

## PENERAPAN NATURAL LANGUAGE PROCESSING UNTUK ANALISIS SENTIMEN ULASAN APLIKASI ALLO BANK MENGUNAKAN ALGORITMA SVM

Singgih Adhi Prasetyo<sup>(1)</sup>, Diana Yusuf<sup>2</sup>, Muhajir Syamsu<sup>3</sup>

<sup>1</sup>Institut Teknologi dan Bisnis Ahmad Dahlan, Jakarta

<sup>2</sup>Institut Teknologi dan Bisnis Ahmad Dahlan, Jakarta

<sup>3</sup>Institut Teknologi dan Bisnis Ahmad Dahlan, Jakarta

### Abstract

*The rapid growth of digital banking applications in Indonesia requires companies to pay close attention to user perceptions and satisfaction. This research focuses on analyzing the sentiment of user reviews of the Allo Bank application obtained from the Google Play Store, applying the Support Vector Machine (SVM) algorithm combined with a Natural Language Processing (NLP) approach. A total of 5,100 reviews in the Indonesian language were collected using web scraping methods. The analysis involved several stages, including text preprocessing, term weighting with TF-IDF, sentiment classification into three categories (positive, negative, and neutral), and performance evaluation of the model. The findings reveal that most user reviews express positive sentiment, with frequently appearing words such as "easy," "secure," and "good." Negative sentiments are commonly represented by words like "complicated," "lag," and "slow," while neutral reviews often include expressions such as "adequate" and "okay." The SVM model achieved strong performance, with an accuracy of 96%, precision of 97%, recall of 75%, and an F1-score of 79%, indicating high effectiveness despite limited sensitivity to minority classes. These classification results can provide valuable insights for identifying areas that need improvement or should be maintained by the application developers. Therefore, this study is expected to serve as a reference for strategic decision-making based on user feedback to enhance the quality of digital banking services.*

**Kata Kunci:** Analisis Sentimen, Allo Bank, Support Vector Machine, NLP, TF-IDF.

### Informasi Artikel:

Dikirim : 03 September 2025

Ditelaah: 24 Juni 2026

Diterima: 28 Juni 2026

Publikasi: 29 Juni 2026

Januari – Juni 2026, Vol 7 (1) : hlm 65-75

©2026 Institut Teknologi dan Bisnis Ahmad Dahlan.

All rights reserved.

## PENDAHULUAN

Perkembangan teknologi yang sangat pesat saat ini hampir menyentuh seluruh sektor kehidupan seperti ekonomi, kesehatan, dan pendidikan (Yusuf et al., 2023). Kehadiran bank digital menjadi jawaban atas kebutuhan masyarakat modern yang menginginkan layanan keuangan praktis, cepat, serta dapat diakses kapan saja melalui perangkat mobile. Di Indonesia, beberapa bank digital seperti Bank Jago, Line Bank, dan Allo Bank telah memainkan peran penting dalam membentuk ekosistem keuangan digital. Allo Bank, yang diluncurkan oleh CT Corp pada tahun 2022, menawarkan layanan keuangan yang terintegrasi dengan gaya hidup, ritel, dan hiburan, sehingga menarik perhatian generasi muda. Layanan yang ditawarkan mencakup pembukaan rekening digital, dompet elektronik, pinjaman mikro, hingga transaksi dengan merchant mitra. Namun, kesuksesan sebuah aplikasi perbankan digital tidak hanya ditentukan oleh kelengkapan fitur, tetapi juga oleh kepuasan dan pengalaman pengguna.

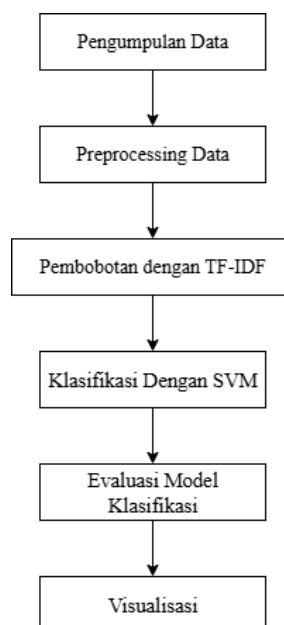
Pemanfaatan teknologi *Natural Language Processing* (NLP) menjadi salah satu cara melihat bagaimana menganalisis respon pengguna aplikasi *allo bank* yang nanti analisis tersebut digunakan dalam pengambil keputusan oleh para manajemen *allo bank*. NLP adalah bidang kecerdasan buatan yang mempelajari bagaimana sebuah dari sistem komputer dapat memahami dan mengontrol bahasa alami yang terkait dengan teks atau ucapan. Dengan mengumpulkan data tentang pemahaman dan penggunaan bahasa manusia, ini digunakan untuk membuat teknik yang tepat bagi sistem komputer untuk mengelola bahasa manusia dan menyelesaikan berbagai tugas (Muhammad et al., 2022). NLP menjadi pendekatan yang sangat efektif. NLP memungkinkan komputer untuk memproses dan memahami bahasa manusia secara otomatis, sehingga dapat digunakan untuk analisis sentimen terhadap opini pengguna. Pendekatan ini telah banyak digunakan di berbagai sektor, termasuk perbankan (Suresh et al., 2024).

Penelitian terdahulu oleh (Anilkumar et al., 2023) membandingkan *Support Vector Machine* (SVM) dan *Naïve Bayes* dalam klasifikasi ulasan aplikasi, dan menemukan bahwa SVM dengan RBF Kernel menghasilkan akurasi hingga 92,86%. Sehingga dalam penelitian ini adalah SVM sebagai klasifikasi sentimen. SVM merupakan algoritma supervised learning yang terbukti efektif untuk tugas klasifikasi teks, termasuk dalam analisis sentimen. Algoritma ini bekerja dengan mencari hyperplane optimal yang memisahkan data ke dalam kelas-kelas sentimen (positif, negatif, netral) dengan margin maksimum. SVM dipilih karena memiliki performa yang baik dalam menangani data teks berdimensi tinggi dan menghasilkan akurasi tinggi (Eldo et al., 2024).

Tujuan dari penelitian ini adalah untuk menganalisis persepsi ulasan pengguna untuk aplikasi Allo Bank di Google Play Store dengan menggunakan teknik NLP dan algoritma SVM. Diharapkan penelitian ini akan memberikan rekomendasi strategis bagi pengembang untuk meningkatkan kualitas layanan perbankan digital melalui pemetaan persepsi pengguna.

## METODE

Pendekatan yang digunakan dalam penelitian ini adalah Metode kuantitatif deskriptif digunakan dalam penelitian ini untuk menganalisis ulasan pengguna untuk aplikasi *Allo Bank* yang tersedia di Google Play Store. Data diklasifikasikan berdasarkan tiga nilai utama: positif, negatif, dan netral. Tokenisasi, penghapusan *stopword*, dan *stemming* adalah langkah-langkah yang digunakan dalam metode pengolahan bahasa alami (NLP) untuk menyelesaikan analisis. Selanjutnya, algoritma Support Vector Machine (SVM) digunakan dalam mengklasifikasikan dan membobotkan data menggunakan TF-IDF. Untuk menilai kinerja model, metrik akurasi, presisi, recall, dan skor F1 digunakan. Metode ini diharapkan menghasilkan analisis yang terukur dan tidak bias yang menggambarkan persepsi pengguna tentang layanan Allo Bank dan membantu mereka membuat keputusan. Adapun gambar 1 merupakan kerangka kerja penelitian:



Gambar 1. Kerangka Kerja Penelitian

Sumber: analisis data, 2025

## HASIL DAN PEMBAHASAN

### 3.1. Dataset

Tahapan pengumpulan *dataset* menggunakan *web scraping* menggunakan *framework* dari python yaitu *google play scraper* yang akan dijalankan dengan *script* python di visual studio code.

```

from google_play_scraper import reviews, Sort
import pandas as pd
import time
import os

app_package = 'com.allnapp.yump'
batch_size = 100
max_reviews = 5100 # Target minimum review
collected_reviews = []
continuation_token = None

print("Mulai scraping review Allo Bank (tanpa filter skor)...")

while len(collected_reviews) < max_reviews:
    result, continuation_token = reviews(
        app_package,
        lang='id',
        country='id',
        sort=Sort.NEWEST,
        count=batch_size,
        filter_score_with=None,
        continuation_token=continuation_token
    )

```

Gambar 1. Code Pengumpulan Dataset

Sumber: analisis data, 2025

Dari hasil proses di atas menghasilkan hasil sebagai berikut:

Tabel 1. Hasil Pengumpulan Dataset

| No.  | Content  | score | At                      |
|------|--|-------|-------------------------|
| 1    | kocak masa dapat limit 1000 perak doang dan harus top up dl biar nambah limit.. kocak aplikasi nya 🤔🤔🤔 | 1     | 2025-06-20<br>20:04:00  |
| 2    | ajjiib niiah dapat limit 1000...gasss luuur  | 5     | 2025-06-21<br>06:35:00  |
| 3    | allo bank sangat mempermudah untuk transaksi online dan banyak promo menarik                           | 5     | 2025 06-30-<br>06:36:28 |
| ...  |  |       |                         |
| 4554 | sangat buruk limit paylater dapat 1000 langsung di hapus tidak berguna                                 | 2     | 2025-07-02<br>20:20:06  |

Sumber: analisis data, 2025

### 3.2. Processing data

*Preprocessing* merupakan tahap krusial dalam analisis data teks yang bertujuan untuk memastikan data yang akan digunakan dalam proses klasifikasi berada dalam kondisi yang bersih, seragam, dan terstruktur.

```

# 1. Load data ulasan asli
df = pd.read_csv("allobank_reviews_5180.csv") # Harus ada kolom 'content' dan 'score'

# 2. Salin data asli sebelum preprocessing
df['raw_text'] = df['content'].astype(str)

# 3. Inisialisasi stopword dan stemmer
stop_factory = StopwordRemoverFactory()
default_stopwords = set(stop_factory.get_stop_words())

custom_stopwords = set([
# Kata informal umum
'ya', 'gk', 'ga', 'tak', 'nggak', 'enggak', 'aja', 'sok', 'abekuk', 'uuh', 'dah',
'si', 'nya', 'ya', 'eh', 'lho', 'sih', 'dong', 'den', 'loh', 'lah', 'lg', 'nih',
'oke', 'ok', 'iya', 'tidak', 'bgt', 'banget', 'bisa', 'akan', 'udah', 'lagi',
'malah', 'kayak', 'kayaknya', 'seperti', 'karna', 'kalo', 'klu', 'tnt', 'pokok',
'juaranya', 'aplikasinya', 'orang', 'atas', 'terlalu', 'lain', 'team',

# Bahasa Inggris umum
'very', 'good', 'recommended', 'fun', 'traveling', 'best', 'the', 'service', 'too',
])

```

Gambar 2. Input File

Sumber: analisis data, 2025

### 3.2.1. Case Folding

Mengubah semua huruf dalam teks menjadi huruf kecil (*lowercase*). Jika teks kosong (`pd.isnull`), mengembalikan string kosong ("").

```

# 1. Case folding
def case_folding(text):
    if pd.isnull(text):
        return ""
    return text.lower()

```

Gambar 3. Code Case Folding

Sumber: analisis data, 2025

Tabel 2. Hasil Case Folding

| No.  | Sebelum Case Folding   | Sesudah Case Folding   |
|------|--|--|
| 1    | merasakan kenyamanan saat<br>berbelanja  | merasakan kenyamanan saat<br>berbelanja  |
| 2    | pimjaman gak jelas masa biaya<br>admin lebih besar dr biaya bunga. aneh<br>memang allo bank ini.   | pimjaman gak jelas masa biaya<br>admin lebih besar dr biaya bunga. aneh<br>memang allo bank ini. |
| 3    | aplikasi allo bank sangat membantu<br>saya dalam bidang<br>bisnis □ terimakasih atas kerja samanya | aplikasi allo bank sangat membantu<br>saya dalam bidang<br>bisnis terimakasih atas kerja samanya |
| ...  |  |  |
| 4554 | menginstal ribet   | menginstal ribet   |

Sumber: analisis data, 2025

### 3.2.2. Stopword Removal

Filtering dilakukan untuk menghapus kata-kata umum seperti kata hubung atau kata bantu, serta karakter khusus, angka, atau simbol yang tidak relevan.

```
# 3. Stopword removal
def remove_stopwords(text):
    words = text.split()
    return ' '.join([word for word in words if word not in stopwords])
```

Gambar 4. Code Stop Removal

Sumber: analisis data, 2025

Tabel 3. Hasil Stop Removal

| No.  | Sebelum <i>Stopword Removal</i>  | Sesudah <i>Stopword Removal</i>                                      |
|------|--|--|
| 1    | merasakan kenyamanan saat berbelanja   | merasakan kenyamanan berbelanja                                      |
| 2    | pimjaman gak jelas masa biaya admin lebih besar dr biaya bunga. aneh memang allo bank ini.   | pimjaman gak jelas masa biaya admin besar biaya bunga aneh allo bank |
| 3    | aplikasi allo bank sangat membantu saya dalam bidang bisnis □ terimakasih atas kerja samanya | aplikasi allo bank membantu bidang bisnis terimakasih kerjasamanya   |
| ...  |  |  |
| 4554 | menginstal ribet   | menginstal ribet   |

Sumber: analisis data, 2025

### 3.2.3. Tokenizing

*Tokenizing* adalah proses pemisahan kalimat atau teks menjadi satuan kata yang lebih kecil yang disebut token

```
# 4. Tokenization
def tokenize(text):
    return text.split()
```

Gambar 5. Code Tokenizing

Sumber: analisis data, 2025

Tabel 4. Hasil Tokenizing

| No. | Sebelum <i>Tokenizing</i>  | Sesudah <i>Tokenizing</i>   |
|-----|--|---|
| 1   | merasakan kenyamanan saat berbelanja   | ['merasakan', 'kenyamanan', 'saat', 'berbelanja']   |
| 2   | pimjaman gak jelas masa biaya admin lebih besar dr biaya bunga. aneh memang allo bank ini. | ['pimjaman', 'gak', 'jelas', 'masa', 'biaya', 'admin', 'lebih', 'besar', 'dr', 'biaya', 'bunga', 'aneh', 'memang', 'allo', 'bank', 'ini'] |
| ... |  |   |

Sumber: analisis data, 2025

### 3.2.4. Stemming

```
# 5. Stemming per token
def stem_tokens(tokens):
    return [stemmer.stem(token) for token in tokens]
```

Gambar 6. Code Stemming

Sumber: analisis data, 2025

| No.  | Sebelum Stemming   | Sesudah Stemming   |
|------|--|--|
| 1    | merasakan kenyamanan saat berbelanja   | rasa nyaman saat belanja   |
| 2    | pimjaman gak jelas masa biaya admin lebih besar dr biaya bunga. aneh memang allo bank ini.   | pinjam gak jelas masa biaya admin besar dari biaya bunga aneh memang allo bank ini |
| 3    | aplikasi allo bank sangat membantu saya dalam bidang bisnis □ terimakasih atas kerja samanya | aplikasi allo bank bantu dalam bidang bisnis terima kasih atas kerja sama          |
| ...  |  |  |
| 4554 | menginstal ribet   | instal ribet   |

Sumber: analisis data, 2025

### 3.3. Pembobotan TF-IDF

TF-IDF Merupakan proses perhitungan atau pengekstrakan kata menjadi sebuah angka berbentuk vektor yang digunakan untuk menentukan bobot dari sebuah kata dalam sebuah dokumen atau korpus. Bobot ini berguna untuk menentukan seberapa penting kata tersebut dalam sebuah dokumen. (Putra et al., 2023)

```
tfidf_vectorizer = TfidfVectorizer(max_features=5100)
X = tfidf_vectorizer.fit_transform(df['clean_content'])
y = df['sentiment']

X_train, y_train, X_test, y_test = train_test_split(X, y, test_size=0.2, stratify=y, random_state=42)

model = LinearSVC()
model.fit(X_train, y_train)
```

Gambar 7. Code Pembobotan TF-IDF

Sumber: analisis data, 2025

Tabel 5. Hasil Pembobotan TF-IDF

| Kata       | D1  | D2 | D3 |
|------------|-----|----|----|
| Merasakan  | 0,5 | 0  | 0  |
| Kenyamanan | 0,5 | 0  | 0  |
| Belanja    | 0,5 | 0  | 0  |

...

|      |   |   |      |
|------|---|---|------|
| Sama | 0 | 0 | 0,15 |
|------|---|---|------|

Sumber: analisis data, 2025

### 3.4. Klasifikasi Menggunakan SVM

```
36 @ti.rache_research
37 def prepare_model(df):
38     tfidf = TfidfVectorizer(max_features=10000, ngram_range=(1, 2))
39     X_all = tfidf.fit_transform(df['strending'])
40     y_all = df['sentimen'].values
41     X_train, X_test, y_train, y_test, idx_train, idx_test = train_test_split(X_all, y_all, df.index, test_size=0.2, random_state=42)
42     train_texts = df.loc[idx_train]
43     svm = SVC(kernel='linear', probability=True)
44     svm.fit(X_train, y_train)
45     return tfidf, svm, X_train, y_train, train_texts
46
47 tfidf, svm_model, X_train, y_train, train_texts = prepare_model(data)
48
49 def preprocess_test(text):
50     text = text.lower()
51     return " ".join([w for w in text if w.isalnum() or w.ispunct])
52
53 def evaluasi_model(model, X, y_true):
54     y_pred = model.predict(X)
55     acc = accuracy_score(y_true, y_pred)
56     prec = precision_score(y_true, y_pred, average='macro')
57     rec = recall_score(y_true, y_pred, average='macro')
58     f1 = f1_score(y_true, y_pred, average='macro')
59     cm = confusion_matrix(y_true, y_pred, labels=['Positif', 'Netral', 'Negatif'])
60     return acc, prec, rec, f1, cm, y_pred
```

Gambar 8. Proses Latih SVM

Sumber: analisis data, 2025

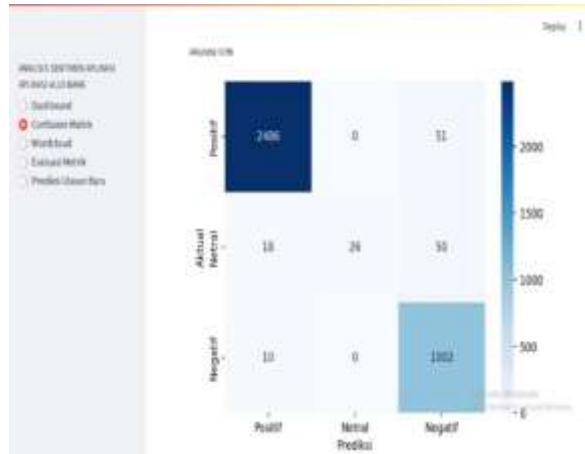
### 3.5. Hasil Visualisasi

Hasil visualisasi membantu dalam memberikan sebuah gambar yang interaktif agar mudah dipahami oleh orang awam. Visualisasi menggunakan streamlit sehingga berbasis website.



Gambar 9. Halaman Dashboard

Sumber: analisis data, 2025



Gambar 10. Halaman *Comusion Matrix*

Sumber: analisis data, 2025



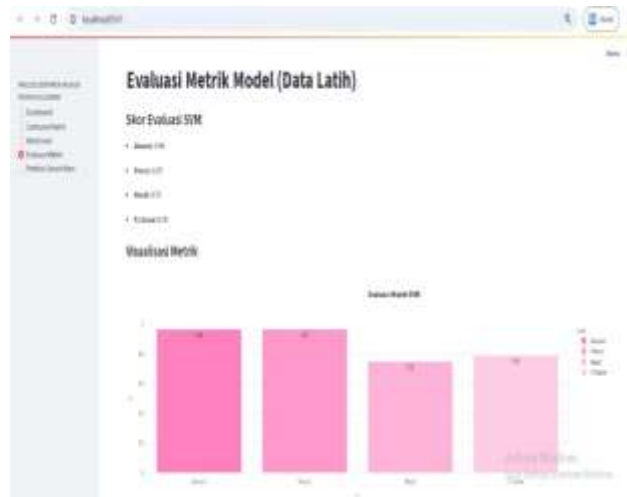
Gambar 11. Halaman Sentimen Positif

Sumber: analisis data, 2025



Gambar 12. Halaman Sentimen Negatif

Sumber: analisis data, 2025



Gambar 13. Halaman Evaluasi Metrik Model

Sumber: analisis data, 2025

## KESIMPULAN

Penelitian ini dapat menyimpulkan sebuah bahwa algoritma yaitu *Support Vector Machine* (SVM) dengan pendekatan *Natural Language Processing* (NLP) mampu memberikan hasil klasifikasi sentimen ulasan pengguna aplikasi Allo Bank dengan performa yang baik, ditunjukkan oleh akurasi 96% dan presisi 97%, meskipun recall (75%) dan F1-score (79%) menunjukkan masih adanya keterbatasan dalam mendeteksi beberapa ulasan pada kelas tertentu. Analisis TF-IDF mengungkapkan bahwa ulasan positif didominasi kata seperti mudah, cepat, dan nyaman, sedangkan ulasan negatif banyak berisi keluhan seperti gagal, error, dan lama, sementara ulasan netral cenderung bersifat deskriptif seperti baru install atau biasa saja. Secara umum, mayoritas ulasan pengguna bersentimen positif, namun masih terdapat kritik yang perlu diperhatikan terkait isu teknis maupun pelayanan. Untuk pengembangan lebih lanjut, disarankan memperluas variasi data latih agar model lebih robust, serta menambahkan fitur visualisasi yang lebih mendalam seperti analisis kata kunci (*word cloud*) atau tren sentimen berdasarkan waktu, sehingga hasilnya dapat memberikan wawasan yang lebih komprehensif.

## DAFTAR PUSTAKA

- Ahmad Fauzi, S.Kom., et al. (2024). **Analisis Sentimen dan Penerapannya dalam NLP**. [https://www.google.co.id/books/edition/Analisis\\_Sentimen\\_Sentiment\\_Analysis/dgoREQAAQBAJ?hl=id&gbpv=0](https://www.google.co.id/books/edition/Analisis_Sentimen_Sentiment_Analysis/dgoREQAAQBAJ?hl=id&gbpv=0). 5.
- Ananda, R., et al. (2024). **Produktivitas Penggunaan VS Code dalam Pengembangan Proyek Data Sains**. <https://ejournal.warunayama.org/kohesi> 5(9), 2-5.

- Anilkumar, C., E., S. V, Kanchana, S., & Kumar, S. B. (2023). **Sentimental Analysis on Product Reviews Using Support Vector Machine and Nave Bayes.** *Applied and Computational Engineering*, 2(1), 1067–1073. <https://doi.org/10.54254/2755-2721/2/20220586>
- Beddy Purnama & Iwan Sofana. (2021). **Implementasi Artificial Intelligence dan Machine Learning Dasar-dasar Python untuk Ilmu Data.** 112.
- Denny Jollyta. (2023). **Klasifikasi Data dalam Pembelajaran Mesin.** [https://www.google.co.id/books/edition/Algoritma\\_Klasifikasi\\_untuk\\_Pemula\\_Solus/y84TEQAAQBAJ?hl=id&gbpv=1](https://www.google.co.id/books/edition/Algoritma_Klasifikasi_untuk_Pemula_Solus/y84TEQAAQBAJ?hl=id&gbpv=1). 28.
- Efori Bu'Ulolo, et al. (2023). **Big Data Analysis Phyton untuk Perguruan Tinggi Pengantar Python, Pandas, dan NumPy dalam Analisis Data.** 13, 43.
- Eldo, H., Ayuliana, A., Suryadi, D., Chrisnawati, G., & Judijanto, L. (2024). **Penggunaan Algoritma Support Vector Machine (SVM) Untuk Deteksi Penipuan pada Transaksi Online.** *Jurnal Minfo Polgan*, 13(2), 1627–1632. <https://doi.org/10.33395/jmp.v13i2.14186>
- Fide, S., Suparti, S., & Sudarno, S. (2021). **Analisis Sentimen Ulasan Aplikasi Tiktok Di Google Play Menggunakan Metode Support vector machine (Svm) Dan Asosiasi.** *Jurnal Gaussian*, 10(3), 346–358. <https://doi.org/10.14710/j.gauss.v10i3.32786>
- Hatta, M., et al. (2024). **Intelligence Systems, Teori dan Implementasi Support Vector Machine (SVM).**
- Muhammad, R., Ardiansyah, M. I., & Afini, A. (2022). **MENGEMBANGKAN SISTEM PERCAKAPAN OTOMATIS BERBASIS LAYANAN PESAN INSTAN.** Uwais Inspirasi Indonesia.
- Munandar, D., Afdal, M., Zarnelly, Z., & Novita, R. (2024). **Analisis Sentimen Ulasan Pengguna Aplikasi Mobile Banking Menggunakan Algoritma K- Nearest Neighbor.** *Jurnal Teknologi Sistem Informasi Dan Aplikasi*, 7(3), 1309–1318. <https://doi.org/10.32493/jtsi.v7i3.41409>
- Putra, K. T., Hariyadi, M. A., & Crysdiyan, C. (2023). **Perbandingan Feature extraction TF-IDF dan BOW Untuk Analisis Sentimen Berbasis SVM.** *Jurnal Cahaya Mandalika (JCM)*, 1449–1463.
- Saputra, M., Wahyuni, S., Informasi, M. T., & Pancabudi, U. P. (2024). **ANALISIS SENTIMEN PENGGUNA PADA APLIKASI BANK DIGITAL KROM.** 10(2), 327–332.
- Suresh, M., Vincent, G., Vijai, C., & Rajendhiran, M. (2024). **Analyse Customer Behaviour and Sentiment Using Natural Language Processing (NLP) Techniques to Improve Customer Service and Personalize Banking.** *Educational Administration: Theory and Practice*, 30(May). <https://doi.org/10.53555/kuey.v30i5.4462>
- Yusuf, D., Guilin, X., & Jiao, D. (2023). **Application of K-Means Clustering Algorithm to Obtain Recommendations for Strategies to Increase the Number of Students in the Information Systems Study Program at ITB Ahmad Dahlan Jakarta.** *Journal of Computer Science Advancements*, 1(4), 204–214. <https://doi.org/10.70177/jsca.v1i4.581>