

Sentiment-Based Knowledge Discovery pada Aplikasi iPusnas Menggunakan Metode Machine Learning dan Deep Learning

Pratiwi Ayuningtiyas¹, Ken Ditha Tania^{2*}, Winda Kurnia Sari³

^{1,2,3} Sistem Informasi, Universitas Sriwijaya

pratiwiayuningtiyas04@gmail.com¹, kanya.tania@gmail.com², windakurniasari@unsri.ac.id³

Article Info

Article history:

Received 2025-07-16

Revised 2025-08-15

Accepted 2025-09-03

Keyword:

*Sentiment Analysis,
Deep Learning,
iPusnas,
Knowledge Discovery,
Machine Learning.*

ABSTRACT

iPusnas is a digital library application developed by the National Library of the Republic of Indonesia since 2016, with over 1.5 million users. Despite its potential to improve literacy, the application has only received a rating of 2.0. This study conducted sentiment analysis on 7.596 reviews obtained through web scraping using the Google Play Scraper Library. The data then underwent preprocessing steps including case folding, data cleaning, tokenization, stopword removal, and stemming. Reviews were automatically labeled based on the rating score, where scores of 1-3 were categorized as negative, with 5.174 entries, and scores 4-5 as positive, with 2.422 entries. The dataset was split in an 80:20 ratio, with 80% for training, and 20% for testing. The machine learning models tested were SVM, Random Forest, CNN, LSTM, and RNN. The evaluation metrics included accuracy, precision, recall, F1-score, and confusion matrix. CNN and LSTM achieved the highest accuracy (82%), Random Forest and CNN achieved the highest precision (81%), RNN the highest recall (79%) and LSTM the highest F1-score (79%). McNemar test showed a significant difference between Random Forest and CNN, Random Forest and LSTM, and between RNN and LSTM, while CNN and LSTM, as well as CNN and RNN, showed no significant difference.



This is an open access article under the [CC-BY-SA](#) license.

I. PENDAHULUAN

Perpustakaan berfungsi menyimpan, mengelola, dan menyediakan berbagai buku sebagai sumber informasi bagi masyarakat [1]. Undang-Undang Nomor 43 Tahun 2007 tentang Perpustakaan, menegaskan peran strategis perpustakaan dalam mencerdaskan kehidupan bangsa [2]. Seiring perkembangan teknologi informasi, perpustakaan dituntut untuk beradaptasi agar tetap diminati dan relevan bagi masyarakat. Hal ini tercermin dalam pergeseran perpustakaan konvensional yang sebelumnya hanya dapat diakses secara fisik, telah bertransformasi menjadi perpustakaan digital yang dapat diakses oleh pengguna dari berbagai lokasi dan waktu secara bebas [3]. Salah satu bentuk transformasi tersebut adalah hadirnya iPusnas, aplikasi perpustakaan digital yang dikembangkan oleh Perpustakaan Nasional Republik Indonesia (PNRI) sejak tahun 2016. Layanan ini menyediakan akses bacaan dan informasi digital secara gratis melalui perangkat seluler, serta telah diakses

oleh lebih dari 1,5 juta pengguna sejak diluncurkan [4]. Fungsionalitas iPusnas yang tersedia di *smartphone*, berbasis web (*web-based*), hibrida berbasis tablet (*tab-based hybrid*), memudahkan akses bagi pengguna di daerah terpencil atau jauh dari perpustakaan, sekaligus mendukung peralihan dari bacaan ilegal ke publikasi resmi [5].

Keberadaan aplikasi iPusnas menunjukkan komitmen pemerintah untuk meningkatkan literasi dan minat baca di Indonesia. Pengalaman pengguna terhadap layanan iPusnas menghasilkan berbagai tanggapan yang terekam dalam bentuk ulasan, baik positif maupun negatif. Tanggapan tersebut dituangkan melalui ulasan pada platform Google Play Store [6] Per Mei 2025, aplikasi iPusnas memiliki penilaian sebesar 2,0 dengan jumlah ulasan mencapai lebih dari 33,78 ribu. Besarnya jumlah ulasan membutuhkan pemrosesan otomatis yang efektif untuk memperoleh informasi. Oleh karena itu, analisis sentimen diterapkan untuk mengenali dan mengklasifikasikan aspek emosional dalam konten ulasan pengguna [7][8]. Dalam praktiknya, analisis

sentimen merupakan bagian dari proses *knowledge discovery*, yaitu pendekatan sistematis untuk mengekstraksi pengetahuan tersembunyi dari berbagai basis data [9]. Besarnya volume data yang dihasilkan dari interaksi pengguna dan aktivitas digital menggambarkan pentingnya peran *knowledge discovery* dalam mengolah informasi tidak terstruktur menjadi pengetahuan yang bernilai [10]. Proses *knowledge discovery* sendiri membutuhkan kontribusi dari berbagai bidang keilmuan serta penguasaan teknologi pengolahan data dan algoritma komputasi untuk menggali pengetahuan yang relevan dari data yang kompleks [11].

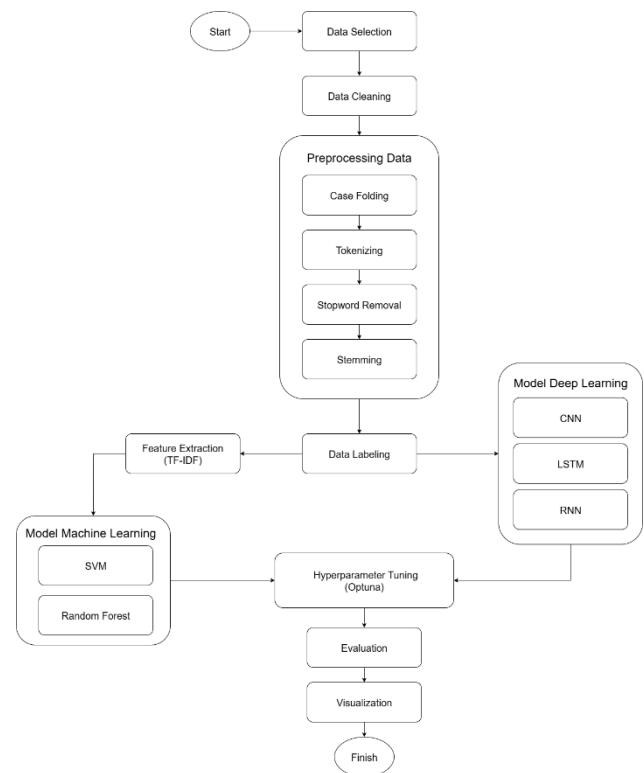
Metode *machine learning* dan *deep learning* menjadi pendekatan yang banyak digunakan untuk tugas-tugas kompleks, termasuk klasifikasi sentimen [12]. Metode *machine learning* diterapkan dalam studi [13] menerapkan algoritma SVM dan *Naïve Bayes* untuk menilai pandangan masyarakat Indonesia terhadap dampak resesi global pada aplikasi twitter. Penelitian ini mendapatkan akurasi terbesar pada model SVM sebesar 79,5% dan disusul *Naïve Bayes* sebesar 72,5%. Selanjutnya, penelitian [14] memanfaatkan model *Random Forest* dan *Word2Vec* terhadap ulasan film Indonesia, akurasi yang didapatkan sebesar 83,59%. Pada pendekatan *deep learning* penelitian [15] mengaplikasikan LSTM pada kasus berita online pariwisata Inggris berbahasa Inggris dan memperoleh akurasi sebesar 81,36%. Penelitian [16] menggunakan CNN untuk menilai saran mahasiswa terhadap kinerja departemen di perguruan tinggi, mencatat akurasi 98%. Sementara itu, penelitian [17] menilai pengaruh negasi dalam kalimat terhadap analisis sentimen dan deteksi polaritas dengan *Naïve Bayes*, SVM, *Artificial Neural Network* (ANN), dan *Recurrent Neural Network* (RNN), penelitian ini memperlihatkan performa tertinggi terdapat pada model RNN dengan akurasi sebesar 95,67%.

Berdasarkan temuan-temuan dari penelitian sebelumnya, algoritma SVM, Random Forest, LSTM, CNN, dan RNN sama-sama menunjukkan kinerja yang baik dan konsisten dalam melakukan klasifikasi sentimen pada berbagai konteks. Dengan demikian, penelitian ini menerapkan kelima pendekatan tersebut untuk menganalisis sentimen ulasan pengguna aplikasi iPusnas. Tujuan utama dari penelitian ini adalah untuk melihat persepsi pengguna terhadap iPusnas dan mengevaluasi akurasi setiap model sebagai tolak ukur efektivitas klasifikasi, sehingga diharapkan mampu memberikan wawasan secara mendalam mengenai pengalaman pengguna terhadap aplikasi iPusnas.

II. METODE

Penelitian ini diawali dengan tahapan *data selection*, yaitu pengambilan data ulasan pengguna aplikasi iPusnas yang dikumpulkan melalui platform Google Play Store. Setelah data terkumpul, dilakukan *preprocessing* yang terdiri dari lima tahapan yaitu *case folding*, *data cleaning*, *tokenizing*, *stemming*, dan *stopword removal*. Data yang telah diproses kemudian diberi label sentimen, yaitu positif dan negatif. Selanjutnya, data dibagi menjadi dua bagian, yaitu data

pelatihan dan data pengujian. Pada tahap pemodelan, digunakan dua pendekatan yaitu *machine learning* dan *deep learning*. Untuk pendekatan *machine learning*, algoritma yang digunakan adalah *Support Vector Machine* (SVM) dan *Random Forest*. Sebelum masuk ke tahap pemodelan, dilakukan proses tokenisasi menggunakan metode *Term Frequency-Inverse Document Frequency* (TF-IDF) untuk merepresentasikan data teks dalam bentuk numerik berbobot, agar dapat diproses oleh algoritma klasifikasi [18]. Sementara itu, pendekatan *deep learning* dilakukan dengan tiga model yaitu *Convolutional Neural Network* (CNN), *Long Short-Term Memory* (LSTM), dan *Recurrent Neural Network* (RNN). Seluruh model kemudian dioptimasi menggunakan Optuna untuk meningkatkan performa model dan memaksimalkan akurasi dari setiap model. Tahap terakhir dari proses ini adalah evaluasi terhadap performa masing-masing model. Evaluasi kinerja model dilakukan berdasarkan metrik *accuracy*, *precision*, *recall*, *F1-score*, dan *confusion matrix*, yang digunakan untuk mengidentifikasi model yang memberikan hasil paling optimal. Selain itu, dilakukan juga visualisasi data menggunakan *word cloud*. Alur penelitian dapat terlihat pada Gambar 1.



Gambar 1. Alur penelitian

A. Data Selection

Tahap pertama dari penelitian ini berupa proses pemilihan data, yang bertujuan untuk memperoleh informasi yang relevan sesuai kebutuhan analisis [19]. Sumber data adalah ulasan pengguna aplikasi iPusnas yang tersedia di Google Play Store. Pengumpulan data dilakukan melalui Google

Collaboratory dengan menggunakan library Google Play Scraper. Data diambil dari periode 1 Mei 2021 hingga 31 Mei 2025. Proses ini menghasilkan 7.650 dataset, yang disimpan dalam format Microsoft Excel untuk analisis. Contoh data ulasan yang diperoleh disajikan pada Tabel I.

TABEL I
DATA ULASAN

userName	content	score
novi a. puspita	jadi sering gangguan sih Min.. dan tiap buka, mesti unduh lagi. Pdhl udah unduh sebelumnya. mendingan yg sebelumnya deh apk ipusnas	1
novalina panja putri	Kemarin tidak bisa dibuka. Akhirnya muncul dengan tampilan baru dan buku2 yang lebih lengkap dan update. Terima kasih ipusnas.	5

B. Preprocessing Data

Tahap selanjutnya adalah pra-pemrosesan data, yang melibatkan beberapa tahapan subproses, yaitu *case folding*, *data cleaning*, *tokenizing*, *stopword removal*, dan *stemming* [20]. Langkah ini memegang peranan penting dalam pelaksanaan analisis sentimen, karena menentukan kualitas data yang akan diolah pada tahap selanjutnya untuk memperoleh informasi yang bernilai [21]

1). *Case Folding*: Proses normalisasi teks yang mengubah semua karakter dalam dokumen menjadi huruf kecil. Sementara, karakter lainnya dianggap sebagai delimiter atau pembatas [22]. Proses ini dilakukan menggunakan fungsi `.str.lower()` dari pustaka pandas yang diterapkan langsung pada setiap baris teks. Langkah ini dilakukan agar kata yang sama namun berbeda dalam penulisan kapital, seperti “Bagus” dan “bagus” tidak dianggap sebagai dua entitas berbeda dalam analisis.

2). *Data Cleaning*: Pembersihan Data adalah proses menghilangkan elemen-elemen yang tidak akurat, berlebihan, tidak konsisten, dan tidak lengkap dari suatu dataset. Tujuan utama dari fase ini adalah mengubah data mentah yang berisi informasi yang tidak relevan atau bermasalah menjadi data bersih yang dapat mendukung proses analisis secara lebih akurat dan optimal [23].

Pada tahap ini dilakukan beberapa langkah pembersihan data. Pertama, menghapus kolom yang tidak berkaitan langsung dengan analisis sentimen. Kolom-kolom tersebut meliputi `reviewId`, `userName`, `userImage`, `thumbsUpCount`, `reviewCreatedVersion`, `replyContent`, `repliedAt`, `appVersion`. Proses penghapusan ini dilakukan secara langsung pada *DataFrame* untuk menjaga konsistensi data yang akan dianalisis. Selanjutnya menghapus tautan (URL) menggunakan *regular expression* dengan pola `https://|S+|www\.\S+`, sehingga seluruh teks yang mengandung tautan seperti `https://` atau `www.` akan dihapus dari data. Teks yang mengandung hastag juga dihapus

menggunakan *regular expression* `#[A-Za-z0-9]+`. Dengan pola ini simbol hastag berserta kata setelahnya akan dihapus. Angka umumnya tidak membawa konteks sentimen, sehingga seluruh angka dalam teks dihapus dengan *regular expression* `[0-9]+`. Karakter khusus seperti subskrip akan dihapus menggunakan *regular expression* `\u2070-\u209F`. Langkah ini bertujuan untuk mengurangi *noise* pada data. Seluruh tanda baca seperti titik, koma, tanda seru, dan lainnya dihapus menggunakan fungsi `translate()` dari python dengan modul `string.punctuation`. Terakhir, menghapus kata-kata non alfabet, proses ini dilakukan menggunakan pola *regular expression* `^[a-zA-Z]+\$` untuk menyaring setiap kata, sehingga kata yang mengandung simbol atau karakter non-alfabet lainnya akan dihapus.

3). *Tokenizing*: Proses membagi kalimat atau teks menjadi bagian-bagian yang lebih kecil, biasa disebut term atau token. Pembagian ini dilakukan berdasarkan spasi kata [24]. Proses tokenisasi dalam penelitian ini dilakukan menggunakan fungsi `word_tokenize()` dari library *Natural Language Toolkit* (NLTK). Fungsi ini akan memecah kalimat menjadi daftar kata berdasarkan pemisah seperti spasi dan tanda baca.

4). *Stopword Removal*: *Stopword* merupakan kata-kata yang memiliki makna umum, tidak membawa informasi penting, dan tidak mengandung sentimen, namun sering muncul dalam jumlah besar pada teks. Proses *stopword removal* dilakukan untuk mengurangi dimensionalitas data, menurunkan beban komputasi, serta meningkatkan efisiensi dan kinerja analisis [25]. Dalam penelitian ini, proses *stopword removal* dilakukan menggunakan library *Natural Language Toolkit* (NLTK), yang menyediakan daftar *stopwords* bahasa indonesia melalui modul `nltk.corpus.stopwords`. Daftar tersebut kemudian akan dikonversi ke dalam struktur data `set()` untuk mempercepat proses pencocokan. Selanjutnya, data teks yang telah melalui tahap tokenisasi diproses menggunakan fungsi `remove_stopwords()`, yang mengevaluasi setiap token dan menghapus kata-kata yang termasuk dalam daftar *stopwords*. Hasil akhir berupa token yang telah bersih datu siap digunakan untuk tahap berikutnya.

5). *Stemming*: *Stemming* adalah proses mengubah kata-kata dalam sebuah ulasan menjadi bentuk kata dasarnya (akar kata) dengan menerapkan aturan-aturan tertentu [26]. Proses stemming dilakukan dengan library python Sastrawi yang merupakan pustaka yang digunakan untuk menghasilkan kata dasar dari afiks dalam bahasa indonesia [27]. Setelah data teks melewati tahap tokenisasi dan *stopword removal*, kumpulan kata-kata tersebut digabung menjadi kalimat utuh. Kalimat ini kemudian diproses oleh algoritma stemming dari Sastrawi, yang secara otomatis mengubah setiap kata menjadi bentuk dasar yang sesuai aturan bahasa Indonesia.

Contoh hasil dari setiap tahapan dalam proses preprocessing data disajikan pada tabel II untuk memberikan gambaran perubahan data mentah menjadi format yang siap untuk dianalisis lebih lanjut.

TABEL II
HASIL PREPROCESSING DATA

Proses	Hasil
Case Folding	kemarin tidak bisa dibuka. akhirnya muncul dengan tampilan baru dan buku2 yang lebih lengkap dan update. terima kasih ipusnas.
Data Cleaning	kemarin tidak bisa akhirnya muncul dengan tampilan baru dan buku yang lebih lengkap dan terima kasih
Tokenizing	kemarin,tidak,bisa,akhirnya,muncul,dengan,tampilan,baru,dan,buku,yang,lebih,lengkap,dan,terima,kasih
Stopword Removal	kemarin,muncul,tampilan,buku,lengkap,terima,kasih
Stemming	kemarin,muncul,tampil,buku,lengkap,terima,kasih

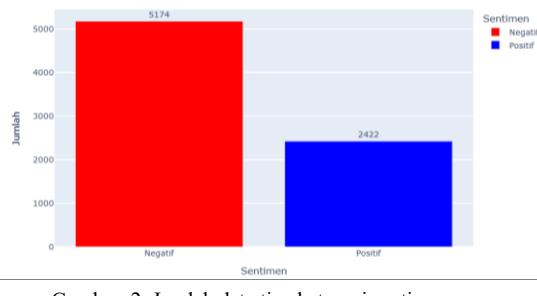
C. Data Labeling

Proses pelabelan data dilakukan untuk mendukung tahap klasifikasi [28]. Skor ulasan dikategorikan ke dalam dua kelompok sentimen yang berbeda. Proses kategorisasi sentimen didasarkan pada nilai *score* di mana, *score* 1 sampai 3 diklasifikasikan sebagai sentimen negatif, sedangkan skor lebih dari 3 diklasifikasikan sebagai sentimen positif.

TABEL III
JUMLAH DATA PADA TIAP RATING

Rating	Jumlah
1	3052
2	1018
3	1104
4	701
5	1721
Total	7596

Berdasarkan tabel III dari total 7.596 data, diperoleh 5.174 data dengan kelas negatif dan 2.422 data dengan kelas positif. Distribusi jumlah data pada masing-masing kategori sentimen ditampilkan pada Gambar 2.

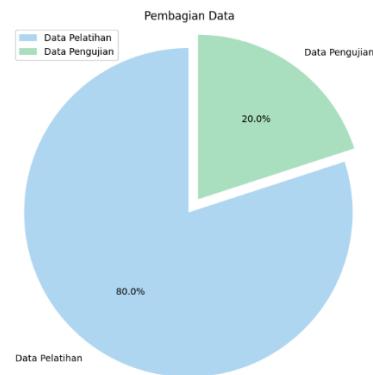


Gambar 2. Jumlah data tiap kategori sentimen

D. Split Data

Setelah tahap *preprocessing* selesai dilakukan pemisahan data yang merupakan metode umum untuk membagi data menjadi dua bagian atau lebih yang membentuk suatu subset data, proses pemisahan data biasanya membagi dataset

menjadi dua bagian, yaitu data pelatihan dan data pengujian [29]. Hal ini dilakukan untuk memverifikasi akurasi model dan memverifikasi akurasi model sebelum digunakan [30]. Pada penelitian ini dataset yang digunakan terdiri dari 7.596 data akan dibagi menjadi dua bagian, yaitu data pelatihan dan data pengujian, dengan perbandingan 80:20. Sebanyak 5.174 data (80%) digunakan sebagai data untuk melatih model, sedangkan 2.422 data (20%) digunakan untuk menguji performa model, pembagian data dapat terlihat pada gambar 3.



Gambar 3. Distribusi data latih dan data uji

E. Model Machine Learning

Pada pendekatan *machine learning*, penelitian ini menerapkan dua algoritma klasifikasi utama, yaitu *Support Vector Machine* (SVM) dan *Random Forest*, serta memanfaatkan metode TF-IDF untuk mentransformasikan data teks menjadi bentuk numerik yang dapat diolah oleh algoritma tersebut.

1). *Support Vector Machine (SVM)*: Salah satu algoritma klasifikasi data berbasis supervised learning yang efektif dalam menangani data berdimensi tinggi dan bersifat nonlinier [31]. Ide utama dari metode ini adalah melakukan klasifikasi dengan menentukan hyperplane optimal yang dapat memisahkan dua kelas secara jelas [32]. SVM berupaya untuk memaksimalkan margin, yaitu jarak antara titik data terdekat dari setiap kelas dengan hyperplane. Hyperplane ditentukan berdasarkan nilai bobot margin yang telah dioptimalkan. Hyperplane dapat dideskripsikan sebagai [33] :

$$w \cdot x + b = 0 \quad (1)$$

Dimana w merupakan vektor bobot, x adalah vektor input, dan b menunjukkan nilai bias [33].

2). *Random Forest*: Algoritma *Random Forest* adalah metode ensemble yang mengkombinasikan beberapa pohon keputusan (*decision tree*). Setiap pohon dalam *Random Forest* dibangun sebagai vektor acak yang didistribusikan secara identik dan independen [34]. Metode ini memiliki tiga komponen utama. Pertama, bootstrapping sampling digunakan untuk membangun setiap pohon prediksi dari

subset data yang diambil secara acak dengan pengembalian. Kedua, pemilihan prediktor dilakukan secara acak pada setiap node dalam pohon untuk meningkatkan diversifikasi model. Ketiga, hasil akhir dari *Random Forest* diperoleh dengan menggabungkan prediksi dari seluruh pohon keputusan yang terbentuk, baik melalui mekanisme voting mayoritas untuk tugas klasifikasi maupun rata-rata untuk regresi [35].

3). *Feature Extraction TF-IDF*: TF-IDF adalah salah satu metode untuk mengukur signifikansi matematis kata-kata dalam dokumen. Bobot yang tinggi akan diberikan pada kata-kata yang sering muncul dalam banyak dokumen. Metode ini dimanfaatkan untuk mengukur tingkat kepentingan suatu kata dalam dokumen [36]. TF-IDF merepresentasikan frekuensi relatif kata t dalam dokumen teks, dan skala frekuensi dokumen terbalik dengan jumlah dokumen, ia dapat memproses sebagai berikut :

$$W_{d,t} = tf \times \log \left(\frac{N}{df} \right) \quad (2)$$

Dimana w menunjukkan nilai bobot untuk term t , dokumen d , jumlah total dokumen dalam korpus didefinisikan sebagai N , tf dan df menunjukkan frekuensi term yang menunjukkan jumlah kali dan jumlah dokumen dalam term tertentu masing masing [37].

F. Model Deep Learning

Pada pendekatan *deep learning*, penelitian ini mengimplementasikan tiga model utama, yaitu *Convolutional Neural Network* (CNN), *Long Short-Term Memory* (LSTM), dan *Recurrent Neural Network* (RNN).

1). *Convolutional Neural Network (CNN)*: Jenis jaringan saraf maju yang digunakan untuk mengekstraksi fitur-fitur utama dari konteks teks [38]. Secara umum, arsitektur CNN terdiri dari tiga lapisan utama, yaitu (1) *input layer*, yang berfungsi untuk menerima data masukan berupa vektor fitur, (2) *hidden layer*, yang terdiri dari lapisan yang disembunyikan berdasarkan model dan ukuran data, (3) *output layer*, yang menghasilkan prediksi akhir berdasarkan kelas target [39]. CNN bekerja dengan mengalirkan data multidimensi seperti gambar, representasi kata, atau penyisipan kata ke dalam lapisan konvolusional yang mengandung sejumlah filter. Filter ini diterapkan secara bertahap untuk mempelajari berbagai aspek dari input secara berbeda. Model ini memanfaatkan *pooling layer* (lapisan pengumpulan), dimana output akan dibagi lagi untuk menghasilkan bagian yang lebih kecil [40].

Model CNN pada penelitian ini terdiri atas tujuh lapisan. Pertama, digunakan embedding layer dengan dimensi output 64 untuk merepresentasikan kata dalam bentuk vector. Selanjutnya, terdapat Conv1D layer dengan jumlah filter sebesar 64 atau 128 dan kernel size 5. Regulasi L2 diterapkan pada lapisan ini dengan nilai koefisien antara 1e-4 hingga 1e-2. Batch Normalization digunakan setelah Conv1D agar distribusi aktivasi stabil. Selanjutnya diterapkan

GlobalMaxPooling1D, dropout dengan Tingkat antara 0.3 hingga 0.5. Lapisan dense dengan 64 neuron dan aktivasi ReLu digunakan sebelum menuju output layer. Lapisan output berupa dense layer dengan dua neuron dan aktivasi softmax untuk klasifikasi dua kelas. Model dioptimasi menggunakan adam dengan learning rate antara 1e-4 hingga 1e-2, menggunakan loss function sparse_categorical_crossentropy. Pelatihan dilakukan dengan maksimal 10 epoch serta early stopping (patience=2).

2). *Long Short-Term Memory (LSTM)*: Salah satu jenis dari *Recurrent Neural Network* (RNN). Metode LSTM digunakan untuk mengklasifikasikan data jangka panjang dengan menyimpannya dalam sel memori [41]. Terdapat tiga elemen penting dalam LSTM, yaitu *input gate*, *forget gate*, dan *output gate*. *Input gate* berfungsi sebagai penyaring untuk mengidentifikasi informasi yang pantas untuk disimpan dan diperbarui ke status berikutnya. *Forget gate* berperan dalam memutuskan informasi yang perlu dilupakan atau dihapus berdasarkan status sel sebelumnya dan data input baru. Sementara itu, *output gate* menentukan informasi yang akan dijadikan keluaran pada status sel berikutnya [42]. Jaringan LSTM secara khusus dirancang untuk menangani permasalahan predksi berurutan. Mekanismenya bekerja dengan mempelajari fungsi “ f ” yang memetakan nilai input (X) ke dalam urutan output (Y) [43].

$$Y(t) = f(X(t)) \quad (3)$$

Penelitian ini menerapkan model LSTM yang terdiri dari embedding layer dengan dimensi output 128, dua lapisan Bidirectional LSTM berurutan dengan unit 64 atau 128, dimana lapisan pertama menggunakan konfigurasi return_sequences=True, kedua lapisan tersebut mengaplikasikan dropout dengan Tingkat antara 0.3 hingga 0.5 serta regulasi L2 dengan koefisien antara 1e-4 hingga 1e-2. Selanjutnya terdapat GlobalMaxPooling1D, dense layer berjumlah 64 unit dengan aktivasi ReLu, dropout, dan regulasi L2. Lapisan output berupa dense layer dengan dua neuron dan softmax untuk klasifikasi dua kelas. Model dioptimasi menggunakan adam dengan learning rate antara 1e-5 dan 5e-4, menggunakan loss sparse categorical crossentropy, Pelatihan dilakukan maksimal 10 epoch dengan mekanisme early stopping (patience=2).

3). *Recurrent Neural Network (RNN)*: RNN memiliki konsep memori, di mana output dipengaruhi oleh urutan input sebelumnya dan kemampuan untuk mengingat input yang penting. Model ini tidak menyimpan seluruh urutan secara eksplisit, melainkan mempertahankan informasi dari langkah waktu sebelumnya melalui status tersembunyi [44]. Agar RNN mampu menghasilkan prediksi yang akurat, diperlukan data sejak awal frasa. Ketergantungan ini dikenal sebagai ketergantungan jangka panjang, karena terdapat jeda waktu yang cukup panjang antara data yang relevan dan saat data tersebut digunakan dalam proses prediksi [45].

Model RNN dalam penelitian ini menggunakan embedding layer dengan dimensi output 128, satu lapisan Bidirectional SimpleRNN dengan jumlah unit antara 32 hingga 256, menggunakan dropout antara 0.1 hingga 0.5 dan regulasi L2 dengan koefisien antara 1e-6 hingga 1e-2. Lapisan RNN diatur untuk mengembalikan seluruh sekvens (return_sequences=True). Output lapisan ini kemudian diproses oleh GlobalMaxPooling1D. Selanjutnya, terdapat dense layer dengan dua neuron dan aktivasi softmax. Optimasi model dilakukan menggunakan Adam dengan learning rate dalam rentang 1e-5 hingga 1e-3. Fungsi loss yang digunakan berupa sparse categorical crossentropy. Pelatihan dilakukan maksimal 10 epoch dengan mekanisme early stopping (patience=3).

G. Hyperparameter Tuning

Hyperparameter tuning digunakan untuk mengidentifikasi parameter yang dapat menghasilkan performa terbaik pada model. Penelitian ini menggunakan hyperparameter optuna. Optuna merumuskan penyetelan hyperparameter sebagai masalah pengoptimalan global yang secara efisien mencari parameter yang paling optimal [46]. Optuna menguji banyak kombinasi hyperparameter dan mengevaluasi kinerjanya pada kumpulan data pengujian yang ditentukan. Optuna menggunakan eksperimen dan evaluasi berulang untuk menemukan kumpulan hyperparameter optimal untuk metrik kinerja tertentu [47]. Pada penelitian ini seluruh model akan dioptimalkan dengan hyperparameter optuna sebanyak 10 percobaan.

H. Evaluation

Kinerja suatu model klasifikasi dapat diukur dengan *confusion matrix*. Secara umum, *confusion matrix* menyajikan informasi yang membandingkan hasil klasifikasi yang dilakukan oleh sistem dengan hasil yang sebenarnya. Dalam proses klasifikasi, terdapat empat istilah, yaitu *True Positive* (TP), *True Negative* (TN), *False Positive* (FP) dan *False Negative* (FN) [48]. Perhitungan rata-rata akurasi dan F1 dihitung seperti yang ditunjukkan pada rumus [49].

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (4)$$

$$\text{precision} = \frac{TP}{TP + FP} \quad (5)$$

$$\text{recall} = \frac{TP}{TP + FN} \quad (6)$$

$$F1 = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \quad (7)$$

III. HASIL DAN PEMBAHASAN

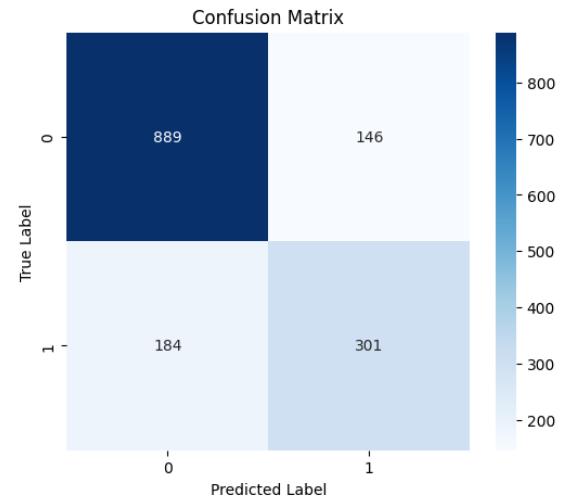
A. Support Vector Machine (SVM)

Berdasarkan hasil evaluasi model SVM yang ditampilkan pada Gambar 4, diketahui bahwa model ini mencapai tingkat *accuracy* sebesar 78%, *precision* 75%, *recall* 74%, dan *F1-score* 74%. Mengacu pada *confusion matrix* yang disajikan pada Gambar 5, dari total 1.520 data, terdapat 301 data yang diprediksi sebagai negatif sesuai dengan label sebenarnya negatif (*True Negative*), serta 889 data diprediksi positif sesuai dengan label sebenarnya positif (*True Positive*). Sementara itu, sebanyak 146 data diprediksi negatif padahal label sebenarnya positif (*False Negative*), dan 184 data diprediksi positif namun sebenarnya berlabel negatif (*False Positive*).

Kesalahan klasifikasi pada model SVM terjadi pada kalimat ['suka', 'aplikasi', 'tapi', 'bug'] yang seharusnya positif namun diprediksi negatif. Kalimat ini mengandung dua sisi sentimen, yaitu apresiasi terhadap aplikasi dan keluhan ringan mengenai bug. Karena SVM menggunakan representasi teks berbasis frekuensi seperti TF-IDF tanpa mempertimbangkan konteks atau urutan kata, model cenderung memberi bobot besar pada kata bermuatan negatif seperti "bug". Akibatnya, walaupun makna kalimat adalah positif, model gagal mengenali dominasi sentimen tersebut dan menghasilkan prediksi yang keliru.

	precision	recall	f1-score	support
Negatif	0.83	0.86	0.84	1035
Positif	0.67	0.62	0.65	485
accuracy			0.78	1520
macro avg	0.75	0.74	0.74	1520
weighted avg	0.78	0.78	0.78	1520

Gambar 4. Classification report model SVM



Gambar 5. Confusion matrix model SVM

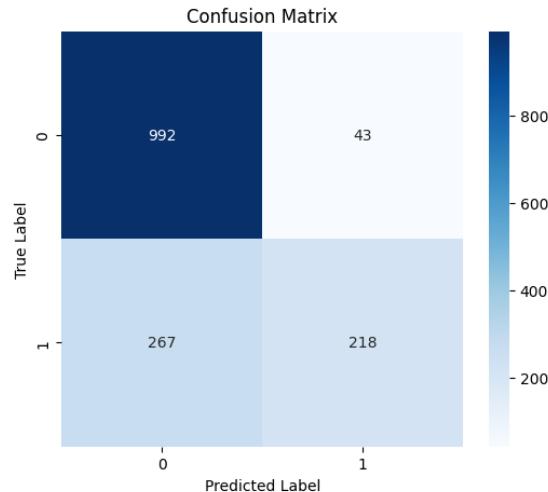
B. Random Forest

Evaluasi terhadap model *Random Forest* yang ditampilkan pada Gambar 6 menunjukkan bahwa model ini mampu mencapai *accuracy* sebesar 80%, *precision* 81%, *recall* 70%, dan *F1-score* 72%. Selain itu, analisis terhadap *confusion matrix* pada Gambar 7, dari total 1.520 data, terdapat 218 data yang berhasil diprediksi sebagai negatif sesuai dengan label sebenarnya (*True Negative*), serta 992 data yang diklasifikasikan sebagai positif dengan label yang positif juga (*True Positive*). Sementara itu, sebanyak 43 data yang seharusnya positif justru diprediksi sebagai negatif (*False Negative*), dan 267 lainnya diprediksi positif padahal label sebenarnya negatif (*False Positive*).

Salah satu kesalahan klasifikasi pada model Random Forest terlihat pada kalimat ['lumayan', 'baca', 'buku', 'aplikasi], yang seharusnya bernada positif namun diprediksi sebagai negatif. Kesalahan ini muncul karena model bekerja berdasarkan representasi fitur individual tanpa memperhitungkan urutan atau konteks antar kata. Akibatnya, makna kata 'lumayan' yang dalam konteks ini bersifat positif tidak ditangkap dengan tepat, dan hubungan antar kata yang membentuk makna keseluruhan kalimat ikut diabaikan, sehingga menghasilkan prediksi yang tidak sesuai,

	precision	recall	f1-score	support
0	0.79	0.96	0.86	1035
1	0.84	0.45	0.58	485
accuracy			0.80	1520
macro avg	0.81	0.70	0.72	1520
weighted avg	0.80	0.80	0.78	1520

Gambar 6. Classification report model random forest



Gambar 7. Confusion matrix model random forest

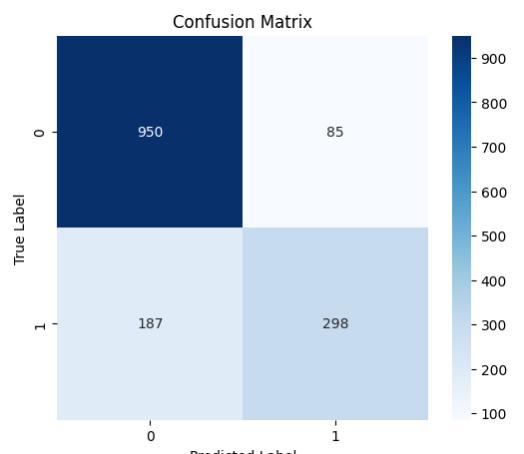
C. Convolutional Neural Network (CNN)

Hasil evaluasi terhadap model CNN yang ditampilkan pada Gambar 8 menunjukkan bahwa model ini memiliki

tingkat *accuracy* sebesar 82%, *precision* 81%, *recall* 77%, dan *F1-score* 78%. Sementara itu, berdasarkan *confusion matrix* pada Gambar 9 dari total 1.520 data, ditemukan bahwa sebanyak 298 data diklasifikasikan dengan benar sebagai kelas negatif (*True Negative*), dan 950 data diidentifikasi dengan tepat sebagai kelas positif (*True Positive*). Namun, model masih menghasilkan 85 kesalahan prediksi pada data yang seharusnya positif namun diprediksi sebagai negatif (*False Negative*), serta 187 kesalahan pada data yang seharusnya negatif namun diprediksi sebagai positif (*False Positive*).

	precision	recall	f1-score	support
0	0.84	0.92	0.87	1035
1	0.78	0.61	0.69	485
accuracy			0.82	1520
macro avg	0.81	0.77	0.78	1520
weighted avg	0.82	0.82	0.81	1520

Gambar 8. Classification report model CNN



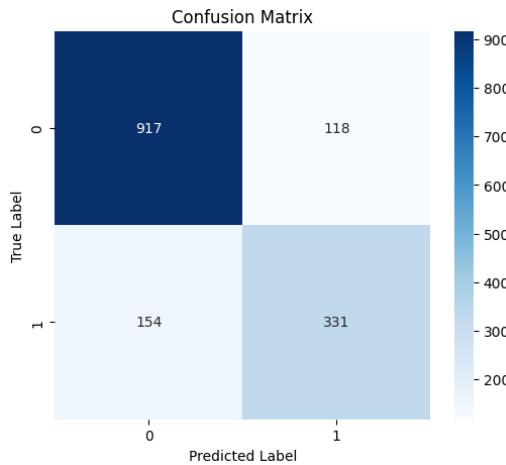
Gambar 9. Confusion matrix model CNN

D. Long Short-Term Memory (LSTM)

Hasil evaluasi terhadap model LSTM yang ditampilkan pada Gambar 10 menunjukkan model ini mampu mencapai *accuracy* sebesar 82%, *precision* 80%, *recall* 78%, *F1-score* 79%. Berdasarkan *confusion matrix* pada Gambar 11, dari total 1.520 data, terdapat 331 data yang diprediksi negatif dan benar berlabel negatif (*True Negative*), sebanyak 917 data diprediksi positif dan berlabel positif (*True Positive*), sebanyak 118 data diprediksi negatif namun sebenarnya berlabel positif (*False Negative*) serta 154 data diprediksi positif tetapi berlabel negatif (*False Positive*).

	precision	recall	f1-score	support
0	0.86	0.89	0.87	1035
1	0.74	0.68	0.71	485
accuracy			0.82	1520
macro avg	0.80	0.78	0.79	1520
weighted avg	0.82	0.82	0.82	1520

Gambar 10. Classification report model LSTM



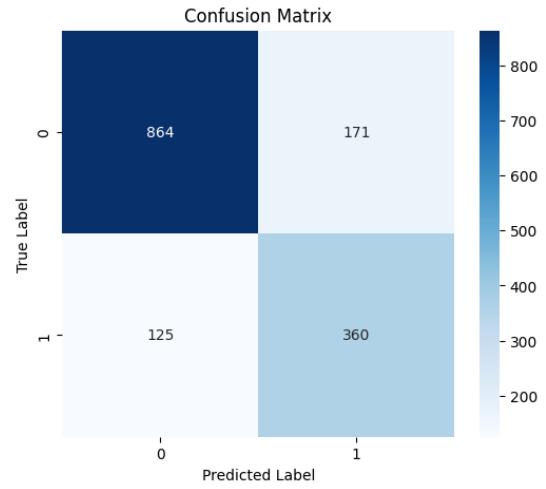
Gambar 11. Confusion matrix model LSTM

E. Recurrent Neural Network (RNN)

Berdasarkan hasil evaluasi terhadap model RNN yang ditampilkan pada Gambar 12, diperoleh tingkat *accuracy* sebesar 81%, *precision* 78%, *recall* 79%, *F1-score* 78%. Selanjutnya, jika merujuk *confusion matrix* pada Gambar 13, dari total 1.520 data, terdapat 360 data yang diprediksi negatif dan benar berlabel negatif (*True Negative*), sebanyak 864 data diprediksi positif dan berlabel positif (*True Positive*), sebanyak 171 data diprediksi negatif namun sebenarnya berlabel positif (*False Negative*), serta 125 data diprediksi positif tetapi berlabel negatif (*False Positive*).

	precision	recall	f1-score	support
Negatif	0.87	0.83	0.85	1035
Positif	0.68	0.74	0.71	485
accuracy			0.81	1520
macro avg	0.78	0.79	0.78	1520
weighted avg	0.81	0.81	0.81	1520

Gambar 12. Classification report model RNN



Gambar 13. Confusion matrix model RNN

Seluruh model deep learning yang digunakan, yaitu CNN, LSTM, dan RNN gagal mengklasifikasikan sentimen dengan benar pada kalimat ['aplikasi', 'lambat', 'moga', 'baik'] yang seharusnya positif namun diprediksi negatif. Kesalahan ini terjadi karena ketiga model terlalu terpengaruh pada kata "lambat" yang muncul di awal dan memiliki konotasi negatif yang kuat. CNN cenderung menangkap pola lokal seperti "aplikasi lambat", sementara LSTM dan RNN, walaupun mempertimbangkan urutan kata, tetap mengalami kesulitan mengenali perubahan sentimen karena kata positif seperti "moga" dan "baik" muncul di akhir dan tidak cukup kuat untuk mengubah representasi memori yang sudah terbentuk negatif sejak awal.

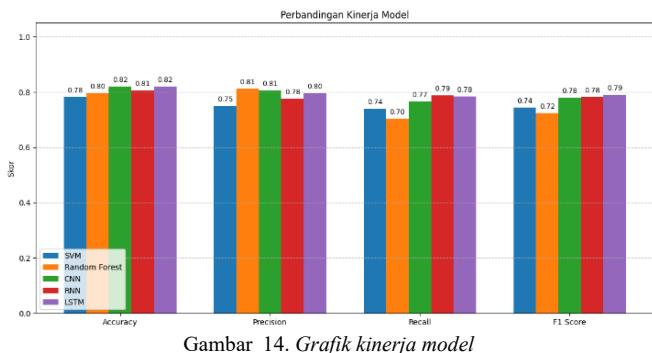
F. Perbandingan Model

TABEL IV
KINERJA MODEL

Model	Accuracy	Precision	Recall	F1-Score
SVM	78%	75%	74%	74%
Random Forest	80%	81%	70%	72%
CNN	82%	81%	77%	78%
LSTM	82%	80%	78%	79%
RNN	81%	78%	79%	78%

Tabel IV menyajikan perbandingan kinerja lima model yang terdiri atas pendekatan *machine learning* dan *deep learning*, yaitu *Support Vector Machine* (SVM), *Random Forest*, *Convolutional Neural Network* (CNN), *Long Short-Term Memory* (LSTM), dan *Recurrent Neural Network* (RNN), berdasarkan empat metrik evaluasi yaitu, *accuracy*, *precision*, *recall*, dan *F1-score*, hasil evaluasi performa menunjukkan bahwa model CNN dan LSTM memiliki performa yang serupa dengan *accuracy* tertinggi, diikuti oleh RNN, Random Forest, dan SVM. Dari sisi *precision*, Random Forest dan CNN menunjukkan nilai yang lebih tinggi

dibandingkan model lainnya, sedangkan *recall* tertinggi ditemukan pada RNN dan LSTM, dengan CNN berada di posisi berikutnya. Nilai *F1-score* menunjukkan hasil tertinggi pada LSTM dan CNN, diikuti oleh RNN, Random Forest, dan SVM. Model SVM secara konsisten menunjukkan nilai metrik yang lebih rendah dibandingkan model-model lainnya. Gambar 14 menunjukkan grafik perbandingan performa antar model berdasarkan berbagai metrik evaluasi tersebut.



Gambar 14. Grafik kinerja model

Berdasarkan hasil evaluasi kinerja model, nilai akurasi antar model menunjukkan perbedaan yang relatif kecil sehingga dibutuhkan analisis statistik untuk menilai signifikansi perbedaan tersebut. Uji McNemar dilakukan pada beberapa pasangan model untuk mengevaluasi signifikansi perbedaan prediksi di antara model-model tersebut.

TABEL V
HASIL UJI STATISTIK MCNEMAR

Perbandingan model	Statistik McNemar	p-value	Kesimpulan
Random Forest vs CNN	7.959	0.0048	Perbedaan signifikan
Random Forest vs RNN	0.555	0.4559	Tidak signifikan
Random Forest vs LSTM	9.153	0.0025	Perbedaan signifikan
CNN vs RNN	2.755	0.0969	Tidak signifikan
CNN vs LSTM	0.219	0.6396	Tidak signifikan
RNN vs LSTM	5.461	0.0194	Perbedaan signifikan
SVM vs Random Forest	1208	0.000	Perbedaan signifikan

Tabel V menunjukkan hasil dari uji McNemar yang memperlihatkan bahwa perbandingan antara Random Forest dengan CNN memiliki p-value sebesar 0.0048, dan dengan LSTM sebesar 0.0025, keduanya menunjukkan perbedaan yang signifikan secara statistik. Perbandingan antara Random Forest dengan RNN menghasilkan p-value sebesar 0.4559, sedangkan perbandingan antara CNN dengan LSTM memiliki p-value 0.6396, yang keduanya menandakan tidak terdapat perbedaan yang signifikan sehingga hasil prediksi kedua model tersebut dapat dianggap sebanding secara statistik. Perbandingan antara CNN dan RNN dengan nilai p-

value 0.096 juga tidak signifikan pada tingkat signifikansi 0,05 menunjukkan performa yang serupa secara statistik. Namun, perbandingan antara RNN dan LSTM dengan p-value 0.0194 menunjukkan perbedaan yang signifikan meskipun nilai akurasi keduanya relatif dekat. Selain itu, perbedaan signifikan juga ditemukan antara SVM dan Random Forest dengan p-value 0.000. Secara statistik, CNN dan LSTM menunjukkan performa yang unggul secara signifikan dibandingkan dengan Random Forest dan SVM, sementara RNN menempati posisi menengah di antara model-model tersebut. Performa RNN tidak berbeda signifikan dengan CNN dan Random Forest, namun lebih rendah dibandingkan LSTM.

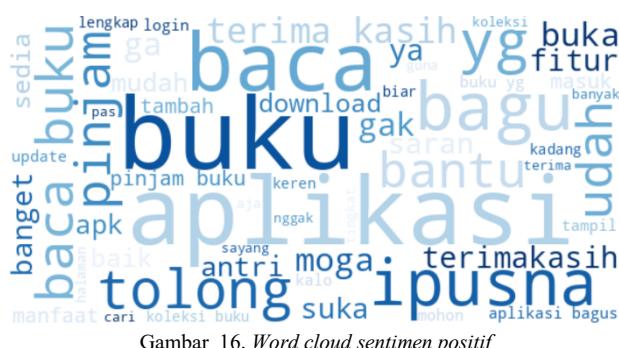
G. Visualisasi Word Cloud

Analisis terhadap visualisasi *word cloud* yang merepresentasikan sentimen negatif dan positif dari ulasan pengguna iBusnas bertujuan untuk memperoleh pemahaman yang dapat dimanfaatkan dalam peningkatan kualitas, penyempurnaan fungsionalitas, dan perencanaan pengembangan aplikasi. Dengan memahami persepsi pengguna secara langsung, arah pengembangan iBusnas dapat ditetapkan secara lebih presisi dan responsif.



Gambar 15. Word cloud sentimen negatif

Word Cloud sentimen negatif yang ditampilkan pada Gambar 15 menunjukkan dominasi keluhan pengguna terhadap aspek teknis dan fungsionalitas aplikasi. Kemunculan kata-kata seperti “login”, “update”, “gagal”, “error”, dan “loading” merepresentasikan berbagai kendala teknis yang secara langsung memengaruhi pengguna dalam memanfaatkan fitur-fitur yang tersedia. Kata “login” dan “loading” menunjukkan permasalahan terjadi pada fase awal interaksi pengguna dengan aplikasi, yaitu proses autentikasi dan pematuhan konten. Permasalahan ini diinterpretasikan sebagai indikasi adanya gangguan teknis seperti ketidakstabilan sistem, gangguan jaringan, atau adanya bug pasca pembaruan aplikasi. Selain itu, kemunculan kata “tolong” pada wordcloud negatif menunjukkan walaupun pengguna mengalami hambatan, mereka tetap memiliki ekspektasi terhadap perbaikan layanan. Hal ini mengisyaratkan bahwa loyalitas pengguna pada aplikasi belum sepenuhnya hilang, namun mulai mengalami penurunan akibat pengalaman negatif yang bersifat teknis.



Gambar 16. Word cloud sentimen positif

Sentimen positif yang ditampilkan pada gambar 16 memperlihatkan sejumlah kata kunci seperti “buku”, “baca”, “pinjam”, “aplikasi”, “bagus”, “fitur”, dan “terima kasih”. Kemunculan kata-kata tersebut menunjukkan bahwa fitur inti dari aplikasi, yaitu peminjaman dan pembacaan buku digital, telah diterima dengan baik oleh sebagian besar pengguna. Hal ini mengindikasikan bahwa iPusnas berhasil memenuhi kebutuhan dasar pengguna dalam hal akses literatur digital. Selanjutnya, munculnya kata-kata seperti “tolong”, dan “tambahkan” menunjukkan adanya keinginan pengguna terhadap pengembangan fitur lebih lanjut, seperti perluasan koleksi buku, penyempurnaan antarmuka, atau fitur notifikasi yang lebih informatif. Hal ini menunjukkan bahwa pengguna tidak hanya menikmati layanan yang tersedia, tetapi juga memiliki harapan terhadap peningkatan akses koleksi, pengembangan fitur baru, serta penyempurnaan terhadap fungsi-fungsi yang telah ada.

Perbandingan antara *word cloud* sentimen positif dan negatif menghasilkan temuan penting melalui proses *knowledge discovery*, yaitu kemunculan kata yang konsisten seperti “buku”, “baca”, “pinjam” dan “aplikasi” pada kedua sentimen, yang berarti pengguna tetap memprioritaskan fungsi utama aplikasi yaitu membaca dan meminjam buku meskipun mengalami hambatan. Ini menunjukkan bahwa fitur ini aplikasi masih relevan dan diminati, namun pengalaman pengguna secara keseluruhan masih terganggu oleh kendala teknis seperti gagal loading, lambatnya loading, atau gangguan setelah pembaruan aplikasi. Temuan ini penting karena menunjukkan bahwa pengembangan aplikasi tidak cukup hanya menambahkan fitur baru, namun harus diawali dengan memastikan keandalan dan stabilitas sistem yang mendukung interaksi dasar pengguna.

Berdasarkan hasil temuan tersebut, terdapat dua hal penting yang dapat menjadi perhatian utama bagi pengembang iPusnas. Pertama, perlu ada fokus pada perbaikan aspek teknis seperti kestabilan saat login, kecepatan loading, dan minimasi error pasca update. Bagian ini menjadi titik kritis yang menentukan apakah pengguna akan melanjutkan atau menghentikan interaksi dengan aplikasi. Kedua, masukan positif berupa pengembangan fitur pelengkap yang dapat meningkatkan kenyamanan pengguna, seperti notifikasi ketersediaan buku dan opsi pengingat bacaan. Fitur-fitur tersebut diharapkan dapat mendukung

kebiasaan membaca pengguna secara lebih teratur serta memudahkan akses terhadap buku yang dinantikan. Dengan fokus pada aspek teknis dan fungsional pengembang dapat menetapkan prioritas perbaikan secara strategis, sehingga iPusnas tidak hanya menjadi aplikasi yang stabil secara teknis, tetapi juga mampu menghadirkan pengalaman pengguna yang lebih optimal dan selaras dengan kebutuhan nyata.

IV. KESIMPULAN

Berdasarkan penerapan lima model klasifikasi dalam analisis sentimen ulasan aplikasi iPusnas, mencakup dua model *machine learning* (*Support Vector Machine* dan *Random Forest*) serta tiga model *deep learning* yaitu CNN, LSTM, dan RNN, didapatkan hasil evaluasi berdasarkan metrik *accuracy*, *precision*, *recall*, *F1-score*, *confusion matrix*, dan uji McNemar. Hasil menunjukkan bahwa model CNN dan LSTM memiliki *accuracy* tertinggi sebesar 82%, sedangkan SVM memiliki *accuracy* terendah sebesar 78%. Untuk *precision*, nilai tertinggi terdapat pada CNN dan Random Forest sebesar 81% dan terendah pada SVM sebesar 75%. Pada *recall*, RNN mencatat nilai sebesar 79% dan SVM terendah sebesar 74%. Sementara itu, *F1-score* tertinggi dimiliki LSTM sebesar 79% sedangkan SVM terendah sebesar 74%. Uji McNemar menunjukkan perbedaan signifikan antara Random Forest dengan CNN, Random Forest dengan LSTM, Random Forest dan SVM, serta antara RNN dan LSTM. Namun, perbedaan antara CNN, RNN, dan LSTM tidak signifikan secara statistik, menunjukkan performa sebanding. Selanjutnya pada analisis word cloud menunjukkan adanya dominasi keluhan teknis pada sentimen negatif, seperti error, loading lambat, dan masalah login. Ulasan positif didominasi oleh kepuasan terhadap fitur peminjaman dan pembacaan buku. Kemunculan kata seerti “buku”, “baca”, dan “aplikasi” di kedua kategori menunjukkan walaupun pengguna mengalami kendala teknis, namun kebutuhan terkait akses digital tetap tinggi.

DAFTAR PUSTAKA

- [1] D. Ilmu, P. Uin, and R. F. Palembang, "Penerapan Strategi Promosi Perpustakaan Wahfiuddin Rahmad Harahap," 2021.
 - [2] R. Indonesia, *Dewan Perwakilan Rakyat Republik Indonesia dan Presiden Republik Indonesia*. Indonesia, 2007, p. 129.
 - [3] S. Anjani and Y. Winoto, "Pemetaan Publikasi Ilmiah Tentang Perpustakaan Digital Tahun 2011-2021 Melalui Aplikasi VOSViewer," *Jurnal Ilmu Perpustakaan (JIPER)*, vol. Vol.4, No.2, 2022.
 - [4] S. Taryani and L. Wijayanti, "Pengukuran Kualitas Layanan Aplikasi Ipusnas Terhadap Kepuasan Pengguna Dengan Menggunakan Metode Webqual 4.0," *VISI PUSTAKA*, vol. Vol. 25 No. 1, 2023.
 - [5] A. Septiani and I. Budi, "Klasifikasi Ulasan Pengguna Aplikasi: Studi Kasus Aplikasi Ipusnas Perpustakaan Nasional Republik Indonesia (PNRI)," *JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, vol. Vol. 07, 2022.
 - [6] F. Sena Lestari, M. Maariful Huda, T. Prabowo, and I. Komputer, "Sentiment Analysis of iPusnas Application Reviews on Google Play Using Support Vector Machine," *The Changing Role of Knowledge and Living Sustainability in ASEAN Community*, 2022.

- [7] K. L. Tan, C. P. Lee, and K. M. Lim, "A Survey of Sentiment Analysis: Approaches, Datasets, and Future Research," *Applied Sciences (Switzerland)*, vol. 13, no. 7, Apr. 2023, doi: 10.3390/app13074550.
- [8] Q. Ain, E. Utami, and A. Nasiri, "Analisis Sentimen: Prediksi Rating Terhadap Reviews Wisatawan Tanjung Puting Pada Tripadvisor Menggunakan Support Vector Machine," *JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, vol. 9, no. 3, pp. 1586–1595, Aug. 2024, doi: 10.29100/jipi.v9i3.5430.
- [9] C. Llatas, B. Soust-Verdaguer, L. C. Torres, and D. Cagigas, "Application of Knowledge Discovery in Databases (KDD) to environmental, economic, and social indicators used in BIM workflow to support sustainable design," *Journal of Building Engineering*, vol. 91, Aug. 2024, doi: 10.1016/j.jobe.2024.109546.
- [10] V. Novalia, K. Ditha Tania, A. Meiriza, and A. Wedhasmara, "Knowledge Discovery of Application Review Using Word Embedding's Comparison with CNN-LSTM Model on Sentiment Analysis," in *ICECOS 2024 - 4th International Conference on Electrical Engineering and Computer Science, Proceeding*, Institute of Electrical and Electronics Engineers Inc., 2024, pp. 234–238. doi: 10.1109/ICECOS63900.2024.10791113.
- [11] N. A. Sofiah, K. D. Tania, A. Meiriza, and A. Wedhasmara, "A Comparative Assessment SARIMA and LSTM Models for the Gurugram Air Quality Index's Knowledge Discovery," in *ICECOS 2024 - 4th International Conference on Electrical Engineering and Computer Science, Proceeding*, Institute of Electrical and Electronics Engineers Inc., 2024, pp. 26–31. doi: 10.1109/ICECOS63900.2024.10791243.
- [12] S. Mazya Permataning Tyas, R. Sarno, and B. Setya Rintyarna, "Analisis Perbandingan Metode Klasifikasi Sentimen Berita Saham: Pendekatan Machine Learning, Deep Learning, Transfer Learning, dan Graf," *Jurnal Penelitian Ipteks*, vol. 9, no. 1, pp. 58–64, 2024, doi: <https://doi.org/10.32528/penelitianipteks.v9i1.1479>.
- [13] S. A. Sutresno, "Analisis Sentimen Masyarakat Indonesia Terhadap Dampak Penurunan Global Sebagai Akibat Resesi di Twitter," *Building of Informatics, Technology and Science (BITS)*, vol. 4, no. 4, Mar. 2023, doi: 10.47065/bits.v4i4.3149.
- [14] F. I. Rafif, M. D. Purbolaksono, and W. Astuti, "Sentiment Analysis using Random Forest and Word2Vec for Indonesian Language Movie Reviews," *Jurnal Media Informatika Budidarma*, vol. 7, no. 3, p. 1109, Jul. 2023, doi: 10.30865/mib.v7i3.6299.
- [15] N. M. K. Sedana, I. N. S. W. Wijaya, and I. K. R. Arthana, "Analisis Sentimen Berbahasa Inggris Dengan Metode Lstm Studi Kasus Berita Online Pariwisata Bali," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 11, no. 6, pp. 1325–1334, Dec. 2024, doi: 10.25126/jtiik.2024118792.
- [16] Y. Yuliska, D. Hidayatul Qudsi, J. Hakim Lubis, K. Umam Syaliman, and N. Fadilah Najwa, "Analisis Sentimen Pada Data Saran Mahasiswa Terhadap Kinerja Departemen Di Perguruan Tinggi Menggunakan Convolutional Neural Network," *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK)*, vol. Vol. 8, 2021, doi: 10.25126/jtiik.202184842.
- [17] P. Mukherjee, Y. Badr, S. Doppalapudi, S. M. Srinivasan, R. S. Sangwan, and R. Sharma, "Effect of Negation in Sentences on Sentiment Analysis and Polarity Detection," in *Procedia Computer Science*, Elsevier B.V., 2021, pp. 370–379. doi: 10.1016/j.procs.2021.05.038.
- [18] A. U. El Majid and R. Nuari, "Perbandingan Kinerja Metrik Bert Dan Model Machine Learning Klasik (Svm, Naive Bayes) Untuk Analisis Sentimen," *Jurnal Infotek Polbeg*, vol. 10, no. 2, 202AD.
- [19] B. Z. Ramadhan, I. Riza, and I. Maulana, "Analisis Sentimen Ulasan Pada Aplikasi E-Commerce Dengan Menggunakan Algoritma Naïve Bayes," 2022. [Online]. Available: <http://jurnal.polibatam.ac.id/index.php/JAIC-Herwinskyah> and A. Witanti, "Analisis Sentimen Masyarakat Terhadap Vaksinasi Covid-19 Pada Media Sosial Twitter Menggunakan Algoritma Support Vector Machine (SVM)," *Jurnal Sistem Informasi dan Informatika (Simika) P-ISSN*, vol. 5, pp. 2622–6901, 2022.
- [21] S. Wu, Y. Liu, Z. Zou, and T. H. Weng, "S_I_LSTM: stock price prediction based on multiple data sources and sentiment analysis," *Conn Sci*, vol. 34, no. 1, pp. 44–62, 2022, doi: 10.1080/09540091.2021.1940101.
- [22] Styawati, N. Hendrastuty, A. Raahman Isnain, and A. Yanti Rahmadhani, "Analisis Sentimen Masyarakat Terhadap Program Kartu Prakerja Pada Twitter Dengan Metode Support Vector Machine," *Jurnal Informatika: Jurnal pengembangan IT (JPIT)*, vol. 6, no. 3, 2021, doi: <https://doi.org/10.30591/jpit.v6i3.2870>.
- [23] H. T. Ismet, T. Mustaqim, and D. Purwitasari, "Aspect Based Sentiment Analysis of Product Review Using Memory Network," *Scientific Journal of Informatics*, vol. 9, no. 1, pp. 73–83, May 2022, doi: 10.15294/sji.v9i1.34094.
- [24] O. I. Gifari, M. Adha, I. Rifky Hendrawan, F. Freddy, and S. Durrand, "Analisis Sentimen Review Film Menggunakan TF-IDF dan Support Vector Machine," *JIFOTECH (Journal Of Information Technology)*, vol. 2, no. 1, 2022.
- [25] H. T. Duong and T. A. Nguyen-Thi, "A review: preprocessing techniques and data augmentation for sentiment analysis," *Comput Soc Netw*, vol. 8, no. 1, Dec. 2021, doi: 10.1186/s40649-020-00080-x.
- [26] A. Bijaksana, P. Negara, H. Muhandi, and I. M. Putri, "Analisis Sentimen Maskapai Penerbangan Menggunakan Metode Naïve Bayes Dan Seleksi Fitur Information Gain Sentiment Analysis On Airlines Using Naïve Bayes Method And Feature Selection Information Gain," *Jurnal Teknologi Informatika dan Ilmu Komputer (JTIK)*, vol. 7, no. 3, pp. 599–606, 2020, doi: 10.25126/jtiik.202071947.
- [27] D. A. Kristiyanti and S. Hardani, "Sentiment Analysis of Public Acceptance of Covid-19 Vaccines Types in Indonesia using Naïve Bayes, Support Vector Machine, and Long Short-Term Memory (LSTM)," *Jurnal RESTI*, vol. 7, no. 3, pp. 722–732, Jun. 2023, doi: 10.29207/resti.v7i3.4737.
- [28] M. A. Raihan and E. B. Setiawan, "Aspect Based Sentiment Analysis with FastText Feature Expansion and Support Vector Machine Method on Twitter," *Jurnal RESTI*, vol. 6, no. 4, pp. 591–598, 2022, doi: 10.29207/resti.v6i4.4187.
- [29] A. Erfina and M. R. N. R. Alamsyah, "Implementation of Naïve Bayes classification algorithm for Twitter user sentiment analysis on ChatGPT using Python programming language," *Data and Metadata*, vol. 2, Jan. 2023, doi: 10.56294/dm202345.
- [30] N. D. Kurniawan, P. R. Ferdian, and N. Hidayati, "Analisis Sentimen Algoritma Naïve Bayes, Support Vector Machine, dan Random Forest Pada Ulasan Aplikasi Ajaib," *Jurnal Nasional Teknologi dan Sistem Informasi*, vol. 11, no. 1, pp. 87–97, May 2025, doi: 10.25077/teknosi.v11i1.2025.87-97.
- [31] F. Nufairi, N. Pratiwi, and F. Herlando, "Analisis Sentimen Pada Ulasan Aplikasi Threads Di Google Play Store Menggunakan Algoritma Support Vector Machine," *JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, vol. 9, no. 1, pp. 339–348, Feb. 2024, doi: 10.29100/jipi.v9i1.4929.
- [32] J. J. Tono and P. Parjito, "Persepsi Publik Terhadap Kepemimpinan Firli Bahuri Di Kpk: Pendekatan Sentimen Twitter Dengan Naïve Bayes Dan SVM," *JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, vol. 10, no. 2, pp. 1272–1285, Mar. 2025, doi: 10.29100/jipi.v10i2.6181.
- [33] D. A. Musleh *et al.*, "Arabic Sentiment Analysis of YouTube Comments: NLP-Based Machine Learning Approaches for Content Evaluation," *Big Data and Cognitive Computing*, vol. 7, no. 3, Sep. 2023, doi: 10.3390/bdcc7030127.
- [34] T. Chen, X. Yin, L. Peng, J. Rong, J. Yang, and G. Cong, "Monitoring and recognizing enterprise public opinion from high-risk users based on user portrait and random forest algorithm," *Axioms*, vol. 10, no. 2, Jun. 2021, doi: 10.3390/axioms10020106.
- [35] J. Asian, M. D. Rosita, and T. Mantoro, "Sentiment Analysis for the Brazilian Anesthesiologist Using Multi-Layer Perceptron Classifier and Random Forest Methods," *Jurnal Online Informatika*, vol. 7, no. 1, pp. 132–141, Jun. 2022, doi: 10.15575/join.v7i1.900.

- [36] D. Purnamasari, A. B. Aji, S. Madenda, I. M. Wiryan, and S. Harmanto, "Sentiment Analysis Methods For Customer Review Of Indonesia E-Commerce," *International Journal of Innovative Computing, Information and Control*, vol. 20, no. 1, pp. 47–60, Feb. 2024, doi: 10.24507/ijicic.20.01.47.
- [37] M. Kamyab, G. Liu, and M. Adjeisah, "Attention-Based CNN and Bi-LSTM Model Based on TF-IDF and GloVe Word Embedding for Sentiment Analysis," *Applied Sciences (Switzerland)*, vol. 11, no. 23, Dec. 2021, doi: 10.3390/app112311255.
- [38] L. Xiaoyan, R. C. Raga, and S. Xuemei, "GloVe-CNN-BiLSTM Model for Sentiment Analysis on Text Reviews," *J Sens*, vol. 2022, 2022, doi: 10.1155/2022/7212366.
- [39] A. Z. R. Adam and E. B. Setiawan, "Social Media Sentiment Analysis Using Convolutional Neural Network (CNN) and Gated Recurrent Unit (GRU)," *Jurnal Ilmiah Teknik Elektro Komputer dan Informatika*, vol. 9, no. 1, pp. 119–131, Feb. 2023, doi: 10.26555/jiteki.v9i1.25813.
- [40] R. Bharal and O. V Vamsi Krishna, "Social Media Sentiment Analysis Using CNN-BiLSTM," *International Journal of Science and Research (IJSR)*, vol. 10, no. 9, pp. 656–661, Sep. 2021, doi: 10.21275/sr21913110537.
- [41] B. A. Pramono, A. Firman Daru, and M. B. Ulum, "Twitter Sentiment Analysis Using Natural Language Processing (NLP) Method and Long Short Term Memory (LSTM) Algorithm in the 2024 Indonesian Presidential Election," 2024. [Online]. Available: <http://ejournal.uksw.edu/ijiteb>
- [42] K. L. Tan, C. P. Lee, K. S. M. Anbananthen, and K. M. Lim, "RoBERTa-LSTM: A Hybrid Model for Sentiment Analysis With Transformer and Recurrent Neural Network," *IEEE Access*, vol. 10, pp. 21517–21525, 2022, doi: 10.1109/ACCESS.2022.3152828.
- [43] K. Mrhar, L. Benhiba, S. Bourekache, and M. Abik, "A Bayesian CNN-LSTM Model for Sentiment Analysis in Massive Open Online Courses MOOCs," *International Journal of Emerging Technologies in Learning*, vol. 16, no. 23, pp. 216–232, 2021, doi: 10.3991/ijet.v16i23.24457.
- [44] S. Imron, E. I. Setiawan, J. Santoso, and M. H. Purnomo, "Aspect Based Sentiment Analysis Marketplace Product Reviews Using BERT, LSTM, and CNN," *Jurnal RESTI*, vol. 7, no. 3, pp. 586–591, Jun. 2023, doi: 10.29207/resti.v7i3.4751.
- [45] M. H. Abdalla *et al.*, "Sentiment Analysis Based on Hybrid Neural Network Techniques Using Binary Coordinate Ascent Algorithm," *IEEE Access*, vol. 11, pp. 134087–134099, 2023, doi: 10.1109/ACCESS.2023.3334980.
- [46] P. K. Sahu and T. Fatma, "Optimized Breast Cancer Classification Using PCA-LASSO Feature Selection and Ensemble Learning Strategies with Optuna Optimization," *IEEE Access*, vol. 13, pp. 35645–35661, 2025, doi: 10.1109/ACCESS.2025.3539746.
- [47] L. G. Atlas *et al.*, "A modernized approach to sentiment analysis of product reviews using BiGRU and RNN based LSTM deep learning models," *Sci Rep*, vol. 15, no. 1, Dec. 2025, doi: 10.1038/s41598-025-01104-0.
- [48] I. G. S. Mas Diyasa, N. M. I. Marini Mandenni, M. I. Fachrurrozi, S. I. Pradika, K. R. Nur Manab, and N. R. Sasmita, "Twitter Sentiment Analysis as an Evaluation and Service Base On Python Textblob," *IOP Conf Ser Mater Sci Eng*, vol. 1125, no. 1, p. 012034, May 2021, doi: 10.1088/1757-899x/1125/1/012034.
- [49] B. Su and J. Peng, "Sentiment Analysis of Comment Texts on Online Courses Based on Hierarchical Attention Mechanism," *Applied Sciences (Switzerland)*, vol. 13, no. 7, Apr. 2023, doi: 10.3390/app13074204.