

# Sistem Deteksi dan Penyaringan Pesan Spam SMS Berbasis Android

**Candra Surya Setiyawan<sup>1</sup>, Mochamad Bressandy Syach Putra<sup>2</sup>**

<sup>1,2</sup>Teknik Informatika, Fakultas Teknik dan Ilmu Komputer, Universitas Nusantara PGRI Kediri

E-mail: [\\*1ndracansurya1927@gmail.com](mailto:ndracansurya1927@gmail.com) , [2putrasyach2@gmail.com](mailto:putrasyach2@gmail.com)

**Abstrak** – Pesan spam pada layanan Short Message Service (SMS) masih sering dimanfaatkan untuk penipuan dan penyalahgunaan data pribadi sehingga diperlukan sistem yang mampu melakukan penyaringan pesan secara otomatis pada perangkat seluler. Penelitian ini mengembangkan aplikasi penyaringan SMS spam berbasis Android dengan mengintegrasikan antarmuka Flutter dan mesin klasifikasi berbasis Python melalui plugin Chaquopy. Metode klasifikasi yang digunakan adalah Logistic Regression dengan tahapan prapemrosesan teks, ekstraksi fitur menggunakan Term Frequency–Inverse Document Frequency (TF-IDF), serta pelatihan dan pengujian model pada dataset SMS berbahasa Indonesia yang telah dilabeli ke dalam kelas spam dan non-spam. Hasil evaluasi menunjukkan bahwa model menghasilkan akurasi sebesar 96%, dengan nilai precision 0,94 dan recall 0,97 pada kelas non-spam, serta precision 0,97 dan recall 0,94 pada kelas spam. Confusion matrix yang diperoleh adalah [[111, 3], [7, 108]], yang menunjukkan bahwa sebagian besar pesan berhasil diklasifikasikan secara tepat pada kedua kelas. Berdasarkan hasil tersebut, Logistic Regression dapat diterapkan secara efektif untuk klasifikasi teks pendek pada pesan SMS dan sesuai digunakan sebagai sistem penyaringan spam yang berjalan langsung pada perangkat Android.

**Kata Kunci** — *Android, Klasifikasi Teks, Logistic Regression, SMS Spam, Pembelajaran Mesin*

## 1. PENDAHULUAN

Layanan pesan singkat (*Short Message Service* / SMS) masih digunakan secara luas dalam berbagai layanan resmi, terutama untuk keperluan verifikasi keamanan dan transaksi perbankan karena tidak bergantung pada koneksi internet. Di sisi lain, penggunaan SMS juga membuka peluang penyalahgunaan sebagai media pengiriman pesan spam, seperti pesan promosi yang tidak diinginkan maupun pesan yang mengarah pada penipuan. Kondisi tersebut menimbulkan kebutuhan akan mekanisme penyaringan pesan yang mampu membedakan pesan spam dan non-spam secara otomatis. Untuk menjawab permasalahan tersebut, berbagai pendekatan berbasis *machine learning* telah dikembangkan. Sejumlah penelitian menunjukkan bahwa algoritma *Logistic Regression* mampu mengklasifikasikan SMS spam dengan tingkat akurasi hingga 96% serta memiliki kompleksitas komputasi yang relatif rendah. Karakteristik ini menjadikan *Logistic Regression* sesuai digunakan untuk pengolahan teks pendek pada perangkat seluler dengan keterbatasan sumber daya.[1], [2], [3]

Selain *Logistic Regression*, sejumlah algoritma lain juga telah dikaji dalam penelitian klasifikasi SMS spam. Salah satu metode yang banyak digunakan adalah *Support Vector Machine* (SVM), yang mampu mencapai tingkat akurasi hingga 96,94% pada klasifikasi SMS berbahasa Indonesia [4], [5]. Meskipun demikian, SVM membutuhkan proses komputasi yang lebih berat, terutama ketika jumlah data meningkat. Pendekatan Deep Learning seperti ResNet dan IndoBERT menunjukkan performa klasifikasi yang sangat tinggi dengan akurasi hingga 99,35% untuk IndoBERT dan 99,08% untuk ResNet [6], [7]. Akan tetapi, metode tersebut memerlukan sumber daya komputasi yang besar sehingga kurang efisien untuk diterapkan secara langsung pada perangkat seluler.

Penelitian Prasetya *et al.* menunjukkan bahwa prapemrosesan teks yang meliputi pembersihan teks, tokenisasi, penghapusan kata umum, dan stemming merupakan tahapan penting dalam pengolahan teks berbahasa Indonesia yang bersifat tidak baku dan berukuran pendek [8]. Penelitian tersebut diterapkan pada ulasan pengguna aplikasi Spotify yang memiliki karakteristik linguistik berupa singkatan, variasi penulisan, dan struktur kalimat informal, yang juga umum ditemukan pada pesan SMS. Oleh karena itu, pendekatan prapemrosesan teks yang digunakan dalam penelitian tersebut relevan untuk diterapkan sebagai dasar pengolahan data pada sistem deteksi spam SMS sebelum dilakukan ekstraksi fitur dan proses klasifikasi.

Permasalahan lain yang masih ditemui pada pengembangan sistem deteksi spam adalah aspek implementasi pada platform Android. Beberapa sistem yang telah dikembangkan masih bergantung pada arsitektur berbasis server atau layanan web sehingga memerlukan pengiriman data pesan ke pihak eksternal [9]. Pendekatan tersebut menimbulkan ketergantungan terhadap infrastruktur tambahan dan membatasi penerapan sistem secara mandiri pada perangkat Android. Oleh karena itu, penelitian ini mengembangkan sistem penyaringan SMS spam berbasis Android dengan menerapkan algoritma *Logistic Regression* sebagai metode klasifikasi yang ringan dan efisien untuk dijalankan langsung pada perangkat.

## 2. METODE PENELITIAN

Penelitian ini menggunakan pendekatan eksperimental yang digabung dengan perancangan sistem aplikasi. Pendekatan eksperimental digunakan untuk membangun dan mengevaluasi model klasifikasi SMS spam, sedangkan perancangan sistem diterapkan untuk mengintegrasikan model ke dalam aplikasi berbasis Android. Tahapan penelitian meliputi pengambilan dataset terlabel dari kaggle, prapemrosesan data teks, ekstraksi fitur menggunakan TF-IDF, pembangunan model klasifikasi *Logistic Regression*, evaluasi kinerja model, serta implementasi sistem ke dalam aplikasi Android. Pada Gambar 1 berikut merupakan tahapan atau langkah-langkah dalam melakukan penelitian ;

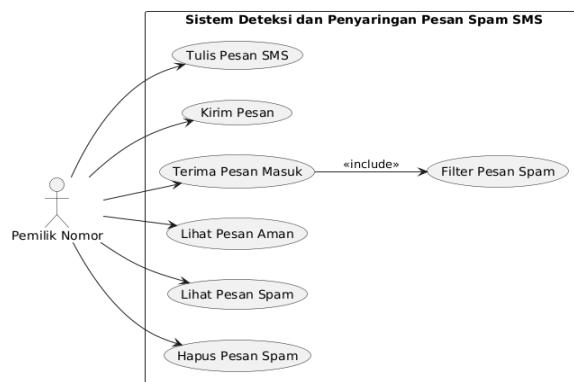


Gambar 1. Alur Penelitian

### 2.1 Perancangan Sistem

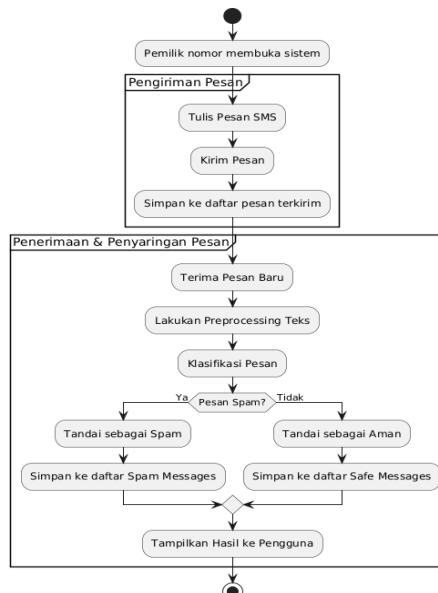
Perancangan sistem dilakukan menggunakan pendekatan Object-Oriented Analysis and Design (OOAD) untuk menggambarkan fungsi dan alur kerja aplikasi pendekripsi SMS spam berbasis Android. Pendekatan ini digunakan untuk memastikan bahwa sistem dirancang secara terstruktur, dengan pemisahan yang jelas antara peran pengguna dan proses yang dijalankan oleh sistem.

Diagram *use case* digunakan untuk memodelkan fungsi utama sistem dan hubungan antara pengguna dengan aplikasi. Diagram ini menggambarkan aktivitas pengguna dalam mengirim dan menerima pesan SMS serta mengakses hasil klasifikasi pesan ke dalam kategori spam dan non-spam. Diagram *use case* ditunjukkan pada Gambar 1 di bawah ini.



Gambar 1. Use Case Penyaringan Spam Sms

Berdasarkan fungsi-fungsi yang telah dimodelkan pada diagram *use case*, alur proses klasifikasi pesan SMS digambarkan menggunakan diagram *activity*. Diagram ini menunjukkan tahapan proses yang terjadi di dalam sistem, mulai dari penerimaan pesan SMS, proses prapemrosesan teks, klasifikasi pesan menggunakan model *Logistic Regression*, hingga penempatan pesan ke dalam kategori spam atau non-spam. Diagram *activity* ditunjukkan pada Gambar 2.



Gambar 2. Activity Diagram Penyaringan Spam SMS

## 2.2 Pengumpulan Data

Data yang digunakan dalam penelitian ini diperoleh dari dataset terbuka yang tersedia pada platform Kaggle [10]. Dataset tersebut berisi pesan SMS berbahasa Indonesia yang telah dilabeli ke dalam dua kelas, yaitu spam dan non-spam. Penelitian ini tidak melakukan proses pelabelan ulang terhadap data, melainkan memanfaatkan label yang telah tersedia sebagai dasar pelatihan dan pengujian model klasifikasi.

## 2.3 Prapemrosesan Data Teks

Prapemrosesan data dilakukan untuk membersihkan dan menormalkan teks SMS sebelum proses ekstraksi fitur. Tahapan ini meliputi case folding dengan mengubah seluruh teks menjadi huruf kecil, penghapusan angka dan karakter non-alfanumerik menggunakan regular expression, serta perbaikan spasi. Selanjutnya dilakukan normalisasi kata tidak baku ke dalam bentuk kata baku menggunakan kamus kata baku bahasa Indonesia guna mengurangi variasi penulisan kata yang umum ditemukan pada pesan SMS[11][12]. Setelah itu, dilakukan penghapusan stopwords bahasa Indonesia untuk menghilangkan kata-kata umum yang tidak memiliki kontribusi signifikan terhadap proses klasifikasi[13]. Tahap prapemrosesan diakhiri dengan proses stemming menggunakan algoritma Sastrawi untuk mengubah kata ke bentuk dasarnya[14].

## 2.4 Ekstraksi Fitur

Metode Term Frequency–Inverse Document Frequency (TF-IDF) digunakan untuk merepresentasikan pesan SMS ke dalam bentuk vektor numerik. TF-IDF berperan dalam memberikan bobot pada setiap kata berdasarkan frekuensi kemunculannya dalam dokumen dan keseluruhan korpus, sehingga teks dapat diubah menjadi representasi numerik yang dapat diproses pada tahap klasifikasi[15]. Bobot TF-IDF dihitung menggunakan persamaan berikut:

di mana:

- $TF(t, d)$  adalah frekuensi kemunculan kata  $t$  dalam dokumen  $d$ .
  - $DF(t)$  adalah jumlah dokumen yang mengandung kata  $t$ .
  - $N$  adalah jumlah total dokumen.

Bobot TF-IDF yang dihasilkan digunakan sebagai fitur masukan untuk model *Logistic Regression*.

## 2.5 Pelatihan Model

Metode klasifikasi yang digunakan dalam penelitian ini adalah *Logistic Regression*, yaitu algoritma klasifikasi statistik yang digunakan untuk memodelkan probabilitas suatu data termasuk ke dalam kelas tertentu[16]. Pada penelitian ini, *Logistic Regression* digunakan untuk mengklasifikasikan pesan SMS ke dalam dua kelas, yaitu spam dan non-spam.

Model *Logistic Regression* memanfaatkan fungsi sigmoid untuk memetakan kombinasi linear fitur ke dalam nilai probabilitas, yang dirumuskan sebagai berikut:

di mana:

- $x_1, x_2, \dots, x_n$  merupakan fitur hasil ekstraksi TF-IDF,
  - $\beta_0$  adalah bias,
  - $\beta_1, \beta_2, \dots, \beta_n$  adalah bobot model.

Nilai probabilitas yang dihasilkan digunakan untuk menentukan kelas pesan berdasarkan ambang batas tertentu. Jika nilai probabilitas  $\geq 0,5$  maka pesan diklasifikasikan sebagai spam, sedangkan jika nilai probabilitas  $< 0,5$  maka pesan diklasifikasikan sebagai non-spam.

## 2.6 Implementasi Sistem

Model klasifikasi yang telah dilatih diintegrasikan ke dalam aplikasi Android. Antarmuka aplikasi dikembangkan menggunakan Flutter, sedangkan mesin klasifikasi diimplementasikan menggunakan bahasa Python. Integrasi antara Flutter dan Python dilakukan melalui plugin Chaquopy, sehingga proses prapemrosesan dan klasifikasi pesan dapat dijalankan langsung di dalam aplikasi. Implementasi ini memungkinkan sistem bekerja secara otomatis dalam memproses pesan SMS yang diterima.

## 2.7 Evaluasi Model

Evaluasi kinerja model dilakukan menggunakan beberapa metrik, yaitu akurasi, presisi, recall, dan F1-score, yang dirumuskan sebagai berikut:

$$\text{Presisi} = \frac{TP}{TP+FP} \dots \dots \dots (4)$$

di mana:

- $TP$  adalah *true positive*
  - $TN$  adalah *true negative*
  - $FP$  adalah *false positive*
  - $FN$  adalah *false negative*

### 3. HASIL DAN PEMBAHASAN

Bab ini menyajikan hasil pengujian model klasifikasi SMS spam yang telah dibuat. Hasil pengujian diperoleh dari proses evaluasi model *Logistic Regression* menggunakan dataset SMS berbahasa Indonesia yang telah melalui tahapan prapemrosesan dan ekstraksi fitur.

### 3.1 Deskripsi Dataset dan Skenario Pengujian

Dataset yang digunakan dalam penelitian ini merupakan dataset SMS berbahasa Indonesia yang diperoleh dari Kaggle dan telah dilengkapi dengan label kategori spam dan non-spam. Dataset pada tabel 1 terdiri dari dua kelas dengan distribusi yang relatif seimbang sehingga tidak diperlukan penanganan khusus terhadap ketidakseimbangan data.

Tabel 1. Distribusi Kelas Dataset SMS

Kategori	Percentase (%)
Spam	50,22
Non-Spam	49,78

Dataset kemudian dibagi menjadi data latih dan data uji dengan rasio 80:20. Data latih digunakan untuk membangun model klasifikasi, sedangkan data uji digunakan untuk mengevaluasi kinerja model terhadap data yang belum pernah dilihat sebelumnya.

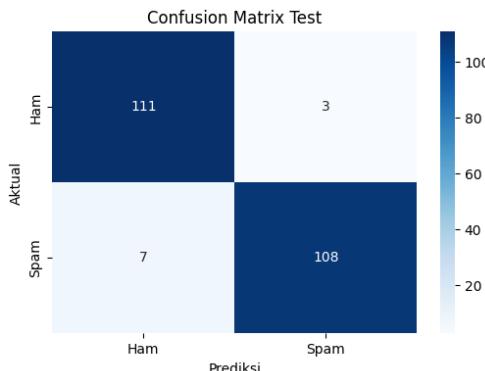
### 3.2 Hasil Pemrosesan Data

Tahap prapemrosesan data dilakukan untuk meningkatkan kualitas data teks sebelum proses ekstraksi fitur. Pesan SMS umumnya memiliki karakteristik teks yang pendek, tidak baku, serta mengandung banyak variasi penulisan, seperti singkatan dan kata informal. Oleh karena itu, proses case folding, penghapusan angka dan karakter non-alfanumerik, serta normalisasi kata tidak baku menggunakan kamus kata baku bahasa Indonesia menjadi tahapan penting untuk mengurangi variasi kata yang tidak relevan.

Penghapusan stopwords bertujuan untuk menghilangkan kata-kata umum yang tidak memiliki kontribusi signifikan dalam membedakan kelas spam dan non-spam. Selanjutnya, proses stemming menggunakan algoritma Sastrawi dilakukan untuk mengubah kata ke bentuk dasarnya sehingga kata dengan makna yang sama dapat direpresentasikan secara konsisten. Hasil prapemrosesan ini menghasilkan teks yang lebih terstruktur dan representatif, yang berdampak langsung pada efektivitas proses ekstraksi fitur TF-IDF.

### 3.3 Hasil Pelatihan dan Pengujian Model

Model *Logistic Regression* dilatih menggunakan data latih hasil ekstraksi fitur TF-IDF, kemudian diuji menggunakan data uji untuk mengevaluasi kemampuan klasifikasi pesan SMS ke dalam kategori spam dan non-spam. Hasil pengujian model disajikan dalam bentuk *confusion matrix* pada gambar 3 yang menggambarkan perbandingan antara label aktual dan label prediksi.



Gambar 3. Confusion Matrix

Berdasarkan confusion matrix pada Gambar 2, diperoleh sebanyak 111 pesan non-spam (ham) yang berhasil diklasifikasikan dengan benar dan 108 pesan spam yang berhasil terdeteksi dengan tepat. Selain itu, terdapat 3 pesan non-spam yang salah diklasifikasikan sebagai spam (*false positive*) dan 7 pesan spam yang salah diklasifikasikan sebagai non-spam (*false negative*).

Hasil ini menunjukkan bahwa model memiliki kemampuan klasifikasi yang baik dalam membedakan pesan spam dan non-spam, dengan jumlah kesalahan klasifikasi yang relatif kecil dibandingkan total data uji.

### 3.4 Evaluasi Model

Evaluasi kinerja model dilakukan untuk menilai kemampuan *Logistic Regression* dalam mengklasifikasikan pesan SMS ke dalam kategori spam dan non-spam. Berdasarkan tabel 2 pengujian dilakukan menggunakan data uji sebanyak 229 pesan, dan kinerja model diukur menggunakan metrik akurasi, *presisi*, *recall*, dan *F1-score*.

Berdasarkan hasil evaluasi, model mencapai akurasi sebesar 96%, yang menunjukkan bahwa sebagian besar pesan SMS pada data uji berhasil diklasifikasikan dengan benar. Nilai ini mencerminkan kinerja model yang stabil pada dataset dengan distribusi kelas yang relatif seimbang.

Untuk kelas non-spam (ham), model menghasilkan nilai presisi sebesar 0,94 dan recall sebesar 0,97, dengan F1-score sebesar 0,96. Nilai recall yang tinggi pada kelas ini menunjukkan bahwa model sangat baik dalam mengenali pesan non-spam, sehingga risiko pesan normal salah terdeteksi sebagai spam dapat diminimalkan. Sementara itu, untuk kelas spam, model memperoleh presisi sebesar 0,97 dan recall sebesar 0,94, dengan F1-score sebesar 0,96. Nilai presisi yang tinggi menunjukkan bahwa sebagian besar pesan yang diprediksi sebagai spam benar-benar merupakan spam, sedangkan nilai recall yang sedikit lebih rendah menunjukkan masih adanya sejumlah kecil pesan spam yang lolos sebagai non-spam.

Secara keseluruhan, nilai macro average dan weighted average untuk presisi, recall, dan F1-score masing-masing sebesar 0,96, yang menandakan bahwa model memiliki performa yang seimbang pada kedua kelas. Hasil ini menunjukkan bahwa *Logistic Regression* mampu memberikan kinerja klasifikasi yang baik dan konsisten pada data SMS berbahasa Indonesia.

Tabel 2. Hasil Evaluasi Kinerja Model

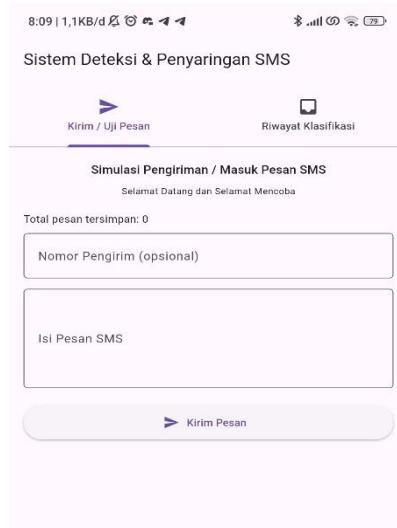
Kelas	Presisi	Recall	F1-Score	Support
Ham (Non-Spam)	0,94	0,97	0,96	114
Spam	0,97	0,94	0,96	115
Akurasi			0,96	229
Macro Average	0,96	0,96	0,96	229
Weighted Average	0,96	0,96	0,96	229

### 3.5 Implementasi Aplikasi Android

Implementasi sistem deteksi dan penyaringan SMS spam direalisasikan dalam bentuk aplikasi berbasis Android. Aplikasi ini dirancang untuk mensimulasikan proses penerimaan pesan SMS, melakukan klasifikasi secara otomatis, serta mengelola hasil klasifikasi ke dalam kategori spam dan non-spam. Antarmuka aplikasi dikembangkan menggunakan Flutter, sedangkan proses klasifikasi teks dijalankan menggunakan bahasa Python yang diintegrasikan melalui plugin Chaquopy.

#### 3.5.1. Halaman Utama

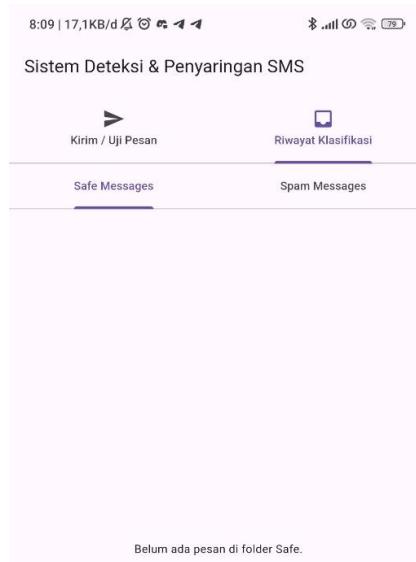
Halaman utama aplikasi berfungsi sebagai antarmuka awal bagi pengguna untuk melakukan pengujian pesan SMS. Pada halaman ini, pengguna dapat memasukkan nomor pengirim (opsional) dan isi pesan SMS yang akan diklasifikasikan. Setelah pesan dikirim, sistem secara otomatis menjalankan proses prapemrosesan teks dan klasifikasi menggunakan model *Logistic Regression* yang telah dilatih sebelumnya. Tampilan halaman utama aplikasi ditunjukkan pada Gambar 4



Gambar 4. Halaman Utama

### 3.5.2. Halaman Riwayat dan Klasifikasi

Aplikasi menyediakan halaman riwayat klasifikasi yang berfungsi untuk menampilkan daftar pesan SMS yang telah diproses oleh sistem. Pada kondisi awal penggunaan, halaman riwayat belum menampilkan data karena belum terdapat pesan yang diklasifikasikan. Tampilan kondisi awal halaman riwayat klasifikasi yang masih kosong ditunjukkan pada Gambar 5.

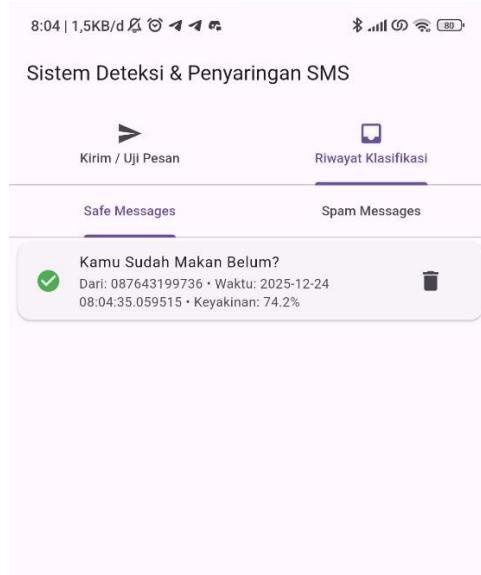


Gambar 5. Halaman Riwayat & Klasifikasi

Setelah pengguna melakukan proses klasifikasi pesan, hasil klasifikasi akan otomatis ditambahkan ke dalam halaman riwayat sesuai dengan kategori pesan. Melalui halaman ini, pengguna dapat memantau kembali hasil klasifikasi pesan spam dan non-spam yang telah diproses sebelumnya.

### 3.5.3. Hasil Klasifikasi Pesan Non-Spam

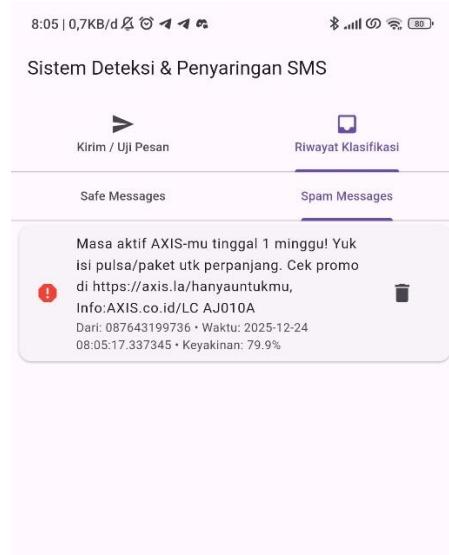
Pesan yang diklasifikasikan sebagai non-spam secara otomatis disimpan ke dalam folder Safe Messages pada menu riwayat klasifikasi. Setiap pesan ditampilkan bersama informasi pendukung, seperti nomor pengirim, waktu penerimaan, serta nilai keyakinan hasil klasifikasi. Penyimpanan pesan non-spam ini bertujuan untuk memastikan bahwa pesan normal tetap dapat diakses oleh pengguna tanpa terganggu oleh pesan spam. Contoh hasil klasifikasi pesan non-spam ditunjukkan pada Gambar 6.



Gambar 6. Hasil Klasifikasi Non-Spam

### 3.5.4. Hasil Klasifikasi Pesan Spam

Pesan yang terdeteksi sebagai spam akan dipisahkan dan disimpan ke dalam folder *Spam Messages*. Pemisahan ini memungkinkan sistem melakukan fungsi penyaringan secara efektif, sehingga pesan yang berpotensi mengganggu atau berbahaya tidak tercampur dengan pesan normal. Setiap pesan spam ditandai dengan ikon peringatan serta nilai keyakinan model yang menunjukkan tingkat probabilitas pesan tersebut tergolong spam. Tampilan folder *Spam Messages* ditunjukkan pada Gambar di bawah ini



Gambar 7. Hasil Klasifikasi Spam

## 4. SIMPULAN

Berdasarkan hasil penelitian dan pembahasan yang telah dilakukan, dapat disimpulkan sebagai berikut:

1. Penelitian ini berhasil membangun sistem deteksi dan penyaringan SMS spam berbasis Android dengan memanfaatkan algoritma *Logistic Regression* yang terintegrasi ke dalam aplikasi menggunakan antarmuka *Flutter* dan mesin klasifikasi berbasis Python melalui plugin *Chaquopy*. Sistem yang dikembangkan mampu melakukan klasifikasi pesan SMS secara otomatis pada aplikasi Android.
2. Model klasifikasi yang digunakan menunjukkan kinerja yang baik dengan tingkat akurasi sebesar 96% pada data uji. Nilai presisi, recall, dan F1-score yang relatif seimbang pada kedua kelas menunjukkan bahwa sistem mampu membedakan pesan spam dan non-spam secara konsisten tanpa kecenderungan bias kelas yang signifikan.

3. Prapemrosesan teks dengan normalisasi kata tidak baku dan stemming bahasa Indonesia memungkinkan sistem menangani pesan SMS yang pendek dan bersifat informal, sehingga proses klasifikasi dapat diterapkan pada berbagai variasi penulisan pesan..
4. Implementasi algoritma Logistic Regression pada aplikasi Android menunjukkan keunggulan dalam efisiensi komputasi. Sistem dapat dijalankan secara responsif tanpa membebani sumber daya perangkat, sehingga mendukung proses klasifikasi pesan SMS secara otomatis di dalam aplikasi..
5. Sistem yang dikembangkan masih memiliki keterbatasan dalam mendeteksi pola spam baru yang belum terwakili dalam data pelatihan. Keterbatasan ini menyebabkan sebagian kecil pesan spam berpotensi terklasifikasi sebagai non-spam.

## 5. SARAN

Berdasarkan keterbatasan yang ditemukan dalam penelitian ini, penelitian selanjutnya disarankan untuk memperluas dan memperbarui dataset pelatihan agar dapat mencakup variasi pola spam yang lebih beragam. Selain itu, metode ekstraksi fitur dapat dikembangkan lebih lanjut, misalnya dengan memanfaatkan n-gram atau pendekatan lain yang mampu menangkap konteks kata dalam pesan SMS. Penelitian lanjutan juga dapat mengeksplorasi penggunaan atau penggabungan algoritma klasifikasi selain Logistic Regression dengan tetap mempertimbangkan efisiensi komputasi, sehingga sistem tetap sesuai untuk diterapkan pada perangkat Android.

## DAFTAR PUSTAKA

- [1] F. Reviantika, Y. Azhar, and G. I. Marthasari, “Analisis Klasifikasi SMS Spam Menggunakan Logistic Regression,” APIC, 2021. [Online]. Available: <https://www.cnbcindonesia.com>
- [2] A. Nur *et al.*, “Prosiding SEMNAS INOTEK (Seminar Nasional Inovasi Teknologi) 80 Implementasi Algoritma Regresi Logistik untuk Binary Classification dalam Spam SMS dan WhatsApp Penulis Korespondensi,” Online, 2023
- [3] S. Mutmainnah, T. Ansyor Lorosae, and S. Ramadhan, “Model Text Embedding dan TF-IDF+Ngram untuk Meningkatkan Kinerja Algoritma Binary Classifier pada Klasifikasi SMS Palsu,” vol. 4, no. 1, pp. 55–64, 2025, [Online]. Available: <https://ojs.trigunadharma.ac.id/index.php/isi>
- [4] M. A. Sofyan, N. Rahaningsih, and R. D. Dana, “DETEKSI SMS SPAM BERBAHASA INDONESIA MENGGUNAKAN ALGORITMA SUPPORT VECTOR MACHINE,” 2024. [Online]. Available: [http://bit.ly/yw\\_sms\\_spam\\_indonesia](http://bit.ly/yw_sms_spam_indonesia)
- [5] M. Dauber Panjaitan, P. P. Adikara, and B. D. Setiawan, “Klasifikasi Spam pada Short Message Service (SMS) menggunakan Support Vector Machine,” 2017. [Online]. Available: <http://j-ptiik.ub.ac.id>
- [6] G. Airlangga, “A Comparative Analysis of Deep Learning Models for SMS Spam Detection: CNN-LSTM, CNN-GRU, and ResNet Approaches,” *Journal of Computer Networks, Architecture and High Performance Computing*, vol. 6, no. 4, pp. 1952–1960, Oct. 2024, doi: 10.47709/cnahpc.v6i4.4827.
- [7] D. Ramdhan, H. Lucky, A. P. Kemala, and A. Chowanda, “SHORT MESSAGE SERVICE (SMS) SPAM FILTERING USING DEEP LEARNING IN BAHASA INDONESIA,” *ICIC Express Letters, Part B: Applications*, vol. 13, no. 10, pp. 1093–1100, Oct. 2022, doi: 10.24507/icicelb.13.10.1093.
- [8] “Implementasi NLP(Natural Language Processing) Dasar pada Analisis Sentiment Review Spotify,” *Seminar Nasional Teknologi & Sains*, vol. 3, no. 1, pp. 145–153, doi: 10.29407/stains.v3i1.4166.
- [9] J. Handoko, Z. Salsabila, and J. SIFO Mikroskil, “Sistem Deteksi Nomor Telepon dan Rekening Bank Terindikasi Penipuan Berbasis Aplikasi Android dan Web.,” *JSM*, vol. 23, no. 2, 2022. DOI: <https://doi.org/10.55601/jsm.v23i2.913>
- [10] Kaggle, “Indonesian SMS Spam.” Accessed: Dec. 24, 2025. [Online]. Available: [https://www.kaggle.com/datasets/gevabriel/indonesian-sms-spam/data?select=sms\\_spam\\_indo.csv](https://www.kaggle.com/datasets/gevabriel/indonesian-sms-spam/data?select=sms_spam_indo.csv)
- [11] Kaggle, “kamus\_slag.” Accessed: Dec. 24, 2025. [Online]. Available: <https://www.kaggle.com/datasets/forningulo/kamus-slag>
- [12] S. Shevira, I. Made, A. D. Suarjaya, and P. Wira Buana, “Pengaruh Kombinasi dan Urutan Pre-Processing pada Tweets Bahasa Indonesia,” *JITTER*, vol. 3, no. 2, Aug. 2022. DOI: <https://doi.org/10.24843/JTRTI.2022.v03.i02.p06>.
- [13] A. W. Pradana and M. Hayaty, “The Effect of Stemming and Removal of Stopwords on the Accuracy of Sentiment Analysis on Indonesian-language Texts,” *Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control*, pp. 375–380, Oct. 2019, doi: 10.22219/kinetik.v4i4.912.

- [14] M. U. Albab, Y. K. P., and M. N. Fawaiq, "Optimization of the Stemming Technique on Text Preprocessing President 3 Periods Topic," *Jurnal Transformatika*, vol. 20, no. 2, pp. 1–12, Jan. 2023, doi: 10.26623/transformatika.v20i2.5374.
- [15] A. H. I. B. A. Azrir, P. Naveen, and S. C. Haw, "Sentiment Analysis using Machine Learning Models on Shopee Reviews," *Journal of System and Management Sciences*, vol. 14, no. 2, pp. 214–228, 2024, doi: 10.33168/JSMS.2024.0213.
- [16] A. Firizkiansah, A. Muhammad, and I. R. Maulana, "Optimasi Klasifikasi Data Teks Menggunakan Algoritma Logistic Regression dengan TF-IDF dan SMOTE," *JIKOMTI*, vol. 2, no. 1, May 2025..