Indonesian Journal of Computer Science
ISSN 2302-4364 (*print*) dan 2549-7286 (*online*)
Jln. Khatib Sulaiman Dalam, No. 1, Padang, Indonesia, Telp. (0751) 7056199, 7058325
Website: ijcs.stmikindonesia.ac.id | E-mail: ijcs@stmikindonesia.ac.id

Analisis Sentimen Layanan Hotel Menggunakan Algoritma Extra Trees: Studi Kasus pada Ulasan Pelanggan

Windy Aprilita Z.[1], Junadhi[2], Agustin[3], Hadi Asnal[4]
windyaprilita@gmail.com[1], junadhi@sar.ac.id[2], agustin@sar.ac.id[3], hadiasnal@sar.ac.id[4]
Universitas Sains dan Teknologi Indonesia, Teknik Informatika

Informasi Artikel

Diterima : 12 Mei 2024
Direview : 16 Mei 2024
Disetujui : 30 Jun 2024

Kata Kunci

Analisis Sentimen Hotel, Algoritma Extra Trees, Ulasan Pelanggan

Abstrak

Penelitian ini bertujuan untuk menganalisis sentimen layanan hotel berdasarkan ulasan pelanggan menggunakan algoritma Extra Trees. Metode ini diujicobakan pada dataset yang berisi ulasan-ulasan pelanggan tentang layanan hotel. Evaluasi dilakukan dengan memperhitungkan akurasi, presisi, recall, dan skor F1 dari model yang dikembangkan. Hasil penelitian menunjukkan bahwa algoritma Extra Trees mampu mencapai akurasi sebesar 85.05%, dengan presisi sebesar 84.46%, recall sebesar 97.00%, dan skor F1 sebesar 90.17%. Temuan ini mengindikasikan bahwa algoritma Extra Trees memiliki kinerja yang baik dalam menganalisis sentimen layanan hotel berdasarkan ulasan pelanggan. Implikasi dari penelitian ini adalah memberikan panduan kepada hotel untuk memahami dan meningkatkan kualitas layanan mereka berdasarkan umpan balik dari pelanggan. Selain itu, penelitian ini juga dapat menjadi dasar untuk pengembangan lebih lanjut dalam bidang analisis sentimen dan pelayanan pelanggan di industri pariwisata.

Keywords

Hotel Sentiment Analysis, Extra Trees Algorithm, Customer Reviews

Abstract

This research aims to analyze the sentiment of hotel services based on customer reviews using the Extra Trees algorithm. This method was tested on a dataset containing customer reviews about hotel services. The evaluation is done by taking into account the accuracy, precision, recall, and F1 score of the developed model. The results showed that the Extra Trees algorithm was able to achieve an accuracy of 85.05%, with a precision of 84.46%, a recall of 97.00%, and an F1 score of 90.17%. These findings indicate that the Extra Trees algorithm has good performance in analyzing hotel service sentiment based on customer reviews. The implication of this research is to provide guidance to hotels to understand and improve their service quality based on feedback from customers. In addition, this research can also be the basis for further development in the field of sentiment analysis and customer service in the tourism industry.

A. Pendahuluan

Era digital saat ini, ulasan pelanggan telah menjadi salah satu aspek yang sangat berpengaruh dalam industri perhotelan. Ulasan tersebut tidak hanya memberikan gambaran tentang pengalaman individu pelanggan, tetapi juga dapat memengaruhi citra dan reputasi suatu hotel di mata calon tamu. Namun, dengan jumlah ulasan yang terus meningkat secara signifikan dan beragamnya pendapat yang terungkap dalam ulasan tersebut, mengelola dan menganalisis ulasan secara manual telah menjadi tugas yang sangat menantang bagi hotel-hotel. Oleh karena itu, ada kebutuhan mendesak untuk mengembangkan pendekatan yang lebih efisien dalam mengelola ulasan pelanggan, terutama dalam konteks penilaian sentimen.

Salah satu pendekatan yang menarik adalah melalui analisis sentimen otomatis menggunakan teknik-teknik dari bidang Machine Learning. Dengan memanfaatkan kemajuan dalam bidang ini, hotel dapat mengotomatisasi proses penilaian ulasan pelanggan untuk memahami sentimen secara keseluruhan, mulai dari pujian hingga keluhan. Algoritma Machine Learning, seperti Extra Trees, adalah salah satu dari banyak alat yang dapat digunakan untuk menganalisis sentimen dengan tingkat akurasi yang tinggi. Namun, meskipun teknik analisis sentimen telah menjadi semakin populer, penerapannya dalam konteks layanan hotel masih belum sepenuhnya dieksplorasi. Dalam industri perhotelan, faktor-faktor khusus seperti keragaman bahasa dan terminologi yang digunakan dalam ulasan, serta variasi dalam tingkat kepuasan pelanggan, dapat menjadi tantangan tersendiri dalam menerapkan algoritma analisis sentimen dengan efektif. Oleh karena itu, penelitian lebih lanjut diperlukan untuk mengevaluasi keefektifan algoritma analisis sentimen, khususnya algoritma Extra Trees, dalam menghadapi tantangan-tantangan tersebut. Dengan mengadopsi pendekatan analisis sentimen menggunakan algoritma Extra Trees, hotel dapat mendapatkan manfaat yang signifikan.

Manajemen Hotel dapat mengidentifikasi pola umum dalam ulasan pelanggan, memahami sentimen umum terhadap layanan yang ditawarkan, serta mengukur tingkat kepuasan pelanggan secara lebih objektif. Lebih dari itu, dengan mengenali aspek-aspek tertentu yang mungkin perlu perbaikan, hotel dapat dengan cepat mengambil tindakan korektif yang diperlukan untuk meningkatkan kualitas layanan. Dengan demikian, penelitian tentang analisis sentimen layanan hotel menggunakan algoritma Extra Trees bukan hanya akan memberikan wawasan yang berharga bagi industri perhotelan, tetapi juga dapat membantu hotel-hotel dalam memperkuat posisi di pasar yang semakin kompetitif.

Beberapa penelitian yang berkaiatan dengan algoritma Extra Trees seperti yang dilakukan oleh (Rachmansyah & Astriratma, 2023) yang menghasilkan model menggunakan algoritma Extra Trees tanpa metode oversampling SMOTE pada rasio 80% data latih dan 20% data uji dengan nilai akurasi sebesar 79,8%, precision 63,1%, dan recall 56,1%. Penelitian yang dilakukan oleh (A. S. Aribowo et al., 2020) yang memperoleh hasil pengembangan model sebanyak lima jenis model diuji coba, yaitu Naive Bayes (NB), Support Vector Machine (SVM), Decision Tree, Random Forest, dan pengklasifikasi Extra Tree. Hasilnya adalah sebuah model dengan akurasi maksimum 89,68% dengan menggunakan kombinasi preprocessing standar (mengubah emoticon dan menangani kata-kata yang tidak

terstruktur), ekstraksi fitur count vectorizer, dan pengklasifikasi model Extra Tree. Berikutnya penelitian yang dilakukan oleh (Iqbal et al., 2023) dengan hasil bahwa algoritma Random Forest (RF) dengan model Seleksi atribut memiliki kinerja terbaik dalam mendeteksi penyakit dementia, dengan akurasi sebesar 91.56%. Ini merupakan angka yang lebih tinggi dibandingkan dengan algoritma lain seperti Extra Tree (ET), Ridge, dan Linear Discriminant Analysis (LDA), yang masing-masing memiliki akurasi 91.44%, 90.44%, dan 90.44%.

B. Metode Penelitian

Proses melakukan sebuah penelitian data dan informasi yang bersifat objektif yang akan digunakan sebagai titik acuan dalam penelitian, dengan adanya data-data tersebut di harapkan penelitian yang di hasilkan adalah penelitian yang berkualitas. Proses dalam melakukan penelitian ini digambarkan sebagai berikut:

Gambar 1. Tahapan Penelitian

Tahapan yang dilakukan dapat dijelaskan sebagai berikut:

1. Identifikasi Masalah

Pembuatan sistem ini diawali dengan menentukan masalah, dalam penelitian ini dapat diidentifikasi masalahnya adalah kualitas layanan hotel yang mungkin belum optimal. Ulasan pelanggan dapat menjadi cerminan dari kepuasan atau ketidakpuasan mereka terhadap layanan yang diberikan oleh hotel. Pemahaman yang kurang mendalam terhadap sentimen pelanggan. Ulasan pelanggan seringkali beragam dan memerlukan analisis yang cermat untuk memahami pandangan pelanggan secara menyeluruh.

2. Pengumpulan Dataset

Sumber data ini dapat berupa platform online seperti situs web reservasi hotel (contohnya Booking.com, Agoda, atau Expedia), platform ulasan seperti TripAdvisor, atau media sosial terkait (seperti Twitter atau Instagram) di mana pengguna memberikan ulasan tentang pengalaman mereka di hotel.

3. Text Processing

Tahap *Text Processing* diartikan sebagai tahap awal untuk untuk membersihkan kata-kata yang tidak perlu atau kata-kata yang tidak memiliki makna. Proses keseluruhan dilakukan menggunakan program python3, sehingga dilakukan secara otomatis. *Preprocessing* yang akan dilakukan mencakup berbagai aktivitas, diantaranya adalah:

1. *Cleaning*
   Menghilangkan karakter yang tidak diinginkan seperti tanda baca, url, emoji, angka, atau karakter khusus dari teks.
2. *Casefolding*
   Mengubah teks menjadi huruf kecil atau huruf besar yang bertujuan untuk menghilangkan perbedaan yang ditimbulkan oleh perbedaan penulisan huruf besar dan kecil dalam dokumen teks.
3. *Tokenization*
   Memecah teks menjadi kata-kata atau frase yang lebih kecil.
4. *Stopword Removal*
   Menghilangkan kata-kata yang tidak memiliki makna yang signifikan seperti “apakah”, “ke”, “luar”, ”biasanya”, dll.
5. *Filtering*
   Menghilangkan kata-kata yang tidak sesuai dengan topik yang dianalisis, hal ini akan membuat data lebih bersih dan akurat sebelum digunakan dalam proses analisis.
6. *Stemming*
   Menghilangkan imbuhan dari kata-kata untuk mengurangi variasi kata yang sama.

4. Pembobotan Kata TF-IDF

Metode TF-IDF (*Terms Frequency-Inverse Document Frequency*) merupakan suatu cara untuk memberikan bobot hubungan suatu kata (*term*) terhadap dokumen. Metode ini menggabungkan dua konsep untuk perhitungan bobot,

yaitu frekuensi kemunculan sebuah kata di dalam sebuah dokumen tertentu dan inverse frekuensi dokumen yang mengandung kata tersebut. Frekuensi kemunculan kata di dalam dokumen yang diberikan menunjukkan seberapa penting kata itu di dalam dokumen tersebut. Frekuensi dokumen yang mengandung kata tersebut menunjukkan seberapa umum kata tersebut. Sehingga bobot hubungan antara sebuah kata dan sebuah dokumen akan tinggi apabila frekuensi kata tersebut tinggi di dalam dokumen dan frekuensi keseluruhan dokumen yang mengandung kata tersebut yang rendah pada kumpulan dokumen. Rumus untuk TF-IDF:

$$tf = 0{,}5 + 0{,}5\ x\ \frac{tf}{\max(tf)}$$

$$idf_t = \log\left(\frac{D}{df_t}\right)$$

$$W_{d,t} = tf_{d,t}\ x\ IDF_{d,t}$$

Keterangan:
tf = banyaknya kata yang dicari pada sebuah dokumen
max (tf) = jumlah kemunculan terbanyak term pada dokumen yang sama.
Nilai *D* = total dokumen
df_t = jumlah dokumen yang mengandung term t.
IDF = *Inversed Document Frequency* (log2 (D/df))
d = dokumen ke-d
t = kata ke-t dari kata kunci
W = bobot dokumen ke-d terhadap kata ke-t

Fungsi metode TF-IDF adalah untuk mencari representasi nilai dari tiap-tiap dokumen dari suatu kumpulan data *training* (*training set*) dimana nantinya dibentuk suatu vektor Antara dokumen dengan kata (*documents with terms*) yang kemudian untuk kesamaan antar dokumen dengan cluster akan ditentukan oleh sebuah prototype vektor yang disebut juga dengan *cluster centroid*.

5. Pelabelan *Lexicon Vader*

Pendekatan berbasis leksikon (*Lexicon-based*) bergantung pada leksikon sentimen, yaitu kumpulan istilah sentimen yang diketahui dan dikompilasi sebelumnya. Sumber daya yang dapat digunakan pada analisis sentimen berbasis leksikon secara umum ada 2 jenis, yaitu berbasis kamus dan berbasis corpus yang menggunakan metode statistik atau semantik untuk menemukan polaritas sentiment. Penulis mengunakan sumber daya kamus pada penelitian ini, dengan cara memanfaatkan library TextBlob pada Pyhton yang menyediakan kamus berisi leksikon sentimen.

6. Klasifikasi Data

Sebelum melakukan klasifikasi data, perlu dilakukannya beberapa tahapan seperti *spliting data,* pembangunan algoritma Extra Tress dan melatih algoritma tersebut untuk melakukan evaluasi.

Spliting data digunakan membagi data menjadi dua bagian, yaitu data latih (*training*) dan data uji (*testing*). Pada penelitian ini, penulis melakukan empat percobaan.

1. Percobaan kedua, 70% data sebagai data latih dan 30% sebagai data uji.
2. Percobaan ketiga, 80% data sebagai data latih dan 20% sebagai data uji.

Setelah algoritma Extra Tress dilatih, tahap selanjutnya adalah melakukan evaluasi terhadap performa dari algoritma Extra Tress, performa akan diuji berdasarkan metode *confusion matrix* dengan menggunakan data uji.

7. Hasil

Setelah data diproses dengan algoritma Extra Tress maka akan di peroleh hasil berupa data yang menunjukkan kategori postitif, negatif, dan netral. Pada saat klasifikasi, algoritma akan mencari probabilitas tertinggi dari semua kategori dokumen yang diujikan. Selanjutnya data divisualisasikan dalam bentuk diagram agar menghasilkan grafik dari masing-masing kategori.

8. Evaluasi Model Convusion Matrix

Cara mengetahui performa dari metode Extra Tress, maka dilakukan pengujian terhadap model. Hasil klasifikasi akan ditampilkan dalam bentuk *confusion matrix*. Nilai akurasi yang didapatkan dihitung berdasarkan jumlah nilai dari diagonal *confusion matrix* dibagi dengan jumlah seluruh data. Selanjutnya evaluasi dilakukan untuk mengetahui kinerja model. Evaluasi model dilakukan dengan cara melihat tingkat akurasi metode melalui *confusion matrix* dan tabel akurasi serta presisi untuk tiap model. Setelah data uji diujikan, maka akan menghasilkan daftar kelas-kelas dari data uji, sebut saja prediksi kelas. Kemudian prediksi kelas dibandingkan dengan kelas yang sebenarnya dari data uji yang disembunyikan sebelumnya. Sehingga dapat dilihat dan dihitung nilai *accuration*, *precision*, *recall*, dan *f1-score*.

C. Hasil dan Pembahasan

1. Pembahasan

Pada tahap implementasi penelitian, proses yang dilakukan adalah sebagai berikut:

A. Import Library

Sebelum menganalisis data, dilakukan impor library yang diperlukan untuk proses tersebut. Kode program Python yang digunakan disajikan dalam Gambar 2.

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
sns.set_style("whitegrid")

#set warning
import warnings
warnings.filterwarnings('ignore')


pd.pandas.set_option('display.max_columns', None)
```

Gambar 2. Import Library

B. Memuat Data

Selanjutnya, dataset yang telah diperoleh melalui teknik crawling dibaca. Kode program Python yang digunakan disajikan pada Gambar 3.

```
filename = "reviewHotel.csv"
df = pd.read_csv(filename, encoding = 'latin-1')
df.head()
```

Gambar 3. Memuat Data

Hasil dari eksekusi kode program dapat dilihat pada Gambar 4 di bawah ini.

	Hotel_name	name	rating	review
0	Hotel Gran Mahakam	Robert Gow	4.0	Old boutique hotel but very well maintained. R...
1	Hotel Gran Mahakam	Oggi Gunadi	5.0	Outstanding service from the first sight. Firs...
2	Hotel Gran Mahakam	Fifi Muliadi	5.0	When I was working, I frequently visited the h...
3	Hotel Gran Mahakam	Renatha P	5.0	Having all you can eat luncheon in private roo...
4	Hotel Gran Mahakam	Swathi V	5.0	My stay was a combination of quarantine and no...

Gambar 4. Hasil Pemuatan Data

C. Cleaning Data

Proses berikutnya adalah membersihkan data, yang bertujuan untuk menghapus data yang duplikat, kata imbuhan, simbol, karakter, dan data yang tidak diperlukan. Proses pembersihan data ini melibatkan beberapa langkah, termasuk case folding, tokenizing, penghapusan stop word, dan stemming. Kode program Python yang digunakan untuk proses ini ditampilkan dalam Gambar 5 di bawah ini.

```
import string
import re
```

```
def clean_text(text):
    return re.sub('[^a-zA-Z]', ' ', text).lower()
df['cleaned_text'] = df['Review'].apply(lambda x: clean_text(x))
df['label'] = df['Rating'].map({1.0:0, 2.0:0, 3.0:0, 4.0:1, 5.0:1})
```

Gambar 5. Cleaning Data

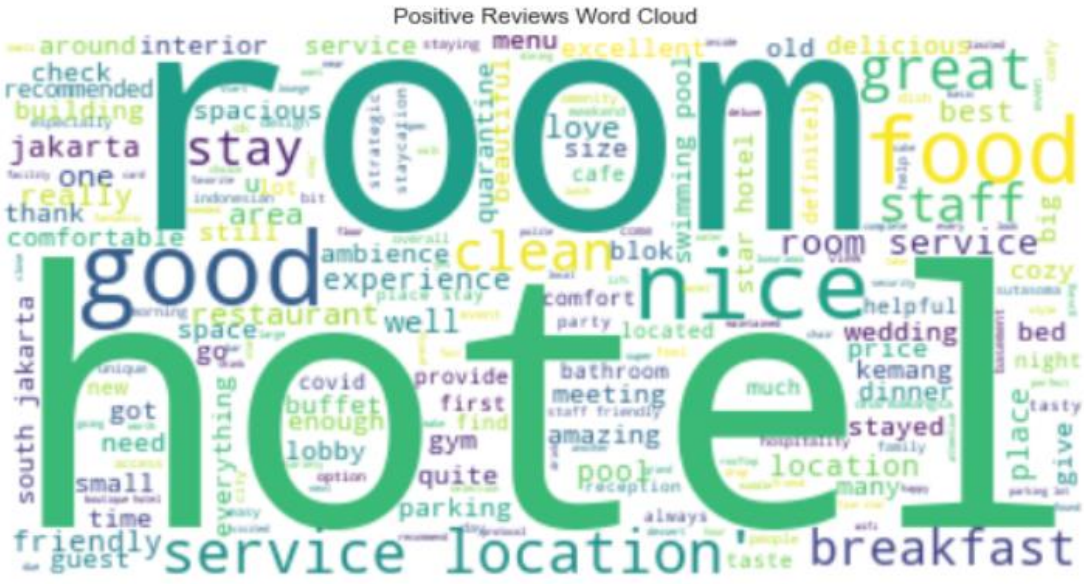
Setelah proses pembersihan data selesai dilakukan, hasil dari program disajikan pada Gambar 6 di bawah ini.

	Rating	Review	cleaned_text	label	Review_len	punct	tokens	lemmatized_review
0	4.0	Old boutique hotel but very well maintained. R...	old boutique hotel but very well maintained r...	1	490	2.4	[old, boutique, hotel, but, very, well, mainta...	old boutique hotel well maintained reception c...
1	5.0	Outstanding service from the first sight. Firs...	outstanding service from the first sight firs...	1	309	1.9	[outstanding, service, from, the, first, sight...	outstanding service first sight first impressi...
2	5.0	When I was working, I frequently visited the h...	when i was working i frequently visited the h...	1	404	2.5	[when, i, was, working, i, frequently, visited...	working frequently visited hotel meeting lunch...
3	5.0	Having all you can eat luncheon in private roo...	having all you can eat luncheon in private roo...	1	171	5.8	[having, all, you, can, eat, luncheon, in, pri...	eat luncheon private room pax round table fine...
4	5.0	My stay was a combination of quarantine and no...	my stay was a combination of quarantine and no...	1	389	2.6	[my, stay, was, a, combination, of, quarantine...	stay combination quarantine normal stay easily...

Gambar 6. Hasil Cleaning Data

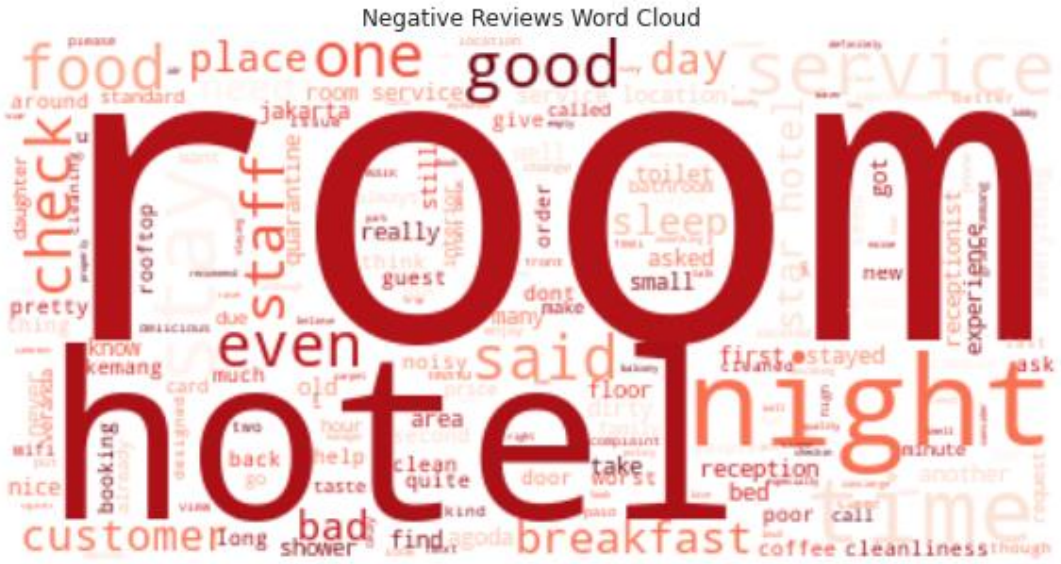
D. Visualisasi Word Cloud

Pada visualisasi Word Cloud yang terbagi menjadi dua kategori, yaitu positif dan negatif, kata-kata yang sering muncul dalam ulasan positif akan ditampilkan dengan ukuran yang lebih besar pada bagian positif, sedangkan kata-kata yang sering muncul dalam ulasan negatif akan ditampilkan dengan ukuran yang lebih besar pada bagian negatif. Berikut adalah hasil word cloud review positif dari penelitian ini. Sentimen yang dominan dalam word cloud ini adalah positif. Hal ini dapat dilihat dari banyaknya kata-kata yang memiliki makna positif, seperti "*good*", "*excellent*", "*comfortable*", "*enjoyable*", dan "*recommended*".



Gambar 7. Positif Review

Selanjutnya disajikan hasil word cloud review positif dari penelitian ini. Sentimen yang dominan dalam word cloud ini adalah negatif. Hal ini dapat dilihat dari banyaknya kata-kata yang memiliki makna negatif, seperti "*bad*", "*dirty*", "*poor*", dan "*disappointed*".



Gambar 8. Negatif Review

2. Hasil

A. Implementasi Model Extra Trees

Implementasi Model Extra Trees merupakan tahap penting dalam proses pengembangan sistem analisis sentimen untuk ulasan layanan hotel. Model Extra Trees adalah salah satu algoritma klasifikasi yang digunakan dalam machine learning untuk memprediksi sentimen dari suatu teks berdasarkan fitur-fitur yang diekstraksi dari teks tersebut. Berikut adalah kode progra yang digunakan pada model Extra Tress.

```
from sklearn.ensemble import ExtraTreesClassifier
classifier = ExtraTreesClassifier(n_estimators=150, random_state=50)
classifier.fit(X_train_vect, y_train)
extra_trees_pred = classifier.predict(X_test_vect)

# Classification report
print(classification_report(y_test, extra_trees_pred))

# Confusion Matrix
class_label = ["negative", "positive"]
df_cm = pd.DataFrame(confusion_matrix(y_test, extra_trees_pred), index=class_label, columns=class_label)
sns.heatmap(df_cm, annot=True, fmt='d')
plt.title("Confusion Matrix")
plt.xlabel("Predicted Label")
plt.ylabel("True Label")
plt.show()
```

Gambar 9. Model Extra Trees

Hasil dari model Extra Trees diatas disajikan pada gambar 10 dibawah ini.

	precision	recall	f1-score	support
0	0.71	0.48	0.57	21
1	0.78	0.90	0.84	42
accuracy			0.76	63
macro avg	0.74	0.69	0.70	63
weighted avg	0.76	0.76	0.75	63

Gambar 10. Hasil Model Extra Trees

Berdasarkan tabel evaluasi diatas, menggunakan algoritma Extra Trees. Algoritma ini memiliki kinerja yang cukup baik dalam mengidentifikasi sampel positif dan negatif, dengan nilai presisi, recall, dan f1-score yang tinggi.

B. K-Fold Cross Validation

K-Fold Cross-Validation adalah metode untuk meningkatkan akurasi model, mengurangi overfitting, dan memastikan generalisasi model yang baik. Dalam proses ini, data latih dibagi menjadi K bagian, dan pada setiap iterasi, satu bagian digunakan sebagai data validasi sementara yang lainnya digunakan sebagai data latih. Prosedur ini diulang K kali untuk memastikan setiap bagian menjadi data validasi satu kali. Ini memungkinkan penggunaan seluruh data latih secara bertahap dan menghasilkan hasil yang lebih akurat. Berikut ini adalah kode program yang digunakan.

```
from sklearn.model_selection import cross_val_score

models = [
          MultinomialNB(),
          LogisticRegression(),
          RandomForestClassifier(n_estimators = 150),
          SVC(kernel = 'linear'),
          KNeighborsClassifier(n_neighbors = 5),
          ExtraTreesClassifier(n_estimators=150, random_state=50)
         ]
names = ["Extra Trees"]
for model, name in zip(models, names):
    print(name)
    for score in ["accuracy", "precision", "recall", "f1"]:
        print(f" {score} - {cross_val_score(model, X_train_vect, y_train, scoring=score, cv=10).mean()} ")
    print()
```

Gambar 11. K-Fold Cross Validation

Selanjutnya kode program diatas dijalankan dan berikut ini adalah hasilnya.

```
Extra Trees
 accuracy - 0.8504761904761905
 precision - 0.8446087246087245
 recall - 0.9700000000000001
 f1 - 0.9017229437229437
```

Gambar 12. Hasil Akurasi setelah K-Fold Cross Validation

Berdasarkan gambar, terlihat bahwa akurasi algoritma Extra Trees mengalami peningkatan setelah dilakukan k-fold cross validation. Hal ini dapat dilihat dari kurva yang menunjukkan tren naik dari k = 2 hingga k = 10. Berdasarkan hasil k-fold cross validation, dapat disimpulkan bahwa k-fold cross validation dapat membantu meningkatkan akurasi algoritma Extra Trees. Hal ini menunjukkan bahwa k-fold cross validation adalah teknik yang bermanfaat untuk mengevaluasi dan meningkatkan kinerja model machine learning.

D. Simpulan

Dari hasil penelitian mengenai analisis sentimen layanan hotel dengan menggunakan algoritma Extra Trees, ditemukan bahwa model yang dikembangkan mampu mencapai tingkat akurasi yang memuaskan, yaitu sebesar 85.05%. Hal ini menandakan bahwa model memiliki kemampuan yang baik dalam mengklasifikasikan sentimen dari ulasan pelanggan terhadap layanan hotel. Lebih lanjut, nilai precision sebesar 84.46% menunjukkan bahwa model dapat mengidentifikasi dengan tepat ulasan yang positif atau negatif, sedangkan recall sebesar 97.00% menunjukkan kemampuan model untuk menemukan sebagian besar ulasan yang seharusnya diklasifikasikan sebagai positif. Skor F1 sebesar 90.17% juga menunjukkan bahwa model memiliki keseimbangan yang baik antara precision dan recall. Oleh karena itu, hasil penelitian menunjukkan bahwa algoritma Extra Trees dapat digunakan secara efektif dalam melakukan analisis sentimen terhadap ulasan pelanggan layanan hotel.

E. Ucapan Terima Kasih

Terima kasih penulis sampaikan kepada Universitas Sains dan Teknologi Indonesia yang telah meluangkan waktunya untuk membantu penulis dalam menyelesaikan penelitian ini serta bapak ibu dosen pembimbing dan penguji yang teribat memberikan masukan dan arahan dalam penelitian ini.

F. Referensi

[1] Pratama, A., Cahya, R. & Ratnawati, D. E., 2017. Implementasi Algoritme Support Vector Machine (SVM) untuk Prediksi Ketepatan Waktu Kelulusan Mahasiswa. Pengembangan Teknologi Informasi dan Ilmu Komputer, Volume 2, pp. 1704-1708.

[2] Rizky Hilman Faturrahman, dkk. 2022. Klasifikasi Sentimen Ulasan Film Menggunakan Support Vector Machine, Information Gain, dan N-Grams. Vol.9, No.3 pp. 1928-1933

[3] Alfita Nuriza, 2020. Klasifikasi ReviewProduk Kecantikan Pada Aplikasi Sociolla Menggunakan AlgoritmeModified K-Nearest Neighbor (MK-NN)dengan Pembobotan BM25. Vol. 4, No. 10, Oktober 2020, hlm. 3426-3431

[4] Muhammad Muadin, dkk. 2023. Implementasi Metode Support Vector Machine Pada Opinion Mining Masyarakat Terkait ChatGPT. Vol. 7, No.1, Juni 2023, Hlm 78-84

[5] Junadhi, Agustin, Rifqi, M., & Anam, M. K. (2022). Sentiment Analysis of Online Lectures using K-Nearest Neighbors based on Feature Selection. Jurnal Nasional Pendidikan Teknik Informatika (JANAPATI), 11(3), 216–225. https://doi.org/10.23887/janapati.v11i3.51531

[6] Alhaq, Z., Mustopa, A., & Santoso, J. D. (2021). Penerapan Metode Support Vector Machine Untuk Analisis Sentimen Pengguna Twitter.
[7] Dwi Cahyo, M. P., Widodo, & Prasetya Adhi, B. (2019). Kinerja Algoritma Support Vector Machine dalam MenentukanKebenaran Informasi Banjir di Twitter. PINTER : Jurnal Pendidikan Teknik Informatika Dan Komputer, 3(2), 116–121. https://doi.org/10.21009/pinter.3.2.5
[8] Fitriyana, V., Hakim, L., Candra Rini Novitasari, D., Hanif Asyhar, A., Studi Matematika, P., Sains Dan Teknologi, F., Sunan Ampel Surabaya, U., & Timur, J. (2023). Analisis Sentimen Ulasan Aplikasi Jamsostek Mobile Menggunakan Metode Support Vector Machine. In Jurnal Buana Informatika (Vol. 14, Issue 1).
[9] R. Maulana, P. A. Rahayuningsih, W. Irmayani, D. Saputra, and W. E. Jayanti, "Improved Accuracy of Sentiment Analysis Movie Review Using Support Vector Machine Based Information Gain," J. Phys. Conf. Ser., vol. 1641, no. 1, pp. 0–6, 2020, doi: 10.1088/1742-6596/1641/1/012060.
[10] N. Octaviani Faomasi Daeli, "Sentiment Analysis on Movie Reviews Using Information Gain and KNearest Neighbor," J. Data Sci. Its Appl., 2020.
[11] O. M. Bafadhal dan A. D. Santoso, "Memetakan Pesan Hoaks Berita Covid-19 Di Indonesia Lintas Kategori, Sumber, Dan Jenis Disinformasi," Bricol. J. Magister Ilmu Komun., vol. 6, no. 02, hal. 235, 2020, doi: 10.30813/bricolage.v6i02.2148.
[12] E. Mailoa, "Analisis Node dengan Centrality dan Follower Rank pada Twitter," J. RESTI (Rekayasa Sist. dan Teknol. Informasi), vol. 4, no. 5, hal. 937–942, 2020, doi: 10.29207/resti.v4i5.2398.
[13] M. Hanafiah, A. Herdiani, W. Astuti, dan M. Kom, "Klasifikasi Spam Tweet Pada Twitter Menggunakan Metode Naïve Bayes (Studi Kasus : Pemilihan Presiden 2019)," in e-Proceeding of Engineering, 2019, vol. 6, no. 2, hal. 9111–9120.
[14] S. Anggelia dan A. Syaifudin, "SENTIMEN WARGANET MAHASISWA TERHADAP COVID-19," LITERASI, vol. 5, hal. 49–57, 2021, doi: http://dx.doi.org/10.25157/literasi.v5i1.5149.