

```

true = true.reset_index(drop=True)
pred = pd.Series(pred).reset_index(drop=True)
rmse = np.sqrt(np.mean((true - pred) ** 2))
mae = np.mean(np.abs(true - pred))
mape = np.mean(np.abs((true - pred) / true)) * 100 if all(true != 0) else None
return rmse, mae, mape

arima_rmse, arima_mae, arima_mape = evaluate_forecast(test,
arima_forecast)
sarima_rmse, sarima_mae, sarima_mape = evaluate_forecast(test,
sarima_forecast)

# Format hasil
result = {
    "input_filter": {
        "kecamatan": kecamatan,
        "tipe_sampah": tipe_sampah
    },
    "metrics": {
        "ARIMA": {"RMSE": arima_rmse, "MAE": arima_mae, "MAPE":
arima_mape},
        "SARIMA": {"RMSE": sarima_rmse, "MAE": sarima_mae, "MAPE":
sarima_mape}
    },
    "forecast": {
        "Train": {
            "Tanggal": train.index.strftime("%Y-%m-%d").tolist(),
            "Nilai": train.values.tolist()
        },
        "Test": {
            "Tanggal": test.index.strftime("%Y-%m-%d").tolist(),
            "Actual": test.values.tolist(),
            "ARIMA": arima_forecast.tolist(),
            "SARIMA": sarima_forecast.tolist()
        },
        "Future": {
            "Tanggal": future_dates.strftime("%Y-%m-%d").tolist(),
            "ARIMA": future_forecast_arima.tolist(),
            "SARIMA": future_forecast_sarima.tolist()
        }
    }
}

return jsonify(result)

```

DAFTAR LAMPIRAN

1. Surat Keterangan Bebas Plagiat



**UNIVERSITAS DARMA PERSADA
UPT PERPUSTAKAAN**

Gedung Rektorat Lantai 3,
Jl. Taman Malaka Selatan, Pondok Kelapa – Jakarta Timur 13450

**SURAT KETERANGAN
HASIL PENGECEKAN TURNITIN**

UPT Perpustakaan Universitas Darma Persada menerangkan telah selesai melakukan pemeriksaan duplikasi/*similarity* menggunakan perangkat lunak Turnitin terhadap hasil karya sebagai berikut:

Judul : PERBANDINGAN ALGORITMA SARIMA DAN ARIMA
UNTUK MEMPREDIKSI POLA PERTUMBUHAN SAMPAH
Penulis : Ahmad Septiawan
NIM : 2019230160
Tgl pemeriksaan : 14 Februari 2026

Dengan hasil Tingkat Kesamaan (*similarity index*) 20%

Demikian Surat Keterangan kami buat, untuk dipergunakan sebagaimana mestinya.

Jakarta, 14 Februari 2026

Ka.UPT Perpustakaan Unsada

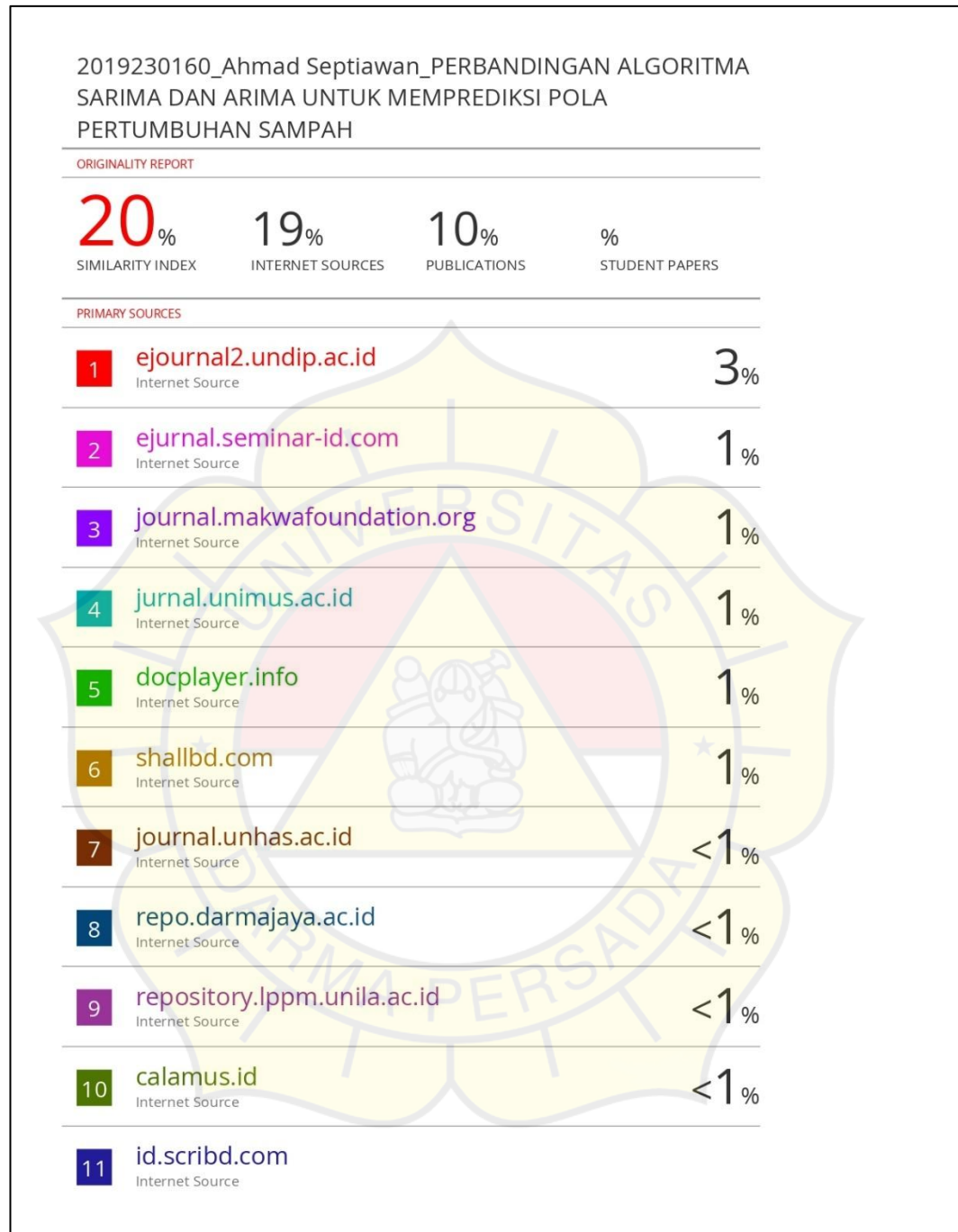


Yus Rusmiyati, SS., MM

Batas maksimal similarity 30% untuk Fakultas Sastra dan Ekonomi

Batas maksimal similarity 25% untuk Fakultas Teknik, Kelautan dan Pasca Sarjana

2. Hasil Turnitin



		<1 %
12	library.binus.ac.id Internet Source	<1 %
13	Amin Padmo Azam Masa, Christina Febriyanti Tobing, Akhmad Irsyad. "PERBANDINGAN METODE HOLT WINTERS EXPONENTIAL SMOOTHING DAN ARIMA PADA PERAMALAN JUMLAH WISATAWAN DI KOTA SAMARINDA", Jurnal Informatika dan Teknik Elektro Terapan, 2026 Publication	<1 %
14	Hira Nur Afifah, Wina Witanti, Fatan Kasyidi. "PREDIKSI PENDAPATAN PADA MITRA TOKO PARFUME TRENDS MENGGUNAKAN METODE VECTOR AUTOREGRESSIVE INTEGRATED MOVING AVERAGE (VARIMA)", Technologia : Jurnal Ilmiah, 2024 Publication	<1 %
15	bkddki.jakarta.go.id Internet Source	<1 %
16	ojsamik.amikmitragama.ac.id Internet Source	<1 %
17	jurnal.polibatam.ac.id Internet Source	<1 %
18	pt.scribd.com Internet Source	<1 %
19	dspace.uui.ac.id Internet Source	<1 %
20	repository.ipb.ac.id:8080 Internet Source	<1 %

21	eprints.undip.ac.id Internet Source	<1 %
22	journal.sinov.id Internet Source	<1 %
23	jurnal.usahid.ac.id Internet Source	<1 %
24	Asri Fornika Sari, Seno D Panjaitan, Bomo W Sanjaya. "Optimasi Pemantauan Kualitas Air Baku Dengan Metode Arima dan Teknologi IOT pada Bak Sedimentasi IPA III", Jurnal Teknologi Lingkungan Lahan Basah, 2024 Publication	<1 %
25	Sri Lestari, Maryana Febryanti. "Analisis Sentimen Mengenai Produk Inovasi Invisible TV Menggunakan Algoritma Naïve Bayes", INTECOMS: Journal of Information Technology and Computer Science, 2024 Publication	<1 %
26	ipm2kpe.or.id Internet Source	<1 %
27	repositori.usu.ac.id Internet Source	<1 %
28	vibdoc.com Internet Source	<1 %
29	Muhammad Redhy Rizani. "QUICK COUNT PEMILIHAN KEPALA DAERAH BERBASIS WEB", Jurnal Sains dan Informatika, 2017 Publication	<1 %
30	Muhammad Rizky Julianto, Yuma Akbar, Tri Wahyudi. "Analisis Sentimen Respon Publik	<1 %

Terhadap Program Internet Gratis di Platform X Melalui Pendekatan Algoritma Naïve Bayes", Jurnal Indonesia : Manajemen Informatika dan Komunikasi, 2024

Publication

31	Muhammad Ziad Iqbal, Yusuf Hendra Pratama, Firmansyah Firmansyah. "Penentuan Peminatan Tugas Akhir Mahasiswa Ilmu Komputer Unizar", Secure And Knowledge-Intelligent Research in Cybersecurity And Multimedia (SAKIRA), 2023	<1 %
32	adenasution.com Internet Source	<1 %
33	smart.stmikplk.ac.id Internet Source	<1 %
34	Tiara Biringpasemba, Irene Tangkawarow, Quido Kainde. "Peramalan Pendapatan Bea Balik Nama Kendaraan Bermotor (BBN-KB) Badan Pendapatan Daerah Sulawesi Utara Menggunakan Metode ARIMA", Jurnal Sistem dan Informatika (JSI), 2025	<1 %
35	repository.stp-bandung.ac.id Internet Source	<1 %
36	widuri.raharja.info Internet Source	<1 %
37	6b1aa66e-13bb-4ed4-a7fe-e6007b3f5822.filesusr.com Internet Source	<1 %

38	Susliansyah Susliansyah, Heny Sumarno, Hendro Priyono, Linda Maulida. "Penerapan Klastering pada Data Mining dalam Menentukan Status Gizi Anak Balita dengan Menggunakan Algoritma K-Medoids", remik, 2026 Publication	<1 %
39	eprints.polsri.ac.id Internet Source	<1 %
40	eprints.sinus.ac.id Internet Source	<1 %
41	repository.bku.ac.id Internet Source	<1 %
42	repository.upi.edu Internet Source	<1 %
43	text-id.123dok.com Internet Source	<1 %
44	www.coursehero.com Internet Source	<1 %
45	eprints.ums.ac.id Internet Source	<1 %
46	journal.universitaspahlawan.ac.id Internet Source	<1 %
47	jurnal.itscience.org Internet Source	<1 %
48	nanopdf.com Internet Source	<1 %
49	openlibrary.telkomuniversity.ac.id Internet Source	<1 %

50	repository.unej.ac.id Internet Source	<1 %
51	w3webdesign.de Internet Source	<1 %
52	aimos.ugm.ac.id Internet Source	<1 %
53	repository.uma.ac.id Internet Source	<1 %
54	123dok.com Internet Source	<1 %
55	Rony Kurniawan Kurniawan, Samari Samari, Sigit Ratnanto. "Komparasi Model Single Moving Avarage & Exponential Smoothing Untuk Peramalan Penjualan AMDK NuCleans", JURNAL NUSANTARA APLIKASI MANAJEMEN BISNIS, 2022 Publication	<1 %
56	citylab.itb.ac.id Internet Source	<1 %
57	core.ac.uk Internet Source	<1 %
58	elreg-03.blogspot.com Internet Source	<1 %
59	epub.imandiri.id Internet Source	<1 %
60	jimfeb.ub.ac.id Internet Source	<1 %
61	pkm.lpkd.or.id Internet Source	<1 %

62	repository.usu.ac.id:8080 Internet Source	<1 %
63	repository.trisakti.ac.id Internet Source	<1 %
64	repository.usd.ac.id Internet Source	<1 %
65	www.parlimen.gov.my Internet Source	<1 %
66	Lita Widiastuti, Wiga Maulana Baihaqi. "Penggunaan Machine Learning untuk Memprediksi Paparan PM2.5 dan Dampaknya terhadap Kesehatan", Jurnal Ilmiah Sains dan Teknologi, 2025 Publication	<1 %
67	M Ilmi Alfaridzi, Rahmad Gunawan, Haris Alfian, Muhammad Fitter Mirano, Hayatun Nazifah, Sri Wahyuni, Kevanda Sondani Illahi. "Perbandingan model SARIMA dan Prophet dalam memprediksi jumlah kunjungan wisatawan mancanegara ke Indonesia berdasarkan data deret waktu bulanan", Jurnal CoSciTech (Computer Science and Information Technology), 2025 Publication	<1 %
68	M Iqbal Mustofa, Dimas Satriadi Iqbal Mustofa, Puput Cendana Sari, Ratih Nur Istiqomah, Alfrido Tegar Bimantoro. "Sistem Informasi Perpustakaan Sekolah Berbasis Dekstop", Binary: Jurnal Teknologi Informasi dan Edukasi, 2024 Publication	<1 %

69 Rifqi Mubarak, Akhmal Angga Syahputra, Abdillah Teguh Permana, Lifa Sholiah, Tarwoto. "Implementasi Data Mining untuk Clustering Lowongan Pekerjaan Menggunakan Metode Algoritma K-Means", Jurnal JTIK (Jurnal Teknologi Informasi dan Komunikasi), 2025

Publication

<1 %

70 Zaid Romegar Mair, Helen Yunita Sari. "Aplikasi Kasir Pada Adibah Boutique Berbasis Desktop", Jurnal Nasional Ilmu Komputer, 2021

Publication

<1 %

71 doaj.org
Internet Source

<1 %

72 eprints.uad.ac.id
Internet Source

<1 %

73 es.scribd.com
Internet Source

<1 %

74 fiskal.kemenkeu.go.id
Internet Source

<1 %

75 fr.scribd.com
Internet Source

<1 %

76 journal.unipdu.ac.id:8080
Internet Source

<1 %

77 journal.unj.ac.id
Internet Source

<1 %

78 journals.unpad.ac.id
Internet Source

<1 %

79 jurnal.kdi.or.id

	Internet Source	<1 %
80	repository.its.ac.id Internet Source	<1 %
81	sinarharapan.id Internet Source	<1 %
82	tdx.maine.edu Internet Source	<1 %
83	Nika Wirana, Noviyanti P.. "Analisis dan Implementasi Sistem Informasi Objek Wisata Bengkayang Menggunakan Pendekatan Berorientasi Objek", Instink: Inovasi Pendidikan, Teknologi Informasi dan Komputer, 2024 Publication	<1 %
84	Suaibah, Shantika Martha, Dadan Kusnandar. "PEMODELAN DOUBLE SEASONAL AUTOREGRESSIVE INTEGRATED MOVING AVERAGE PADA PERAMALAN PENGGUNAAN BEBAN LISTRIK", Bimaster : Buletin Ilmiah Matematika, Statistika dan Terapannya, 2020 Publication	<1 %
85	Syarifah Syila Alkadrie, Madona Yunita Wijaya, Nina Fitriyati. "Evaluasi Performa Metode Exponential Smoothing pada Data Runtun Waktu Hierarkis", The Indonesian Journal of Computer Science, 2025 Publication	<1 %
86	M Rizki, Dadang Priyanto, Galih Hendro Martono, Neny Sulistianingsih, Moch Syahrir. "Perbandingan Algoritma Sarima dan Prophet	<1 %

Untuk Peramalan Trend Penjualan Voucher
Game Online", Jurnal Minfo Polgan, 2025

Publication

87

etheses.uin-malang.ac.id

Internet Source

<1%

Exclude quotes Off

Exclude matches Off

Exclude bibliography Off



3. Coding

```
from flask import Flask, request, jsonify
import pandas as pd
import numpy as np
import os
import mysql.connector
import json

# Machine Learning
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import silhouette_score

# Time Series Modeling (SARIMA dan Arima)
from statsmodels.tsa.arima.model import ARIMA
from statsmodels.tsa.statespace.sarimax import SARIMAX
from sklearn.metrics import mean_absolute_error, mean_squared_error
from math import sqrt

# Visualisasi
import matplotlib.pyplot as plt
import seaborn as sns

app = Flask(__name__)

# Konfigurasi koneksi DB
DB_USER = os.getenv("DB_USER", "root")
DB_PASSWORD = os.getenv("DB_PASSWORD", "")
DB_HOST = os.getenv("DB_HOST", "localhost")
DB_NAME = os.getenv("DB_NAME", "tpst_btg")

# Fungsi buat koneksi DB
def create_connection():
    return mysql.connector.connect(
        user=DB_USER,
        password=DB_PASSWORD,
        host=DB_HOST,
        database=DB_NAME,
        use_pure=True,
        autocommit=True
    )

# Fungsi untuk Evaluasi Forecast
def evaluate_forecast(true, pred):
```

```

rmse = np.sqrt(mean_squared_error(true, pred))
mae = mean_absolute_error(true, pred)
mape = np.mean(np.abs((true - pred) / true)) * 100
return rmse, mae, mape

@app.route('/data', methods=['POST'])
def get_processed_data():
    try:
        data_json = request.get_json()

        if not data_json or 'filename' not in data_json:
            return jsonify({'error': 'Invalid JSON or missing filename'}), 400

        filename = data_json['filename']
        if not filename:
            return jsonify({'error': 'Filename is required'}), 400

        file_path = os.path.join('C:/xampp/htdocs/sarima/upload', filename)
        if not os.path.exists(file_path):
            return jsonify({'error': 'File not found'}), 404

        # Load Excel file
        data = pd.read_excel(file_path, engine='openpyxl')

        # Preprocessing
        if 'Kota / Kabupaten' in data.columns:
            data['Kota / Kabupaten'] = data['Kota /
Kabupaten'].astype(str).str.strip().str.lower()
            data = data[data['Kota / Kabupaten'] != 'lembaga']

        # Pilih kolom numerik yang relevan (hindari kolom ID/urutan)
        num_cols = data.select_dtypes(include=['float64',
'int64']).columns.difference(['No', 'Tahun', 'Sort'])

        # Ganti 0 dan NaN dengan mean per 'Kecamatan / Lokasi Kerja'
        for col in num_cols:
            data[col] = data.groupby('Kecamatan / Lokasi Kerja')[col].transform(
                lambda x: x.replace(0, pd.NA).fillna(x.mean())
            )

        # Ubah ke integer setelah pembulatan
        data[num_cols] = data[num_cols].round().astype('int64')

        # Convert to pure Python dict for JSON safety

```

```

result = json.loads(data.to_json(orient='records'))
return jsonify(result), 200

except Exception as e:
    return jsonify({'error': str(e)}), 500

@app.route('/forecast', methods=['POST'])
def forecast():
    try:
        # Ambil input dari form
        kecamatan = request.form.get('kecamatan')
        tipe_sampah = request.form.get('tipe_sampah')

        if not kecamatan or not tipe_sampah:
            return jsonify({"error": "Kecamatan dan Tipe Sampah harus dipilih"}), 400

        # Ambil data dari DB
        conn = create_connection()
        cur = conn.cursor()

        if kecamatan.lower() == "semua":
            cur.execute("SELECT * FROM data_sampah")
        else:
            cur.execute("SELECT * FROM data_sampah WHERE kecamatan = %s",
(kecamatan,))

        data = cur.fetchall()
        headers = [desc[0] for desc in cur.description]
        cur.close()
        conn.close()

        if not data:
            return jsonify({"error": "Tidak ada data ditemukan."}), 404

        # Buat DataFrame
        df = pd.DataFrame(data, columns=headers)
        df.set_index(headers[0], inplace=True)

        # Cek kolom tersedia
        if tipe_sampah not in df.columns:
            return jsonify({"error": f"Tipe sampah '{tipe_sampah}' tidak ditemukan."}),
400

        # Konversi bulan ke Inggris
        bulan_mapping = {

```

```

        'Januari': 'January', 'Februari': 'February', 'Maret': 'March',
        'April': 'April', 'Mei': 'May', 'Juni': 'June', 'Juli': 'July',
        'Agustus': 'August', 'September': 'September', 'Oktober': 'October',
        'November': 'November', 'Desember': 'December'
    }
    df['Bulan_En'] = df['bulan'].map(bulan_mapping)
    df['Tanggal'] = pd.to_datetime(df['Bulan_En'] + ' ' + df['tahun'].astype(str),
format='%B %Y')
    df['Tanggal'] = df['Tanggal'].dt.to_period('M').dt.to_timestamp()

    # Buat time series
    ts = df.groupby('Tanggal')[tipe_sampah].sum().sort_index()

    # Paksa menjadi range bulanan
    ts = ts.asfreq('MS')

    # Jika ada NaN karena bulan yang hilang, bisa diisi 0 atau interpolasi
    ts = ts.fillna(0)

    if ts.empty:
        return jsonify({"error": "Data time series kosong setelah pemrosesan."}),
404

    # Train-test split
    train_size = int(len(ts) * 0.8)
    train, test = ts[:train_size], ts[train_size:]

    # Model ARIMA
    arima_model = ARIMA(train, order=(1, 1, 1)).fit()
    arima_forecast = arima_model.forecast(steps=len(test))

    # Model SARIMA
    sarima_model = SARIMAX(train, order=(1, 1, 1), seasonal_order=(1, 1, 1,
12)).fit()
    sarima_forecast = sarima_model.forecast(steps=len(test))

    # Forecast masa depan
    future_steps = 12
    future_dates = pd.date_range(start=ts.index[-1] + pd.DateOffset(months=1),
periods=future_steps, freq='MS')
    future_forecast_arima = arima_model.forecast(steps=future_steps)
    future_forecast_sarima = sarima_model.forecast(steps=future_steps)

    # Evaluasi
    def evaluate_forecast(true, pred):

```

```
except Exception as e:
    if 'conn' in locals() and conn:
        conn.close()
    print("Forecast Error:", str(e))
    return jsonify({"error": f"Terjadi kesalahan: {str(e)}"}), 500
# except Exception as e:
#     import traceback
#     traceback.print_exc() # ini akan tampilkan semua stack trace error
#     return jsonify({"error": str(e)}), 500

if __name__ == '__main__':
    app.run(debug=True)
```

