KOMPARASI ALGORITMA XGBOOST DAN RANDOM FOREST DALAM MENGKLASIFIKASIKAN SENTIMEN PENGGUNA IPUSNAS

Alana Nur Fathu[1*], Ria Faulina[2]
[1,2]Program Studi Statistika, Fakultas Sains dan Teknologi, Universitas Terbuka, Kota Tangerang Selatan, Indonesia

[*]Penulis korespondensi: alananurf@gmail.com

ABSTRAK

Perkembangan teknologi digital mendorong perpustakaan untuk menyediakan layanan informasi yang mudah diakses, termasuk melalui aplikasi iPusnas. Banyaknya ulasan pengguna pada *Google Play Store* mencerminkan beragam pengalaman penggunaan yang perlu dianalisis untuk memahami persepsi masyarakat secara objektif. Penelitian ini bertujuan untuk membandingkan performa dua algoritma *machine learning*, yaitu *XGBoost* dan *Random Forest*, dalam mengklasifikasikan sentimen ulasan pengguna iPusnas. Sebanyak 10.000 data ulasan dikumpulkan melalui teknik *web scraping* dan diproses melalui tahapan *preprocessing*, pelabelan berbasis leksikon, serta transformasi fitur menggunakan *Term Frequency–Inverse Document Frequency* (TF-IDF). Evaluasi model dilakukan dengan metrik *accuracy*, *precision*, *recall*, dan *F1-score* pada dua skenario pembagian data, yaitu 80:20 dan 90:10. Hasil penelitian ini menunjukkan bahwa *XGBoost* secara konsisten menunjukkan performa yang lebih baik dibandingkan *Random Forest*, terutama pada metrik *precision*, *recall*, dan *F1-score*. Selain itu, hasil visualisasi *wordcloud* menunjukkan bahwa kata "buku", "baca", dan "aplikasi" memiliki frekuensi kemunculan tertinggi dalam ulasan pengguna, yang menggambarkan fokus utama pembahasan ulasan terhadap fungsi iPusnas sebagai aplikasi perpustakaan digital.

Kata Kunci: analisis sentimen, iPusnas, *XGBoost*, *Random Forest*, TF-IDF.

1 PENDAHULUAN

Perkembangan teknologi informasi dan komunikasi pada era digital saat ini telah membawa perubahan signifikan terhadap berbagai sektor kehidupan, termasuk dalam bidang perpustakaan (Atika & Sayekti, 2023). Dinamika perkembangan tersebut mendorong perubahan dalam pengelolaan dan penyebaran informasi, di mana kebutuhan pemustaka semakin beragam, kompleks, serta menuntut kecepatan dan kemudahan dalam proses informasi. Sebagai salah satu penyedia informasi, perpustakaan dituntut untuk melakukan inovasi berkelanjutan melalui pemanfaatan teknologi digital dan layanan daring guna meningkatkan efisiensi, kualitas, dan jangkauan layanan informasi kepada masyarakat (Natalea & Christiani, 2020).

Sebagai bentuk adaptasi terhadap perkembangan teknologi digital, Perpustakaan Nasional Republik Indonesia, yang biasa disebut PNRI, meluncurkan aplikasi perpustakaan digital bernama iPusnas pada tahun 2016. Aplikasi ini dapat diunduh pada perangkat *Android*, *iOS*, dan *Windows* (Septiani & Budi, 2022). Melalui platform *Google Play Store* pada perangkat *Android*, pengguna tidak hanya dapat mengunduh aplikasi, tetapi juga memberikan ulasan dan penilaian terhadap pengalaman penggunaan aplikasi tersebut (Aprianti dkk., 2025). Sampai saat ini, aplikasi iPusnas telah diunduh lebih dari 1 juta kali dan memperoleh penilaian aplikasi 2,2 dari 5 bintang. Nilai tersebut mencerminkan persepsi dan tingkat kepuasan pemustaka. Namun, ulasan yang tersedia belum sepenuhnya menggambarkan tingkat kepuasan yang akurat karena tidak menggunakan analisis indikator yang terukur dan jelas (Sarah dkk., 2023).

Banyaknya ulasan yang diberikan pengguna terhadap aplikasi iPusnas pada platform *Google Play Store* menunjukkan adanya beragam persepsi dan pengalaman pengguna terhadap

kualitas layanan tersebut. Kondisi ini memerlukan pendekatan yang sistematis untuk mengidentifikasi dan mengklasifikasikan sentimen pengguna secara objektif, salah satunya yakni penerapan metode *machine learning* (Hasibuan & Heriyanto, 2022). Penelitian sebelumnya pernah dilakukan oleh Septiani dan Budi (2022) tentang klasifikasi pengguna iPusnas hanya menggunakan 2.430 *dataset* ulasan periode bulan Januari hingga Desember 2021. Penelitian tersebut menggunakan algoritma klasifikasi *Naive Bayes*, *Support Vector Machine* (SVM), dan *Logistic Regression*, serta mengelompokkan ulasan tersebut ke dalam tiga kategori utama, yaitu *problem*, *request*, dan *other*. Meskipun demikian, penelitian tersebut memiliki keterbatasan pada jumlah data yang relatif kecil serta ruang lingkup klasifikasi yang belum sepenuhnya merepresentasikan persepsi dan sentimen pengguna secara menyeluruh.

Penelitian ini mengembangkan kajian sebelumnya (Septiani & Budi, 2022), dengan menggunakan 10.000 data ulasan pengguna iPusnas dan menerapkan dua algoritma *ensemble learning*, yaitu *XGBoost* dan *Random Forest*, untuk membandingkan performa klasifikasi sentimen pengguna iPusnas. Kedua algoritma ini dipilih karena *XGBoost* dikenal unggul dalam efisiensi dan kemampuan mengatasi *overfitting* melalui teknik *gradient boosting* (Prasetya & Rosa, 2024), sedangkan *Random Forest* memiliki keunggulan dalam mengelola data berukuran besar dan menghasilkan prediksi yang stabil (Firzatullah & Nuroji, 2025). Penelitian ini juga mengukur performa model menggunakan metrik akurasi, presisi, *recall*, dan *F1-score* untuk memperoleh hasil klasifikasi yang akurat dan komprehensif.

2 METODE PENELITIAN

2.1. Data

Penelitian ini memanfaatkan data sekunder berupa data publik yang diperoleh dari *Google Play Store*. Pengambilan data dilakukan melalui teknik *web scraping* menggunakan bahasa pemrograman Python pada platform *Google Colab*. Tahap awal penelitian dilakukan dengan mengakses halaman aplikasi iPusnas pada *Google Play Store* dengan *package identifier* "mam.reader.ipusnas". Data yang dikumpulkan difokuskan pada ulasan pengguna aplikasi tersebut. Jumlah data yang dikumpulkan sebanyak 10.000 data sampel ulasan, dengan cakupan periode pengambilan yang cukup luas, sehingga merepresentasikan variasi pengalaman pengguna.

2.2. Seleksi Data

Setelah data diperoleh melalui *scraping* dengan *package identifier* "mam.reader.ipusnas", dilakukan tahap seleksi atribut untuk menentukan variabel yang sesuai dengan tujuan penelitian. Berdasarkan proses seleksi tersebut, ditetapkan dua atribut utama yang digunakan dalam analisis, yaitu *content* sebagai representasi teks ulasan aplikasi iPusnas dan *score* sebagai nilai penilaian pengguna terhadap aplikasi iPusnas.

2.3. *Preprocessing*

Preprocessing merupakan tahapan pengolahan awal yang bertujuan mengubah data tidak terstruktur menjadi data yang terstruktur sehingga dapat diolah lebih lanjut dalam proses *text mining*. Berdasarkan kajian dari beberapa literatur (Jo, 2019; Cahyani, 2022; Qamar & Raza, 2024), tahapan *preprocessing* umumnya mencakup proses berikut:

1) *Cleansing* merupakan proses pembersihan data dari karakter yang tidak diperlukan untuk mengurangi *noise*.
2) *Case folding* merupakan proses mengonversi seluruh huruf dalam dokumen menjadi huruf kecil serta menghilangkan karakter yang bukan merupakan huruf.
3) *Tokenization* atau tokenisasi merupakan proses pemisahan teks menjadi bentuk kata tunggal yang disebut token.

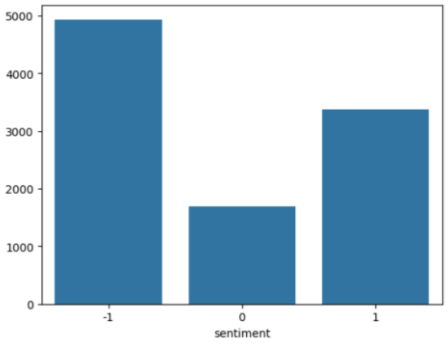
4) *Stopword removal* merupakan tahap eliminasi kata-kata umum atau fungsi yang tidak memiliki kontribusi signifikan terhadap konteks makna dalam teks.
5) *Stemming* merupakan proses pengembalian kata ke bentuk dasar dengan cara menghilangkan afiks, seperti awalan, akhiran, sisipan, maupun gabungan keduanya.

2.4. Transformasi

Pada tahap transformasi, dilakukan dua proses utama, yaitu pelabelan sentimen dan pembobotan teks menggunakan TF-IDF. Pelabelan dilakukan dengan pendekatan *lexicon-based* menggunakan *NRC Emoticon Lexicon*, yang memetakan setiap kata ke dalam polaritas positif, netral, atau negatif melalui pemberian skor +1, 0, –1 (Purwitasari dkk., 2023). Jumlah skor dari seluruh kata dalam ulasan menentukan kategori sentimen akhir, sehingga proses analisis dapat dialihkan dari *unsupervised learning* menjadi *supervised learning* dengan tiga kelas, yaitu negatif, netral, dan positif. Setelah label diperoleh, teks diubah menjadi representasi numerik menggunakan metode *Term Frequency–Inverse Document Frequency* (TF-IDF). *Term Frequency* (TF) menggambarkan tingkat kemunculan suatu kata dalam dokumen, sedangkan *Inverse Document Frequency* (IDF) menunjukkan tingkat kelangkaan kata tersebut di seluruh dokumen (Qamar & Raza, 2024). Kombinasi keduanya menghasilkan bobot TF-IDF yang menilai tingkat relevansi setiap kata. Bobot yang tinggi menandakan bahwa kata tersebut memiliki kontribusi penting dalam menggambarkan isi ulasan, sehingga representasi ini efektif untuk proses klasifikasi sentimen berikutnya.

2.5. *Data Mining*

Pada tahap *data mining*, proses klasifikasi sentimen dilakukan dengan membangun model menggunakan algoritma *XGBoost* dan *Random Forest*. Setelah ulasan direpresentasikan ke dalam vektor TF-IDF, data dibagi menjadi data latih dan data uji untuk memastikan evaluasi model yang objektif. *XGBoost* adalah algoritma *gradient boosting decision tree* yang memanfaatkan *gradient* dan *hessian* untuk memperbarui model secara bertahap, sehingga efektif untuk klasifikasi multi-kelas seperti analisis sentimen (Widiarta dkk., 2023). Sementara itu, *Random Forest* bekerja dengan membangun banyak *decision tree* secara paralel melalui teknik *bagging*, sehingga menghasilkan prediksi yang stabil pada proses klasifikasi (Lestari, 2024). Kedua algoritma ini kemudian dilatih untuk mempelajari pola hubungan antara teks ulasan dan kategori sentimennya. Dari pengolahan *dataset*, diperoleh distribusi sentimen yaitu 3.374 positif, 1.692 netral, dan 4.934 negatif, yang menggambarkan kecenderungan persepsi pengguna terhadap aplikasi iPusnas. Gambar 1 menyajikan distribusi kategori sentimen yang dihasilkan dari proses pelabelan data.



Gambar 1. Distribusi sentimen ulasan pengguna iPusnas

2.6. Evaluasi

Tahap evaluasi dilakukan untuk menilai performa model klasifikasi yang dihasilkan oleh *XGBoost* dan *Random Forest*. Evaluasi dilakukan menggunakan *confusion matrix* yang menjadi

dasar perhitungan beberapa metrik kinerja, yaitu akurasi, *precision*, *recall*, dan *F1-score* (Cahyani, 2022). Penggunaan metrik tersebut bertujuan untuk mengukur ketepatan model dalam mengklasifikasi sentimen dan memastikan bahwa model tidak hanya akurat secara keseluruhan, tetapi juga mampu memberikan hasil yang seimbang dalam membedakan kelas sentimen positif, netral, dan negatif. Tabel 1 menyajikan bentuk *confusion matrix* yang dapat digunakan dalam proses evaluasi.

Tabel 1. *Confusion Matrix*

Confusion Matrix		Prediksi	
		Benar	Salah
Aktual	Benar	*True Positive* (TP)	*False Negative* (FN)
	Salah	*False Positive* (FP)	*True Negative* (TN)

Sumber: (Cahyani, 2022)

Pengujian menggunakan *confusion matrix* menghasilkan perhitungan yang menghasilkan empat keluaran, antara lain seperti pada Tabel 2.

Tabel 2. *Metric* pengukuran untuk evaluasi (Cahyani, 2022)

Metric	Rumus
Akurasi	$\text{Akurasi} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$
Recall	$Recall = \frac{\text{TP}}{\text{TP} + \text{FN}}$
Precision	$Precision = \frac{\text{TP}}{\text{TP} + \text{FP}}$
F-Measure	$F - Measure = 2 \times \frac{Recall \times Precision}{Recall + Precision}$

Sumber: (Cahyani, 2022)

3 HASIL DAN PEMBAHASAN

3.1. Data

Penelitian ini memanfaatkan data sekunder yang diperoleh secara langsung dari internet. Pengumpulan data dilakukan melalui teknik *web scraping* menggunakan bahasa pemrograman Python pada platform *Google Colab*. Tahap awal dilakukan dengan mengakses laman resmi *Google Play Store* untuk aplikasi "iPusnas" dengan nama paket "mam.reader.ipusnas". Selanjutnya, data yang dikumpulkan difokuskan pada ulasan pengguna aplikasi tersebut. Sebanyak 10.000 sampel ulasan pengguna diambil secara acak (*random sampling*) untuk digunakan dalam penelitian ini. Data ulasan tersebut kemudian dimanfaatkan sebagai bahan analisis dalam komparasi kinerja algoritma *XGBoost* dan *Random Forest* dalam mengklasifikasikan sentimen pengguna aplikasi iPusnas. Berikut merupakan hasil pengumpulan data yang diperoleh melalui proses *web scraping* menggunakan *Google Colab*.

	reviewId	userName	userImage	content	score	thumbsUpCount	reviewCreatedVersion	at	replyContent	repliedAt	appVersion
0	c1f90eb4-4a75-406b-951a-7dd7c461d9f3	Yuliana Martha Tresia	https://play-lh.googleusercontent.com/a-/ALV-U...	Setelah disuruh memperbaharui app, malah app-n...	1	1505	2.0.5	2025-09-04 06:51:36	None	NaT	2.0.5
1	28eefcfe-d727-45cb-ba1d-e2dc0da8b474	Laily Khoirina	https://play-lh.googleusercontent.com/a/ACg8oc...	Aplikasinya bermanfaat, tapi kenapa aplikasiny...	1	43	2.0.5	2025-10-10 04:54:55	None	NaT	2.0.5
2	2f1ff843-b473-4be0-a6cc-93819835b748	Gading P.C	https://play-lh.googleusercontent.com/a-/ALV-U...	Fitur filternya gak bisa dipakai loading terus...	1	0	2.0.5	2025-10-15 12:11:21	None	NaT	2.0.5
3	35874bcd-175f-4319-9921-e5fee550a0b3	Anggun Julitaaa	https://play-lh.googleusercontent.com/a/ACg8oc...	Kenapa ga segera di perbarui sih fiturnya? gim...	1	14	2.0.5	2025-10-09 02:23:44	None	NaT	2.0.5
4	7fbd47ff-02cf-46cc-8df4-2e15cfce4636	Vira Pradita	https://play-lh.googleusercontent.com/a-/ALV-U...	Aplikasi iPusnas sebenarnya sangat bermanfaat ...	2	21	2.0.5	2025-10-02 11:41:50	None	NaT	2.0.5
5	ffed749e-30ad-4ffb-becb-5399ba22a5fd	Ni Putu Ayu Maharani Putri	https://play-lh.googleusercontent.com/a/ACg8oc...	Error terus akhir-akhir ini, beberapa buku tid...	1	9	2.0.5	2025-10-10 15:23:21	None	NaT	2.0.5
6	a7fe607a-ff12-4b1e-9e3c-cb59649bb0f4	Retno Astuti	https://play-lh.googleusercontent.com/a/ACg8oc...	Sedih banget, sudah lama antri buku, saat suda...	1	28	2.0.5	2025-10-05 15:51:05	None	NaT	2.0.5
7	4d22b910-aadf-4095-892e-3151908b4fed	Muhammad Wahyu	https://play-lh.googleusercontent.com/a-/ALV-U...	Awalnya senang dengan aplikasi ini karena dapa...	1	2	2.0.5	2025-10-09 18:24:16	None	NaT	2.0.5
8	95231e1c-20f6-4c09-bb7d-670609f315e5	francisca adita	https://play-lh.googleusercontent.com/a-/ALV-U...	Sebelum diperbarui tampilannya ipusnas salah s...	1	31	2.0.5	2025-10-06 00:33:04	None	NaT	2.0.5
9	5d58bfe1-c6e2-4842-bc5c-8b1d0b4b37fa	Melinda Elvira Kalina	https://play-lh.googleusercontent.com/a/ACg8oc...	Semenjak di update, saya jadi ga bisa lagi bac...	1	397	2.0.5	2025-09-05 07:23:23	None	NaT	2.0.5
10	3652e864-6157-4762-b72d-b07c7d09fc53	Ilmi Nurchasanah	https://play-lh.googleusercontent.com/a-/ALV-U...	Maaf jadi nurunin rating. Biasanya kalau di up...	1	16	2.0.5	2025-09-23 02:21:56	None	NaT	2.0.5
11	eb3709ff-008f-4925-b646-477b30d6e054	Nabilah Nur Lathifah	https://play-lh.googleusercontent.com/a/ACg8oc...	pas pertama kali download app nya lancar" aja,...	1	298	2.0.5	2025-09-01 13:16:56	None	NaT	2.0.5
12	cf69c786-83a1-4677-866e-7a1a10f75ed5	Candra Lesmana	https://play-lh.googleusercontent.com/a-/ALV-U...	Seharusnya pengelola bisa lebih bijak dan tang...	2	49	2.0.5	2025-09-11 10:45:39	None	NaT	2.0.5
13	5be48cd3-6806-40d3-883e-2ccc80a743e6	Dwi Sumitry Warahmah	https://play-lh.googleusercontent.com/a-/ALV-U...	susah di akses, susah log in, mau baca keluar ...	1	85	2.0.5	2025-08-05 14:34:31	None	NaT	2.0.5
14	2f9d37f7-fd62-4533-9b9c-c52566332e26	Irfan Giffary	https://play-lh.googleusercontent.com/a-/ALV-U...	setelah update terbaru kenapa buku yang sudah ...	1	10	2.0.5	2025-08-17 07:01:33	None	NaT	2.0.5

Gambar 2. Hasil *scraping* data ulasan pengguna iPusnas

3.2. Seleksi Data

Data yang digunakan dalam penelitian ini berasal dari ulasan pengguna aplikasi iPusnas yang bersifat data publik dengan sebanyak 10.000 data. Dari sejumlah atribut yang diperoleh melalui proses *web scraping*, hanya dua atribut utama yang digunakan, yaitu *content* dan *score*. Proses penyaringan dilakukan karena atribut lainnya tidak relevan dengan fokus penelitian yang berfokus pada analisis sentimen terhadap ulasan pengguna aplikasi iPusnas. Berikut merupakan hasil penyaringan data yang telah dilakukan.

	content	score
0	Setelah disuruh memperbaharui app, malah app-n...	1
1	kenapa errornya konsisten banget? tolong bange...	3
2	Fitur filternya gak bisa dipakai loading terus...	1
3	haloo iPusnas, sebenernya aplikasinya sudah ba...	4
4	tidak bisa mendownload buku, padahal sudah dip...	1

Gambar 3. Hasil seleksi data ulasan pengguna iPusnas

3.3. *Preprocessing*

Pada tahap *preprocessing*, data ulasan yang telah dikumpulkan dan dilabeli diproses melalui lima tahapan agar siap digunakan pada langkah berikutnya. Tahapan tersebut meliputi *cleansing*, *case folding*, *tokenizing*, *stopword removal*, dan *stemming*. Seluruh proses *preprocessing* dilakukan dengan memanfaatkan platform *Google Colaboratory* menggunakan bahasa pemrograman Python.

a. *Cleansing*

Pada tahap ini dilakukan proses *cleansing* untuk menghapus elemen-elemen yang tidak relevan dengan proses klasifikasi sentimen. Data hasil *scraping* masih mengandung berbagai unsur seperti URL, tag pengguna, tanda pagar, angka, karakter khusus, serta simbol yang tidak memiliki makna dalam analisis sentimen. Oleh karena itu, dilakukan proses pembersihan teks menggunakan pendekatan *regular expression (regex)*. Tahap ini bertujuan mengurangi *noise*

pada data dan memastikan bahwa teks hanya berisi karakter alfabet yang bermakna sehingga dapat digunakan secara optimal pada proses *preprocessing* berikutnya.

b. *Case Folding*

Pada tahap *case folding*, seluruh teks dalam data ulasan diubah menjadi huruf kecil (*lowercase*) untuk menghindari perbedaan makna antara kata yang sama tetapi berbeda penulisan huruf besar atau kecil. Selain itu, pada tahap ini juga dilakukan penghapusan elemen-elemen non-huruf seperti tanda baca, simbol, dan angka menggunakan pola *regular expression (regex)*. Dengan demikian, teks menjadi lebih konsisten dan siap diproses pada tahap tokenisasi berikutnya. Pada Tabel 3 ditunjukkan perubahan teks ulasan sebelum dan sesudah proses *case folding*, sehingga teks menjadi seragam (*lowercase*) dan siap untuk proses tokenisasi.

Tabel 3. Hasil *case folding*

Sebelum	Sesudah
Setelah disuruh memperbaharui app, malah app-nya error mulu dan berulang kali berhenti/nutup sendiri sampai usernya frustasi. Buka app aja gak bisa karena ketutup sendiri mulu, gimana lagi mau baca bukunya? Gimana developernya? Kok app diperbaharui bukannya jadi lebih oke, kok malah makin error? Tim iPusnas, minta tolong sekali bagaimana perbaikan pelayanannya, masyarakat mengandalkan layanan perpus digital ini lho.	setelah disuruh memperbaharui app malah app nya error mulu dan berulang kali berhenti nutup sendiri sampai usernya frustasi buka app aja gak bisa karena ketutup sendiri mulu gimana lagi mau baca bukunya gimana developernya kok app diperbaharui bukannya jadi lebih oke kok malah makin error tim ipusnas minta tolong sekali bagaimana perbaikan pelayanannya masyarakat mengandalkan layanan perpus digital ini lho
haloo iPusnas, sebenernya aplikasinya sudah bagus, tapi sayang koleksi bukunya jarang diperbarui. Banyak buku akademik yang saya butuhkan tidak tersedia. Semoga ke depannya iPusnas bisa lebih rutin menambah dan memperbarui koleksi agar mudah mencari buku akademik atau buku yang lain, mengingat ini kan aplikasi resmi dari perpustakaan nasional.	haloo ipusnas sebenernya aplikasinya sudah bagus tapi sayang koleksi bukunya jarang diperbarui banyak buku akademik yang saya butuhkan tidak tersedia semoga ke depannya ipusnas bisa lebih rutin menambah dan memperbarui koleksi agar mudah mencari buku akademik atau buku yang lain mengingat ini kan aplikasi resmi dari perpustakaan nasional

c. *Tokenization*

Pada tahap tokenisasi, teks yang telah melalui proses *cleansing* dan *case folding* dipecah menjadi potongan-potongan kata yang disebut token. Proses ini dilakukan dengan menggunakan *RegexpTokenizer* dari pustaka *Natural Language Toolkit* (NLTK), yang berfungsi memisahkan teks berdasarkan pola huruf dan mengabaikan tanda baca maupun karakter yang bukan alfabet. Tahap ini bertujuan untuk mempermudah analisis dan pengolahan teks pada langkah-langkah *preprocessing* selanjutnya, seperti *stopword removal* dan *stemming*. Tabel 4 menampilkan contoh hasil tokenisasi, yaitu perubahan teks ulasan menjadi daftar token (kata-kata) setelah melalui proses *cleansing* dan *case folding*, sehingga teks lebih siap untuk tahapan praproses berikutnya.

Tabel 4. Hasil tokenisasi

Sebelum	Sesudah
setelah disuruh memperbaharui app malah app nya error mulu dan berulang kali berhenti nutup sendiri sampai usernya	['disuruh', 'memperbaharui', 'app', 'app', 'nya', 'error', 'mulu', 'berulang', 'kali', 'berhenti', 'nutup', 'usernya', 'frustasi', 'buka', 'app', 'aja',

Sebelum	Sesudah
frustasi buka app aja gak bisa karena ketutup sendiri mulu gimana lagi mau baca bukunya gimana developernya kok app diperbaharui bukannya jadi lebih oke kok malah makin error tim ipusnas minta tolong sekali bagaimana perbaikan pelayanannya masyarakat mengandalkan layanan perpus digital ini lho	'gak', 'ketutup', 'mulu', 'gimana', 'baca', 'bukunya', 'gimana', 'developernya', 'app', 'diperbaharui', 'oke', 'error', 'tim', 'ipusnas', 'tolong', 'perbaikan', 'pelayanannya', 'masyarakat', 'mengandalkan', 'layanan', 'perpus', 'digital', 'lho']
haloo ipusnas sebenernya aplikasinya sudah bagus tapi sayang koleksi bukunya jarang diperbarui banyak buku akademik yang saya butuhkan tidak tersedia semoga ke depannya ipusnas bisa lebih rutin menambah dan memperbarui koleksi agar mudah mencari buku akademik atau buku yang lain mengingat ini kan aplikasi resmi dari perpustakaan nasional	['haloo', 'ipusnas', 'sebenernya', 'aplikasinya', 'bagus', 'sayang', 'koleksi', 'bukunya', 'jarang', 'diperbarui', 'buku', 'akademik', 'butuhkan', 'tersedia', 'semoga', 'depannya', 'ipusnas', 'rutin', 'menambah', 'memperbarui', 'koleksi', 'mudah', 'mencari', 'buku', 'akademik', 'buku', 'aplikasi', 'resmi', 'perpustakaan', 'nasional']

d. *Stopword Removal*

Pada tahap *stopword removal*, dilakukan penghapusan kata-kata umum yang tidak memiliki makna signifikan dalam analisis sentimen, seperti “yang”, “dan”, “di”, atau “ke”. Proses ini menggunakan daftar *stopword* bahasa Indonesia dari pustaka *Natural Language Toolkit* (NLTK). Setiap token dalam teks difilter untuk menghilangkan kata-kata tersebut, sehingga hanya kata-kata yang memiliki makna penting dalam analisis sentimen yang dipertahankan.

e. *Stemming*

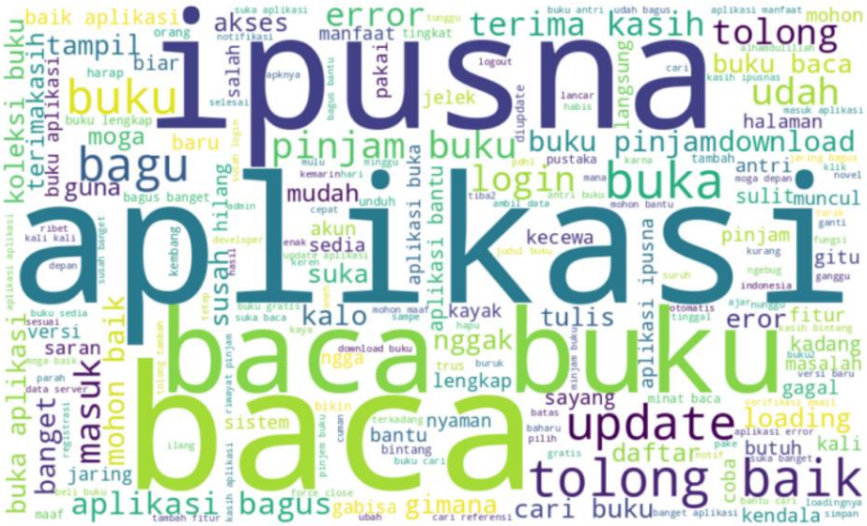
Pada tahap *stemming*, setiap kata yang telah melalui proses *tokenizing* dan *stopword removal* dikembalikan ke bentuk dasarnya agar maknanya lebih seragam. Proses ini dilakukan menggunakan pustaka Sastrawi, yang mengubah kata berimbuhan menjadi bentuk dasarnya, misalnya “membaca” menjadi “baca”. Tahap ini bertujuan untuk mengurangi variasi kata dan meningkatkan konsistensi data teks, sehingga hasil analisis sentimen menjadi lebih akurat. Tabel 5 menyajikan hasil stemming pada data ulasan, yang menunjukkan perbandingan token sebelum dan sesudah dikembalikan ke bentuk kata dasar untuk mengurangi variasi kata dan meningkatkan konsistensi representasi teks.

Tabel 5. Hasil *stemming*

Sebelum	Sesudah
['disuruh', 'memperbaharui', 'app', 'app', 'nya', 'error', 'mulu', 'berulang', 'kali', 'berhenti', 'nutup', 'usernya', 'frustasi', 'buka', 'app', 'aja', 'gak', 'ketutup', 'mulu', 'gimana', 'baca', 'bukunya', 'gimana', 'developernya', 'app', 'diperbaharui', 'oke', 'error', 'tim', 'ipusnas', 'tolong', 'perbaikan', 'pelayanannya', 'masyarakat', 'mengandalkan', 'layanan', 'perpus', 'digital', 'lho']	['suruh', 'baharu', 'app', 'app', 'nya', 'error', 'mulu', 'ulang', 'kali', 'henti', 'nutup', 'usernya', 'frustasi', 'buka', 'app', 'aja', 'gak', 'tutup', 'mulu', 'gimana', 'baca', 'buku', 'gimana', 'developer', 'app', 'baharu', 'oke', 'error', 'tim', 'ipusnas', 'tolong', 'baik', 'layan', 'masyarakat', 'andal', 'layan', 'pus', 'digital', 'lho']
['haloo', 'ipusnas', 'sebenernya', 'aplikasinya', 'bagus', 'sayang', 'koleksi', 'bukunya', 'jarang',	['haloo', 'ipusnas', 'sebenernya', 'aplikasi', 'bagus', 'sayang', 'koleksi', 'buku', 'jarang',

'diperbarui', 'buku', 'akademik', 'butuhkan', 'tersedia', 'semoga', 'depannya', 'ipusnas', 'rutin', 'menambah', 'memperbarui', 'koleksi', 'mudah', 'mencari', 'buku', 'akademik', 'buku', 'aplikasi', 'resmi', 'perpustakaan', 'nasional']	'baru', 'buku', 'akademik', 'butuh', 'sedia', 'moga', 'depan', 'ipusnas', 'rutin', 'tambah', 'baru', 'koleksi', 'mudah', 'cari', 'buku', 'akademik', 'buku', 'aplikasi', 'resmi', 'pustaka', 'nasional']

Setelah proses *stemming*, dilakukan visualisasi frekuensi kata menggunakan *wordcloud* untuk mengidentifikasi kata yang paling sering muncul dalam ulasan pengguna. Tahapan ini bertujuan untuk memberikan gambaran umum mengenai tema dominan dalam teks sebelum dilakukan proses klasifikasi sentimen.



Gambar 4. Hasil *wordcloud*

3.4. Transformasi

Tahap transformasi bertujuan untuk mengubah data teks menjadi representasi numerik yang dapat diolah pada tahap *data mining*. Pada tahap ini, data dibagi ke dalam dua skenario pembagian, yaitu skenario pertama dengan proporsi 80% data latih dan 20% data uji, serta skenario kedua dengan proporsi 90% data latih dan 10% data uji. Selanjutnya, pembobotan dilakukan menggunakan metode *Term Frequency–Inverse Document Frequency* (TF-IDF) untuk merepresentasikan teks ulasan dalam bentuk vektor numerik berdasarkan frekuensi kemunculan kata dan tingkat kepentingan kata dalam kumpulan ulasan. Tabel 6 memperlihatkan hasil pembobotan fitur menggunakan TF-IDF, berupa nilai *Term Frequency* (TF) dan bobot TF-IDF pada beberapa kata/term dominan, sebagai representasi numerik teks yang digunakan pada proses klasifikasi sentimen.

Tabel 6. Hasil TF-IDF

Term	TF	TF-IDF
buku	0,7743	0,075316
aplikasi	0,5299	0,064315
baca	0,3848	0,046929
baik	0,2354	0,039534
buka	0,1941	0,039042

3.5. *Data Mining*

Pada tahap *data mining*, dilakukan klasifikasi sentimen dengan menerapkan dua algoritma *machine learning*, yaitu *Random Forest* dan *Extreme Gradient Boosting (XGBoost)*,

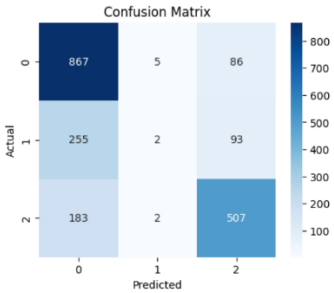
melalui dua skenario pembagian data. Skenario pertama menggunakan proporsi 80% data latih dan 20% data uji, sedangkan skenario kedua menggunakan 90% data latih dan 10% data uji. Kedua model dibangun berdasarkan representasi fitur *Term Frequency–Inverse Document Frequency* (TF-IDF) yang dihasilkan pada tahap transformasi. Dalam proses pelatihan, algoritma *Random Forest* menggunakan 200 pohon keputusan *(n_estimators = 200)* dengan *random state* 42 untuk menjaga reprodusibilitas hasil, sementara *XGBoost* menerapkan metrik evaluasi *multi-class logarithmic loss* (mlogloss) dan melakukan penyesuaian label sentimen dari (-1, 0, 1) menjadi (0, 1, 2) agar selaras dengan format input model.

3.6. Evaluasi

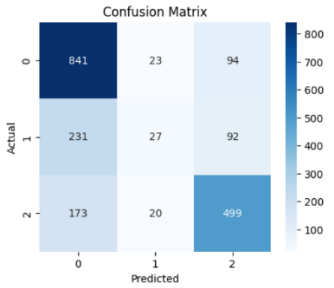
Tabel 7. Hasil metrik model *Random Forest* dan *XGBoost*

Model	*Split*	*Accuracy*	*Precision*	*Recall*	*F1-Score*
Random Forest	80:20	**0,688**	0,542	0,548	0,504
XGBoost	80:20	0,684	**0,597**	**0,559**	**0,539**
Random Forest	90:10	0,681	0,518	0,543	0,500
XGBoost	90:10	**0,688**	**0,644**	**0,567**	**0,552**

Berdasarkan hasil evaluasi pada Tabel 7, performa model menunjukkan variasi antara *Random Forest* dan *XGBoost* pada dua skenario pembagian data. Pada skenario 80:20, model *Random Forest* memperoleh akurasi sebesar 0,688, sedangkan *XGBoost* memiliki akurasi yang sedikit lebih rendah yaitu 0,684. Meskipun demikian, *XGBoost* menunjukkan kinerja lebih baik pada metrik presisi (0,597), *recall* (0,559), dan *F1-score* (0,539) dibandingkan *Random Forest*. Hal ini mengindikasikan bahwa *XGBoost* lebih mampu mengenali pola sentimen secara seimbang antar kelas, meskipun tidak memberikan akurasi tertinggi pada skenario ini.



Gambar 5. *Confusion Matrix Random Forest Split* 80:20

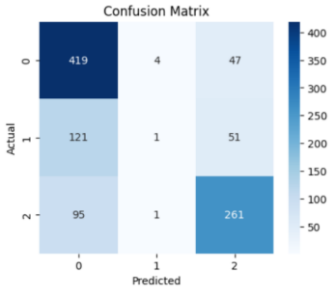


Gambar 6. *Confusion Matrix XGBoost Split* 80:20

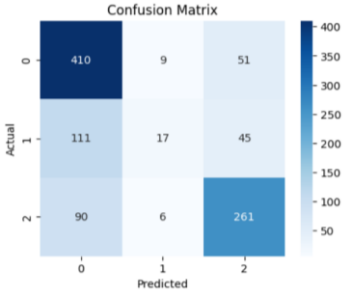
Model *Random Forest split* 80:20 menunjukkan performa yang sangat tidak seimbang. Sentimen negatif dan positif berhasil diprediksi dengan relatif baik, namun model gagal total mendeteksi sentimen netral. Sebanyak 255 data netral salah diklasifikasikan sebagai negatif, hal ini menguatkan bukti bias model yang kuat terhadap kelas negatif. Hanya 2 data netral yang berhasil diprediksi dengan benar dari seluruh data netral yang ada. Sementara itu, *XGBoost split* 80:20 menunjukkan performa yang tidak seimbang. Model ini unggul prediksi negatif dan positif, namun gagal deteksi sentimen netral. Sebanyak 231 data netral salah diklasifikasikan sebagai negatif, hanya 27 prediksi benar. Model ini menunjukkan bias yang kuat terhadap kelas negatif. Keunggulan keduanya datang dari peningkatan kinerja pada kelas negatif dan positif, sementara kelas netral tetap menjadi titik lemah yang signifikan.

Pada skenario 90:10, *XGBoost* memberikan hasil yang lebih unggul dibandingkan *Random Forest* di seluruh metrik evaluasi. *XGBoost* mencapai akurasi sebesar 0,688 dengan

presisi (0,644), *recall* (0,567), dan *F1-score* (0,552), yang menunjukkan peningkatan performa ketika proporsi data latih diperbesar. Sebaliknya, *Random Forest* menunjukkan akurasi 0,681 dengan nilai *F1-score* yang lebih rendah (0,500), menandakan bahwa model ini kurang optimal dalam menangani variasi sentimen terutama pada data latih yang lebih besar.



Gambar 7. *Confusion Matrix Random Forest Split* 90:10



Gambar 8. *Confusion Matrix XGBoost Split* 90:10

Model *Random Forest split* 90:10 menunjukkan performa yang tidak seimbang. Sentimen negatif dan positif berhasil diprediksi dengan baik, namun model gagal total mendeteksi sentimen netral. Sebanyak 121 data netral salah diklasifikasikan sebagai negatif, hanya 1 prediksi benar. Hal ini menguatkan bukti bias model terhadap kelas negatif. Sementara itu, *XGBoost split* 90:10 menunjukkan performa yang tidak seimbang. Sentimen negatif dan positif diprediksi dengan cukup baik, namun model kembali gagal mendeteksi sentimen netral. Sebanyak 111 data netral salah diklasifikasikan sebagai negatif, hanya 17 prediksi benar. Hal ini mengkonfirmasi bias model terhadap kelas negatif. Keunggulan keduanya datang dari peningkatan kinerja pada kelas negatif dan positif, sementara kelas netral tetap menjadi titik lemah yang signifikan.

Secara keseluruhan, hasil evaluasi menunjukkan bahwa *XGBoost* memiliki performa yang lebih unggul dan konsisten dari *Random Forest*, terutama pada metrik-metrik yang menggambarkan kemampuan model dalam menyeimbangkan prediksi setiap kelas, seperti *recall* dan *F1-score*. Performa yang lebih baik dari *XGBoost* kemungkinan dipengaruhi oleh kemampuannya dalam memanfaatkan informasi *gradient* dan *hessian*, sehingga pembaruan model pada setiap iterasi menjadi lebih stabil dan efektif dalam mengenali pola sentimen yang kompleks pada teks ulasan.

4 KESIMPULAN

Berdasarkan penerapan dua model klasifikasi, yaitu *XGBoost* dan *Random Forest*, evaluasi dilakukan menggunakan metrik *accuracy*, *precision*, *recall*, dan *F1-score* pada skenario pembagian data 80:20 dan 90:10. Pada skenario 80:20, Random Forest mencatat *accuracy* tertinggi sebesar 0,688, sedangkan XGBoost menunjukkan kinerja yang lebih baik pada *precision*, *recall*, dan *F1-score*. Pada skenario 90:10, XGBoost kembali unggul dengan *accuracy* sebesar 0,688 dan nilai *precision*, *recall*, serta *F1-score* yang lebih tinggi dibandingkan *Random Forest*. Hal ini mengindikasikan bahwa *XGBoost* lebih stabil dalam mengenali pola sentimen dan lebih adaptif terhadap peningkatan proporsi data latih. Analisis *wordcloud* menunjukkan bahwa kata “buku”, “baca”, dan “aplikasi” memiliki frekuensi kemunculan tertinggi dalam ulasan pengguna. Hal ini menunjukkan bahwa ulasan pengguna didominasi oleh pembahasan mengenai fungsi utama iPusnas sebagai aplikasi perpustakaan digital, khususnya dalam menyediakan akses terhadap koleksi buku dan mendukung aktivitas membaca.

DAFTAR PUSTAKA

Atika, M., & Sayekti, R. (2023). Studi Literatur Review Sistem Informasi Perpustakaan Berbasis Artificial Intelligence (AI) Library Information System Based on Artificial Intelligence (AI): Literatur Review. *Journal of Information and Library Science, 14*(1). https://doi.org/10.20473/pjil.v14i1.46405

Natalea, D. I., & Christiani, L. (2020). Analisis tingkat kepuasan pengguna dalam pemanfaatan aplikasi perpustakaan digital Kabupaten Wonosobo. *Jurnal Ilmu Perpustakaan, 8*(2), 112-120. https://doi.org/10.14710/jip.v8i2.112-120

Septiani, A., & Budi, I. (2022). Klasifikasi Ulasan Pengguna Aplikasi: Studi Kasus Aplikasi Ipusnas Perpustakaan Nasional Republik Indonesia (PNRI). *JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika), 7*(4), 1110-1120. https://doi.org/ 10.29100/jipi.v7i4.3216

Aprianti, Y., Hananto, A. L., & Hilabi, S. S. (2025). Klasifikasi Sentimen Komentar Pengguna pada Aplikasi Ruangguru Menggunakan Algoritma Naive Bayes. *METIK JURNAL (AKREDITASI SINTA 3), 9*(1), 101-110. https://doi.org/10.47002/metik.v9i1.1023

Sarah, M. S., Saepudin, E., & Anwar, R. K. (2023). Hubungan Kualitas Sistem Informasi Aplikasi Ipusnas Dengan Kepuasan Pemustaka Menggunakan Pieces Framework. *Jurnal Ilmiah Multidisiplin, 2*(6), 29-39. https://doi.org/10.56127/jukim.v2i6.968

Hasibuan, E., & Heriyanto, E. A. (2022). Analisis Sentimen Pada Ulasan Aplikasi Amazon Shopping Di Google Play Store Menggunakan Naive Bayes Classifier. *Jurnal Teknik dan Science, 1*(3), 13-24. https://doi.org/10.56127/jts.v1i3.434

PRASETYA, F., & Rosa, P. H. P. (2024, September). Implementasi Algoritma XGBOOST Untuk Klasifikasi Kegagalan Pembayaran Kredit Nasabah Bank. *In Prosiding Seminar Nasional Informatika Bela Negara* (Vol. 4, pp. 110-115). https://santika.upnjatim.ac.id/submissions/index.php/santika/article/view/506

Firzatullah, F. N., & Nuroji, N. (2025). Analisis Sentimen Pengguna Aplikasi Byond BSI Pada Google Play Store Menggunakan Algoritma SVM Dan Random Forest. *METIK JURNAL (AKREDITASI SINTA 3), 9*(2), 356-363. https://doi.org/10.47002/6vtc4567

Cahyani, L. (2022). *Aplikasi text mining di bidang pendidikan*. CV. Literasi Nusantara Abadi.

Qamar, U., & Raza, M. S. (2024). *Applied Text Mining*. Springer.

Jo, T. (2019). *Text mining*. Studies in Big Data, 45, 1-373.

Purwitasari, D., Ansyah, A. S. S., Kurniawan, A. P., & Kholifah, A. N. (2023). A hybrid method on emotion detection for Indonesian tweets of COVID-19. *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi), 7*(2), 254-262. https://doi.org/10.29207/resti.v7i2.4816

Widiarta, I. P. A. P., Dwiyansaputra, R., & Aranta, A. (2023). Analisis Sentimen Masyarakat Terhadap Kebijakan Penerapan Ppkm Di Media Sosial Twitter Dengan Menggunakan Metode Xgboost. *Jurnal Teknologi Informasi, Komputer, dan Aplikasinya (JTIKA), 5*(2), 154-163. https://jtika.if.unram.ac.id/index.php/JTIKA/article/view/342

Lestari, S. (2024). Prediction of Stunting in Toddlers Using Bagging and Random Forest Algorithms. *Sinkron: jurnal dan penelitian teknik informatika, 8*(2), 947-955. https://doi.org/10.33395/sinkron.v8i2.13448