Jurnal Ilmiah Informatika (JIF)
| ISSN (Print) 2337-8379 | ISSN (Online) 2615-1049 |
Jurnal online di akses di http://ejournal.upbatam.ac.id/index.php/jif

Implementasi Algoritma *Naïve Bayes* dalam Sistem Pengarsipan Surat Berbasis AI di GPI Papua Klasis Mimika Papua Tengah

Jennifer Florenzia Indey[1], Supangat[2]

[21]*Universitas 17 Agustus 1945 Surabaya, l. Semolowaru No.45, Menur Pumpungan, Kec. Sukolilo, Surabaya, Jawa Timur, 60118*

INFORMASI ARTIKEL

Sejarah Artikel:
Diterima Redaksi: 13-07-2024
Revisi Akhir: 24-08-2024
Diterbitkan *Online*:05-09-2024

KATA KUNCI

Pengarsipan Surat

Naïve Bayes

Text Mining

Preprocessing

Klasifikasi

KORESPONDENSI

E-mail: Jenniferindeyflore21@gmail.com

A B S T R A C T

This research develops an artificial intelligence-based letter archiving management information system for the GPI Papua Classis Mimika Institute in Central Papua, with a focus on classifying incoming letters using the Naive Bayes algorithm. The aim of this research is to make it easier to search and monitor documents and increase the efficiency and accuracy of mail archive management. The dataset consists of 50 data, of which 20 are spam mail data and 30 are non-spam mail data. Test results using the Confusion Matrix method show that the system has 78% accuracy, 88% precision for the spam category, 73.3% recall for the spam category, and an F1 Score of 79.9%. It is hoped that the implementation of AI technology will help the GPI Papua Klasis Mimika Institute in Central Papua in optimizing the incoming letter classification process, minimizing errors and increasing the efficiency of document management. By using an AI-based approach, the system can also provide practical solutions for more effective and efficient records management, enabling fast and precise access to the required information. Overall, the application of AI technology in mail archiving is expected to not only increase productivity and accuracy, but also provide a strong foundation for improving services and operational efficiency in various sectors and institutions.

1. PENDAHULUAN

Komputerisasi mendorong perkembangan teknologi informasi yang terus meningkat [1]. Arsip memainkan peran penting dalam administrasi dan manajemen sebuah organisasi. Sebagai sumber informasi vital, arsip mencatat semua aktivitas organisasi, termasuk proposal, surat-menyurat, dan dokumen lainnya. Informasi yang tercatat dalam arsip tidak hanya berfungsi sebagai bukti, tetapi juga menjadi memori yang berharga bagi organisasi tersebut [2].

Lembaga GPI Papua Klasis Mimika Papua Tengah adalah sebuah lembaga yang berlokasi di Mimika Papua Tengah, Indonesia. Untuk mencatat surat masuk dan keluar, GPI menggunakan buku agenda arsip, sementara pengarsipan dilakukan dengan cara mengelompokkan dokumen sejenis secara manual. Metode penyimpanan ini berpotensi menyebabkan kehilangan atau kerusakan data, yang dapat menyulitkan proses pencarian di masa mendatang. Pembuatan laporan surat masuk dan keluar juga masih dilakukan secara manual, yang tidak efisien karena memerlukan pengurutan secara manual [3].

Untuk mengatasi masalah tersebut, diperlukan pengembangan sistem informasi berbasis *web* yang akan meningkatkan efisiensi, akurasi, dan keamanan dalam pengarsipan surat elektronik. Sistem arsip elektronik (*e-arsip*) berbasis kecerdasan buatan (AI) akan merekam informasi pada setiap lembaran arsip secara elektronik, memungkinkan akses yang lebih mudah, mempercepat penyampaian informasi, dan menjaga keamanan data penting [4].

Algoritma *Naive Bayes* menggunakan teori probabilitas untuk mengklasifikasikan surat berdasarkan frekuensi kata-kata dalam teks. Klasifikasi adalah proses menentukan atribut dengan menggunakan metode yang memungkinkan algoritma untuk menilai dan memprediksi kelompok atau kategori data [5]. Dengan algoritma ini, dapat dikembangkan sistem pengarsipan surat berbasis kecerdasan buatan yang efektif untuk mengelola surat dengan otomatisasi berdasarkan kategori. Meskipun pentingnya teknologi informasi dalam administrasi telah banyak dibahas, penerapan dan adaptasi teknologi ini masih menjadi tantangan di lembaga seperti GPI Papua Klasis Mimika. Studi sebelumnya cenderung fokus pada keuntungan umum dari

digitalisasi arsip tanpa mengeksplorasi masalah khusus yang dihadapi lembaga keagamaan dan komunitas lokal.

Penelitian ini bertujuan untuk mengatasi kekurangan tersebut dengan mempelajari masalah khusus yang dihadapi oleh GPI Papua Klasis Mimika Papua Tengah dalam penggunaan pengarsipan surat elektronik berbasis kecerdasan buatan (AI). Selain itu, penelitian ini juga mengusulkan strategi untuk membantu transisi dari sistem manual ke sistem pengarsipan elektronik, dengan mempertimbangkan keterbatasan dan konteks khusus lembaga ini.

Oleh karena itu, penelitian ini akan memberikan wawasan tentang penggunaan teknologi informasi dalam pengarsipan di lembaga kecil dan terpencil, dengan menyoroti sistem informasi pengarsipan surat berbasis kecerdasan buatan yang menggunakan algoritma *Naive Bayes*. Sistem ini diharapkan dapat mempermudah pencarian, pengelolaan surat, serta memfasilitasi pengawasan dan pengambilan keputusan.

2. TINJAUAN PUSTAKA

2.1 Penulisan Referensi Lembaga Gereja Protestan Indonesia (GPI)

Gereja Klasis GPI di Mimika, Papua Tengah, mengelola dan membimbing rohani jemaat serta mengatur kegiatan gereja di daerah tersebut. Mereka aktif dalam mendukung masyarakat melalui program kesehatan, pendidikan, dan ekonomi, serta dalam pengembangan fisik dan non-fisik gereja. Struktur organisasinya mencakup Majelis Klasis, pendeta, pengurus gereja, dan komisi seperti pendidikan, diakonia, pemuda, wanita, dan lanjut usia. Klasis ini tidak hanya berfungsi sebagai tempat ibadah, tetapi juga sebagai pusat kegiatan sosial yang meningkatkan kualitas hidup masyarakat di sekitarnya. Meskipun menghadapi tantangan geografis dan keterbatasan sumber daya, Klasis GPI Mimika terdiri dari delapan gereja yang berupaya memperluas pelayanannya dengan komitmen yang kuat.

2.2 Kabupaten Mimika

Kabupaten Mimika, dengan ibu kota Timika, terletak antara 134°31'-138°31' BT dan 4°60'-5°18' LS. Kabupaten Mimika di Provinsi Papua adalah tempat di mana PT Freeport Indonesia mengelola tambang emas terbesar di Indonesia [6]. Kabupaten ini memiliki luas wilayah 19.592 km², atau sekitar 4,75% dari total luas Provinsi Papua, dan terdiri dari 12 distrik seperti Mimika Barat, Mimika Timur, dan Kuala Kencana. Wilayah ini terkenal sebagai salah satu kawasan pertambangan utama di Indonesia. Kota Timika berfungsi sebagai pusat ekonomi utama, terutama karena keberadaan PT. Freeport Indonesia, yang menarik banyak pendatang untuk tinggal dan bekerja di sana.

2.3 Sistem Informasi Manajemen

Sistem Informasi Manajemen (SIM) adalah kerangka kerja yang menggunakan teknologi informasi untuk mengumpulkan, menyimpan, memproses, dan menampilkan informasi yang relevan bagi manajemen perusahaan [7]. Tujuannya adalah membantu dalam pengambilan keputusan, pengelolaan sumber daya, dan meningkatkan efisiensi operasional. SIM terdiri dari Sistem Informasi Keputusan (DSS), Sistem Informasi Eksekutif (ESS), Sistem Informasi Operasional (OSS), Sistem Informasi Pemasaran (MIS), dan Sistem Informasi Keuangan (FIS). Komponen-komponen utama SIM meliputi pengumpulan data dari berbagai sumber, penyimpanan dalam basis data, pengolahan data menjadi informasi, serta penyediaan informasi kepada pengguna. Selain itu, Sistem informasi manajemen berbasis komputer membuat pekerjaan manusia jauh lebih mudah dibandingkan dengan sistem informasi yang masih menggunakan metode manual [8].

2.4 E-arsip

E-arsip, singkatan dari "arsip elektronik", merujuk pada manajemen dan penyimpanan dokumen dan informasi secara elektronik dengan menggunakan teknologi informasi dan komunikasi. Ketika arsip elektronik digunakan, arsip dapat ditemukan dengan cepat dan akurat ketika dibutuhkan [9]. Sistem e-arsip menyimpan dokumen dan informasi dalam format digital dan dapat diakses melalui komputer atau jaringan komputer. Tujuannya adalah untuk meningkatkan kualitas pengelolaan arsip tradisional yang berbasis kertas.

2.5 Text Mining

Text mining adalah proses penggalian informasi dari teks, biasanya dari dokumen, dengan tujuan untuk menemukan kata-kata yang dapat mewakili isi dokumen dan menganalisis hubungan antara kata-kata tersebut. Dokumen yang dianalisis dalam *text mining* biasanya diformat untuk mempermudah ekstraksi informasi yang bermanfaat dari sumber data [10].

2.6 Naïve Bayes

Algoritma *Naive Bayes Classification (NBC)* adalah salah satu algoritma klasifikasi yang berbasis teorema statistika, dikembangkan dari *teorema Bayes* oleh ilmuwan Inggris bernama Thomas Bayes. *Naive Bayes* adalah algoritma yang menghitung proses pelatihan dan melakukan prediksi berdasarkan data uji [11]. NBC menggunakan *probabilitas* untuk memprediksi kelas dari data dengan asumsi independensi yang kuat antar fitur atau variabel yang diklasifikasikan. Meskipun sederhana dan efisien, asumsi independensi yang sangat kuat dalam *Naive Bayes* sering tidak sesuai dengan realitas di dunia nyata. Meskipun demikian, algoritma ini tetap populer dan banyak digunakan dalam berbagai aplikasi klasifikasi seperti klasifikasi teks, deteksi spam, dan analisis sentimen [12]. Jadi, dalam penelitian ini, metode *Naive Bayes Classification (NBC)* digunakan untuk klasifikasi data yang tidak terstruktur, seperti teks atau pemrosesan teks. NBC memiliki keunggulan karena membutuhkan jumlah data pelatihan yang relatif kecil untuk menentukan parameter yang diperlukan dalam proses klasifikasi. Secara umum, *Naive Bayes* sering menunjukkan kinerja yang baik dalam berbagai situasi dunia nyata yang kompleks.

Teorema Bayes memiliki bentuk umum yang ditunjukkan pada persamaan :

$$P(cj|X) = \frac{p(X|cj).p(cj)}{p(X)}$$

Di mana :

X : Data dengan class yang belum diketahui

C	:	Hipotesis data merupakan suatu class spesifik
P(C|X)	:	Probabilitas hipotesis C berdasar kondisi X (posteriori probabilitas)
P(C)	:	Probabilitas hipotesis C (prior probabilitas)
P(C)	:	Probabilitas hipotesis C (prior probabilitas)
P(X)	:	Probabilitas X

Dalam klasifikasi dokumen teks, seperti yang ditunjukkan dalam Pendekatan *Bayesian*, kita memilih kelas dengan probabilitas tertinggi (CMAP). Di sini, cj merupakan kelas teks rahasia, p(cj) adalah probabilitas prior dari kelas teks cj, dan d adalah dokumen teks yang terdiri dari kumpulan kata W1, W2, ..., Wn, di mana W1 adalah kata pertama, W2 adalah kata kedua, dan seterusnya. Dalam klasifikasi dokumen teks, pendekatan Bayesian memilih kelas berdasarkan probabilitas tertinggi (CMAP) sebagaimana terdapat dalam Persamaan 2 berikut ini :

$$C_{MAP} = \text{argmax} \frac{p(cj).p(X|cj)}{p(X)}$$

Nilai P(X) dapat diabaikan karena nilainya konstan untuk semua cj, sehingga dapat ditulis sebagai Persamaan 3 :

$$C_{MAX} = \text{argmax}\ p(cj).p(X|cj)$$

Perhitungan distribusi p(X|cj) menjadi sulit karena jumlah kombinasi kata yang mungkin sangat besar, mengingat setiap kata dalam setiap kategori dianggap independen satu sama lain dalam metode *Naive Bayesian*. Perhitungan ini dapat disederhanakan lebih lanjut dan diwujudkan dalam Persamaan 4 berikut menggunakan metode *Naive Bayesian*, yang mengasumsikan bahwa setiap kata dalam setiap kategori adalah independen satu sama lain.

$$P(X|cj) = \Pi^{n}_{i=1}\, p(wj|cj)$$

Dalam perhitungan *Naive Bayes*, evaluasi atau pengujian dilakukan menggunakan metode akurasi untuk menilai seberapa tepat hasil klasifikasi, mengukur jumlah prediksi yang benar dari proses klasifikasi.

$$\text{Akurasi} = \frac{Jumlah\ dokumen\ terklasifikasi\ dengan\ benar}{Jumlah\ dokumen\ uji}\ X\ 100$$

2.7 Preprocessing

Sebelum proses klasifikasi, *preprocessing* dilakukan untuk menghilangkan, mengubah, atau mengganti karakter non-abjad dan kata-kata yang tidak relevan dari sumber informasi. Tujuannya adalah memastikan bahwa data yang digunakan dalam proses klasifikasi dapat dimanfaatkan secara optimal [13]. Langkah-langkah pra-pemrosesan ini mungkin bervariasi tergantung pada kasusnya. Proses *preprocessing* umumnya terdiri dari empat langkah: *case folding*, *tokenisasi*, *filtrasi*, dan *stemming*.

1. *Case folding*
   Tidak semua dokumen teks menggunakan huruf besar-kecil yang konsisten. Oleh karena itu, penggunaan *case folding* sangat penting untuk mengubah seluruh teks dalam dokumen ke format standar, yang biasanya berupa huruf kecil atau *lowercase* [14].
2. *Tokenisasi*
   Tokenisasi adalah langkah di mana *string* input dipotong berdasarkan setiap kata penyusunnya. Proses ini biasanya membagi teks menjadi kelompok karakter yang membentuk unit kata, dengan memeriksa apakah karakter tertentu dapat berfungsi sebagai pemisah kata atau tidak. Misalnya, spasi, *tab*, dan sisipan dianggap sebagai pemisah kata. Selain itu, tanda kutip tunggal ('), titik (.), titik koma (;), titik dua (:), dan karakter lainnya juga dapat berperan sebagai pemisah kata.
3. *Filtrasi*
   Tahap *filtering* merupakan proses untuk mengekstraksi kata-kata penting dari hasil *tokenisasi*. Langkah ini melibatkan penggunaan algoritma *stoplist* untuk menghilangkan kata-kata yang kurang penting atau *wordlist* untuk menyimpan kata-kata yang relevan. Oleh karena itu, tahap ini juga dikenal sebagai proses penghapusan *stopword*. *Stoplist* atau *stopword* adalah kata-kata yang tidak memberikan informasi penting dan bisa diabaikan dalam pendekatan *bag-of-words*. Kata-kata seperti 'dan', 'atau', 'di', dan 'the' adalah contoh kata umum yang sering muncul dalam hampir semua dokumen. Menghapus *stopword* dapat mengurangi ukuran indeks dan waktu pemrosesan serta menurunkan tingkat kebisingan.
4. *Stemming*
   Proses dalam sistem IR (*Information Retrieval*) yang mengubah kata-kata dalam dokumen menjadi bentuk dasar (*root word*) dengan menggunakan aturan tertentu disebut stemming. Proses ini memetakan berbagai bentuk kata ke bentuk kata utamanya atau batang kata. Stemming banyak digunakan untuk meningkatkan kualitas informasi yang diperoleh selama pencarian informasi dengan mengidentifikasi hubungan antara variasi kata yang serupa. Misalnya, kata-kata seperti "dirampok", "dirampokkan", dan "perampok" dapat diubah menjadi "rampok" yang memiliki makna yang sama. Selain itu, penggunaan kata dasar membantu mengurangi ukuran file indeks. Penting untuk dicatat bahwa proses stemming dalam teks bahasa Indonesia berbeda dengan teks bahasa Inggris, karena dalam bahasa Indonesia juga melibatkan penghapusan prefiks dan sufiks selain akhiran .

2.8 Klasifikasi

Klasifikasi adalah proses mencari pola atau fungsi yang dapat menjelaskan serta memisahkan kategori atau kelas dari data. Tujuan utama dari klasifikasi adalah menggunakan model yang telah dikembangkan untuk memprediksi kelas objek yang belum memiliki label. Metode klasifikasi digunakan untuk mengidentifikasi atau membedakan kelas-kelas yang belum diketahui dalam suatu objek [15]. Model klasifikasi dapat berupa pohon keputusan, aturan "jika-maka", rumus matematika seperti *Naive Bayes*, atau mesin vektor pendukung. Proses klasifikasi umumnya terdiri dari dua tahap utama: pembelajaran dan pengujian. Pada tahap pembelajaran, model prediktif dibuat dengan menggunakan data yang telah diketahui kelasnya. Klasifikasi juga dikenal sebagai metode prediksi karena menggunakan data yang ada untuk membuat prediksi terhadap data yang baru, serta menguji keakuratan model prediktif terhadap data yang belum terlihat sebelumnya [16].

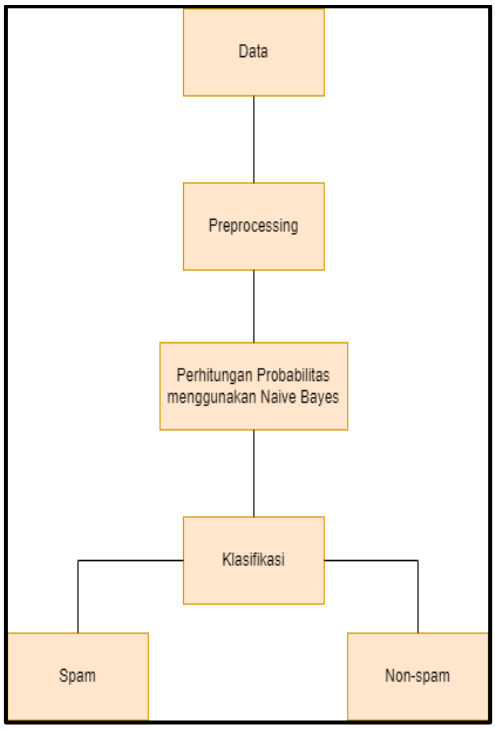
3. METODOLOGI

3.1. Objek Penelitian

Objek penelitian ini adalah data surat masuk dan surat keluar di lembaga GPI Papua Klasis Mimika Papua Tengah. Data dikumpulkan dari lembaga tersebut dan kemudian akan diklasifikasikan menggunakan algoritma *Naive Bayes* untuk menentukan apakah data tersebut termasuk dalam kategori surat *spam* atau *non-spam*. Proses ini didasarkan pada kriteria deskripsi yang disederhanakan melalui *preprocessing* teks yang mencakup beberapa tahap, seperti *stemming* dan *tokenisasi* menggunakan *Naive Bayes*.

3.2. Metodologi Pengolahan Data

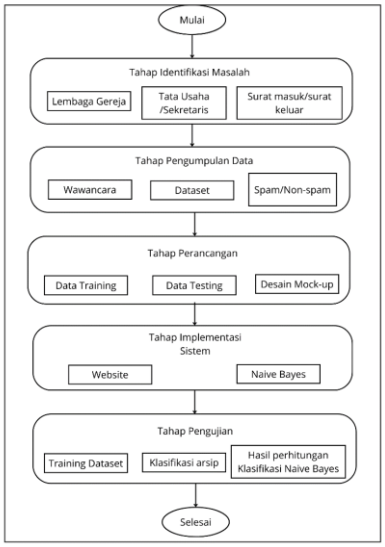
Tampilan diagram untuk metode pengolahan data dalam proses klasifikasi menggunakan *Naïve Bayes* terlihat pada gambar di bawah ini :



Gambar 1. Pengolahan Data

3.3. Tahapan Pelaksanaan Penelitian

Tahapan penelitian yaitu suatu gambaran yang menjelaskan mengenai alur logika berjalannya penelitian secara garis besar. Berikut gambar dan keterangan tahap-tahap yang dilakukan dalam kerangka berpikir penelitian, antara lain.



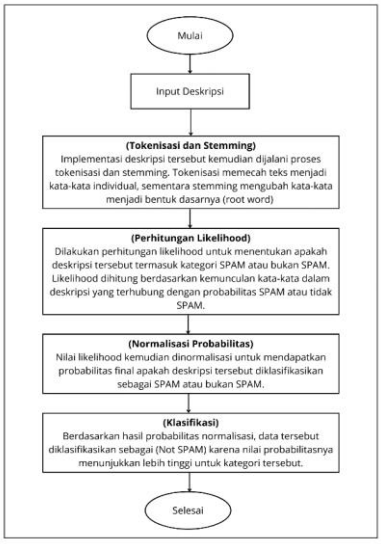
Gambar 2. Kerangka Pemikiran

Tahap-tahap kerangka pemikiran memiliki penjelasan sebagai berikut:

- Tahap Identifikasi Masalah
  Pada tahap ini, masalah diidentifikasi dengan tujuan memahami berbagai masalah yang terjadi dan mencari solusi yang tepat. Proses penelitian ini merupakan bagian integral dalam pengembangan sistem informasi untuk manajemen pengarsipan surat berbasis kecerdasan buatan (AI) di lembaga GPI Papua Klasis Mimika Papua Tengah.
- Tahap Pengumpulan Data
  Data dikumpulkan untuk mendukung pengembangan sistem. Pengumpulan data dilakukan melalui wawancara langsung dengan tata usaha dan sekretaris di lembaga tersebut, untuk memperoleh data yang akurat mengenai apakah surat termasuk spam atau non-spam, serta pemahaman mendalam terhadap proses manajemen yang sedang berjalan.
- Tahap Perancangan
  Pada tahap ini, sistem yang akan digunakan untuk memecahkan masalah atau memenuhi kebutuhan yang telah diidentifikasi dibahas. Ini melibatkan pemilihan metode, algoritma, dan teknologi yang tepat. Berdasarkan masalah yang ada, dibuatlah sistem informasi untuk manajemen pengarsipan surat berbasis kecerdasan buatan (AI) di lembaga GPI Papua Klasis Mimika Papua Tengah.
- Tahap Implementasi
  Berdasarkan data yang telah terkumpul, sebuah aplikasi berbasis *web* dikembangkan menggunakan metode *Naïve Bayes* dan bahasa pemrograman *Laravel*.
- Tahap Pengujian
  Pada tahap akhir, dilakukan pengujian *black box* terhadap aplikasi berbasis web menggunakan metode *Naïve Bayes*. Tujuan pengujian ini adalah untuk memastikan apakah sistem berfungsi sesuai dengan yang diharapkan dan berjalan dengan benar sesuai dengan tujuan penelitian ini.

3.4. Rancangan Algoritma

Rancangan algoritma adalah langkah-langkah yang menyediakan struktur dan proses yang diperlukan untuk mengimplementasikan suatu algoritma. Berikut adalah gambar rancangan algoritma *Naive Bayes* yang digunakan dalam penelitian ini.



Gambar 3. *Flowchart* Rancangan Algoritma

Flowchart ini menggambarkan langkah-langkah dalam algoritma untuk mendeteksi apakah deskripsi teks termasuk kategori *spam* atau *non-spam*. Proses dimulai dengan menginput deskripsi teks, yang kemudian dipecah menjadi kata-kata individual melalui tokenisasi dan diubah ke bentuk dasar melalui stemming. Selanjutnya, dihitung likelihood untuk menentukan kemungkinan deskripsi tersebut sebagai spam berdasarkan kemunculan kata-kata tertentu. Nilai *likelihood* ini dinormalisasi untuk mendapatkan probabilitas akhir. Berdasarkan probabilitas tersebut, deskripsi diklasifikasikan sebagai *spam* atau *non-spam*. Setelah klasifikasi, proses selesai.

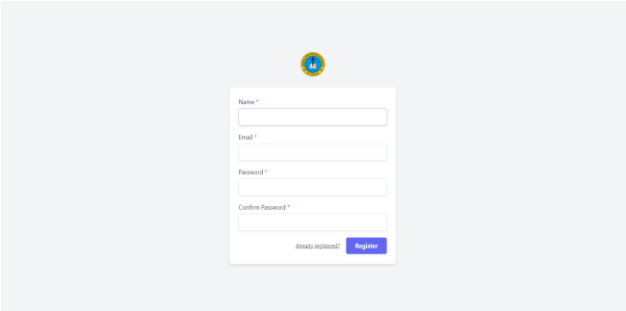
4. HASIL DAN PEMBAHASAN

4.1. Impelementasi Sistem

Proses membangun sistem berdasarkan desain dan rancangan sebelumnya, yang dapat dibagi menjadi beberapa bagian sesuai dengan fungsi yang telah direncanakan sebelumnya disebut implementasi sistem.

4.1.1. Halaman Register

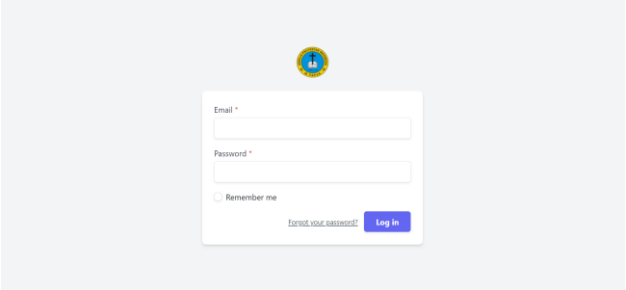
Halaman registrasi adalah bagian dari sebuah *platform* atau aplikasi di mana pengguna dapat membuat akun baru. Pada halaman ini, pengguna diminta untuk mengisi informasi penting seperti nama lengkap, alamat email, kata sandi, dan konfirmasi kata sandi untuk tujuan keamanan. Informasi ini digunakan untuk membuat identitas pengguna yang unik dalam sistem, memastikan bahwa hanya pemilik akun yang dapat mengakses dan menggunakan layanan yang tersedia setelah proses registrasi selesai. Halaman registrasi adalah langkah awal yang penting untuk mengakses fitur-fitur eksklusif yang memerlukan autentikasi.



Gambar 4. Halaman registrasi

4.1.2. Halaman *Login*

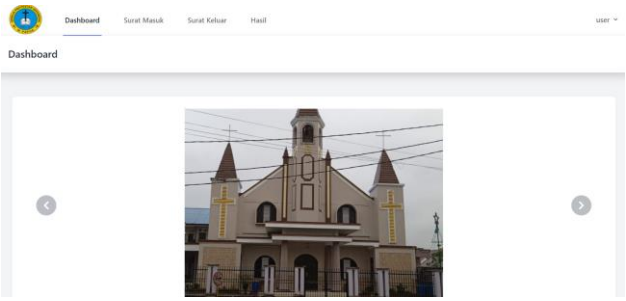
Halaman *login* yang meminta pengguna mengisi kolom email dan kata sandi, memainkan peran penting dalam sistem keamanan. Berikut adalah desain halaman *login* sistem tersebut.



Gambar 5. Halaman *login*

4.1.3. Halaman *Dashboard*

Halaman *dashboard website* ini menampilkan beberapa foto gereja dari GPI Papua Klasis Mimika Papua Tengah. Ini memberikan pengunjung gambaran visual tentang keragaman budaya dan kehidupan rohani komunitas gereja di daerah tersebut.

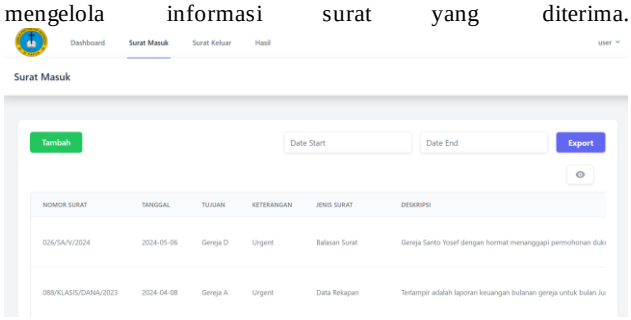


Gambar 6. Halaman *Dashboard*

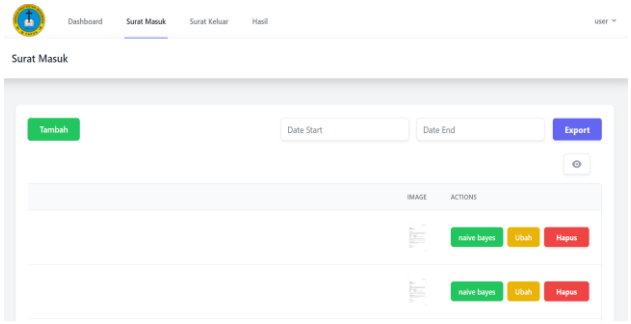
4.1.4. Halaman Surat Masuk

Halaman surat masuk adalah komponen penting dari sistem administrasi; memungkinkan pengguna untuk mencatat dan

mengelola informasi surat yang diterima.

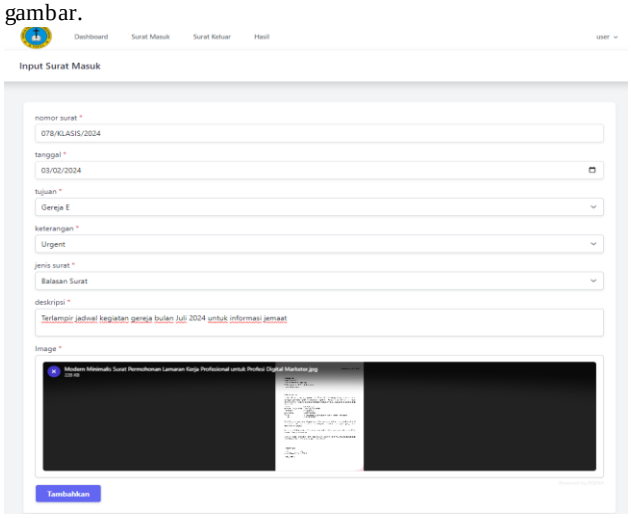


Gambar 7. Halaman utama surat masuk



Gambar 8. Halaman utama surat masuk

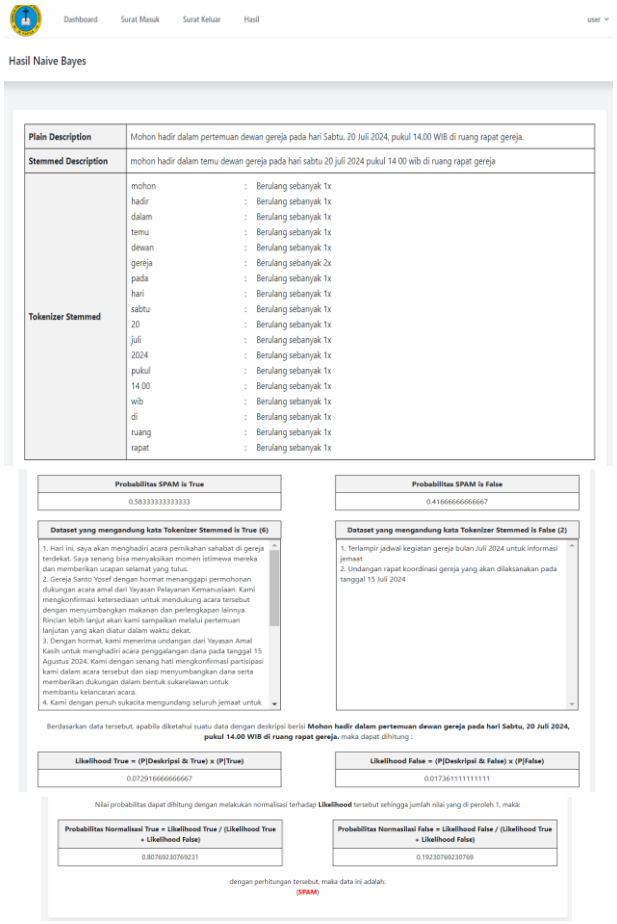
Dengan menekan tombol "Tambah" pada halaman menu tambah, pengguna dapat menambah data surat seperti nomor surat, tanggal, tujuan, keterangan, jenis surat, deskripsi, dan lampiran gambar.



Gambar 9. Halaman input surat masuk

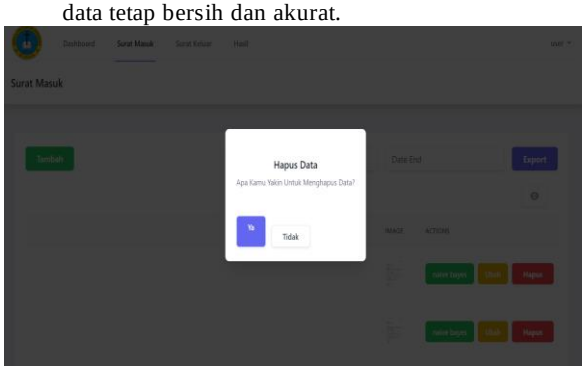
Halaman surat masuk juga memiliki beberapa fitur penting lainnya. Beberapa di antaranya adalah fitur *Naive Bayes* yang menampilkan halaman perhitungan *Naive Bayes*, fitur untuk mengubah data, dan fitur untuk menghapus data. Penjelasan lebih lanjut tentang fitur-fitur ini dapat ditemukan di bawah ini.

- Fitur *Naive Bayes*
  - *Plain Description* adalah teks atau kalimat asli tanpa modifikasi, seperti stemming atau tokenisasi. *Plain Description* berfungsi sebagai input atau data mentah yang akan dianalisis dalam situasi ini. Ini menyajikan informasi secara lengkap dan utuh sesuai dengan penulisan awalnya sebelum diproses oleh algoritma *Naive Bayes*.
  - *Stemmed Description* adalah teks atau kalimat yang telah mengalami proses *stemming*, yaitu proses mengubah kata menjadi bentuk dasar atau akarnya. Tujuan dari proses ini adalah untuk mengurangi variasi kata sehingga kata-kata dengan makna yang sama dianggap sebagai satu entitas. Proses ini penting untuk analisis teks karena membantu menyederhanakan dan menyatukan berbagai bentuk kata yang berbeda tetapi memiliki akar yang sama.
  - *Tokenizer Stemmed* adalah hasil dari proses *tokenisasi* pada teks yang telah *distemming* dan diubah ke bentuk dasar. *Tokenisasi* membagi teks menjadi bagian-bagian yang lebih kecil, seperti kata atau *frasa*. Setelah teks di-*stemming*, *tokenisasi* memisahkan kata-kata tertentu untuk analisis. Metode ini membantu menghitung berapa kali setiap kata yang telah di-*stemming* muncul.
  - *Probabilitas SPAM is True* dan *Probabilitas SPAM is False* merupakan hasil perhitungan algoritma *Naive Bayes*, yang digunakan untuk menentukan apakah sebuah pesan (deskripsi) tergolong *spam* atau *non-spam*. Algoritma ini menggunakan *probabilitas* dan *statistik*, menganalisis data pelatihan sebelumnya untuk menentukan kemungkinan bahwa suatu pesan akan termasuk dalam kategori tertentu.
  - *Dataset yang Mengandung Kata Tokenizer Stemmed is True* dan *Dataset yang Mengandung Kata Tokenizer Stemmed is False* adalah bagian dari proses analisis teks untuk mengidentifikasi apakah pesan termasuk *spam* atau non-spam. Kedua menyebut bagian dari dataset pelatihan yang digunakan untuk mengajarkan model *Naive Bayes*. Dalam sistem ini, setiap entri *database* harus memiliki minimal 5 *stem*. Entri yang kurang dari 5 *stem* tidak akan ditampilkan dalam sistem, tetapi entri yang memiliki 5 *stem* atau lebih akan ditampilkan.
  - *Likelihood True* dan *Likelihood False* adalah dua bagian penting dari algoritma *Naive Bayes* yang digunakan untuk menentukan kemungkinan bahwa sebuah pesan akan dikategorikan sebagai *spam* atau *non-spam*.
  - *Probabilitas Normalisasi True* dan *Probabilitas Normalisasi False* adalah tahap akhir perhitungan algoritma *Naive Bayes* yang digunakan untuk menentukan kategori pesan, seperti *spam* atau *non-spam*. Proses normalisasi mengubah nilai *Likelihood True* dan *Likelihood False* menjadi *probabilitas* total 1, yang membuat interpretasi hasil akhir lebih mudah.



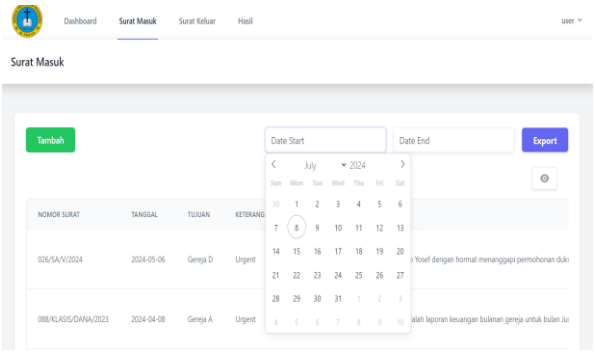
Gambar 10. Tampilan halaman hasil *Naïve Bayes*

- Halaman surat masuk memiliki fitur hapus surat yang memungkinkan pengguna menghapus surat yang telah tercatat dalam sistem. Fitur ini memungkinkan pengguna untuk mengurangi atau menghapus arsip surat yang tidak lagi relevan atau diperlukan. Fitur ini memastikan bahwa sistem administrasi dapat dengan mudah mengelola surat masuk dan membantu menjaga data tetap bersih dan akurat.



Gambar 11. Fitur Hapus Data Surat Masuk

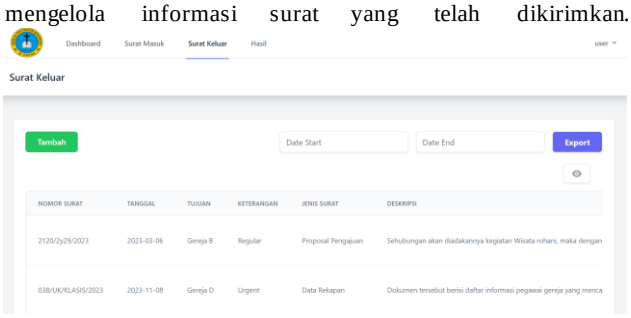
- Dengan menggunakan fitur filter yang mengatur tanggal mulai dan tanggal selesai, Anda dapat menemukan surat dalam rentang waktu tertentu. Selain itu, opsi ekspor *Microsoft Excel* mempermudah pembuatan laporan dan analisis. Dengan demikian, halaman ini tidak hanya membantu mengelola surat masuk dengan lebih cepat, tetapi juga membantu mengatur dan menggunakan data dengan lebih terorganisir dan efisien.



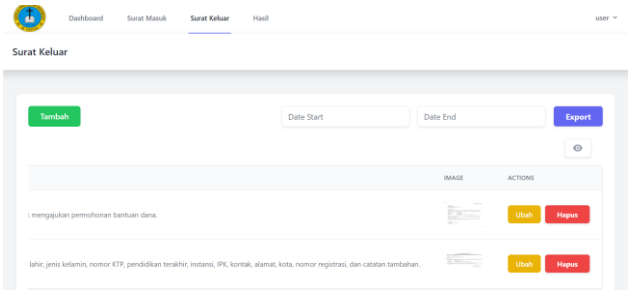
Gambar 12. Fitur *Date Start* dan *Date End*

4.1.5. Halaman Surat Keluar

Halaman surat keluar, komponen penting dari sistem administrasi, memungkinkan pengguna untuk mencatat dan mengelola informasi surat yang telah dikirimkan.

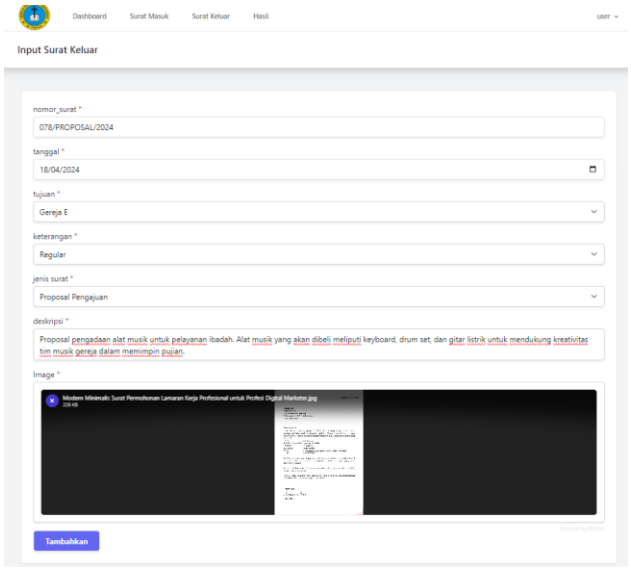


Gambar 13. Halaman utama surat masuk



Gambar 14. Halaman utama surat masuk

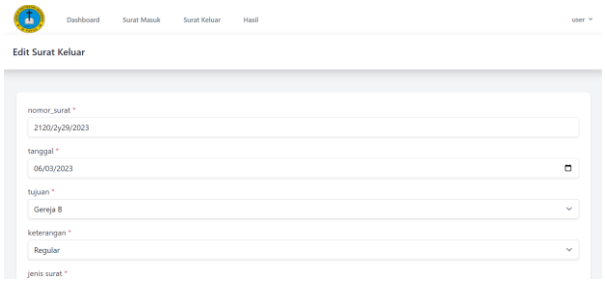
Menekan tombol "Tambah" di halaman menu tambah memungkinkan pengguna untuk menambahkan nomor surat, tanggal, tujuan, keterangan, jenis surat, dan lampiran gambar. Fitur ini memungkinkan pengguna untuk menambahkan data surat.



Gambar 15. Halaman input surat masuk

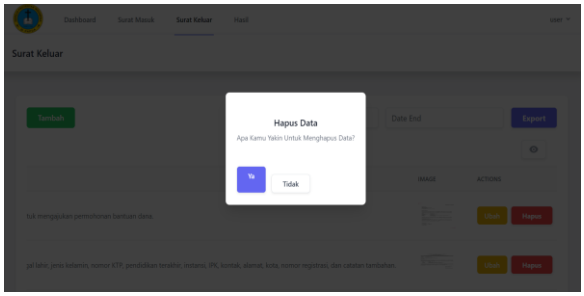
Halaman surat keluar juga memiliki fitur penting lainnya. Misalnya, mereka dapat mengubah dan menghapus data surat serta memfilter berdasarkan tanggal mulai dan selesai.

- Pada halaman surat keluar, pengguna dapat mengubah informasi surat yang tercatat dengan menggunakan fitur "ubah surat". Fitur ini memungkinkan Anda mengubah tanggal, tujuan, keterangan, nomor, jenis, dan gambar. Fitur ini membantu menjaga keakuratan data dan memudahkan pengelolaan surat keluar dalam sistem administrasi.



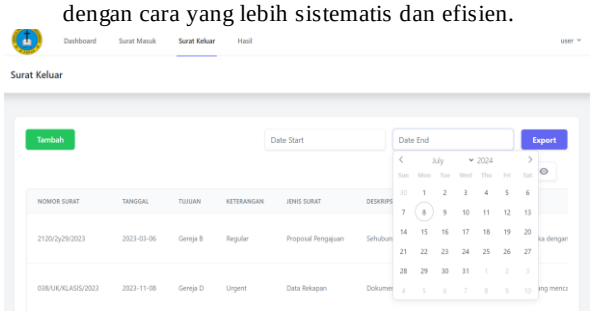
Gambar 16. Halaman Ubah Surat Keluar

- Pada halaman surat keluar, fitur "Hapus Surat" memungkinkan pengguna menghapus surat yang sudah tercatat dalam sistem. Mereka dapat memilih surat mana yang ingin dihapus dan melakukannya untuk mengurangi atau membersihkan arsip surat yang tidak lagi relevan atau diperlukan. Fitur ini memudahkan pengguna mengelola surat keluar dalam sistem administrasi dan membantu menjaga data tetap bersih dan akurat.



Gambar 17. Fitur Hapus Data Surat Keluar

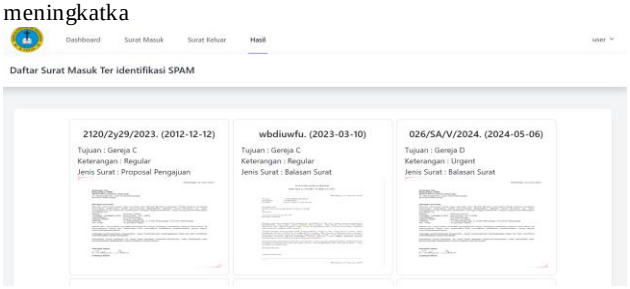
- Untuk melakukan pencarian pada halaman surat keluar dalam rentang waktu tertentu, pengguna dapat menggunakan filter berdasarkan tanggal mulai dan tanggal selesai. Selain itu, opsi ekspor ke *Microsoft Excel* memungkinkan pembuatan laporan dan analisis terkait surat keluar yang lebih mudah. Fitur ini tidak hanya meningkatkan efisiensi pengelolaan surat keluar tetapi juga membantu mengatur dan menggunakan data dengan cara yang lebih sistematis dan efisien.



Gambar 18. Fitur *Date Start* dan *Date End*

4.1.6. Halaman Hasil

Halaman sistem ini sangat penting karena dapat menangani hasil pendeteksian surat masuk sebagai *spam* atau *non-spam*. Ini memungkinkan perusahaan untuk mengelola dan menyaring korespondensi berdasarkan klasifikasi tersebut. Sistem ini memiliki fitur yang tidak hanya meningkatkan efisiensi pengelolaan dokumen tetapi juga memastikan keamanan data dengan menyaring potensi ancaman keamanan yang terkait dengan surat-surat *spam*. Fitur-fiturnya termasuk deteksi otomatis, pengelompokan berdasarkan nomor, tanggal, dan jenis surat, dan kemampuan untuk menyimpan gambar surat fisik. Metode *Naive Bayes*, salah satu metode yang paling umum untuk mendeteksi *spam*, memberikan kontribusi yang signifikan dalam meningkatka



Gambar 19. Halaman Daftar Hasil SPAM

4.2. Implementasi Algoritma Naive Bayes

Untuk menentukan apakah arsip yang dinilai ulang seharusnya dikategorikan sebagai *spam* atau *non-spam*, penelitian ini akan menggunakan Algoritma *Naive Bayes* untuk mengklasifikasikan deskripsi pengarsipan surat. Dataset yang digunakan terdiri dari 50 data, dengan 20 di antaranya diidentifikasi sebagai *spam* dan 30 lainnya sebagai *non-spam*. Penelitian ini bertujuan untuk menghasilkan rekomendasi klasifikasi yang dapat meningkatkan efisiensi dan akurasi dalam manajemen arsip serta membantu pengambilan keputusan administratif yang lebih terinformasi terkait tindakan yang harus diambil terkait setiap arsip yang dinilai ulang. Ini dicapai melalui proses pemrosesan data yang

mencakup *stemming* dan *tokenisasi*, serta menggunakan *probabilitas prior* dan kemungkinan dari model *Naive Bayes*.

4.2.1. *Plain Description*

Pada tahap ini, data dikumpulkan langsung dari lembaga terkait untuk keperluan pengelolaan dokumen dan analisis. Proses pengumpulan data ini sangat penting untuk memastikan bahwa semua informasi yang diperlukan dikumpulkan dengan lengkap dan akurat. Setelah data dikumpulkan, langkah berikutnya adalah memasukkan data ke dalam sistem untuk diproses menggunakan metode perhitungan *Naive Bayes*. Metode ini menganalisis teks mentah dari data yang telah dimasukkan, yang disebut deskripsi sederhana. Dalam analisis ini, sistem menentukan apakah setiap dokumen dapat dikategorikan sebagai *spam* atau *non-spam* berdasarkan fitur *linguistik* yang ada dalam teks. Proses ini sangat penting untuk membantu pengambilan keputusan yang akurat dan informasi tentang pengelolaan dan pemrosesan dokumen dalam konteks penelitian atau administratif. Berikut adalah contoh *database* yang digunakan untuk melatih *Naive Bayes*.

Mohon hadir dalam pertemuan dewan gereja pada hari Sabtu, 20 Juli 2024, pukul 14.00 WIB di ruang rapat gereja.

4.2.2. *Stemmed Description*

Pada tahap ini, teks deskripsi asli yang telah diproses dengan metode *stemming* digunakan. Proses *stemming* adalah pengurangan kata ke bentuk dasar atau akarnya. Ini dilakukan dalam pengolahan teks dan analisis bahasa alami untuk menyederhanakan kata-kata dengan variasi morfologis sehingga kata-kata yang berasal dari akar yang sama dianggap sebagai satu entitas. Seperti yang ditunjukkan dalam penjelasan sederhana di atas, contoh *database* yang hasil dari proses *stemming*-nya adalah

mohon hadir dalam temu dewan gereja pada hari sabtu 20 juli 2024 pukul 14 00 wib di ruang rapat gereja

Hasilnya adalah bahwa kata-kata seperti "pertemuan" berubah menjadi "temu", dan kata-kata yang berulang seperti "gereja" tidak berubah karena sudah ada dalam bentuk dasar mereka. Terutama ketika algoritma pembelajaran mesin seperti *Naive Bayes* digunakan untuk klasifikasi dokumen, proses stemming membantu meningkatkan akurasi analisis teks dan mengurangi kompleksitas teks. Dalam hal ini, deskripsi yang telah di-*stemming* digunakan untuk menentukan kemungkinan bahwa teks tersebut akan dikategorikan sebagai *spam* atau *non-spam*.

4.2.3. *Tokenizer Stemmed*

Pada titik ini, teks yang telah distemming dipecahkan menjadi unit yang lebih kecil yang dikenal sebagai token. Setelah *stemming* mengubah teks ke bentuk aslinya, langkah berikutnya adalah memecah teks menjadi kata-kata individu atau token. *Tokenisasi* adalah langkah penting dalam pemrosesan bahasa alami karena memungkinkan analisis teks yang lebih mendalam dan khusus. Misalnya, dalam deskripsi yang di-*stemming* dengan judul "mohon hadir dalam temu dewan gereja pada hari sabtu 20 juli 2024 pukul 14 00 wib di ruang rapat gereja", setiap kata digunakan dengan cara yang berbeda. "Mohon", "hadir", "dalam", "temu", "dewan", "gereja," dll. Frekuensi kemunculan setiap token juga dicatat selama proses *tokenisasi*. Sebagai contoh, kata "gereja" muncul dua kali dalam teks. Selanjutnya, model pembelajaran mesin seperti *Naive Bayes* menggunakan data frekuensi ini untuk perhitungan probabilitas. Memecah teks menjadi bagian-bagian yang lebih kecil memungkinkan analisis untuk menemukan pola penting yang menentukan apakah teks tersebut termasuk dalam kategori *spam* atau *non-spam*. *Tokenizer* Stemmed memastikan bahwa setiap kata dasar diperhitungkan dengan benar dalam model statistik yang digunakan untuk klasifikasi dokumen, dengan menyederhanakan dan mengorganisasikan teks untuk analisis lebih lanjut. Tabel data yang sudah ada di *Tokenizer Stemmed* dapat ditemukan di bawah ini.

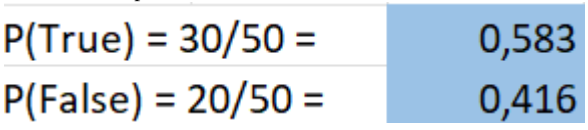
Tabel 1. Tabel *Tokenizer Stemmed*

mohon	:	Berulang sebanyak 1x
hadir	:	Berulang sebanyak 1x
dalam	:	Berulang sebanyak 1x
temu	:	Berulang sebanyak 1x
dewan	:	Berulang sebanyak 1x
gereja	:	Berulang sebanyak 2x
pada	:	Berulang sebanyak 1x
hari	:	Berulang sebanyak 1x
sabtu	:	Berulang sebanyak 1x
20	:	Berulang sebanyak 1x
juli	:	Berulang sebanyak 1x
2024	:	Berulang sebanyak 1x
pukul	:	Berulang sebanyak 1x
14 00	:	Berulang sebanyak 1x
wib	:	Berulang sebanyak 1x
di	:	Berulang sebanyak 1x
ruang	:	Berulang sebanyak 1x
rapat	:	Berulang sebanyak 1x

4.2.4. *Probabilitas Prior*

Pada titik ini, *probabilitas prior* digunakan sebagai bagian dari perhitungan yang lebih besar dalam *Naïve Bayes* untuk menentukan *probabilitas posterior*. Perhitungan ini menggabungkan *probabilitas prior* dengan *likelihood*—kemungkinan kata-kata tertentu muncul dalam dokumen *spam* atau *non-spam* dan memberikan *baseline* awal yang kemudian diubah oleh data aktual dari dokumen yang dianalisis.

Dalam model *Naive Bayes*, konsep digunakan untuk mengkategorikan dokumen, terutama untuk menentukan apakah sebuah dokumen atau pesan termasuk dalam kategori *spam* atau *non-spam*. Dalam hal ini, "*Probabilitas SPAM is True*" menunjukkan kemungkinan bahwa dokumen tersebut adalah *spam*, sementara "*Probabilitas SPAM is False*" menunjukkan kemungkinan bahwa dokumen tersebut bukan spam, atau dengan kata lain, *non-spam*.



Gambar 20. Perhitungan manual di *Microsoft Excel*

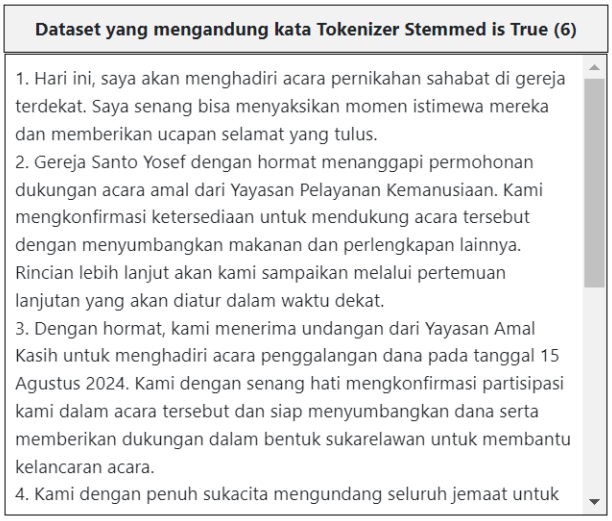
Gambar 21. Perhitungan *Probabilitas* pada system

Langkah awal penting dalam proses klasifikasi dokumen menggunakan *Naive Bayes* adalah *Probabilitas SPAM is True* dan *Probabilitas SPAM is False*. Mereka memberikan dasar statistik tentang kemungkinan spesifik dari kata-kata dalam dokumen untuk memperbaiki prediksi apakah dokumen tersebut kemungkinan besar adalah *spam* atau *non-spam*.

4.2.5. Dataset yang mengandung kata *Tokenizer Stemmed is True/False*

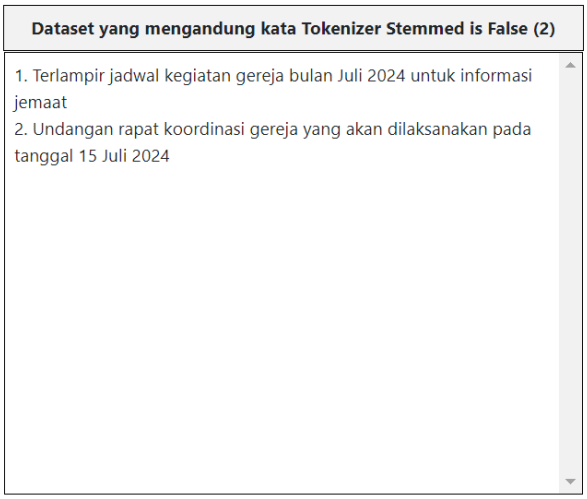
konsep yang digunakan dalam analisis teks dan pembelajaran mesin untuk mengetahui kehadiran atau ketidakhadiran kata-kata tertentu dalam sebuah dataset. Setelah *stemming* dan *tokenisasi* dokumen atau deskripsi, kita memecah teks menjadi kata-kata dasar atau token. Kemudian, kita ingin mengetahui berapa banyak dokumen dalam dataset yang mengandung token-token tersebut.

Dataset yang Mengandung Kata *Tokenizer Stemmed is True* adalah kumpulan dokumen dalam dataset yang mengandung satu atau lebih kata-kata yang di-*stemming* dan di-*tokenisasi* dari deskripsi yang diberikan. Misalnya, jika deskripsi mengandung kata-kata seperti "mohon", "hadir", "dalam", "temu", dan seterusnya, kita akan menghitung jumlah dokumen dalam dataset yang mengandung kata-kata ini, yang membantu kita menentu.



Gambar 22. Dataset yang Mengandung Kata *Tokenizer Stemmed is True*

Sebaliknya, ini mengacu pada kumpulan dokumen dalam dataset yang tidak mengandung kata-kata yang telah di-*stemming* dan di-*tokenisasi* dari deskripsi yang diberikan; dalam contoh berikut, dua dokumen dalam dataset tidak mengandung kata-kata tersebut, yang menunjukkan bahwa token-token tersebut tidak muncul di dua dokumen yang berbeda dalam dataset.



Gambar 23. Dataset yang Mengandung Kata *Tokenizer Stemmed is False*

4.2.6. *Likelihood True/ Likelihood False*

Komponen penting dari model *Naïve Bayes* digunakan untuk menghitung kemungkinan sebuah dokumen termasuk dalam kategori tertentu (misalnya, *spam* atau *non-spam*) berdasarkan kata-kata yang ada dalam dokumen. Kemungkinan ini mengukur seberapa besar kemungkinan kata-kata tertentu muncul dalam dokumen yang termasuk dalam kategori tertentu. *Likelihood True* mengacu pada kemungkinan bahwa token tertentu muncul dalam dokumen *spam* (atau "Benar" untuk kategori *spam*). Untuk menghitung *Likelihood True*, kita harus mengetahui berapa kali setiap token muncul dalam dokumen *spam* di seluruh dataset. Sebaliknya, *Likelihood False* menunjukkan kemungkinan bahwa token tertentu muncul dalam dokumen yang dikategorikan sebagai non-spam; perhitungannya sama dengan *Likelihood True* tetapi menggunakan data dari dokumen *non-spam*. Hasilnya adalah sebagai berikut:

Likelihood True = (6/50) x 0,583 =	0,0729
Likelihood False = (2/50) x 0,416 =	0,0174

Gambar 24. Perhitungan manual di *Microsoft Excel*

Gambar 25. Perhitungan *Likelihood* pada system

4.2.7. *Probabilitas Normalisasi*

Langkah penting dalam model *Naive Bayes* adalah menentukan kategori akhir sebuah dokumen dengan menggunakan perhitungan kemungkinan. Setelah menghitung kemungkinan untuk masing-masing kategori (*spam* atau *non-spam*), kita perlu menormalisasi nilai-nilai tersebut sehingga jumlah kemungkinan total menjadi 1. Normalisasi ini memungkinkan kita untuk membandingkan kemungkinan relatif dari masing-masing kategori dan membuat ramalan yang lebih akurat. Kami menggunakan rumus berikut untuk menghitung kemungkinan normalisasi :

$$\text{Probabilitas Normalisasi True} = \frac{\text{Likelihood True}}{\text{Likelihood True} + \text{Likelihood False}}$$

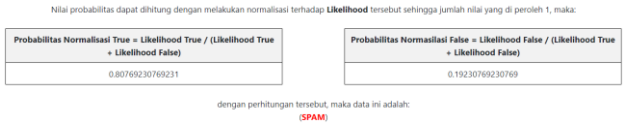
$$\text{Probabilitas Normalisasi False} = \frac{\text{Likelihood False}}{\text{Likelihood True} + \text{Likelihood False}}$$

Gambar 26. Rumus menghitung *probabilitas normalisasi*

Maka,

Probabilitas Normalisasi True = 0,0729 / (0,0729 + 0,0174) =	0.8077
Probabilitas Normalisasi True = 0,017361 / (0,0729 + 0,0174) =	0.1923

Gambar 27. Perhitungan manual di *Microsoft Excel*



Gambar 28. Perhitungan *probabilitas normalisasi* pada sistem

Hasil menunjukkan bahwa kemungkinan normalisasi benar adalah sekitar 0,8077 dan kemungkinan normalisasi salah adalah sekitar 0,1923. Dengan kata lain, perhitungan ini menunjukkan 80.77% kemungkinan bahwa dokumen tersebut adalah *spam* (Benar) dan 19.23% kemungkinan bahwa dokumen tersebut tidak spam.

Model *Naive Bayes* akan mengklasifikasikan dokumen ini sebagai spam karena kemungkinan kategori *spam* lebih tinggi. Normalisasi probabilitas memungkinkan model untuk membuat prediksi yang lebih akurat dan akurat karena memastikan bahwa keputusan akhir didasarkan pada perbandingan yang adil dan proporsional dari kemungkinan untuk setiap kategori.

4.3. Pengujian *Confusion Matrix*

Alat yang digunakan untuk mengevaluasi kinerja model klasifikasi adalah *Confusion Matrix*. *Matrix* ini menunjukkan bagaimana model klasifikasi Anda melakukan tugas klasifikasinya dengan membagi jumlah prediksi yang benar dan salah menjadi empat kategori: *True Positive* (TP), *False Positive* (FP), *True Negative* (TN), dan *False Negative* (FN). Berikut adalah definisi masing-masing kategori dalam konteks klasifikasi *spam*:

- True Positive (TP): Jumlah dokumen yang dianggap benar-benar *spam*.
- False Positive (FP): Banyak dokumen yang dianggap *spam* tetapi sebenarnya tidak.
- True Negative (TN): Sebagian besar dokumen yang dikategorikan sebagai *spam* tetapi tidak benar-benar spam.
- False Negative (FN): Banyak dokumen yang dianggap tidak *spam* tetapi sebenarnya adalah *spam*.

Sebagai contoh, *Confusion Matrix* dapat dibentuk

Tabel 2. Tabel *Confusion Matrix*

	Predicted Spam	*Predicted Non-spam*
Actual Spam	22 (TP)	8 (FN)
Actual Non-spam	3 (FP)	17 (TN)

Kita dapat menghitung beberapa metrik evaluasi penting dari *confusion matrix*:

- ***Accuracy***: Persentase prediksi yang benar dari total prediksi.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} = \frac{22 + 17}{22 + 17 + 3 + 8} = \frac{39}{50} = 0{,}78\ atau\ 78\%$$

- **Precision *(Spam)*:** Proporsi prediksi spam yang benar-benar *spam*

$$Precision = \frac{TP}{TP + FP} = \frac{22}{22 + 3} = \frac{22}{25} = 0{,}88\ atau\ 88\%$$

- ***Recall (Spam)*:** Proporsi data spam yang berhasil diidentifikasi dengan benar sebagai *spam*.

$$Recall = \frac{TP}{TP + FN} = \frac{22}{22 + 8} = \frac{22}{30} = 0{,}733\ atau\ 73{,}3\%$$

- ***F1 Score* (Spam)**: *Harmonic mean* dari *precision* dan *recall*.

$$F1\ Score = 2\ x\ \frac{Precision\ x\ Recall}{Precision + Recall} = 2\ x\ \frac{0{,}88\ x\ 0{,}733}{0{,}88 + 0{,}733} = 2\ x\ \frac{0{,}645}{1{,}613} = 0{,}799\ atau\ 79{,}9\%$$

Dengan menggunakan *confusion matrix* dan menghitung berbagai metrik evaluasi, kita dapat melihat bagaimana model klasifikasi berfungsi dalam memprediksi data *spam* dan *non-spam*. Model menunjukkan *accuracy* 78%, *precision* 88%, *recall* 73.3%, dan *F1 Score* 79.9%.

5. KESIMPULAN DAN SARAN

Penelitian ini menemukan bahwa sistem informasi manajemen e-arsip surat berbasis *web* yang menggunakan metode *Naive Bayes* dan proses *preprocessing* seperti *stemming* dan *tokenizer* meningkatkan efisiensi dan akurasi pengelolaan arsip surat di Lembaga GPI Papua Klasis Mimika. Proses *preprocessing* teks terbukti efektif dalam mengklasifikasikan surat menjadi kategori yang tepat, yaitu spam dan non-spam, dengan hasil pengujian menunjukkan bahwa proses ini meningkatkan jumlah surat yang disimpan dalam arsip Model klasifikasi memiliki *accuracy* 78%, *precision* 88%, *recall* 73,3%, dan F1 Score 79.9%, menurut pengujian menggunakan *Confusion Matrix*. Berdasarkan hasil implementasi aplikasi, saran pengembangan selanjutnya termasuk meningkatkan fitur modul pelaporan dan analisis data arsip; integrasi dengan sistem informasi lainnya; evaluasi dan pemeliharaan sistem secara berkala; dan eksplorasi algoritma klasifikasi seperti *KNN*, *SVM*, dan *K-Means* untuk meningkatkan akurasi prediksi. Studi mendatang juga harus mempertimbangkan penggunaan *Optical Character Recognition (OCR)* dalam sistem pengarsipan agar lebih mudah mengelola berbagai format surat.

DAFTAR PUSTAKA

[1] A. Betiana, "SISTEM INFORMASI E-ARSIP SURAT PADA KANTOR KECAMATAN LIMAU DENGAN MENERAPKAN METODE CHRONOLOGICAL FILING SYSTEM," vol. 2, no. 1, pp. 7–10, 2021.

[2] E. L. Pratiwi, H. Anwar, J. Administrasi, B. Politeknik, and N. Banjarmasin, "SISTEM INFORMASI E-ARSIP BERBASIS WEB PADA PT. GEDE LANGGENG MAKMUR," Online, 2022. [Online]. Available:

http://ejurnal.poliban.ac.id/index.php/intekna/issue/archive

[3] C. Tristianto, J. Raya, P. Serpong, N. 10 Tangerang, and S. Banten, "PERANCANGAN SISTEM INFORMASI PENGARSIPAN SURAT MENGGUNAKAN METODE JOHARI WINDOW DAN RAPID APPLICATION DEVELOPMENT BERBASIS WEB," *Jurnal Ilmu Komputer JIK*, pp. 7–12, 2022.

[4] W. Suratman, F. Fauziah, and R. T. K. Sari, "Aplikasi Elektronik Arsip (E-Arsip) Surat Berbasis Web Menggunakan Metode First In First Out (FIFO)," *Paradigma - Jurnal Komputer dan Informatika*, vol. 23, no. 2, Sep. 2021, doi: 10.31294/p.v23i2.10749.

[5] D. Abisono Punkastyo, F. Septian, and A. Syaripudin, "Implementasi Data Mining Menggunakan Algoritma Naïve Bayes Untuk Prediksi Kelulusan Siswa," 2024.

[6] S. Rumkorem Sitorus and M. Huda, "PENERAPAN VALUE ENGINEERING PADA PROYEK PENINGKATAN JALAN TIMIKA BATAS TUGU PAPUA," vol. 8, no. 1, pp. 11–018, 2020.

[7] R. Kurniawan Ritonga and R. Firdaus, "JICN: Jurnal Intelek dan Cendikiawan Nusantara PENTINGNYA SISTEM INFORMASI MANAJEMEN DALAM ERA DIGITAL THE IMPORTANCE OF MANAGEMENT INFORMATION SYSTEMS IN THE DIGITAL ERA," vol. 1, no. 3, 2024, [Online]. Available: https://jicnusantara.com/index.php/jicn

[8] Andrian Syahputra, Ragil Wiranti, and W. A. Widiya Astita, "PERAN SISTEM INFORMASI MANAJEMEN ORGANISASI DALAM PENGAMBILAN KEPUTUSAN," *Jurnal Manajemen Sistem Informasi (JMASIF)*, vol. 1, no. 1, pp. 26–31, Apr. 2022, doi: 10.35870/jmasif.v1i1.67.

[9] A. Tri Amalia, P. Administasi Perkantoran, and F. Ekonomika dan Bisnis, "Sistem Informasi Manajemen Arsip Elektronik (E-Arsip) Berbasis Microsoft Access Terhadap Efektivitas Penemuan Kembali Arsip Pada SMKN 4 Surabaya Lifa Farida Panduwinata," 2022. [Online]. Available: https://journal.unesa.ac.id/index.php/jpap

[10] A. Firdaus and W. I. Firdaus, "Text Mining Dan Pola Algoritma Dalam Penyelesaian Masalah Informasi: (Sebuah Ulasan)," 2021.

[11] G. Firmansyah and A. Hermawan, "Implementasi Algortima Naive Bayes Untuk Klasifikasi Kesegaran Buah Jeruk," *Jurnal Informatika*, vol. 10, no. 2, pp. 180–184, Oct. 2023, doi: 10.31294/inf.v10i2.16115.

[12] A. Anggakara, R. Helilintar, and R. A. Ramadhani, "Implementasi Metode Naïve Bayes Untuk Menentukan Kelas Unggulan Pada Siswa SMP," 2022.

[13] F. A. Muttaqin and A. Mukaharil Bachtiar, "IMPLEMENTASI TEKS MINING PADA APLIKASI PENGAWASAN PENGGUNAAN INTERNET ANAK 'DODO KIDS BROWSER,'" *Jurnal Ilmiah Komputer dan Informatika (KOMPUTA)*, pp. 1–8, 2019, [Online]. Available: http://www.bing.com/

[14] J. A. Rieuwpassa, S. Sugito, and T. Widiharih, "IMPLEMENTASI METODE NAIVE BAYES CLASSIFIER UNTUK KLASIFIKASI SENTIMEN ULASAN PENGGUNA APLIKASI NETFLIX PADA GOOGLE PLAY," *Jurnal Gaussian*, vol. 12, no. 3, pp. 362–371, Feb. 2024, doi: 10.14710/j.gauss.12.3.362-371.

[15] A. D. Imanuel, N. N. Pusparini, and A. Sani, "KLASIFIKASI UNTUK MEMPREDIKSI TINGKAT KELULUSAN MAHASISWA STMIK WIDURI MENGGUNAKAN ALGORITMA NAÏVE BAYES," 2024.

BIODATA PENULIS

Jennifer Florenzia Indey
Mahasiswa Program Studi Informatika, Fakultas Teknik, Universitas 17 Agustus 1945 Surabaya.

Supangat, M.Kom., Ph.D., ITIL., COBIT., CLA
Dosen Program Studi Informatika, Fakultas Teknik, Universitas 17 Agustus 1945 Surabaya.