Vol. 6 No.1 Tahun 2025
Jurnal Kecerdasan Buatan, Komputasi dan Teknologi Informasi
https://ejournal.unuja.ac.id/index.php/core
E-ISSN : 2774-7875 and P-ISSN : 2775-0124

Sentiment Analysis ff Netflix Review on Google Play with Machine Learning

Andi Wijaya [1], Muhammad Alfin Dwiyanto [2], Zainul Muttaqin [3], M. Alvin Fikrul Haq [4]
[1,2,3,4] Fakultas Teknik, Universitas Nurul Jadid, Indonesia

Info Artikel

Riwayat Artikel
Diterima: 19-05-2025
Disetujui: 25-06-2025

Kata Kunci
Netflix;
Analisis Sentimen;
Naive Bayes;
Support Vector Machine;

mr.andiwijaya@gmail.com

ABSTRAK

Penelitian ini bertujuan mengklasifikasikan sentimen pengguna terhadap aplikasi *Netflix* di *Google Play Store*, mengatasi tantangan volume ulasan dan keragaman bahasa. Analisis sentimen otomatis diperlukan untuk mengekstraksi informasi emosional dari teks ulasan. Sebanyak 20.000 ulasan dikumpulkan menggunakan web *scraping*. Data kemudian melalui tahap praproses seperti *case folding, cleaning, tokenizing*, dan *stopword removal*. Klasifikasi sentimen berbasis leksikon dilakukan dengan menerjemahkan teks Indonesia ke Inggris menggunakan *Google Translate*, lalu dianalisis VADER *Sentiment Intensity Analyzer*. Pembobotan kata diterapkan menggunakan TF-IDF untuk menentukan signifikansi kata dalam dokumen. Perbandingan hasil klasifikasi sentimen dilakukan menggunakan algoritma *Support Vector Machine* (SVM) dan *Naive Bayes*. Hasil menunjukkan SVM mencapai akurasi 82%, presisi 83%, *recall* 89%, dan skor F1 86%. Naive Bayes menghasilkan akurasi 76%, presisi 74%, recall 94%, dan skor F1 83%. Secara keseluruhan, SVM terbukti lebih unggul dalam memberikan klasifikasi sentimen yang seimbang dan akurat dalam studi ini.

1. PENDAHULUAN (11pt, bold)

Perkembangan teknologi digital dan penetrasi internet yang masif telah mengubah cara masyarakat mengonsumsi hiburan. Salah satu platform yang paling populer dalam industri hiburan digital adalah *Netflix*, yang menyediakan layanan *streaming* film dan serial televisi secara *daring*. Seiring dengan meningkatnya jumlah pengguna, ulasan pengguna *(user reviews)* di *platform* distribusi aplikasi seperti *Google Play Store* menjadi sumber informasi penting untuk mengevaluasi kepuasan, ekspektasi, dan persepsi konsumen terhadap aplikasi *Netflix*.

Ulasan tersebut tidak hanya memberikan gambaran tentang kualitas teknis aplikasi, namun juga mencerminkan opini dan sentimen pengguna yang bersifat subjektif. Namun demikian, tingginya volume dan keanekaragaman bahasa dalam ulasan menjadikan analisis manual menjadi tidak efisien dan rentan terhadap bias. Oleh karena itu, diperlukan pendekatan otomatis seperti analisis sentimen untuk mengekstraksi informasi emosional dari teks ulasan secara sistematis.

Analisis sentimen merupakan cabang dari *Natural Language Processing* (NLP) yang bertujuan untuk mengklasifikasikan opini menjadi sentimen positif, negatif, atau netral [1][2]. Dalam penelitian ini, machine learning digunakan sebagai metode utama untuk mengklasifikasikan data, khususnya dengan menerapkan algoritma *Naive Bayes* dan *Support Vector Machine (SVM).* Algoritma ini dikenal memiliki performa yang baik dalam klasifikasi teks[3] dan telah banyak diterapkan dalam studi analisis sentimen aplikasi lain seperti *mobile banking* [4], hotel *booking* [5], dan aplikasi pemebelajaran daring [6].

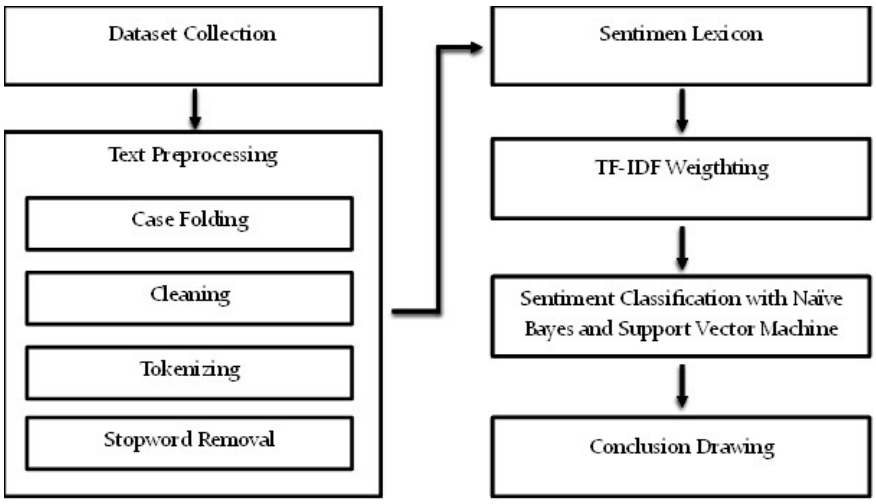
Studi sebelumnya oleh Ranjan dan Mishra (2020) menunjukkan bahwa SVM dengan fitur TF-IDF berhasil mencapai akurasi hingga 93,37% dalam klasifikasi ulasan aplikasi di Google Play Store[7][8]. Sementara itu, *Naive Bayes* dikenal unggul dalam menangani data berdimensi

tinggi dengan komputasi yang efisien, dan menjadi pilihan populer dalam pengolahan bahasa alami. Meskipun telah banyak penelitian mengenai analisis sentimen pada aplikasi secara umum, kajian khusus terhadap ulasan pengguna *Netflix di Google Play Store* masih sangat terbatas. Padahal, ulasan-ulasan tersebut memiliki nilai strategis dalam memberikan masukan bagi pengembangan fitur, peningkatan performa, dan personalisasi layanan.

Penelitian ini bertujuan untuk mengidentifikasi dan mengklasifikasikan sentimen pengguna terhadap aplikasi *Netflix* menggunakan pendekatan *machine learning* berbasis *Naive Bayes* dan *SVM*. Dengan pendekatan ini, diharapkan dapat diperoleh wawasan yang lebih mendalam mengenai persepsi pengguna serta faktor-faktor yang memengaruhi tingkat kepuasan dan ketidakpuasan pengguna aplikasi *Netflix*.

2. METODE

Penelitian ini dilakukan dengan menggunakan beberapa tahapan utama yang dijelaskan dalam diagram alir metode berikut:



Gambar 1. Research Flow

a) *Dataset Collection*
Tahap awal adalah pengumpulan data ulasan pengguna terhadap aplikasi *Netflix* dari *Google Play Store*. Data dikumpulkan menggunakan teknik web scraping dengan memanfaatkan *tools* seperti *Python* dan *library BeautifulSoup atau Scrapy*. Data yang dikumpulkan berupa teks ulasan, rating, dan metadata lainnya.

b) *Text Preprocessing*
Agar data teks dapat diproses oleh algoritma machine learning secara optimal, dilakukan tahap pra-pemrosesan data *(text preprocessing)* dengan beberapa langkah:
- *Case Folding*: Mengubah seluruh huruf menjadi huruf kecil *(lowercase)* untuk menyamakan bentuk kata.
- *Cleaning*: Menghapus karakter atau simbol yang tidak diperlukan, seperti angka, tanda baca, emoji, URL, dan tag HTML.
- *Tokenizing*: Memecah kalimat menjadi bagian-bagian kecil berupa kata (tokens).
- *Stopword Removal*: Menghapus kata-kata umum (seperti “dan”, “yang”, “ke”) yang tidak memiliki pengaruh signifikan terhadap klasifikasi sentimen.

c) *Sentimen Lexicon*
Selanjutnya, digunakan kamus sentimen *(sentiment lexicon)* sebagai acuan dalam proses analisis. *Lexicon* ini dapat berupa daftar kata dengan bobot sentimen positif, negatif, atau netral[9], yang berfungsi untuk mendukung validasi hasil klasifikasi sentimen secara otomatis atau sebagai fitur tambahan.

d) TF-IDF Weighting

Setelah teks dibersihkan dan dipisah menjadi token, dilakukan ekstraksi fitur menggunakan metode *Term Frequency-Inverse Document Frequency (TF-IDF)*. Metode ini digunakan untuk memberikan bobot pada kata-kata yang memiliki pengaruh besar dalam dokumen tetapi tidak terlalu umum dalam keseluruhan korpus dokumen.

e) Sentiment Classification with Naïve Bayes and Support Vector Machine

Proses klasifikasi dilakukan dengan membandingkan dua algoritma *machine learning*:
Naïve Bayes (NB): Menggunakan pendekatan probabilistik untuk menentukan kategori sentimen (positif, negatif, atau netral)[10] berdasarkan distribusi kata dalam data.
Support Vector Machine (SVM): Mengklasifikasikan data dengan mencari *hyperplane* terbaik yang memisahkan kelas sentimen[11] dalam ruang fitur.
Kedua model akan dievaluasi berdasarkan akurasi, presisi, *recall*, dan *F1-score* untuk menentukan algoritma yang paling optimal.

f) Conclusion Drawing

Tahap terakhir adalah menarik kesimpulan dari hasil analisis. Kesimpulan mencakup interpretasi terhadap hasil klasifikasi, perbandingan performa algoritma, serta insight penting dari ulasan pengguna terkait aplikasi *Netflix* di *Google Play Store*.

3. HASIL DAN PEMBAHASAN

Berdasarkan tahapan-tahapan yang telah dijelaskan sebelumnya, berikut ini adalah hasil yang diperoleh serta pembahasan mengenai implementasi dan temuan pada masing-masing tahapan metode penelitian.

1) Dataset Collection

Pada tahap ini, data dikumpulkan dari ulasan pengguna aplikasi *Netflix* yang tersedia di *Google Play Store*. Proses pengumpulan data dilakukan menggunakan metode web *scraping* dengan bantuan bahasa pemrograman Python untuk mengekstraksi data secara otomatis dari halaman aplikasi *Netflix*. Dataset yang berhasil dikumpulkan terdiri dari 20.000 ulasan pengguna. Data yang diperoleh masih dalam bentuk mentah (raw data), sehingga memerlukan tahapan pra-pemrosesan sebelum dapat digunakan untuk analisis sentimen. Pemilihan jumlah ulasan sebanyak 20.000 dilakukan untuk memastikan dataset memiliki representasi yang cukup terhadap beragam opini pengguna, baik positif, negatif, maupun netral. Ulasan-ulasan ini selanjutnya digunakan sebagai data latih dan data uji dalam proses klasifikasi sentimen menggunakan algoritma machine learning.

2) Text Preprocessing

Pada tahap praproses teks, berbagai langkah dilakukan untuk membersihkan dan memformat teks sehingga siap untuk analisis lebih lanjut.
Case Folding. Tahap pertama dalam proses prapemrosesan data pada penelitian ini adalah case folding. Pada tahap ini, seluruh huruf dalam teks diubah menjadi huruf kecil guna menyamakan format penulisan. Tujuannya adalah agar perbedaan antara huruf besar dan kecil tidak memengaruhi analisis, sehingga proses pencarian menjadi lebih sederhana, konsistensi data meningkat, dan kompleksitas dimensi teks dapat dikurangi.

Table 1. Case Folding Process

No	Response	Case Folding
1	rugi udh berlangganan tapi gabisa nonton,tolon...	rugi udh berlangganan tapi gabisa nonton tolon...
2	baguss sihh tapi aku kira lengkap teryata gak ...	baguss sihh tapi aku kira lengkap teryata gak ...
3	tolong aplikasi nya diperbaiki lagi, mau log i...	tolong aplikasi nya diperbaiki lagi mau log i...
4	langganan sudah dibatalkan, dan uninstall male...	langganan sudah dibatalkan dan uninstall male...
5	aplikasi bagus menurut saya sangat recommended...	aplikasi bagus menurut saya sangat recommended...
6	ini kemarin saya udah berhenti berlangganan ta...	ini kemarin saya udah berhenti berlangganan ta...

Pembersihan. Dalam proses pembersihan, hal ini dilakukan untuk menghilangkan semua tanda baca dan menghilangkan *noise* dari kalimat yang tidak diperlukan. Proses pembersihan data

teks bertujuan untuk meningkatkan kualitas data dengan menghilangkan elemen yang tidak diperlukan.

Table 2. Case Folding Process

No	Case Folding	Cleaning
1	rugi udh berlangganan tapi gabisa nonton tolon...	rugi udh berlangganan tapi gabisa nonton tolon...
2	baguss sihh tapi aku kira lengkap teryata gak ...	baguss sihh tapi aku kira lengkap teryata gak ...
3	tolong aplikasi nya diperbaiki lagi mau log i...	tolong aplikasi nya diperbaiki lagi mau log in...
4	aplikasi bagus menurut saya sangat recommended...	aplikasi bagus menurut saya sangat recommended...
5	aplikasi sering sekali eror padahal langganan...	aplikasi sering sekali eror padahal langganan ...
6	ini kemarin saya udah berhenti berlangganan ta...	ini kemarin saya udah berhenti berlangganan ta...

Tokenizing. Setelah melalui tahap *Cleaning*, selanjutnya masuk ke tahap *Tokenizing,* dari hasil *Cleaning* kemudian diolah pada tahap *Tokenizing*. *Tokenizing* merupakan suatu proses dalam *text processing*, di mana teks dipecah menjadi unit-unit kecil yang disebut token[12]. Token-token ini biasanya berupa kata-kata tersendiri, tetapi dapat juga berupa frasa, kalimat, atau elemen lainnya tergantung pada tujuan analisis. Tokenizing merupakan langkah penting dalam banyak tugas *natural language processing* (NLP) karena memungkinkan dilakukannya analisis yang lebih rinci.

Table 3. Tokenizing Process

No	Cleaning	Tokenizing
1	rugi udh berlangganan tapi gabisa nonton tolon...	rugi, udh, berlangganan, tapi, gabisa, nonton...
2	baguss sihh tapi aku kira lengkap teryata gak ...	baguss, sihh, tapi, aku, kira, lengkap, terya...
3	tolong aplikasi nya diperbaiki lagi mau log in...	tolong, aplikasi, nya, diperbaiki, lagi, mau,...
4	aplikasi bagus menurut saya sangat recommended.	aplikasi, bagus, menurut, saya, sangat, recom...
5	aplikasi sering sekali eror padahal langganan ...	aplikasi, sering, sekali, eror, padahal, lang...
6	ini kemarin saya udah berhenti berlangganan ta...	ini, kemarin, saya, udah, berhenti, berlangga...

Stopword Removal. Proses selanjutnya adalah *stopword removal* dimana data akan diolah dan diproses untuk menghilangkan kata-kata yang dianggap tidak relevan seperti awalan “ny”, “di”, “yang” dan lain-lain. Proses *stopword* mempengaruhi nilai akurasi dan persentase sentimen kata[13].

Table 4. Stopword Removal Process

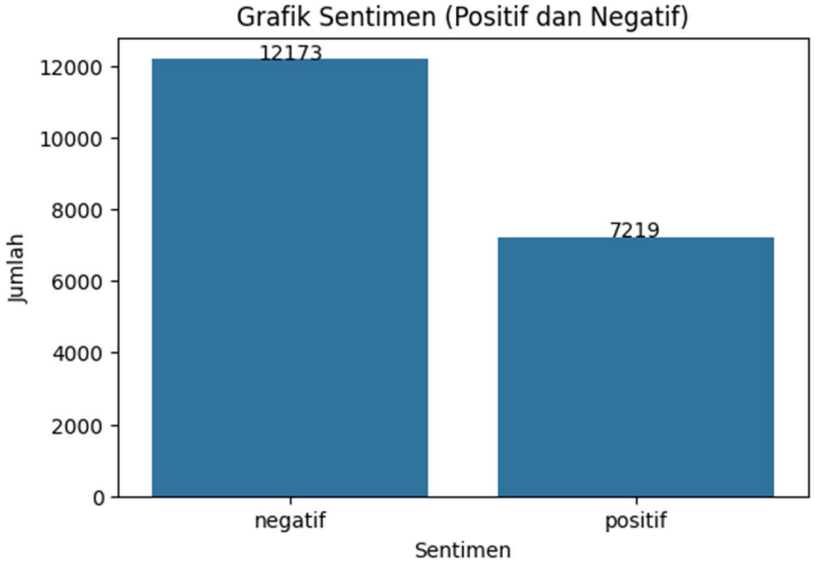
No	Tokenizing	Stopword Removal
1	rugi udh berlangganan tapi gabisa nonton tolon...	rugi, udh, berlangganan, tapi, gabisa, nonton...
2	baguss sihh tapi aku kira lengkap teryata gak ...	baguss, sihh, tapi, aku, kira, lengkap, terya...
3	tolong aplikasi nya diperbaiki lagi mau log in...	tolong, aplikasi, nya, diperbaiki, lagi, mau,...
4	aplikasi bagus menurut saya sangat recommended.	aplikasi, bagus, menurut, saya, sangat, recom...
5	aplikasi sering sekali eror padahal langganan ...	aplikasi, sering, sekali, eror, padahal, lang...
6	ini kemarin saya udah berhenti berlangganan ta...	ini, kemarin, saya, udah, berhenti, berlangga...

3) Sentimen Lexicon

Penelitian ini mengimplementasikan klasifikasi sentimen berbasis leksikon setelah tahap praproses data opini selesai. Proses ini melibatkan penerjemahan teks opini berbahasa Indonesia ke bahasa Inggris menggunakan *Google Translate*. Selanjutnya, teks terjemahan akan dianalisis sentimennya menggunakan VADER *Sentiment Intensity Analyzer*, yang membandingkan teks dengan kamus leksikon bahasa Inggris. VADER kemudian menghasilkan skor sentimen komposit yang mengindikasikan polaritas (positif, negatif, atau netral)[14] serta intensitas sentimen pada teks tersebut, memungkinkan klasifikasi sentimen yang akurat dan berlabel. Data yang digunakan dalam proses leksikon ini merupakan hasil dari tahap *stemming*.

Table 5. Examples of Stemming and Lexicon Labeling Process

No	Stopword Removal	Stemming	Polarity Text	Lexicon Labeling
1	sih, netflix, beranda, memutar, memutar, woi, bayar, pakai, uang	sih netflix beranda putar putar woi bayar pakai uang	-4	Negatif
2	tolong, film, baru , admin, tayang, ulang	tolong film baru admin tayang ulang	1	Positif
3	ganti, paketan, netflix, disuruhnya, lanjutkan, paketan, tolong, cerah, nya	ganti paket netflix suruh lanjut paket tolong cerah nya	1	Positif
4	aplikasi, akun, login, dikenal juga sebagai, persulit	aplikasi akun login kenal juga bagai sulit	-3	Negatif
5	puas, pelayanan, nya	puas layan nya	3	Positif



Gambar 2. Comparison of Positive and Negative Sentiment Data

4) TF-IDF Weighting

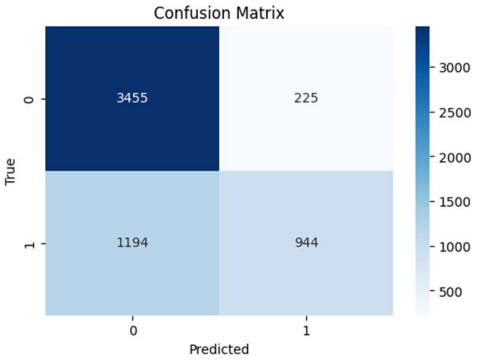
Penelitian ini melanjutkan dengan pembobotan *Term Frequency-Inverse Document Frequency* (TF-IDF), sebuah metode yang krusial untuk mengevaluasi signifikansi kata dalam sebuah dokumen[15]. *Term Frequency* (TF) mengukur frekuensi kemunculan sebuah kata dalam dokumen tertentu, memberikan bobot lebih pada kata yang sering muncul karena relevansinya yang tinggi. Kemudian, *Document Frequency* (DF) mencatat jumlah dokumen yang mengandung kata tersebut dalam seluruh korpus. Selanjutnya, *Inverse Document Frequency* (IDF) menilai keunikan sebuah kata dengan membagi total dokumen dengan jumlah dokumen yang memuat kata tersebut. Nilai TF-IDF sendiri diperoleh dari perkalian nilai TF dan IDF, menghasilkan representasi numerik yang menggambarkan pentingnya sebuah kata dalam dokumen. Melalui TF-IDF, teks dapat diubah menjadi format numerik, memfasilitasi analisis dan pemahaman dokumen yang lebih mendalam.

5) Klasifikasi *Support Vector Machine* dan *Naive Bayes*

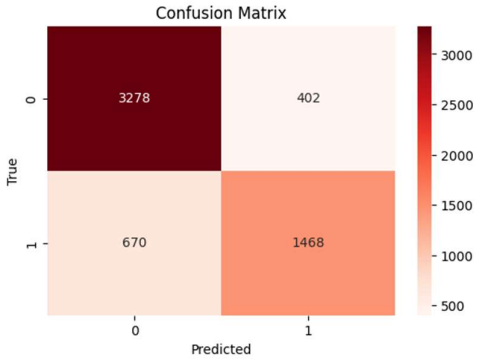
Setelah tahap preprocessing, klasifikasi leksikon, dan pembobotan kata TF-IDF selesai, selanjutnya dilakukan perbandingan hasil klasifikasi sentimen menggunakan *Support Vector Machine* dan *Naive Bayes* untuk mengetahui hasil prediksi sentimen dari data. Algoritma *Support Vector Machine* menghasilkan akurasi 82%, presisi 83%, recall 89%, dan skor F1 86%, *Naive Bayes* menghasilkan akurasi 76%, presisi 74%, recall 94%, dan skor F1 83%. Hasil proses klasifikasi pada data uji dapat dilihat pada matriks konfusi dan tabel berikut:

Table 6. Classification Result

No.	Algoritma	Kelas	Precision	Recall	F1-Score	Support	Akurasi
1.	Naive Bayes	Negatif	0.74	0.94	0.83	3 680	76%
		Positif	0.81	0.44	0.57	2 138	
2.	Support Vector Machine	Negatif	0.83	0.89	0.86	3 680	82%
		Positif	0.79	0.69	0.73	2 138	



Gambar 3. Confusion Matrix -Naive Bayes



Gambar 4. Confusion Matrix - Support Vector Machine

6) Conclusion Drawing

SVM mencapai skor F1 86%, yang mengindikasikan keseimbangan optimal antara presisi dan *recall* dalam memprediksi sentimen. Sementara itu, *Naive Bayes* menghasilkan skor F1 83%, sedikit di bawah SVM. Perbedaan ini menunjukkan bahwa meskipun *Naive Bayes* mampu mengidentifikasi banyak instansi positif (*recall* tinggi), SVM memberikan keseimbangan yang lebih baik antara identifikasi positif yang akurat dan minimnya kesalahan prediksi. Oleh karena itu, SVM terbukti lebih unggul dalam memberikan klasifikasi sentimen yang seimbang dan akurat dalam studi ini.

4. KESIMPULAN DAN SARAN

Penelitian ini berhasil menerapkan klasifikasi sentimen pada data opini setelah melalui serangkaian tahapan praproses, klasifikasi leksikon, dan pembobotan kata menggunakan TF-IDF. Perbandingan dua algoritma klasifikasi sentimen, *Support Vector Machine (SVM)* dan *Naive Bayes*, menunjukkan hasil yang berbeda. SVM terbukti lebih unggul dengan skor F1 86%, mengindikasikan keseimbangan yang optimal antara presisi dan recall dalam memprediksi sentimen. Sementara itu, *Naive Bayes* mencapai skor F1 83%, sedikit di bawah performa SVM. Hal ini menunjukkan bahwa SVM memberikan klasifikasi sentimen yang lebih seimbang dan akurat dalam konteks penelitian ini.

Untuk pengembangan penelitian ke depan, disarankan untuk menjelajahi algoritma klasifikasi lain di luar SVM dan *Naive Bayes* guna potensi peningkatan performa. Mengingat tantangan penerjemahan bahasa, pengembangan atau penggunaan kamus leksikon sentimen khusus bahasa Indonesia sangat direkomendasikan untuk akurasi yang lebih tinggi. Selain itu, peningkatan kualitas praproses dengan menangani aspek kompleks seperti negasi atau sarkasme, serta evaluasi model pada berbagai domain data, akan memperkaya validitas dan generalisasi temuan penelitian

5. DAFTAR PUSTAKA

[1] K. Barik, S. Misra, A. K. Ray, and A. Sukla, "A blockchain-based evaluation approach to analyse customer satisfaction using AI techniques," *Heliyon*, vol. 9, no. 6, Jun. 2023, doi: 10.1016/j.heliyon.2023.e16766.

[2] S. Ranjan and S. Mishra, "Perceiving University Student's Opinions from Google App Reviews," Dec. 2023, doi: 10.1002/cpe.6800.

[3] V. S. Anoop and S. Sreelakshmi, "Public discourse and sentiment during Mpox outbreak: an analysis using natural language processing," *Public Health*, vol. 218, pp. 114–120, May 2023, doi: 10.1016/j.puhe.2023.02.018.

[4] Y. Amirkhalili and H. Y. Wong, "Banking on Feedback: Text Analysis of Mobile Banking iOS and Google App Reviews," Mar. 2025, [Online]. Available: http://arxiv.org/abs/2503.11861

[5] N. Darraz, I. Karabila, A. El-Ansari, N. Alami, and M. El Mallahi, "Advancing recommendation systems with DeepMF and hybrid sentiment analysis: Deep learning and Lexicon-based integration," *Expert Syst Appl*, vol. 279, Jun. 2025, doi: 10.1016/j.eswa.2025.127432.

[6] N. Vedavathi and K. M. Anil Kumar, "E-learning course recommendation based on sentiment analysis using hybrid Elman similarity," *Knowl Based Syst*, vol. 259, Jan. 2023, doi: 10.1016/j.knosys.2022.110086.

[7] S. Ranjan and S. Mishra, "Comparative Sentiment Analysis of App Reviews."

[8] Z. Rahman, P. Sakinah, Y. Hendra, B. Satria, F. Maulana, and A. Q. Ayun, "Sentiment Analysis of Gojek App Reviews on Google Play Store with Natural Language Processing Using Naive Bayes' Algorithm-Zumardi Rahman et.al Sentiment Analysis of Gojek App Reviews on Google Play Store with Natural Language Processing Using Naive Bayes' Algorithm," vol. 06, 2024, doi: 10.54209/jatilima.v6i03.1189.

[9] H. Liu and A. Hosseini, "SASLS: Semantic analysis of sentiment in social networks using Lexicon-Based methodology and Semi-Supervised sentiment annotation," *Ain Shams Engineering Journal*, vol. 16, no. 6, May 2025, doi: 10.1016/j.asej.2025.103378.

[10] E. Azeraf, E. Monfrini, and W. Pieczynski, "Improving usual Naive Bayes classifier performances with Neural Naive Bayes based models," Nov. 2021, [Online]. Available: http://arxiv.org/abs/2111.07307

[11] Hermanto, A. Y. Kuntoro, T. Asra, E. B. Pratama, L. Effendi, and R. Ocanitra, "Gojek and Grab User Sentiment Analysis on Google Play Using Naive Bayes Algorithm and Support Vector Machine Based Smote Technique," in *Journal of Physics: Conference Series*, IOP Publishing Ltd, Nov. 2020. doi: 10.1088/1742-6596/1641/1/012102.

[12] Y. Zhang, M. Sun, Y. Ren, and J. Shen, "Sentiment Analysis of Sina Weibo Users under the Impact of Super Typhoon Lekima Using Natural Language Processing Tools: A Multi-Tags Case Study," in *Procedia Computer Science*, Elsevier B.V., 2020, pp. 478–490. doi: 10.1016/j.procs.2020.06.116.

[13] C. Li and X. Hu, "Medical Artificial Intelligence in Scholarly and Public Perspective: BERTopic-Based Analysis of Topic-Sentiment Collaborative Mining," *Data Science and Informetrics*, May 2025, doi: 10.1016/j.dsim.2025.05.001.

[14] Md. N. Hoque, U. Salma, Md. J. Uddin, Md. M. Ahamad, and S. Aktar, "Exploring transformer models in the sentiment analysis task for the under-resource Bengali language," *Natural Language Processing Journal*, vol. 8, p. 100091, Sep. 2024, doi: 10.1016/j.nlp.2024.100091.

[15] B. Das Sarit Chakraborty Student Member and I. Member, "An Improved Text Sentiment Classification Model Using TF-IDF and Next Word Negation."