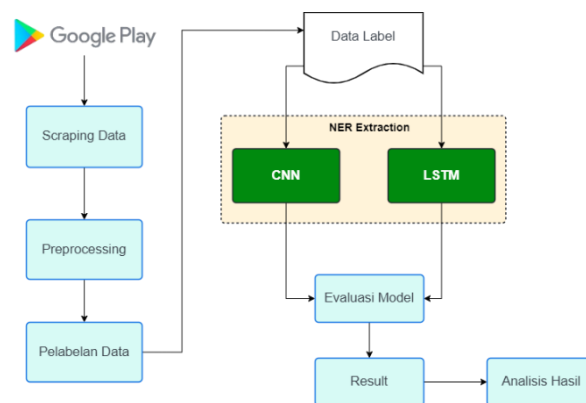
menunjukkan performa yang tinggi dalam berbagai aplikasi seperti klasifikasi sentimen ulasan film [16], produk layanan transportasi daring [7], komentar pengguna aplikasi [17]. Penelitian tersebut memiliki performa kinerja klasifikasi cukup tinggi yaitu diatas 90%. Oleh karena itu, penelitian ini memanfaatkan cara kerja dan performa akurasi tinggi yang dimiliki CNN untuk menentukan label entitas dalam proses NER.

Sementara itu, *Long ShortTerm Memory* (LSTM) merupakan jenis khusus dari *Recurrent Neural Network* (RNN) yang mampu menangani ketergantungan jangka panjang, juga digunakan untuk membandingkan performa dengan CNN. LSTM telah berhasil digunakan dalam klasifikasi sentimen [16], [18], klasifikasi kategori berita [19]–[21], serta deteksi berita hoax [22]. Hasil penelitian-penelitian tersebut menunjukkan performa akurasi klasifikasi yang cukup tinggi.

Penelitian ini bertujuan mengekstraksi informasi dari ulasan pengguna guna mengidentifikasi masalah-masalah kritis terkait aksesibilitas aplikasi transportasi *online*. Kebaruan dari penelitian ini adalah menerapkan CNN dan LSTM untuk mengidentifikasi entitas aksesibilitas dari ulasan pengguna yang sebelumnya belum digunakan dalam proses NER. Hasil identifikasi masalah aksesibilitas diharapkan dapat digunakan untuk meningkatkan kepuasan pengguna dalam menggunakan aplikasi, termasuk bagi pengguna dengan keterbatasan. Sehingga aplikasi dapat diakses dan bermanfaat bagi semua kalangan masyarakat.

## 2. Metode Penelitian

Penelitian ini mencakup beberapa tahapan penelitian diantaranya pengumpulan data ulasan, *preprocessing*, pelabelan manual, ekstraksi fitur dengan CNN dan LSTM, evaluasi model baik CNN maupun LSTM, dan analisis hasil. Tahapan penelitian ditunjukkan pada Gambar 1.



Gambar 1. Tahapan Penelitian

### 2.1. Pengumpulan Data

Tahap awal dalam penelitian ini adalah pengumpulan data ulasan dari aplikasi transportasi *online* yang tersedia di Google Playstore, dengan fokus pada aplikasi Gojek, Grab, Maxim, dan Indriver. Proses

pengumpulan data ulasan dilakukan dengan metode *web scraping* menggunakan *library google\_play\_scraper* pada bahasa pemrograman Python. Data yang diambil merupakan ulasan dalam Bahasa Indonesia menggunakan kata kunci yang berkaitan dengan aksesibilitas seperti disabilitas, 'aksesibel', 'difabel', 'gangguan penglihatan', dan kata kunci sejenis. Hasil perolehan data dari proses *scraping* sebanyak 6.255 ulasan. Tabel 1 menunjukkan *sample* data ulasan yang dikumpulkan.

Tabel 1. Contoh Data Ulasan Aplikasi Transportasi Online

No	Ulasan
1	Kok sekarang buat kami yg tunanetra ga bisa pake chat ke pengemudi Tombol kirim yg terletak dikanan tidak bisa di akses
2	keluhan buat konsumen yang mau menggunakan gocar ,, selalu pada kendala bagasi mobil yang tidak bisa memuat kursi roda untuk konsumen yang lagi sakit

### 2.2. Preprocessing Data

Tahap selanjutnya adalah proses membersihkan data dari elemen yang tidak relevan untuk diolah menggunakan model NER. Beberapa langkah *preprocessing* adalah *casefolding* untuk mengubah teks menjadi bentuk standar atau *lowercase*, penghapusan tanda baca dan karakter selain huruf, tokenisasi untuk memecah kalimat menjadi kata-kata (token), bigram untuk menangkap hubungan antar kata, *stopword removal* untuk menghapus kata-kata umum yang tidak mengandung informasi penting, dan *stemming* untuk mengubah kata menjadi bentuk dasar [23].

Pada penelitian ini, penggunaan bigram merupakan langkah penting dalam *preprocessing* data karena banyak entitas terkait aksesibilitas yang terdiri dari lebih dari satu kata. *Bigram* adalah kombinasi dua kata berurutan yang memungkinkan model menangkap konteks antar kata dengan lebih baik [24], seperti "tombol navigasi" atau "warna kontras." Tanpa bigram, entitas penting yang terdiri dari beberapa kata bisa terabaikan atau terdeteksi secara tidak akurat.

Pada penggunaan bigram, model dapat mendeteksi hubungan antar kata dan mengenali frasa tersebut sebagai satu entitas yang bermakna. Tanpa bigram, beberapa entitas penting terkait aksesibilitas yang terdiri dari lebih dari satu kata mungkin terabaikan atau terdeteksi secara salah. Penggunaan bigram memungkinkan pemrosesan yang lebih akurat dan kaya dalam analisis teks, terutama saat menangani ulasan yang mengandung kombinasi kata yang relevan dengan masalah aksesibilitas.

### 2.3. Pelabelan Manual

Setelah *preprocessing*, data perlu dilabeli secara manual untuk membedakan entitas yang ada dalam teks berdasarkan label entitas yang telah ditentukan. Pelabelan dikelompokkan menjadi empat label entitas yaitu Antarmuka (AM), Kemudahan Akses (KA), Keluhan (K), dan Fitur Aplikasi (FA). Tabel 2 menunjukkan label entitas dan kata kunci setiap label.

Label 'Antarmuka' digunakan untuk melabeli ulasan yang mengacu pada *user interface* (UI) aplikasi terutama bagaimana UI dapat memengaruhi pengalaman pengguna disabilitas. Label 'Kemudahan Akses' digunakan untuk mengidentifikasi ulasan yang membahas bahwa aplikasi mudah digunakan oleh pengguna dengan kebutuhan khusus. Label 'Keluhan' digunakan untuk memahami aspek aplikasi yang paling bermasalah khususnya dari perspektif pengguna disabilitas. Sedangkan Label 'Fitur Aplikasi' mengacu pada fitur-fitur spesifik yang ditawarkan oleh aplikasi untuk menciptakan ide fitur yang ramah disabilitas. Pemilihan label entitas dalam penelitian ini dilakukan berdasarkan relevansi topik dengan pengalaman pengguna yang menggunakan aplikasi transportasi online terutama dalam konteks pengguna disabilitas. Setiap label memiliki kata kunci yang digunakan sebagai *corpus* untuk melabeli data latihan.

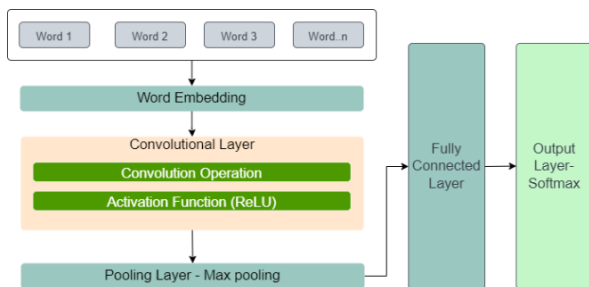
Tabel 2. Pemetaan Kode dan Kata Kunci Label Entitas

Label Entitas	Kata Kunci
AM	'navigasi', 'tombol', 'gesture kontrol', 'respon', 'tampilan', 'opsi layar', 'ukuran teks', 'teks kecil', 'baca', 'warna kontras', 'ikon', dst
KA	'aksesibilitas', 'layanan khusus', 'jangkau', 'peringat', 'dukungan pelanggan', 'mudah paham', 'prioritas', 'audio', 'layar sentuh', dst
K	'susah digunakan', 'lambat', 'keluhan', 'sulit akses', 'kesulitan akses', 'lemot', 'susah', 'sulit baca', 'parah', 'hilang', 'ribet', 'error', dst
FA	'fitur', 'update', 'pembaruan', 'pesan cepat', 'bayar elektronik', 'gopay', 'talkback', 'telepon', 'jemput', 'tunai', 'e-wallet', 'notifikasi', dst

### 2.4. NER Extraction Menggunakan Model CNN

Pada tahap ini, model CNN dilatih menggunakan data yang telah dilabeli. CNN bekerja dengan cara mendeteksi fitur dari kata-kata dalam teks berdasarkan hubungan spasial antar token yang berguna untuk mendeteksi entitas dalam kalimat. Proses pelatihan model CNN untuk NER dibagi menjadi beberapa tahapan, mulai dari *word embedding* untuk mengubah setiap token dari data *input* ke dalam bentuk representasi numerik agar dapat diproses oleh model CNN. Selanjutnya *convolutional layer* bertugas untuk mengidentifikasi pola spasial antar token yang menunjukkan entitas yang relevan. *Convolution* menerapkan filter untuk mendeteksi *n-gram* penting seperti ['sulit digunakan', 'mode gelap'] sebagai kandidat entitas penting. Setelah proses *convolution*, *pooling layer* bertugas mereduksi dimensi fitur yang dihasilkan. Tahap tersebut bertujuan untuk mengurangi dimensi dan menangkap fitur paling penting dari setiap kalimat. Keluaran dari pooling akan diratakan menjadi satu dimensi dan diproses di *Fully Connected Layer*. Pada tahap inilah, label entitas ditentukan untuk setiap token dalam kalimat. Setelah proses klasifikasi di *Fully Connected Layer*, *Output Layer* menghasilkan prediksi entitas NER untuk setiap kata dalam kalimat berdasarkan skor probabilitasnya. Setelah model dilatih, evaluasi dilakukan dengan menghitung *Accuracy*, *Precision*, *Recall*, dan *F1-Score* pada data

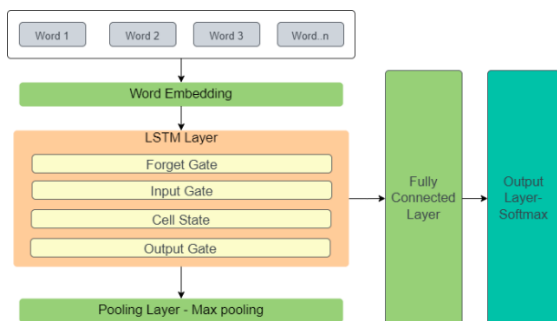
uji untuk memastikan bahwa model dapat mengenali entitas dengan baik. Tahapan proses pelatihan model CNN untuk NER ditunjukkan pada Gambar 2.



Gambar 2. Tahapan Proses Pelatihan Model CNN untuk NER

### 2.5. NER Extraction Menggunakan Model LSTM

Tahapan model LSTM hampir sama dengan model CNN. Langkah awal dimulai dengan tokenisasi terhadap kalimat dan mengonversi setiap token menjadi representasi numerik (*embedding*). LSTM memiliki memori jangka panjang dan pendek yang cocok untuk mendeteksi urutan kata dalam kalimat. Pada LSTM *layer* memproses setiap kata dalam urutan kalimat, memperhitungkan konteks antar kata. LSTM menjaga informasi urutan kata menggunakan *gates* (*input gate*, *forget gate*, *output gate*) sehingga bisa menangkap hubungan antar kata-kata sebelumnya dan kata saat ini, misalnya antara “ramah” dan “tunanetra”. Pada setiap kata dalam kalimat, LSTM memproses informasi dari kata sebelumnya dan memperbarui status tersembunyi serta memori. Hasil dari setiap urutan kata diteruskan ke *Fully Connected Layer* untuk memprediksi label entitas dari setiap token dalam komentar. Lapisan terakhir adalah *Output Layer* yang menggunakan fungsi *Softmax*. *Softmax* mengubah hasil dari *Fully Connected Layer* menjadi probabilitas untuk setiap entitas yang telah didefinisikan. Model LSTM juga dievaluasi menggunakan metrik *Accuracy*, *Precision*, *Recall*, dan *F1-Score* untuk mengukur kinerja model pada data uji. Tahapan proses pelatihan model LSTM untuk NER ditunjukkan pada Gambar 3.



Gambar 3. Tahapan Proses Pelatihan Model LSTM untuk NER

## 3. Hasil dan Pembahasan

### 3.1. Evaluasi Model CNN dan NER

Ekstraksi NER pada penelitian ini menggunakan dua pendekatan model pembelajaran mendalam, yaitu CNN

dan LSTM. Kedua model dievaluasi menggunakan skema *5-Fold Cross Validation* pada 3.000 data latih untuk menguji kinerja model dalam melakukan ekstraksi entitas pada teks. Tabel 3 berikut menunjukkan hasil evaluasi dari kedua model:

Tabel 3. Hasil Evaluasi Model CNN dan LSTM

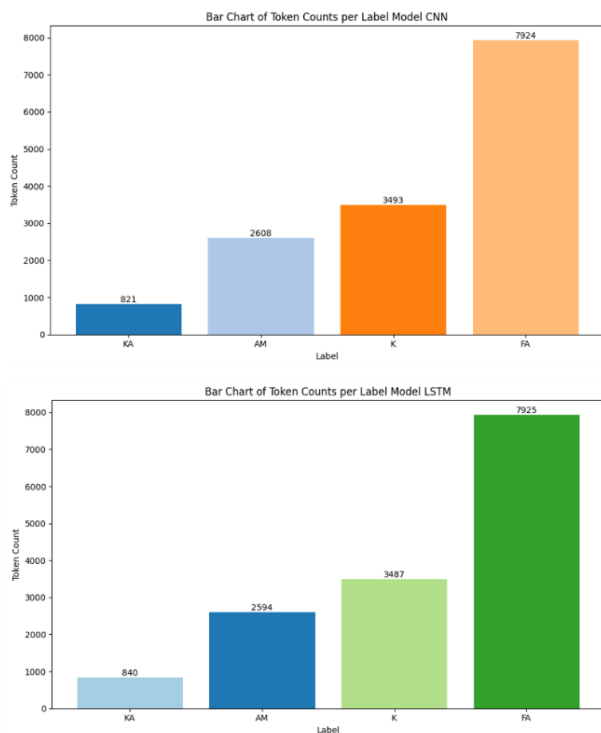
Model	Accuracy	Precision	Recall	F-1 Score
CNN	99,84%	74,87%	74,82%	74,84%
LSTM	99,48%	74,79%	74,44%	74,61%

Berdasarkan tabel tersebut, dapat dilihat bahwa CNN menunjukkan performa yang lebih unggul daripada LSTM dalam semua metrik meskipun selisihnya sangat kecil. Dari segi akurasi, kedua model menunjukkan kinerja yang sangat baik dalam hal ketepatan secara keseluruhan. Namun, akurasi yang tinggi tidak selalu mencerminkan kinerja yang optimal di semua metrik. Perbedaan ini disebabkan oleh ketidakseimbangan data, di mana sebagian besar token tidak mewakili entitas (termasuk dalam kelas "Other (O)"). Model dapat dengan mudah mencapai akurasi tinggi dengan memprediksi kelas mayoritas ini. Namun, *precision*, *recall*, dan *F1-score* lebih rendah karena model mengalami kesulitan mengenali entitas yang lebih jarang seperti nama orang atau lokasi. Akibatnya, performa untuk kelas yang lebih jarang menjadi lebih rendah, yang terlihat dari metrik lainnya. Oleh karena itu, meskipun akurasinya tinggi, *precision*, *recall*, dan *F1-score* memberikan gambaran yang lebih realistis tentang kemampuan model dalam mendeteksi entitas yang lebih kompleks. Secara keseluruhan, meski CNN lebih unggul dalam semua metrik evaluasi, LSTM tetap dapat digunakan sebagai model alternatif yang efisien karena perbedaannya tidak signifikan.

### 3.2. NER Extraction

Model CNN dan LSTM telah digunakan untuk mengklasifikasikan token ke dalam beberapa label entitas dalam proses ekstraksi NER, seperti Fitur Aplikasi, Keluhan, Antarmuka, dan Kemudahan Akses. Data ulasan yang diperoleh dari aplikasi transportasi *online* dianalisis secara keseluruhan karena keterbatasan jumlah data dan menjadi tidak seimbang ketika dibahas secara terpisah. Gambar 4 menunjukkan hasil klasifikasi token setiap label entitas pada model CNN dan LSTM.

Berdasarkan hasil klasifikasi kedua model menunjukkan bahwa distribusi token dalam setiap label konsisten, dimana label Fitur Aplikasi (FA) selalu mendominasi diikuti oleh Keluhan (K), Antarmuka (A), dan Kemudahan Akses (KA). Perbedaan utama di antara kedua model terletak pada jumlah token yang terklasifikasi dalam setiap label. Model CNN cenderung mengklasifikasikan jumlah token yang lebih tinggi untuk setiap label dibandingkan model LSTM, yang mungkin disebabkan oleh karakteristik arsitektur masing-masing model dalam memproses data urutan.



Gambar 4. Hasil Klasifikasi Token Setiap Label Entitas

Secara keseluruhan, model CNN menunjukkan performa yang lebih unggul dalam hal jumlah token yang terklasifikasi dengan benar, terutama untuk kategori FA dan K, yang merupakan dua kategori terpenting dalam ulasan aplikasi transportasi *online*. Namun, model LSTM juga memberikan hasil yang cukup baik, dengan distribusi token yang tetap konsisten meskipun jumlahnya sedikit lebih rendah dibandingkan CNN. Hal ini menunjukkan bahwa kedua model dapat digunakan untuk tugas NER dengan baik, meskipun CNN menunjukkan kecenderungan sedikit lebih akurat dalam menangkap entitas dari kategori yang lebih dominan.

### 3.3 Analisis Kemunculan Kata Setiap Label Entitas

Hasil klasifikasi dari kedua model selanjutnya diidentifikasi untuk melihat frekuensi kemunculan kata di setiap label entitas menggunakan *wordcloud*. Visualisasi *wordcloud* memberikan gambaran yang jelas mengenai isu-isu utama yang dihadapi pengguna aplikasi transportasi *online* terkait antarmuka, kemudahan akses, keluhan, dan fitur aplikasi, dengan fokus yang kuat pada aksesibilitas bagi pengguna disabilitas. Gambar 5 menunjukkan perbandingan kemunculan kata pada label entitas AM.

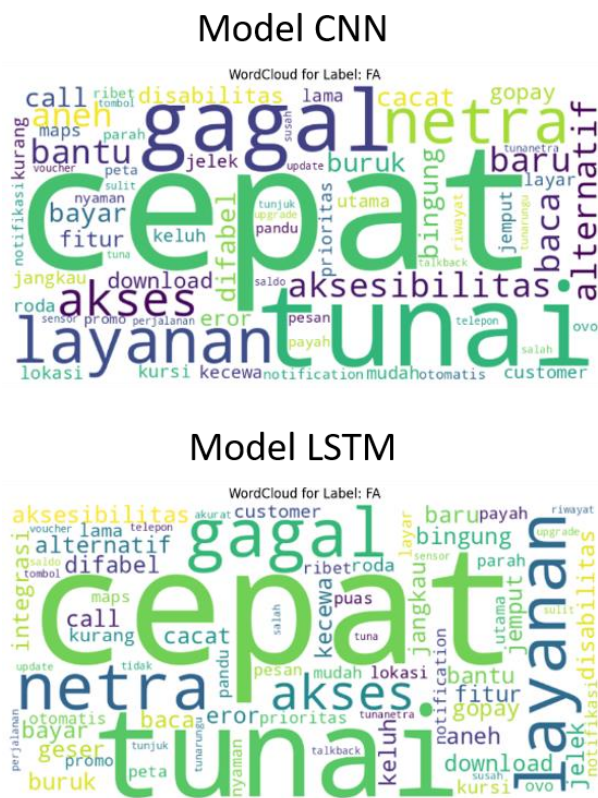


Gambar 5. *Wordcloud* Kemunculan Kata Label Entitas AM

Dapat dilihat dari kedua *wordcloud*, kemunculan kata kunci seperti "tuna", "netra", "bantu", "baca", dan "antarmuka" mendominasi. Ini menunjukkan bahwa kedua model fokus pada topik-topik yang relevan dengan aksesibilitas antarmuka bagi penyandang tunanetra. Pada *wordcloud* CNN terlihat lebih tersegmentasi dengan kata-kata yang lebih tersebar dan beragam, seperti "buruk", "kecewa", "fitur", "ikon", "layar", dan kata-kata lainnya yang berhubungan dengan antarmuka dan tanggapan pengguna. Sementara itu, *wordcloud* LSTM cenderung menangkap kata-kata yang lebih umum seperti bantu, *feedback*, disabilitas, fitur, cacat, yang menunjukkan bahwa LSTM lebih fokus pada representasi pengalaman pengguna secara keseluruhan.

Kemunculan kata pada label entitas KA ditunjukkan pada Gambar 6. Model CNN menekankan kata-kata seperti "difabel," "cacat," "netra," dan "tuna", menandakan bahwa kata-kata ini sering muncul dan sangat relevan dalam label ini. Kata-kata seperti "fisik," "bantu," dan "error" juga muncul, mencerminkan tantangan akses. Model LSTM menekankan kata-kata serupa, namun "bantu," "call," dan "puas" lebih menonjol dibanding CNN.





Gambar 8. Wordcloud Kemunculan Kata Label Entitas FA

Secara keseluruhan, model CNN dan LSTM dalam penelitian ini berhasil mengekstraksi informasi dari ulasan pengguna aplikasi transportasi online. Pada label antarmuka, informasi yang dihasilkan berkaitan dengan tantangan aksesibilitas, terutama masalah visual, di mana pengguna, khususnya penyandang tunanetra, sering memberikan tanggapan negatif. Pada label keluhan, ditemukan bahwa masalah performa aplikasi menjadi keluhan utama. Label kemudahan akses mencerminkan hambatan yang dihadapi pengguna dengan keterbatasan fisik. Sementara itu, label fitur aplikasi menekankan pentingnya kecepatan layanan dan metode pembayaran. Hasil tersebut dapat digunakan untuk meningkatkan pengalaman pengguna, terutama dalam hal aksesibilitas dan efisiensi aplikasi.

#### 4. Kesimpulan

Dari penelitian ini, model CNN dan LSTM berhasil diterapkan untuk implementasi NER dalam mengekstraksi informasi dari ulasan aplikasi transportasi online. Kedua model menunjukkan akurasi yang tinggi yaitu CNN 99.84%, dan LSTM: 99.48%, namun masih memiliki kelemahan dalam mendeteksi entitas yang jarang muncul atau memiliki konteks kompleks. Metrik *Precision*, *Recall*, dan *F1-Score* untuk kedua model berkisar di angka 74%, menunjukkan bahwa meskipun akurasi tinggi, model perlu peningkatan untuk deteksi entitas secara lebih komprehensif. Hasil visualisasi kemunculan kata menunjukkan bahwa baik CNN maupun LSTM dapat menangkap kata-kata relevan dalam kategori seperti

fitur aplikasi, keluhan, antarmuka, dan kemudahan akses. Namun, CNN cenderung lebih baik dalam menangkap pola visual yang lebih spesifik, sementara LSTM lebih berfokus pada variasi kata yang lebih luas dalam memahami pengalaman pengguna. Penelitian selanjutnya disarankan untuk mengembangkan model yang mampu mendeteksi entitas dengan kemunculan teks yang kompleks. Hal ini bisa dilakukan dengan menambah jumlah data yang lebih beragam serta memastikan bahwa distribusi konteks teks dalam label-label tersebut seimbang.

#### Daftar Rujukan

- [1] Adi Ahdiat, "Gojek vs Grab, Mana yang Konsumennya Lebih Banyak?," *databoks.katadata.co.id*, 2022. <https://databoks.katadata.co.id/datapublish/2022/12/08/gojek-vs-grab-mana-yang-konsumennya-lebih-banyak>.
- [2] Kementerian Koordinator Bidang Pembangunan Manusia dan Kebudayaan RI, "Pemerintah Penuhi Hak Penyandang Disabilitas di Indonesia," <https://www.kemendikbud.go.id/>, 2023. <https://www.kemendikbud.go.id/pemerintah-penuhi-hak-penyandang-disabilitas-di-indonesia> (accessed Nov. 19, 2024).
- [3] S. D. Poerwanti, S. Makmun, and A. D. Dewantara, "Jalan Panjang Menuju Inklusi Digital bagi Penyandang Disabilitas di Indonesia," *J. Urban Sociol.*, vol. 7, no. 1, p. 44, 2024, doi: 10.30742/jus.v1i1.3536.
- [4] C. Matulesy and A. Purnama, "Analisis Sentimen Terhadap Review Pengguna Indrive Di Google Playstore Menggunakan Algoritma Support Vector Machine," *J. Darma Agung*, vol. 31, no. 1, p. 1015, 2023, doi: 10.46930/ojsuda.v31i1.3078.
- [5] P. Kurniawati, R. Y. Fa'rifah, and D. Witasryah, "Sentiment Analysis of Maxim Online Transportation App Reviews using Support Vector Machine (SVM) Algorithm," *Build. Informatics, Technol. Sci.*, vol. 5, no. 2, pp. 466–475, 2023, doi: 10.47065/bits.v5i2.4265.
- [6] D. S. Annam, A. Hananto, F. Nurapriani, and T. Tukino, "Clustering User Sentiment Transportasi Online Gojek Dan Grab Dengan Metode K-Means," *J. Tika*, vol. 8, no. 2, pp. 164–171, 2023, doi: 10.51179/tika.v8i2.2165.
- [7] D. S. Ashari, B. Irawan, and C. Setianingsih, "Sentiment Analysis on Online Transportation Services Using Convolutional Neural Network Method," *Int. Conf. Electr. Eng. Comput. Sci. Informatics*, pp. 335–340, 2021, doi: 10.23919/EECSI53397.2021.9624261.
- [8] I. D. F.- IxDF, "What is Accessibility?," *Interaction Design Foundation - IxDF*, 2016. <https://www.interaction-design.org/literature/topics/accessibility>.
- [9] F. S. Gharehchopogh and Z. A. Khalifelu, "Analysis and evaluation of unstructured data: Text mining versus natural language processing," *2011 5th Int. Conf. Appl. Inf. Commun. Technol. AICT 2011*, pp. 7–10, 2011, doi: 10.1109/ICAICT.2011.6111017.
- [10] K. A. Rokhman, B. Berlilana, and P. Arsi, "Perbandingan Metode Support Vector Machine Dan Decision Tree Untuk Analisis Sentimen Review Komentar Pada Aplikasi Transportasi Online," *J. Inf. Syst. Manag.*, vol. 3, no. 1, pp. 1–7, 2021, doi: 10.24076/joism.2021v3i1.341.
- [11] S. Kale and S. Govilkar, "Survey of Named Entity Recognition Techniques for Various Indian Regional Languages," *Int. J. Comput. Appl.*, vol. 164, no. 4, pp. 37–43, 2017, doi: 10.5120/ijca2017913621.
- [12] Github, "Mobile Content Accessibility Guidelines (MCAG)," [getvined.github.io](https://getvined.github.io/), 2024. <https://getvined.github.io/mcag/about>.

- [13] R. R. Sedik and A. D. E. Romadhony, "Information Extraction from Indonesian Crime News with Named Entity Recognition," *15th Int. Conf. Knowl. Smart Technol. KST 2023*, pp. 1–5, 2023, doi: 10.1109/KST57286.2023.10086789.
- [14] M. S. Hossain, N. Nayla, and A. A. Rassel, "Product Market Demand Analysis Using Nlp in Banglish Text With Sentiment Analysis and Named Entity Recognition," *2022 56th Annu. Conf. Inf. Sci. Syst. CISS 2022*, pp. 166–171, 2022, doi: 10.1109/CISS53076.2022.9751188.
- [15] M. Y. S. Dirgantara, M. A. Fauzi, and R. S. Perdana, "Penerapan Named Entity Recognition Untuk Mengenali Fitur Produk Pada Ecommerce Menggunakan Rule Template Dan Naive Bayes ...," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 2, no. 10, pp. 3912–3920, 2018, [Online]. Available: <https://rama.unimal.ac.id/id/eprint/273/>.
- [16] M. R. Haque, S. Akter Lima, and S. Z. Mishu, "Performance Analysis of Different Neural Networks for Sentiment Analysis on IMDb Movie Reviews," *3rd Int. Conf. Electr. Comput. Telecommun. Eng. ICECTE 2019*, pp. 161–164, 2019, doi: 10.1109/ICECTE48615.2019.9303573.
- [17] S. Gunathilaka and N. De Silva, "Aspect-based Sentiment Analysis on Mobile Application Reviews," *22nd Int. Conf. Adv. ICT Emerg. Reg. ICTer 2022*, pp. 183–188, 2022, doi: 10.1109/ICTer58063.2022.10024070.
- [18] A. Sri Widagdo, A. Ardiansyah, K. N. Qodri, F. E. Nugroho, and N. A. Rizky, "Analisis Sentimen Mobil Listrik di Indonesia Menggunakan Long-Short Term Memory (LSTM)," *J. Fasilkom*, vol. 13, no. 3, pp. 416–423, 2023, doi: 10.37859/jf.v13i3.6303.
- [19] A. A. N. A. Nugraha and I. B. M. Mahendra, "Implementasi Long-Short Term Memory (LSTM) pada Klasifikasi Kategori Berita," *JELIKU (Jurnal Elektron. Ilmu Komput. Udayana)*, vol. 12, no. 3, pp. 563–568, 2023, [Online]. Available: <https://ojs.unud.ac.id/index.php/JLK/article/view/92566>.
- [20] Yudi Widhiyasana, Transmissia Semiawan, Ilham Gibran Achmad Mudzakir, and Muhammad Randi Noor, "Penerapan Convolutional Long Short-Term Memory untuk Klasifikasi Teks Berita Bahasa Indonesia," *J. Nas. Tek. Elektro dan Teknol. Inf.*, vol. 10, no. 4, pp. 354–361, 2021, doi: 10.22146/jnteti.v10i4.2438.
- [21] A. Cardova and A. Hermawan, "Implementasi Metode LSTM Untuk Mengklasifikasi Berita Palsu Pada PolitiFact," *J. Fasilkom*, vol. 13, no. 3, pp. 471–479, 2023, doi: 10.37859/jf.v13i3.6175.
- [22] R. A. Sunan, H. F. E. K., and C. S. K. Aditya, "Klasifikasi Hoax Berita Politik Menggunakan Algoritma Long Short-Term Memory (LSTM) dengan Penambahan Fitur Embedding Global Vector (GloVe)," *J. Edukasi dan Penelit. Inform.*, vol. 10, no. 2, p. 287, 2024, doi: 10.26418/jp.v10i2.76042.
- [23] I. Dwijayanti, M. Habibi, K. Kusumaningtyas, and S. Riyadi, "Tweets Classification of Mental Health Disorder in Indonesia Using LDA and Cosine Similarity," *Telemat. J. Inform. dan Teknol. Inf.*, vol. 21, no. 1, pp. 41–54, 2024, doi: 10.31515/telematika.v21i1.10725.
- [24] B. Zaman, E. Hariyanti, and E. Purwanti, "Sistem Deteksi Bahasa pada Dokumen menggunakan N-Gram," *J. Multinetics*, vol. 1, no. 2, p. 21, 2015, doi: 10.32722/vol1.no2.2015.pp21-26.