DETEKSI SUREL *SPAM* DAN *NON-SPAM* BAHASA INDONESIA MENGGUNAKAN METODE NAÏVE BAYES

Disusun Oleh:

N a m a : Azmiardhy Zulkifli Farmadiansyah
NIM : 17523225

PROGRAM STUDI INFORMATIKA – PROGRAM SARJANA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA
2021

HALAMAN PENGESAHAN DOSEN PEMBIMBING

DETEKSI SUREL *SPAM* DAN *NON-SPAM* BAHASA INDONESIA MENGGUNAKAN METODE NAÏVE BAYES

TUGAS AKHIR

Disusun Oleh:

N a m a : Azmiardhy Zulkifli Farmadiansyah

NIM : 17523225

Yogyakarta, 19 Juli 2021

Pembimbing I,

(Fayruz Rahma, S.T., M.Eng.)

Pembimbing II,

(Ahmad Fathan Hidayatullah, S.T., M.Cs.)

HALAMAN PENGESAHAN DOSEN PENGUJI

DETEKSI SUREL *SPAM* DAN *NON-SPAM* BAHASA INDONESIA MENGGUNAKAN METODE NAÏVE BAYES

TUGAS AKHIR

Telah dipertahankan di depan sidang penguji sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer dari Program Studi Informatika – Program Sarjana di Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 2 Agustus 2021

Tim Penguji

Ahmad Fathan Hidayatullah, S.T., M.Cs. _____

Anggota 1

Dr. Syarif Hidayat, S.Kom., M.I.T. _____

Anggota 2

Sri Mulyati, S.Kom., M.Kom. _____

Mengetahui,
Ketua Program Studi Informatika – Program Sarjana
Fakultas Teknologi Industri
Universitas Islam Indonesia

(Dr. Raden Teduh Dirgahayu, S.T., M.Sc.)

HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan di bawah ini:

Nama : Teknik Informatika

NIM : 17523225

Tugas akhir dengan judul:

DETEKSI SUREL *SPAM* DAN *NON-SPAM* BAHASA INDONESIA MENGGUNAKAN METODE NAÏVE BAYES

Menyatakan bahwa seluruh komponen dan isi dalam tugas akhir ini adalah hasil karya saya sendiri. Apabila di kemudian hari terbukti ada beberapa bagian dari karya ini adalah bukan hasil karya sendiri, tugas akhir yang diajukan sebagai hasil karya sendiri ini siap ditarik kembali dan siap menanggung risiko dan konsekuensi apapun.

Demikian surat pernyataan ini dibuat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 19 Juli 2021

(Azmiardhy Zulkifli Farmadiansyah)

HALAMAN PERSEMBAHAN

Alhamdulillahirobbil'alamin atas segala nikmat yang telah diberikan kepada kita. Shalawat serta salam kita haturkan kepada junjungan kita Nabi Muhammad SAW yang kita nantikan safa'atnya di yaumul akhir nanti.

Terima kasih yang amat besar saya haturkan kepada orang tua saya yang telah mengasuh dan mendidik saya sejak di dalam kandungan sampai pada saat ini juga dan kedua kakak saya yang telah memberikan dukungan baik moril maupun materil. Semoga senantiasa diberikan kesehatan, kebahagiaan dan panjang umur.

Terima kasih kepada Dosen Pembimbing saya, Bu Fayruz dan Pak Fathan yang selalu melatih, membimbing saya dengan sabar dan selalu memperhatikan saya di setiap pertemuan bimbingan.

Terima kasih kepada semua pihak yang tidak bisa disebutkan satu persatu atas dukungan dan bantuannya baik secara langsung maupun tidak langsung.

HALAMAN MOTO

"Ketahuilah bahwa kemenangan bersama kesabaran, kelapangan bersama kesempitan, dan kesulitan bersama kemudahan." (HR. Tirmidzi).

All big things come from small beginnings – Atomic Habits

Persistence is very important. You should not give up unless you are forced to give up – Elon Musk

KATA PENGANTAR

Assalamualaikum Warahmatullahi Wabarakatuh

Puji dan syukur atas ke hadirat Allah Swt. atas segala nikmat, rahmat, dan karunia-Nya sehingga penulis dapat menyelesaikan Tugas Akhir ini. Tugas Akhir yang berjudul "Deteksi Surel Spam Dan Non Spam Bahasa Indonesia Menggunakan Metode Naïve Bayes" disusun untuk memenuhi salah satu syarat untuk mencapai gelar sarjana (S1) pada Jurusan Informatika Fakultas Teknologi Industri Universitas Islam Indonesisa.

Pada kesempatan ini, penulis mengucapkan banyak terima kasih kepada berbagai pihak yang telah memberikan bantuan dan dukungan baik secara langsung maupun secara tidak langsung dalam penyelesaian Tugas Akhir ini, yaitu:

1. Orang tua dan kedua kakak penulis serta keluarga besar yang telah memberikan berbagai macam dukungan baik moril maupun materil kepada penulis.
2. Dr. Raden Teduh Dirgahayu, S. T., M. Sc. selaku Ketua Prodi Informatika – Program Sarjana UII.
3. Fayruz Rahma, S.T., M.Eng. selaku Dosen Pembimbing I dan Ahmad Fathan Hidayatullah, S. T., M. Sc. selaku Dosen Pembimbing II yang telah memberikan pengarahan, bimbingan, bantuan, serta masukan kepada penulis sehingga Tugas Akhir ini dapat diselesaikan dengan baik.
4. Kepada teman-teman HBS Group yang selalu memberikan semangat dan bantuan dalam menyelesaikan Tugas Akhir ini.
5. Kepada teman-teman Sondong yang selalu ada sebagai tempat berbagi cerita dan dukungan kepada penulis.
6. Teman seperjuangan penulis seluruh mahasiswa Informatika angkatan 2017.
7. Seluruh pihak yang telah membantu yang tidak dapat disebutkan satu persatu.

Penulis sadar banyak terdapat kekurangan dalam pembuatan tugas akhir ini. Namun penulis selalu berharap tugas akhir ini dapat bermanfaat atau mungkin bisa dikembangkan menjadi hal yang lebih besar lagi.

Yogyakarta, 16 Juli 2021

(Azmiardhy Zulkifli Farmadiansyah)

SARI

Penggunaan surel yang mudah saat ini sering dimanfaatkan banyak orang sehingga menimbulkan dampak positif maupun negatif. Surel negatif biasa disebut dengan surel spam yang berisi berupa iklan, penipuan, virus dan *malware* yang berpotensi untuk merugikan orang lain. Masalah tersebut memerlukan penanganan untuk mengatasinya. Penelitian ini bertujuan untuk membuat sebuah model klasifikasi surel *spam* dan *non spam* berbahasa Indonesia menggunakan algoritma Naïve Bayes. Dalam pembuatan model klasifikasi menggunakan data *training* sebanyak 80% atau 493 surel dari 617 surel yang didapat pada penelitian ini. Berdasarkan hasil penelitian yang telah dilakukan, ditemukan bahwa algoritma Naïve Bayes menggunakan fitur N-gram telah berhasil melakukan klasifikasi sangat baik dengan nilai akurasi 87% hingga 95%, nilai *precision* 80% hingga 93% dan *recall* 93% hingga 100%. Model terbaik dalam mengklasifikasikan surel ditemukan menggunakan 2-gram dengan nilai akurasi tertinggi yaitu 95%.

Kata kunci: Klasifikasi, Naïve Bayes, *Spam*, Surel.

GLOSARIUM

Bandwidth	Maksimal besar transfer yang dapat dilakukan pada satu waktu dalam pertukaran data.
Dataset	Data yang digunakan pada proses pembentukan model.
Model	Hasil dari *training* digunakan untuk mengklasifikasikan bahasa.
Pattern Discovery	Menemukan pola-pola menarik yang meliputi pola periodik dan abnormal, dari data temporal.
Pre-processing	Tahap awal perlakuan awal terhadap data dan dijadikan bahan *training*.
User Interface	Tampilan visual sebuah produk yang menjembatani sistem dengan pengguna

DAFTAR ISI

HALAMAN JUDUL i
HALAMAN PENGESAHAN DOSEN PEMBIMBING ii
HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR iv
HALAMAN PERSEMBAHAN v
HALAMAN MOTO vi
KATA PENGANTAR vii
SARI viii
GLOSARIUM ix
DAFTAR ISI x
DAFTAR TABEL xii
DAFTAR GAMBAR xiii
BAB I PENDAHULUAN 1
1.1 Latar Belakang 1
1.2 Rumusan Masalah 2
1.3 Tujuan Penelitian 2
1.4 Batasan Masalah 2
1.5 Manfaat Penelitian 2
1.6 Sistematika Penulisan 2
BAB II LANDASAN TEORI 4
2.1 *Crowdsourcing* 4
2.2 Surel *Spam* 5
2.3 *Machine Learning* 6
2.4 *Text mining* 7
2.5 N-Gram 8
2.6 Klasifikasi 8
2.7 Naïve Bayes 8
2.8 *Performance Evaluation Measure* 9
2.9 Penelitian Terkait 11
BAB III METODOLOGI PENELITIAN 19
3.1 Alur Pengerjaan Tugas Akhir 19
3.2 Uraian Metodologi 20
3.2.1 Pengambilan Data 20
3.2.2 Langkah-langkah *Preprocessing* 22
3.2.3 Ekstraksi Fitur 25
3.2.4 Klasifikasi Naïve Bayes 25
3.2.5 Evaluasi Model 25
3.2.6 *User interface* 25
BAB IV HASIL DAN PEMBAHASAN 26
4.1 Pengambilan Data 26
4.2 *Preprocessing* 27
4.3 Ekstraksi Fitur 29
4.4 Implementasi Naïve Bayes 31
4.5 Evaluasi Model 33
4.6 *User interface* 37
BAB V KESIMPULAN DAN SARAN 40
5.1 Kesimpulan 40
5.2 Saran 40

DAFTAR PUSTAKA.....41
LAMPIRAN 43

DAFTAR TABEL

Tabel 2. 1 *Confusion matrix*.....10
Tabel 2. 2 Penelitian Terdahulu 11
Tabel 3. 1 Contoh Data Hasil *Labelling* 21
Tabel 3. 2 Proses *Cleaning* 22
Tabel 3. 3 Proses *Remove stopword* 22
Tabel 3. 4 Proses *Tokenization* 23
Tabel 3. 5 Proses *Stemming* 23
Tabel 3. 6 Contoh Penerapan N-Gram.....24
Tabel 4. 1 Hasil Akurasi Model.....32
Tabel 4. 2 Hasil *Confusion matrix* n=1.....34
Tabel 4. 3 Hasil *Confusion matrix* n=2.....35
Tabel 4. 4 Hasil *Confusion matrix* n=3.....35
Tabel 4. 5 Hasil *Confusion matrix* n=4.....35
Tabel 4. 6 Hasil *Confusion matrix* n=5.....36
Tabel 4. 7 Hasil Evaluasi Model.....36

DAFTAR GAMBAR

Gambar 2. 1 Diagram Konsep *Crowdsourcing* 5
Gambar 2. 2 Perbandingan Tingkat Akurasi dan Presisi 11
Gambar 3. 1 Alur Pengerjaan 19
Gambar 3. 2 Logo Mozilla Thunderbird 21
Gambar 3. 3 Hasil Pengumpulan Data Surel 21
Gambar 4. 1 Ekstensi ImportExportsTools NG 26
Gambar 4. 2 Hasil *Labelling* Data Surel 27
Gambar 4. 3 Kode Program Proses *Cleaning* 28
Gambar 4. 4 Instalasi *library nltk* 28
Gambar 4. 5 Import *library nltk* 28
Gambar 4. 6 Kode Program Proses *Remove stopword* 28
Gambar 4. 7 Instalasi *Library Sastrawi* 29
Gambar 4. 8 Import *library Sastrawi* 29
Gambar 4. 9 Kode Program Proses *Stemming* 29
Gambar 4. 10 Kode Program Proses *Tokenization* 30
Gambar 4. 11 Data Hasil *preprocessing* 30
Gambar 4. 12 Kode Program Pembagian Data 31
Gambar 4. 13 Implementasi Ekstraksi Fitur N-Gram 32
Gambar 4. 14 Perubahan Data Teks 32
Gambar 4. 15 Kode Program Klasifikasi Naïve Bayes 33
Gambar 4. 16 Grafik akurasi model 34
Gambar 4. 17 Data Salah Prediksi 37
Gambar 4. 18 Kode Program *User Interface* 38
Gambar 4. 19 Hasil *User Interface* 39

BAB I
PENDAHULUAN

1.1 Latar Belakang

Surel merupakan sarana komunikasi dalam jaringan internal maupun internet untuk pertukaran informasi. Surel masih digunakan hingga saat ini karena kemudahan dalam hal penggunaanya. Saat ini, selain digunakan untuk komunikasi surel juga digunakan untuk kebutuhan otentikasi aplikasi dan sinkronisasi media sosial seperti Instagram, Facebook dan Twitter. Berdasarkan penelitian yang dilakukan Radicati group, jumlah akun surel tahun 2012 diperkirakan sebanyak 3,3 miliar akun. Dengan rincian 75% pemilik akun adalah pribadi atau perseorangan, dan sisanya sebanyak 25% dipergunakan oleh perusahaan (Pratiwi & Ulama, 2016).

Penggunaan surel yang tinggi bisa berdampak positif dan berdampak negatif karena tidak semua orang menggunakan surel dengan baik dan bahkan ada banyak sekali penyalahgunaan surel sehingga berpotensi merugikan pengguna surel lainnya. Surel yang disalahgunakan ini disebut sebagai *spam* atau surel sampah yang mana memiliki konten tentang iklan, penipuan, ancaman dan virus.

Surel *spam* yang beredar di kalangan pengguna sebenarnya memiliki pola tertentu hanya saja banyak sekali pengguna awam tidak banyak mengetahui. Biasanya kasus yang banyak terjadi adalah surel *spam* berjenis iklan yang memenuhi kotak masuk surel korban padahal surel tersebut tidak diinginkan. *Spam* dapat menyebabkan ketidakefisienan *bandwidth* karena merupakan kapasitas dari sebuah jaringan agar dapat dilewati oleh paket data (Sudibyo et al., 2018). Bagi banyak orang hal ini sangat mengganggu sehingga dibutuhkan penanganan mengatasi surel *spam* ini.

Permasalahan ini dapat diminimalisir dengan membuat sebuah model anti *spam* yang bertujuan untuk mengklasifikasikan surel dan memberikan informasi terhadap pengguna surel apabila terdapat pesan yang diprediksi sebagai pesan *spam* (Hayuningtyas, 2017). Salah satu metode menciptakan anti *spam* adalah dengan metode Naive Bayes untuk mengklasifikasikan surel *spam* dan *non spam*. Penelitian menggunakan metode Naïve Bayes sebenarnya telah banyak dilakukan untuk mengklasifikasikan surel *spam* berbahasa Inggris namun penelitian untuk surel yang berbahasa Indonesia masih jarang dilakukan.

Oleh karena itu, penelitian ini akan melakukan klasifikasi surel *spam* dan *non-spam* berbahasa Indonesia menggunakan metode Naïve Bayes.

1.2 Rumusan Masalah

Berdasarkan latar belakang masalah yang telah dibahas, maka rumusan masalah yang menjadi fokus utama dalam penelitian ini yaitu bagaimana cara mengklasifikasikan surel *spam* dan *non spam* berbahasa Indonesia menggunakan metode Naïve Bayes?

1.3 Tujuan Penelitian

Adapun tujuan dari penelitian ini dilakukan adalah untuk mengetahui bagaimana cara identifikasi surel *spam/non-spam* dengan metode Naïve Bayes.

1.4 Batasan Masalah

Untuk mendapatkan gambaran yang lebih jelas, diberikan batasan sebagai berikut:

a. Dataset surel yang digunakan hanya berbahasa Indonesia.
b. Metode klasifikasi yang digunakan adalah Naive Bayes.

1.5 Manfaat Penelitian

Manfaat dari penelitian ini yaitu mengetahui performa algoritma Naïve Bayes dalam melakukan klasifikasi surel *spam* dan *non spam* berbahasa Indonesia.

1.6 Sistematika Penulisan

Sistematika penulisan memiliki tujuan untuk mempermudah para pembaca dalam hal memahami isi dari laporan penelitian. Berikut merupakan gambaran sistematika penulisan laporan penelitian:

BAB I PENDAHULUAN

Bagian pendahuluan berisi pembahasan tentang latar belakang penelitian sehingga dapat diketahui sebab penelitian ini dilakukan dan dilanjutkan dengan rumusan masalah, tujuan penelitian, batasan masalah, manfaat penelitian,

BAB II LANDASAN TEORI

Berisi tentang teori-teori dan penelitian terkait yang berkaitan dengan penelitian yang dilakukan dan juga teori pendukung untuk menunjang dalam melakukan penelitian ini.

BAB III METODOLOGI PENELITIAN

Bagian ini membahas tentang tahapan yang dilakukan dalam proses untuk mendapatkan output penelitian yang diinginkan.

BAB IV HASIL DAN PEMBAHASAN

Bagian ini menjelaskan tentang hasil yang didapatkan dari pengolahan data, pengujian, serta menjelaskan kelebihan dan kekurangan dari output yang dihasilkan.

BAB V PENUTUP

Berisi seluruh rangkuman dari hasil penelitian sehingga terdapat kesimpulan di dalamnya dan berisi saran yang untuk pengembangan penelitian yang lebih baik selanjutnya.

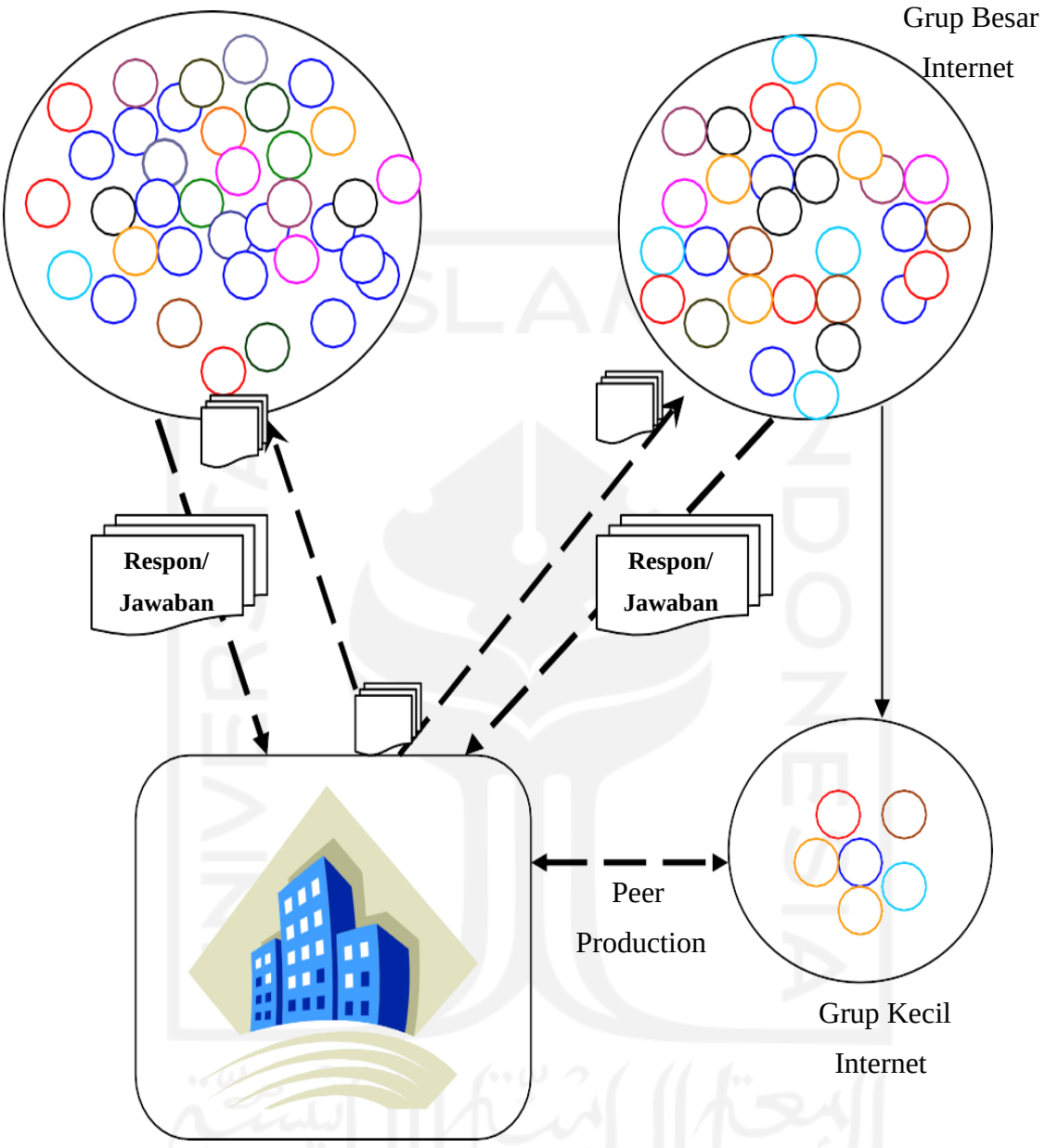
BAB II
LANDASAN TEORI

21 Crowdsourcing

Crowdsourcing merupakan suatu aktivitas yang dilakukan untuk mendapatkan sebuah ide, data ataupun informasi untuk menyelesaikan masalah yang kompleks dengan tidak memandang latar belakang pendidikan, kewarganegaraan, agama, amatir maupun profesional. setiap individu diperbolehkan untuk ikut berpartisipasi dengan pengetahuan dan pengalaman sehingga dalam permasalahan yang ada dapat ditangani secara cepat, tepat, dan hemat biaya. Pengguna nantinya akan mendapatkan kepuasan dari hasil yang telah didapat baik itu ekonomi, pengakuan sosial, maupun pengembangan keterampilan individu. *Crowdsourcing* menjadi hal yang dimanfaatkan oleh berbagai pihak baik individu, perusahaan dan, institusi yang terlibat seperti digambarkan pada Gambar 2. 1. Permasalahan ataupun kekurangan yang ada akan semakin diperkecil atau direduksi, seperti halnya masalah interoperlabilitas, reliabilitas, dan lain sebagainya. *Crowdsourcing* tetap menjadi tendensi untuk beberapa tahun atau decade ke depan dengan pengembangan dan penambahan fitur baru seperti konsep semantic web, data mining dan information integration yang akan saling mendukung konsep *crowdsourcing* itu sendiri.

Dalam metode *crowdsourcing* juga memiliki beberapa kekurangan diantaranya (Andriansyah et al., 2016):

1. Lisensi: hal ini yang menjadi perhatian situs atau pihak-pihak yang menampung hasil kiriman produksi individu. Lisensi kadang bermasalah, dimana pengawasan menjadi semakin terlalu luas lingkup dan daya jangkaunya.
2. Keamanan: dengan semakin terbuka dan cepatnya penyebaran informasi, segal sesuatu yang terbuka akan lebih rawan untuk diasupi atau disisipi oleh seseorang atau sesuatu yang melanggar batas-batas kewajaran, seperti privasi atau keamanan itu sendiri.
3. Kehandalan: hal tersebut menjadi perdebatan suatu contoh dalam wikipedia, definisi suatu kata tingkat keakurasian dan nilai ilmiahnya masih menjadi pertanyaan dan perdebatan, karena siapapun dapat mengakses sistem tersebut, walaupun sudah ada tim yang mencoba mengatasi masalah tersebut, dan berujung pada level kepercayaan dan keandalan akan suatu konsep dan sistem itu sendiri.



Gambar 2. 1 Diagram Konsep *Crowdsourcing*
Sumber: (Andriansyah et al., 2016)

22 Surel *Spam*

Spam atau *stupid pointless annoying messages* merupakan serangan pesan yang dikirimkan ke sejumlah pengguna layanan pesan yang tidak secara khusus meminta pesan tersebut. *Spam* juga dapat didefinisikan sebagai pengiriman pesan secara berulang-ulang. Berikut merupakan tipe-tipe surel *spam* (Hayuningtyas, 2017):

a. Iklan: digunakan untuk mempromosikan suatu barang atau layanan yang dimiliki suatu perusahaan maupun individu perorangan.
b. Phising: menyamar sebagai perusahaan besar/lembaga terpercaya untuk memikat para korban untuk mengunjungi situs web palsu yang tertera dalam pesan dan mengambil data pribadi korban.
c. *Malware*: memperdaya korban dengan mengirimkan sebuah file yang berisikan sebuah virus malware.
d. *Scam*: upaya penyamaran yang dilakukan untuk mendapatkan simpati korban sehingga bisa mendapatkan sesuatu hal yang berharga seperti data maupun uang.

Perbedaan *spam* dan *non spam* dapat dilihat dari struktur surel sebagai berikut:

a. Subject

Merupakan judul topik yang mewakili isi surel biasanya dalam surel *spam* terdapat kata-kata "Ada Diskon" yang sering dijumpai pada korban yang terkena serangan surel *spam*.

b. Body

Merupakan inti dari pesan surel yang diberikan dan isi surel *spam* sangat mudah dikenali dengan melihat kata-kata yang dikirimkan oleh pengirim.

23 Machine Learning

Machine Learning adalah ilmu dalam membuat sebuah sistem automasi belajar mandiri tanpa harus diprogram kembali oleh manusia. *Machine Learning* merupakan disiplin ilmu dalam kecerdasan buatan atau biasa disebut dengan Artificial Intelligent. *Machine learning* membutuhkan sejumlah data latih untuk dapat belajar kemudian hasil dari belajar tersebut akan diuji dengan data tipe yang sama ataupun berbeda. Dalam pembelajaran di *machine learning* terdapat beberapa jenis, yaitu (Febriyanti, 2018):

1. Supervised Learning

Pada jenis ini *learning* dalam pembuatan model menggunakan input berupa data yang telah diberi label. Setelah itu dites untuk memberikan prediksi terhadap data yang tidak diberi label.

2. Unsupervised Learning

Pada jenis ini *learning* tidak diberikan label dan setelah itu pengelompokan data berdasarkan karakteristik-karakteristik yang ditemukan.

3. Reinforcement Learning

Penggunaan jenis ini fase *learning* dan tes dicampurkan sehingga mesin akan melatih dirinya secara aktif terus menerus melakukan *learning* berdasarkan lingkungan.

24 Text mining

Text mining adalah proses yang dilakukan untuk menambang data dengan format teks. *Text mining* memiliki tujuan untuk mengambil kata-kata dan mendapatkan sebuah informasi sehingga dari hasil yang didapat bisa dilakukan sebuah analisis yang memiliki nilai untuk kepentingan tertentu. Terdapat beberapa tahapan proses dalam implementasi *text mining* yaitu *text preprocessing, text transformation, feature selection,* dan *pattern discovery* (Imron, 2019).

A. Text

Merupakan permasalahan pertama yang dalam pemrosesan *text mining* adalah jumlah data teks yang sangat banyak/besar, data *noise*, struktur yang tidak beraturan. hal ini yang menyebabkan ketidakakuratan dalam pemrosesan data teks nantinya.

B. Text Preprocessing

Merupakan proses tahapan awal yang dilakukan dalam penambangan kata untuk mempersiapkan data menjadi lebih mudah untuk diprediksi/klasifikasikan. Berikut beberapa contoh dalam *pre-processing*:

1. *Case Folding*, merupakan tahapan proses yang dilakukan untuk mengubah semua huruf dalam data yang dimasukkan menjadi huruf kecil.
2. *Tokenizing*, merupakan tahapan pemotongan kata berdasarkan tiap kata yang menyusunnya.
3. *Stop-Word Removal*, merupakan proses menghilangkan kata-kata yang tidak memiliki banyak kontribusi dalam data.

C. Text Transformation

Merupakan transformasi teks dan juga sekaligus mengubah kata-kata menjadi bentuk dasarnya sehingga terdapat pengurangan dimensi kata dalam data dokumen.

D. Feature Selection

Merupakan tahap lanjut dari pengurangan dimensi pada transformasi teks sebelumnya.

E. Pattern Discovery

Merupakan tahapan dalam penemuan pola dan pengetahuan yang ada dalam dokumen teks yang ada.

25 N-Gram

N-Gram merupakan model probabilistik yang dikembangkan untuk memprediksi urutan item selanjutnya pada item yang berurutan. Item dapat berupa karakter/huruf, kata dan lain sebagainya. Penggunaan N-Gram pada item kata digunakan untuk mengambil potongan kata berdasarkan nilai n yang ditentukan (Sugianto et al., 2013). Berikut merupakan contoh penggunaan N-gram pada kalimat "Saya sedang membaca jurnal penelitian tersebut" dapat dituliskan dalam metode N-Gram sebagai berikut:

- Unigram (n=1): Saya, sedang, membaca, penelitian, jurnal, tersebut.
- Bigram (n=2): Saya sedang, membaca jurnal, penelitian tersebut.
- Trigram (n=3): Saya sedang membaca, jurnal penelitian tersebut. dan seterusnya.

Metode N-Gram juga memiliki keunggulan yaitu tidak sensitif terhadap kesalahan penulisan yang ada pada suatu data.

26 Klasifikasi

Klasifikasi adalah proses dalam menemukan sekumpulan model dengan membedakan kelas-kelas data dengan tujuan memprediksikan kelas yang belum diketahui kelasnya (*supervised learning*). Dalam klasifikasi terdiri dari dua proses yaitu *learning* dan tes. Pada proses *learning* sebagian data yang telah dikumpulkan dan diberikan label kelas digunakan sebagai data *training* untuk membentuk model. Selanjutnya pada proses tes model yang telah terbentuk diuji dengan sisa data yang telah dikumpulkan tanpa memberikan label kelas untuk mengetahui akurasi model tersebut.

27 Naïve Bayes

Merupakan sebuah algoritma yang dikemukakan oleh ilmuwan Inggris Thomas Bayes untuk pengklasifikasian sederhana yang menghitung sekumpulan probabilitas dengan menjumlahkan antara frekuensi dan kombinasi nilai dari dataset yang diberikan. Keuntungan dalam penggunaan algoritma Naïve Bayes adalah metode ini hanya membutuhkan *training* yang sedikit. Berikut merupakan persamaan Naïve Bayes ditunjukkan pada persamaan (2.1) (Hayuningtyas, 2017).

$$P(H|X) = \frac{P(X|H)P(H)}{P(X)} \qquad \textbf{(}2.1\textbf{)}$$

X : Data dengan kelas yang belum diketahui.

H : Hipotesis data merupakan suatu kelas spesifik

P(H|X): Probabilitas hipotesis H berdasarkan kondisi X (posterior probabilitas)

P(H): Probabilitas hipotesis H (prior probabilitas)

P(X|H): Probabilitas X berdasarkan kondisi pada hipotesis H

P(X): Probabilitas X

Dalam Bayes terdapat aturan P(H1|x) < P(H2|x), maka x diklasifikasikan sebagai H2. Pernyataan P(H1|x) mengindikasikan probabilitas hipotesis H1 berdasarkan kondisi x terjadi, begitu pula dengan H2. Sebenarnya dapat klasifikasi dari x sesuai dengan probabilitas terbesar di antara probabilitas x terhadap semua kelas.

2.8 Performance Evaluation Measure

PEM memiliki tujuan untuk mengevaluasi model yang telah dibuat dan menampilkan kembali prediksi kondisi sebenarnya (aktual) dari data yang dihasilkan oleh algoritma yang digunakan. Banyak perhitungan untuk mendapat hasil dari nilai PEM yaitu (Imron, 2019):

- *Precision*

  Menghitung tingkat kepastian atau jumlah data *testing* yang diklasifikasikan dengan benar oleh model klasifikasi yang dibangun.

  Rumus *Precision* (pre) ditunjukkan pada persamaaan (2.2):

$$pre = \frac{TP}{FP + TP} \qquad (2.2)$$

- *Accuracy*

  Pembanding antara informasi yang ditampilkan oleh sistem dengan benar oleh keseluruhan informasi.

  Rumus *Accuracy* ditunjukkan pada persamaan (2.3):

$$acc = \frac{TN}{FN + TP + TN + TP} \qquad (2.3)$$

- *Recall*

  Recall menghitung sensitivitas atau rasio dari setiap label yang diklasifikasikan dengan benar terhadap data yang salah diklasifikasikan ke label lainnya.

  Rumus *Recall* ditunjukkan pada persamaan (2.4):

$$rec = \frac{TP}{FN + TP} \qquad (2.4)$$

Performance Evaluation Measure ditunjukkan dalam *confusion matrix*, yaitu berbentuk tabel yang berupa hasil pengujian model yang telah dilakukan perbandingan dengan *dataset*, yang berisi dari kelas *true* dan *false*.

Tabel 2. 1 *Confusion matrix*

	Predicted Class	
True Class	***Positive***	***Negative***
Positive	TP	FN
Negative	FP	TN

Keterangan:

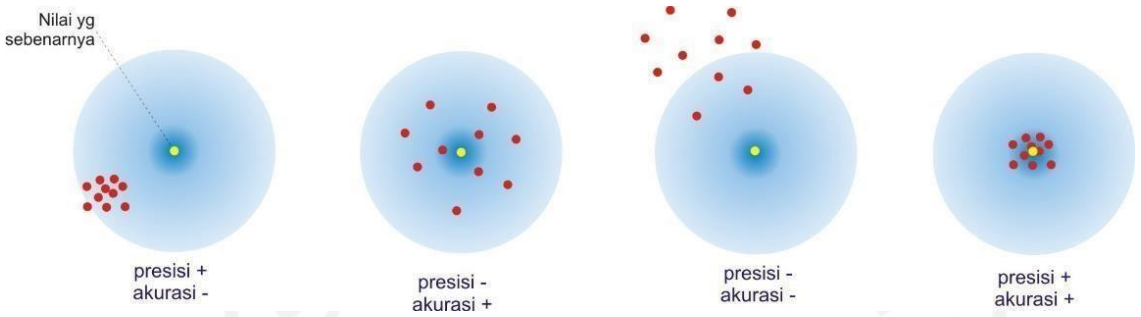
TP (*true positive*): data bernilai positif yang diprediksi benar sebagai positif

TN (*true negative*): data bernilai negatif yang diprediksi benar sebagai negatif

FP (*false positive*): data bernilai negatif yang diprediksi salah sebagai positif

FN (*false negative*): data bernilai positif yang diprediksi salah sebagai negatif

Berikut merupakan contoh persebaran data dengan perhitungan akurasi dan presisi ditunjukkan pada Gambar 2. 2 Perbandingan Tingkat Akurasi dan Presisi.



Gambar 2. 2 Perbandingan Tingkat Akurasi dan Presisi

Sumber: (Imron, 2019)

29 Penelitian Terkait

Dalam penelitian ini terdapat penelitian terdahulu yang telah dilakukan oleh peneliti lain yang mirip dan digunakan sebagai acuan pada penelitian ini. Berikut beberapa penelitian terdahulu dapat dilihat pada Tabel 2. 2

Tabel 2. 2 Penelitian Terdahulu

No	Judul	Dataset	Bahasa	Algoritma	Hasil
1.	Klasifikasi Email *Spam* dengan Menggunakan Metode Support Vector *Machine* dan k-Nearest Neighbor (Pratiwi & Ulama, 2016)	Cs mining Group	Inggris	SVM, K-NN	Metode KNN pada k = 3,5,7,9,11 dengan 10-fold cv menghasilkan ketepatan klasifikasi terbaik pada saat k=3 dengan hasil ketepatan Klasifikasi sebesar 92.28% dengan error 7.72% sedangkan kombinasi metode SVM menggunakan

					kernel linier dan RBF dengan 10-fold cv menghasilkan ketepatan klasifikasi terbaik dengan menggunakan SVM linier dengan ketepatan klasifikasi yang diberikan sebesar 96.6% dengan error 3.4% sehingga disimpulkan metode SVM lebih baik dibanding metode KNN
2.	Unsupervised Feature *Learning* for *Spam* Email Filtering (Diale et al., 2019)	Enron Dataset	Inggris	SVM, Random Forest,C45	Menghasilkan klasifikasi yang baik dengan konsistensi yang lebih baik dibandingkan dengan representasi fitur state-of-art pendekatan dalam tugas pemfilteran *spam* dengan kesamaan kosinus dan Autoencoder, telah terbukti andal berdasarkan kumpulan data yang menjadi

					pertimbangan untuk evaluasi kinerja. SVM 93%, RF 95%,C45 89%
3.	Antlion optimization and boosting classifier for *spam* email detection (Naem et al., 2018)	CSDMC2010 dataset	-	ALO, KNN, SVM, Bagging	ALO-Boosting mencapai nilai terbaik 98.91% akurasi, sensitivitas 99.96%, Spesifisitas 97.83%, 99.88% presisi, 98,89% G-mean, dan 99,91% F-measure dan dengan jumlah fitur yang dipilih paling sedikit.
4.	Klasifikasi Algoritma Naive Bayes dan SVM berbasis PSO dalam memprediksi *spam* email pada hotline sapto (Hengki & Wahyudi, 2020)	Hotline Sapto	-	Naive Bayes,SVM	Support Vector *Machine*s (SVM) berbasis Particle Swarm Optimization (PSO) didapat nilai accuracy 85.25% dengan nilai AUC 0.892. Untuk pengujian Support Vector *Machine*s (SVM) nilai accuracy 84.59% dengan nilai AUC

					0.792. Sedangkan pengujian model Naïve Bayes berbasis Particle Swarm Optimization (PSO) didapat nilai accuracy 81.24% dengan nilai AUC 0.892 dan untuk pengujian model Naïve Bayes didapat accuracy 80.59% dengan nilai AUC 0.942.
5.	Aplikasi Filtering of *Spam* Email Menggunakan Naïve Bayes (Hayuningtyas, 2017)	Uci *Machine Learning* Repository	Inggris	Naive Bayes	Dari hasil penelitian menggunakan *confusion matrix* bahwa pengujian yang dilakukan Naïve Bayes sudah baik karena memiliki akurasi 75,9%.
6.	Analisis *Spam* menggunakan Naive Bayes (Juang, 2016)	Ling *Spam* Dataset	Inggris	Naive Bayes	Dari hasil penelitian yang sudah dilakukan bahwa algoritma naïve bayes dapat mengklasifikasikan suatu pesan ke dalam

					dua kelas yaitu *spam* dan *non spam*. Dari pengklasifikasian tersebut sangat dipengaruhi oleh proses *training*.
7.	Comparative Analysis of Classification Algorithms for Email *Spam* Detection (Abdulhamid et al., 2018)	Uci *Machine Learning* Repository	Inggris	Bayesian Logistic Regression, Hidden Naïve Bayes, Radial BasisFunction (RBF) Network, Voted Perceptron, Lazy Bayesian Rule, Logit Boost, Rotation Forest, NNge, Logistic Model Tree, REP Tree, Naïve Bayes, Multilayer Perceptron, Random Tree and J48.	Hasil penelitian, Akurasi tertinggi sebesar 0,942 didapatkan 10-fold cross validation diterapkan pada Rotasi Algoritma forest dan terendah diperoleh 0,891 saat Pembagian 66% digunakan dengan algoritma REPTree

8.	Klasifikasi Seleksi Atribut Pada Serangan *Spam* Menggunakan Metode Algoritma Decision Tree (Sudibyo et al., 2018)	Uci *Machine Learning* Repository	Inggris	Decision Tree	Penelitian tentang serangan *spam* didapat dari dataset *spam* sebanyak 4601 record yang terdiri 1813 record dianggap *spam* dan 278 data bukan *spam* dengan atribut awal sebanyak 57 atribut dengan 1 atribut class, pada eksperimen yang dilakukan menggunakan select attribute dengan decision tree menjadi 15 atribut dengan 1 atribut class dilakukan 3 percobaan pengujian dengan persentase atribut 30%, 50% dan 70% select atribut didapat hasil fitur select atribut sebesar 70% didapat hasil lebih baik dari 30% ataupun 50% dengan nilai accuracy sebesar 92.469%.

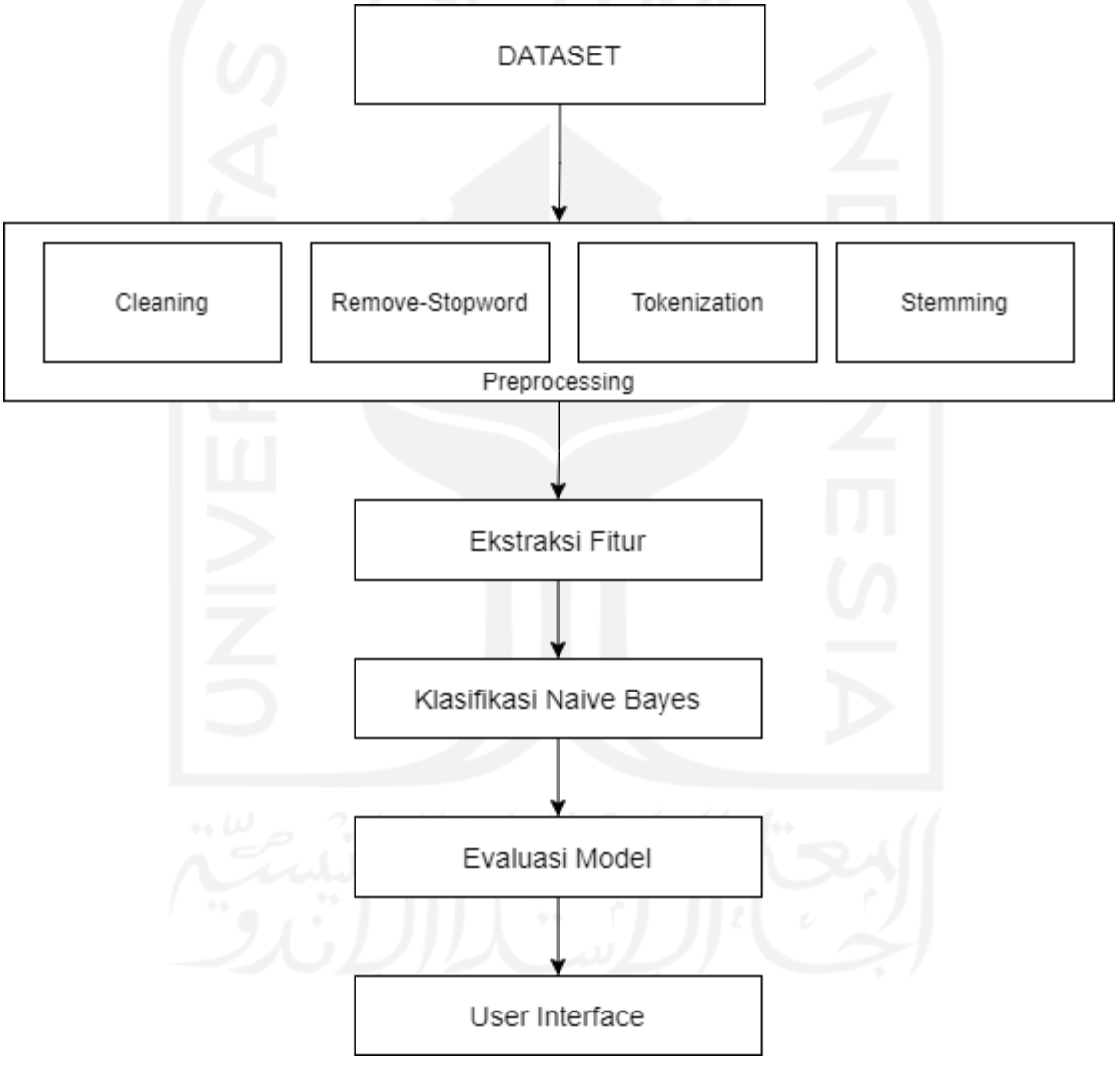
9.	*Spam* Filtering dengan Metode Pos Tagger dan Klasifikasi Naive Bayes (Chandra et al., 2016)	*Spam*Assasin	Inggris	Pos Tagger, Naive Bayes	Hasil Penelitian dengan menggunakan metode pos Tagger dan Klasifikasi Naive Bayes memiliki akurasi terendah 78.72% dan nilai akurasi tertinggi 84.30%
10.	Arabic *Spam* Filtering using Bayesian Model (Al-Alwani & Beseiso, 2013)	Arabic Dataset	Arab	Bayesian	Keakuratan *spam* filtering berbahasa arab yang mereka bangun sangat baik mencapai 80%
11.	Indonesian language email spam detection using N-gram and Naïve Bayes algorithm (Vernanda et al., 2020)	-	Indonesia	Naïve Bayes	Dari hasil eksperimen, bisa jadi menyimpulkan bahwa nilai akurasi berkisar antara 0,615 hingga 0,94, nilai presisi berkisar antara 0,566 hingga 0,924, nilai penarikan berkisar antara 0,96 hingga 1, dan nilai f-measure berkisar antara 0,721 hingga 0,942.

Berdasarkan penelitian terdahulu yang ditunjukkan pada Tabel 2. 2, dalam pengumpulan dataset *spam* dan *non spam* diperoleh dari berbagai sumber dan dalam penelitian yang ditemukan paling banyak digunakan adalah dataset bahasa inggris. *Preprocessing* juga dilakukan pada setiap penelitian yang ada untuk mempersiapkan data agar proses klasifikasi mendapatkan model yang maksimal. Algoritma yang ada pada penelitian sebelumnya juga memiliki kelebihan dan kekurangan masing-masing dan akurasi tertinggi yang ditemukan ditunjukkan oleh algoritma ALO dengan nilai akurasi sebesar 98,91%. Dengan adanya penelitian terkait ini maka penulis mendapatkan gambaran dan referensi untuk membangun sebuah model deteksi surel menggunakan dataset berbahasa Indonesia dengan algoritma Naïve Bayes.

BAB III
METODOLOGI PENELITIAN

3.1 Alur Pengerjaan Tugas Akhir

Pengerjaan tugas akhir ini memiliki alur yang berisi gambaran umum terkait penelitian yang dilakukan dari awal pengerjaan hingga akhir pengerjaan tugas akhir. Alur pengerjaan dapat dilihat pada Gambar 3. 1



Gambar 3. 1 Alur Pengerjaan

Berdasarkan diagram alur pengerjaan, terdapat beberapa langkah-langkah pada penelitian ini, yaitu:

1. Dataset

   Langkah pertama dalam pengerjaan tugas akhir adalah dengan mencari data surel *spam* dan *non spam* berbahasa Indonesia menggunakan teknik *crowdsourcing* dan data yang telah dikumpulkan dilakukan *labelling* data untuk menentukan jenis surel *spam* dan *non spam*.

2. *Preprocessing*

   Langkah kedua, dilakukan *preprocessing* data yang bertujuan untuk menjadikan data yang belum terstruktur menjadi data yang terstruktur. Pada tahap *preprocessing* terdiri dari empat tahapan, yaitu *Cleaning*, *Remove stopword*, *Tokenization* dan *Stemming*.

3. Ekstraksi Fitur

   Langkah ketiga, setelah data teks menjadi lebih terstruktur selanjutnya melakukan ekstraksi fitur menggunakan n-gram.

4. Klasifikasi Naive Bayes

   Langkah keempat, melakukan proses *training* menggunakan metode Naïve Bayes. Proses pengklasifikasian akan menghasilkan kelas *spam* dan *non spam*.

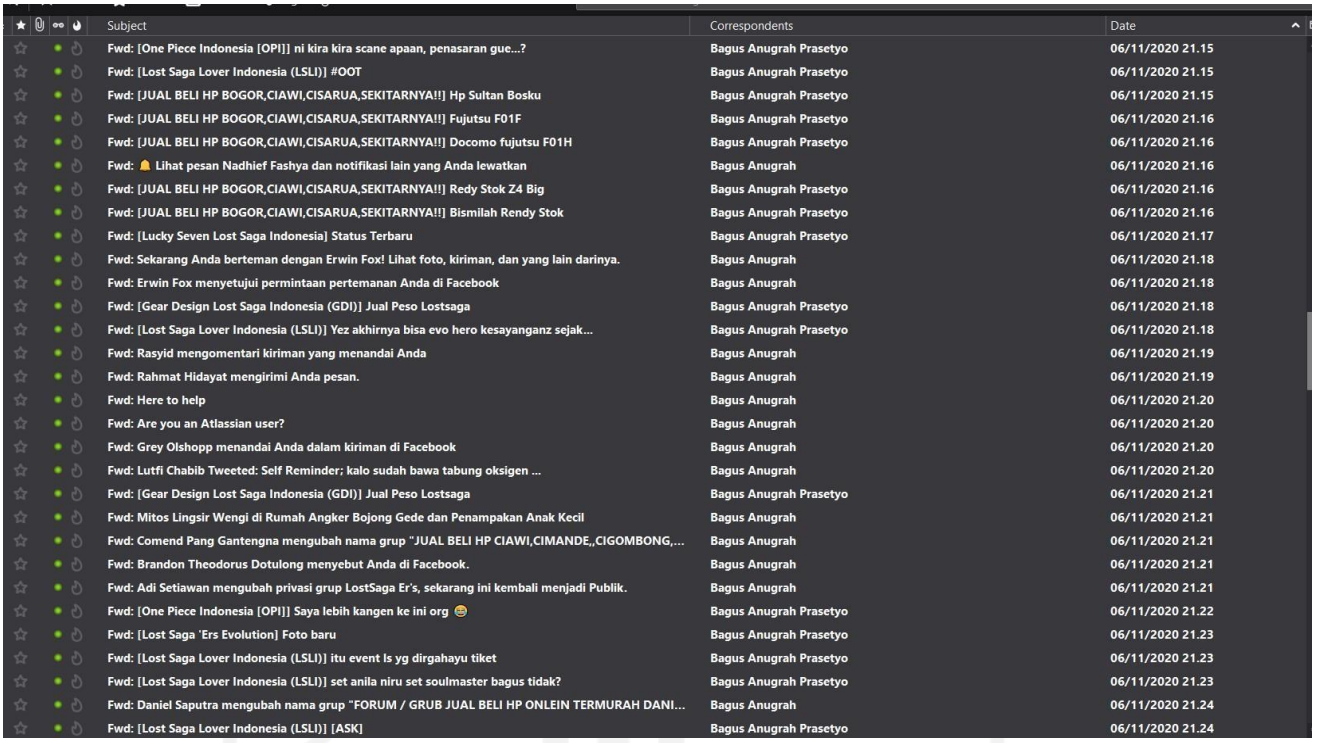
5. Evaluasi Model

Langkah kelima, melakukan proses evaluasi model untuk mengetahui seberapa baik akurasi model dalam memprediksi, dengan melihatnya pada tabel *confusion matrix.*

3.2 Uraian Metodologi

3.2.1 Pengambilan Data

Data pada penelitian kali ini menggunakan data surel *spam* dan *non spam* berbahasa Indonesia yang didapatkan dari pengumpulan data menggunakan metode *crowdsourcing* pada setiap individu yang terindikasi terkena serangan *spam* dan yang mendapatkan *non spam*. Individu yang ikut berpartisipasi pada pengumpulan data melakukan *forward* surel kepada peneliti. Peneliti menggunakan aplikasi Mozilla Thunderbird untuk memudahkan dalam pengumpulan data surel yang ditunjukkan pada Gambar 3. 2.



Subject	Correspondents	Date
Fwd: [One Piece Indonesia [OPI]] ni kira kira scane apaan, penasaran gue...?	Bagus Anugrah Prasetyo	06/11/2020 21.15
Fwd: [Lost Saga Lover Indonesia (LSLI)] #OOT	Bagus Anugrah Prasetyo	06/11/2020 21.15
Fwd: [JUAL BELI HP BOGOR,CIAWI,CISARUA,SEKITARNYA!!] Hp Sultan Bosku	Bagus Anugrah Prasetyo	06/11/2020 21.15
Fwd: [JUAL BELI HP BOGOR,CIAWI,CISARUA,SEKITARNYA!!] Fujutsu F01F	Bagus Anugrah Prasetyo	06/11/2020 21.16
Fwd: [JUAL BELI HP BOGOR,CIAWI,CISARUA,SEKITARNYA!!] Docomo fujutsu F01H	Bagus Anugrah Prasetyo	06/11/2020 21.16
Fwd: 🔔 Lihat pesan Nadhief Fashya dan notifikasi lain yang Anda lewatkan	Bagus Anugrah	06/11/2020 21.16
Fwd: [JUAL BELI HP BOGOR,CIAWI,CISARUA,SEKITARNYA!!] Redy Stok Z4 Big	Bagus Anugrah Prasetyo	06/11/2020 21.16
Fwd: [JUAL BELI HP BOGOR,CIAWI,CISARUA,SEKITARNYA!!] Bismilah Rendy Stok	Bagus Anugrah Prasetyo	06/11/2020 21.16
Fwd: [Lucky Seven Lost Saga Indonesia] Status Terbaru	Bagus Anugrah Prasetyo	06/11/2020 21.17
Fwd: Sekarang Anda berteman dengan Erwin Fox! Lihat foto, kiriman, dan yang lain darinya.	Bagus Anugrah	06/11/2020 21.18
Fwd: Erwin Fox menyetujui permintaan pertemanan Anda di Facebook	Bagus Anugrah	06/11/2020 21.18
Fwd: [Gear Design Lost Saga Indonesia (GDI)] Jual Peso Lostsaga	Bagus Anugrah Prasetyo	06/11/2020 21.18
Fwd: [Lost Saga Lover Indonesia (LSLI)] Yez akhirnya bisa evo hero kesayanganz sejak...	Bagus Anugrah Prasetyo	06/11/2020 21.18
Fwd: Rasyid mengomentari kiriman yang menandai Anda	Bagus Anugrah	06/11/2020 21.19
Fwd: Rahmat Hidayat mengirimi Anda pesan.	Bagus Anugrah	06/11/2020 21.19
Fwd: Here to help	Bagus Anugrah	06/11/2020 21.20
Fwd: Are you an Atlassian user?	Bagus Anugrah	06/11/2020 21.20
Fwd: Grey Olshopp menandai Anda dalam kiriman di Facebook	Bagus Anugrah	06/11/2020 21.20
Fwd: Lutfi Chabib Tweeted: Self Reminder; kalo sudah bawa tabung oksigen ...	Bagus Anugrah	06/11/2020 21.20
Fwd: [Gear Design Lost Saga Indonesia (GDI)] Jual Peso Lostsaga	Bagus Anugrah Prasetyo	06/11/2020 21.21
Fwd: Mitos Lingsir Wengi di Rumah Angker Bojong Gede dan Penampakan Anak Kecil	Bagus Anugrah	06/11/2020 21.21
Fwd: Comend Pang Gantengna mengubah nama grup "JUAL BELI HP CIAWI,CIMANDE,,CIGOMBONG,...	Bagus Anugrah	06/11/2020 21.21
Fwd: Brandon Theodorus Dotulong menyebut Anda di Facebook.	Bagus Anugrah	06/11/2020 21.21
Fwd: Adi Setiawan mengubah privasi grup LostSaga Er's, sekarang ini kembali menjadi Publik.	Bagus Anugrah	06/11/2020 21.21
Fwd: [One Piece Indonesia [OPI]] Saya lebih kangen ke ini org 😂	Bagus Anugrah Prasetyo	06/11/2020 21.22
Fwd: [Lost Saga 'Ers Evolution] Foto baru	Bagus Anugrah Prasetyo	06/11/2020 21.23
Fwd: [Lost Saga Lover Indonesia (LSLI)] itu event ls yg dirgahayu tiket	Bagus Anugrah Prasetyo	06/11/2020 21.23
Fwd: [Lost Saga Lover Indonesia (LSLI)] set anila niru set soulmaster bagus tidak?	Bagus Anugrah Prasetyo	06/11/2020 21.23
Fwd: Daniel Saputra mengubah nama grup "FORUM / GRUB JUAL BELI HP ONLEIN TERMURAH DANI...	Bagus Anugrah	06/11/2020 21.24
Fwd: [Lost Saga Lover Indonesia (LSLI)] [ASK]	Bagus Anugrah Prasetyo	06/11/2020 21.24

Gambar 3. 2 Hasil Pengumpulan Data Surel

Seluruh data hasil pengumpulan disimpan dengan format .txt dan kemudian dilakukan *labelling* data terhadap data yang telah dikumpulkan dengan memisahkan menjadi 2 *class* yaitu *spam* dan *non spam*. Berikut contoh data yang didapat dan hasil *labelling* seperti pada Tabel 3. 1.

Tabel 3. 1 Contoh Data Hasil *Labelling*

Isi Surel	**Label**
Seminggu kemudian, saya sudah menginstal = virus Trojan pada Sistem Operasi semua peranti yang Anda gunakan untuk = mengakses surel.Sama sekali tidak sulit (karena Anda mengikuti tautan = dari kotak masuk surel).	*Spam*
Ini saya lampirkan lagi skripsi lengkapnya pak.Baik, nanti saya sms & surel soalnya hari senin besok baru daftar sidang. Terima Kasih pak.	*Non Spam*
Saya tidak mau Anda mengalaminya karena saya lihat Anda orang baik dan jujur. Jadi, saya tawarkan kesepakatan: Transfer USD 1450 dalam Bitcoin kepada saya dan setelah pembayaran diterima, saya akan segera menghapus bukti itu.	*Spam*
Perkenalkan kak saya Farhan dari mahasiswa DKV Universitas Esa Unggul mau menanyakan kak ada lowongan untuk magang atau tidak ya kak untuk mahasiswa DKV? Terimakasih sebelumnya:)	*Non Spam*

Proses pelabelan data memiliki tujuan untuk mempermudah dalam proses *training* yang akan dilakukan dalam membuat sebuah output model.

3.2.2 Langkah-langkah *Preprocessing*

Tahapan ini memiliki tujuan yaitu untuk membersihkan kata-kata yang tidak memiliki makna dalam bahasa Indonesia dan membuat data menjadi data yang lebih terstruktur. Berikut merupakan urutan tahapan dalam *preprocessing* yang dilakukan:

1. *Cleaning*

Banyaknya *noise* dalam sebuah data *text* membuat data menjadi sangat tidak efektif sehingga diperlukan *cleaning* data. Pada tahapan ini akan dilakukan penghapusan simbol, tanda baca, dan angka. Berikut contoh proses *cleaning* terhadap data dapat dilihat pada Tabel 3. 2

Tabel 3. 2 Proses *Cleaning*

Sebelum	Sesudah
Seminggu kemudian, saya sudah menginstal = virus Trojan pada Sistem Operasi semua peranti yang Anda gunakan untuk = mengakses email.Sama sekali tidak sulit (karena Anda mengikuti tautan = dari kotak masuk email).	seminggu kemudian saya sudah menginstal virus trojan pada sistem operasi semua peranti yang anda gunakan untuk mengakses email sama sekali tidak sulit karena Anda mengikuti tautan dari kotak masuk email
Ini saya lampirkan lagi skripsi lengkapnya pak. Baik, nanti saya sms & email soalnya hari senin besok baru daftar sidang. Terima Kasih pak.	ini saya lampirkan lagi skripsi lengkapnya pak baik nanti saya sms email soalnya hari senin besok baru daftar sidang terima kasih pak

2. *Remove stopwords*

Tahapan ini akan dilakukan penghapusan kata-kata yang kurang bermakna sehingga mendapatkan output yang lebih terstruktur contoh penghapusan kata seperti: dan, atau, saya. Berikut contoh proses *remove stopword* terhadap data dapat dilihat pada Tabel 3. 3.

Tabel 3. 3 Proses *Remove stopword*

Sebelum	Sesudah
seminggu kemudian saya sudah menginstal virus trojan pada sistem operasi semua peranti yang anda gunakan untuk mengakses surel sama sekali tidak sulit karena Anda mengikuti tautan dari kotak masuk email	seminggu kemudian menginstal virus trojan sistem operasi semua peranti gunakan mengakses surel tidak sulit karena mengikuti tautan dari kotak masuk email
ini saya lampirkan lagi skripsi lengkapnya pak baik nanti saya sms email soalnya hari senin besok baru daftar sidang terima kasih pak	lampirkan skripsi lengkapnya pak baik nanti sms email soalnya hari senin besok baru daftar sidang terima kasih pak

3. Tokenization

Tahapan ini bertujuan untuk mengubah kalimat yang panjang menjadi token-token yang lebih kecil sehingga satu token dianggap sebagai suatu bentuk kata atau elemen yang berarti. Berikut contoh proses *tokenization* terhadap data dapat dilihat pada Tabel 3. 4.

Tabel 3. 4 Proses *Tokenization*

Sebelum	**Sesudah**
seminggu kemudian menginstal virus trojan sistem operasi semua peranti gunakan mengakses email tidak sulit karena mengikuti tautan dari kotak masuk email	“seminggu” “kemudian” “menginstal” “virus” “trojan” “sistem” “operasi” “semua” “peranti” “gunakan” “mengakses” “email” “tidak” “sulit” “karena” “mengikuti” “tautan” “dari” “kotak” “masuk” “email”
lampirkan skripsi lengkapnya pak baik nanti sms email soalnya hari senin besok baru daftar sidang terima kasih pak	“lampirkan” “skripsi” “lengkapnya” “pak” “baik” “nanti” “sms” “email” “soalnya” “hari” “senin” “besok” “baru” “daftar” “sidang” “terima” “kasih” “pak”

4. Stemming

Tahapan ini bertujuan untuk mengubah sebuah kata berimbuhan menjadi bentuk kata dasar/asli. Berikut merupakan hasil contoh proses *stemming* terhadap data dapat dilihat pada Tabel 3. 5.

Tabel 3. 5 Proses *Stemming*

Sebelum	**Sesudah**
“seminggu” “kemudian” “menginstal” “virus” “trojan” “sistem” “operasi” “semua” “peranti” “gunakan” “mengakses” “email” “tidak” “sulit” “karena” “mengikuti” “tautan” “dari” “kotak” “masuk” “email”	minggu kemudian instal virus trojan sistem operasi semua peranti guna akses surel tidak sulit karena ikut tautan dari kotak masuk surel
“lampirkan” “skripsi” “lengkapnya” “pak” “baik” “nanti” “sms” “email” “soalnya” “hari” “senin” “besok” “baru” “daftar” “sidang” “terima” “kasih” “pak”	lampir skripsi lengkap pak baik nanti sms surel soal hari senin besok baru daftar sidang terima kasih pak

3.2.3 Ekstraksi Fitur

Setelah data surel melalui tahap *preprocessing*, selanjutnya dilakukan proses ekstraksi fitur menggunakan N-gram bernilai n = 1 hingga n = 5. N-gram merupakan penggabungan kata untuk memprediksi urutan item selanjutnya pada item yang berurutan. Contoh penerapan N-gram dapat dilihat pada Tabel 3. 6.

Tabel 3. 6 Contoh Penerapan N-Gram

1-gram	Saya, juga, janji, akan, me*non*aktifkan, dan, menghapus, semua, software, berbahaya, dari, peranti, anda.
2-gram	Saya juga, juga janji, janji akan, akan me*non*aktifkan, me*non*aktifkan dan, dan menghapus, menghapus semua, semua software, software berbahaya, berbahaya dari, dari peranti, peranti anda.
3-gram	Saya juga janji, juga janji akan, janji akan me*non*aktifkan, akan me*non*aktifkan dan, me*non*aktifkan dan menghapus, dan menghapus semua, menghapus semua software, semua software berbahaya, software berbahaya dari, berbahaya dari peranti, dari peranti anda.
4-gram	Saya juga janji akan, juga janji akan me*non*aktifkan, janji akan me*non*aktifkan dan, akan me*non*aktifkan dan menghapus, me*non*aktifkan dan menghapus semua, dan menghapus semua software, menghapus semua software berbahaya, semua software berbahaya dari, software berbahaya dari peranti, berbahaya dari peranti anda.
5-gram	Saya juga janji akan me*non*aktifkan, juga janji akan me*non*aktifkan dan, janji akan me*non*aktifkan dan menghapus, akan me*non*aktifkan dan menghapus semua, me*non*aktifkan dan menghapus semua software, dan menghapus semua software berbahaya, menghapus semua software berbahaya dari, semua software berbahaya dari peranti, software berbahaya dari peranti anda.

3.2.4 Klasifikasi Naïve Bayes

Metode Naïve Bayes adalah metode yang digunakan untuk mengklasifikasikan data surel untuk mendapatkan prediksi *spam* atau *non spam*. Metode Naïve Bayes dipilih pada penelitian ini karena dapat digunakan untuk pengklasifikasian dengan hasil yang baik, selain itu metode ini digunakan untuk memprediksi suatu kejadian pada masa yang akan datang, dengan cara membandingkan data atau evidence (bukti) yang ada pada masa lampau. Proses klasifikasi akan menggunakan data hasil *preprocessing* dan ekstraksi fitur dengan n-gram yang dilakukan pada tahapan sebelumnya. Data akan dibagi menjadi dua bagian yaitu *training* sebanyak 80% dan data *test* sebanyak 20%. Pada proses *training* model ini menggunakan bantuan *library sklearn* pada bahasa pemrograman *python*.

3.2.5 Evaluasi Model

Evaluasi model dilakukan dengan menghitung nilai *precision*, *recall*, dan *f-score*. *Precision* mengukur tingkat kepastian yang diklasifikasikan dengan benar dapat dihitung dengan rumus persamaan nomor (2.2). *Recall* merupakan rasio prediksi dari data pada setiap label yang diklasifikasikan dengan benar terhadap data yang salah diklasifikasikan ke label lainya dapat dihitung dengan rumus persamaan nomor (2.3). *F-score* merupakan *harmonic mean* antara nilai *precision* dan nilai *recall* dapat dihitung dengan rumus persamaan nomor (2.4).

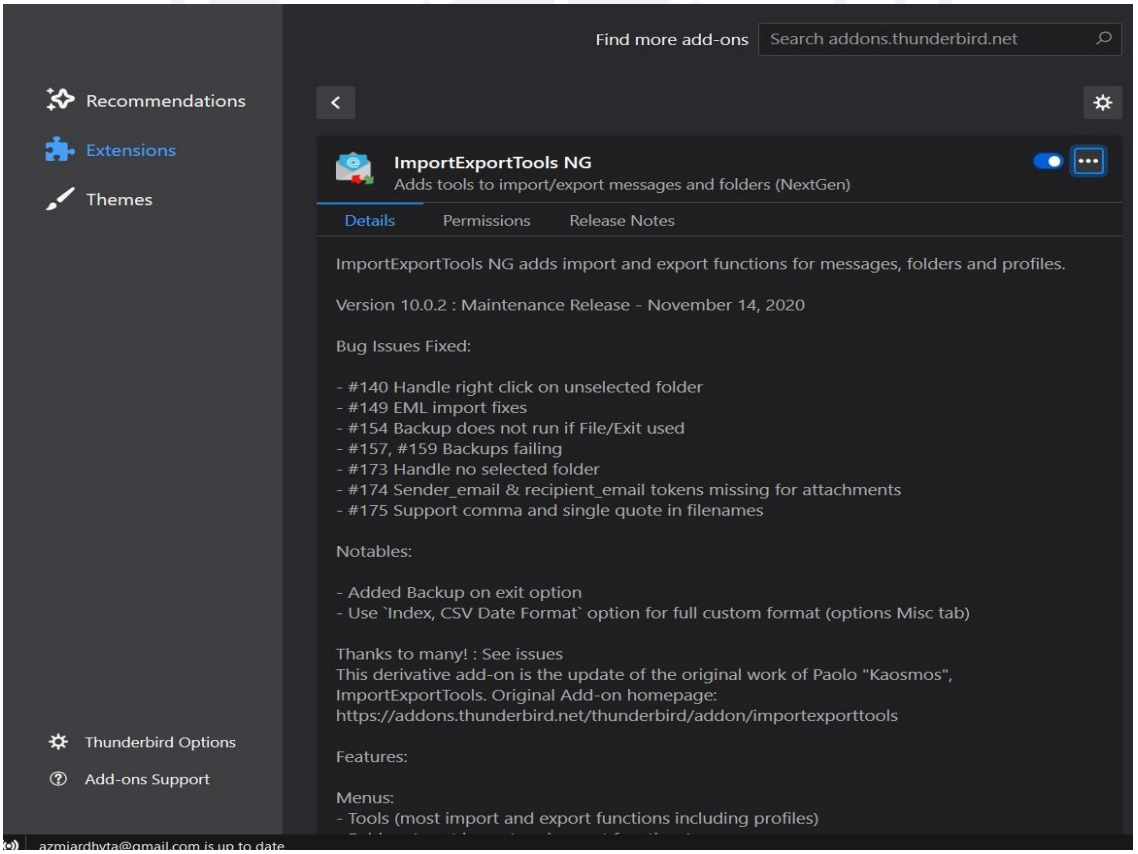
3.2.6 User interface

Membangun *user interface* dilakukan dengan tujuan untuk mempermudah *user* dalam melakukan pengecekan surel baru yang didapat terhadap model yang telah ada dan akan mendapatkan hasil apakah surel baru tersebut termasuk jenis *Spam* atau *non-spam*.

BAB IV
HASIL DAN PEMBAHASAN

41 Pengambilan Data

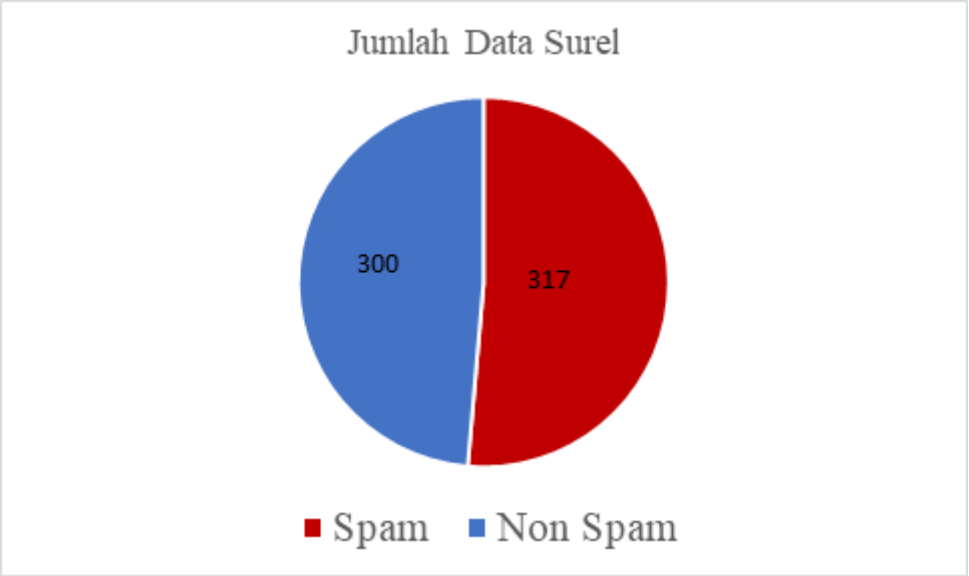
Pengambilan data surel dilaksanakan secara manual menggunakan metode *crowdsourcing* menggunakan bantuan *Mail User Agent* (MUA) Mozilla Thunderbird sebagai tempat pengumpulan surel. Langkah pertama yang dilakukan adalah melakukan instalasi pada perangkat dan memasukkan akun yang akan digunakan sebagai tempat pengumpulan. Proses ekstraksi data surel yang didapat dari setiap individu dilakukan dengan bantuan ekstensi pada Mozilla Thunderbird, yaitu ImportExportTools NG yang ditunjukkan pada Gambar 4. 1 dan data surel disimpan dalam format .txt.



Gambar 4. 1 Ekstensi ImportExportsTools NG

Dari proses pengumpulan data didapatkan sebanyak 617 data surel. Selanjutnya dilakukan proses *labelling* data untuk mengetahui termasuk data surel *spam* atau *non spam*. Proses

labelling menghasilkan persentase rincian data *spam* dan *non spam* yang dapat dilihat pada Gambar 4. 2.



Gambar 4. 2 Hasil *Labelling* Data Surel

4.2 *Preprocessing*

Tahapan ini berisi beberapa proses karena data surel memiliki data yang tidak terstruktur. *Preprocessing* dilaksanakan dengan memakai bantuan *library* yang ada pada bahasa pemrograman *python.* Berikut ini tahapan *preprocessing* yang dilakukan, antara lain sebagai berikut:

1. *Cleaning Data*

   Proses ini dilakukan pembersihan data surel dari tanda baca atau *punctuation*, *hashtag* maupun *mention* dan mengubah semua kata menjadi huruf kecil atau *lowercase.* Berikut merupakan kode program yang diimplementasikan pada tahap *cleaning data* terdapat pada Gambar 4. 3.

```
    def cleaning(str):
        #remove non-ascii
        str = unicodedata.normalize('NFKD', str).encode('ascii',
'ignore').decode('utf-8', 'ignore')
        #remove URLs
        str   =   re.sub(r'(?i)\b((?:https?://|www\d{0,3}[.]|[a-z0-
9.\-]+[.][a-
z]{2,4}/)(?:[^\s()<>]+|\(([^\s()<>]+|(\([^\s()<>]+\)))*\))+(?:\(
([^\s()<>]+|(\([^\s()<>]+\)))*\)|[^\s`!()\[\]{};:\'".,<>?«»“”‘’]
))', '', str)
        #remove punctuations
```

```
        str = re.sub(r'[^\w]|_',' ',str)
        #remove digit from string
        str = re.sub("\S*\d\S*", "", str).strip()
        #remove digit or numbers
        str = re.sub(r"\b\d+\b", " ", str)
        #to lowercase
        str = str.lower()
        #Remove additional white spaces
        str = re.sub('[\s]+', ' ', str)

        return str
```

Gambar 4. 3 Kode Program Proses *Cleaning*

2. *Remove stopword*

Tahapan *remove stopword* memiliki tujuan untuk penghapusan kata-kata yang kurang memiliki makna yang berarti seperti kata: dan, saya, atau. Proses ini memakai bantuan *library* nltk pada bahasa pemrograman Python. Dalam tahap ini, pertama dilakukan instalasi *library* seperti ditunjukan pada Gambar 4. 4.

```
    pip install nltk
```

Gambar 4. 4 Instalasi *library nltk*

Setelah proses instalasi selesai dilakukan, dilakukan impor *library* yang akan digunakan seperti pada Gambar 4. 5.

```
    import nltk
    from nltk import word_tokenize
    from nltk.corpus import stopwords
```

Gambar 4. 5 Import *library nltk*

Berikut merupakan kode program implementasi dari proses *remove stopword* ditunjukkan pada Gambar 4. 6.

```
    def removeStopword(str):
        stop_words = set(stopwords.words('indonesian'))
        word_tokens = word_tokenize(str)
        filtered_sentence = [w for w in word_tokens if not w in
stop_words]
        return ' '.join(filtered_sentence)
```

Gambar 4. 6 Kode Program Proses *Remove stopword*

3. *Stemming*

Tahapan *stemming* memiliki tujuan untuk mengubah kata-kata yang memiliki imbuhan menjadi sebuah kata dasar aslinya. Pada proses *stemming* digunakan bantuan *library* Sastrawi pada bahasa pemrograman Python. Dalam tahap ini, pertama melakukan instalasi *library* seperti pada Gambar 4. 7.

```
pip install Sastrawi
```

Gambar 4. 7 Instalasi *Library Sastrawi*

Setelah proses instalasi selesai selanjutnya adalah proses impor *library* yang digunakan seperti ditunjukkan pada Gambar 4. 8

```
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory
```

Gambar 4. 8 Import *library Sastrawi*

Setelah melakukan impor *library* sastrawi langkah selanjutnya membuat fungsi untuk mengimplementasikan tahapan ini. Berikut merupakan kode program implementasi dari proses *remove stopword* ditunjukkan pada Gambar 4. 9.

```
def stemming(str):
    factory =  StemmerFactory()
    stemmer = factory.create_stemmer()
    return stemmer.stem(str)
```

Gambar 4. 9 Kode Program Proses *Stemming*

4. *Tokenization*

Proses ini bertujuan untuk memecah dokumen menjadi bagian-bagian yang lebih kecil sehingga memudahkan untuk analisis. Proses ini menggunakan *library* nltk pada bahasa pemrograman Python. Berikut merupakan potongan kode program untuk implementasi dari proses *tokenization* ditunjukan pada Gambar 4. 10.

```
stop_words = set(stopwords.words('indonesian'))
    word_tokens = word_tokenize(str)
    filtered_sentence = [w for w in word_tokens if not w in
stop_words]
```

```
        return ' '.join(filtered_sentence)
```

Gambar 4. 10 Kode Program Proses *Tokenization*

Sesudah *preprocessing* selesai dilakukan, hasil *preprocessing* data akan disimpan menjadi sebuah file baru yang dijadikan sebagai data untuk proses klasifikasi. Berikut merupakan hasil dari *preprocessing* ditunjukkan pada Gambar 4. 11.

A1 *fx*

	A	B	C	D
1		message	filename	label
2	0	sempat menang hadiah cari cek nama klik	Spam (182).txt	Spam
3	1	dapat tiket jungleland harga spesial khusu	Spam (183).txt	Spam
4	2	kami ingin membantu anda bepergian di k	Spam (184).txt	Spam
5	3	ayo unduh daftar kartu debit jadi gunakan	Spam (180).txt	Spam
6	4	dapatkan lebih banyak pengunjung ke bro	Spam (199).txt	Spam
7	5	tengah bulan mau belanja Bisa kok Must I	Spam (200).txt	Spam
8	6	cinta senin diskon pesawat hotel senin se	Spam (201).txt	Spam
9	7	bursa kerja nasional peningkatan pelayan	Spam (202).txt	Spam
10	8	halo Ivan, kami senang anda bergabung c	Spam (179).txt	Spam
11	9	dalam rangka ingat tahun merdeka kopitia	Spam (181).txt	Spam
12	10	teman teman bingung usaha modal saran	Spam (109).txt	Spam
13	11	senang hati memberitahu salah menang p	Spam (97).txt	Spam
14	12	aplikasi email tampil email tampil isi email	Spam (185).txt	Spam
15	13	aplikasi online web browser klik link henti	Spam (186).txt	Spam
16	14	semprot canggih repot bawa selang bera	Spam (187).txt	Spam
17	15	terima kasih pindah hati setia belanja buk	Spam (188).txt	Spam
18	16	sangga punggung perut bantu ringan saki	Spam (189).txt	Spam

Gambar 4. 11 Data Hasil *preprocessing*

43 Ekstraksi Fitur

Pada proses ekstraksi fitur menggunakan n-gram dengan bantuan *library* python menggunakan *count vectorizer* menggunakan *library sklearn*. Proses ini menggunakan nilai n=1 / *unigram* hingga n=5 / *fivegram.* Sebelum melakukan proses ekstraksi fitur dilakukan pembagian data dengan rincian yang dapat dilihat pada tabel Tabel 4. 1.

Tabel 4. 1 Pembagian Data *Training* dan Data *Testing*

Pembagian	Presentase	Total
Data Training	80%	493
Data Testing	20%	124
Total	100%	617

Pembagian data ini dipilih secara *random* dengan bantuan *library sklearn*. Berikut merupakan kode program untuk pembagian data ditunjukkan pada Gambar 4. 12.

```
   from sklearn.model_selection import train_test_split
   X_train, X_test, y_train, y_test = train_test_split(X,y,test_size=
0.2)
```

Gambar 4. 12 Kode Program Pembagian Data

Setelah pembagian data dilanjutkan dengan proses ekstraksi fitur menggunakan n-gram dan memakai bantuan *library scikit learn*. Berikut merupakan implementasi kode program dari proses ekstraksi fitur menggunakan n-gram ditunjukkan pada Gambar 4. 13.

```
   from sklearn.feature_extraction.text import CountVectorizer
   unigram_vectorizer = CountVectorizer()
   X_train_unigram = unigram_vectorizer.fit_transform(X_train)
   X_test_unigram = unigram_vectorizer.transform(X_test)

   bigram_vectorizer = CountVectorizer(ngram_range=(2,2))
   X_train_bigram = bigram_vectorizer.fit_transform(X_train)
   X_test_bigram = bigram_vectorizer.transform(X_test)

   trigram_vectorizer = CountVectorizer(ngram_range=(3,3))
   X_train_trigram = trigram_vectorizer.fit_transform(X_train)
   X_test_trigram = trigram_vectorizer.transform(X_test)

   trigram_vectorizer = CountVectorizer(ngram_range=(3,3))
   X_train_trigram = trigram_vectorizer.fit_transform(X_train)
   X_test_trigram = trigram_vectorizer.transform(X_test)



   fourgram_vectorizer = CountVectorizer(ngram_range=(4,4))
   X_train_fourgram = fourgram_vectorizer.fit_transform(X_train)
   X_test_fourgram = fourgram_vectorizer.transform(X_test)
```

```
fivegram_vectorizer = CountVectorizer(ngram_range=(5,5))
X_train_fivegram = fivegram_vectorizer.fit_transform(X_train)
X_test_fivegram = fivegram_vectorizer.transform(X_test)
```

Gambar 4. 13 Implementasi Ekstraksi Fitur N-Gram

Pada tahap ini akan menghasilkan *word vector* dengan nilai yang sudah terbobot berdasarkan frekuensi dari kemunculan sebuah *term* dalam dokumen yang ada. Berikut merupakan hasil perubahan data teks ke dalam numerik ditunjukkan pada Gambar 4. 14.

	akan menerima	informasi tentang	ingin itu	instal virus	itu terjadi	kumpul informasi	menerima transfer	pada sistem
0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	1
2	0	1	0	0	0	1	0	0
3	0	0	1	0	1	0	0	0
4	1	0	0	0	0	0	1	0

Gambar 4. 14 Perubahan Data Teks

44 Implementasi Naïve Bayes

Proses klasifikasi menggunakan algoritma Naïve Bayes diawali dengan memanggil *library sklearn* dan mengimpor *library* bernama *MultinomialNB* yang ada pada bahasa pemrograman python. Dalam proses ini digunakan data *training* sebesar 80% dari jumlah data yang sudah dibagi pada tahap sebelumnya. Adapun implementasi kode program untuk proses *training* model ditunjukan pada Gambar 4. 15.

```
from sklearn.naive_bayes import MultinomialNB
classifier_unigram = MultinomialNB()
classifier_unigram.fit(X_train_unigram,y_train)

classifier_bigram = MultinomialNB()
classifier_bigram.fit(X_train_bigram,y_train)

classifier_trigram = MultinomialNB()
classifier_trigram.fit(X_train_trigram,y_train)

classifier_fourgram = MultinomialNB()
classifier_fourgram.fit(X_train_fourgram,y_train)
```

```
classifier_fivegram = MultinomialNB()
classifier_fivegram.fit(X_train_fivegram,y_train)
```

Gambar 4. 15 Kode Program Klasifikasi Naïve Bayes

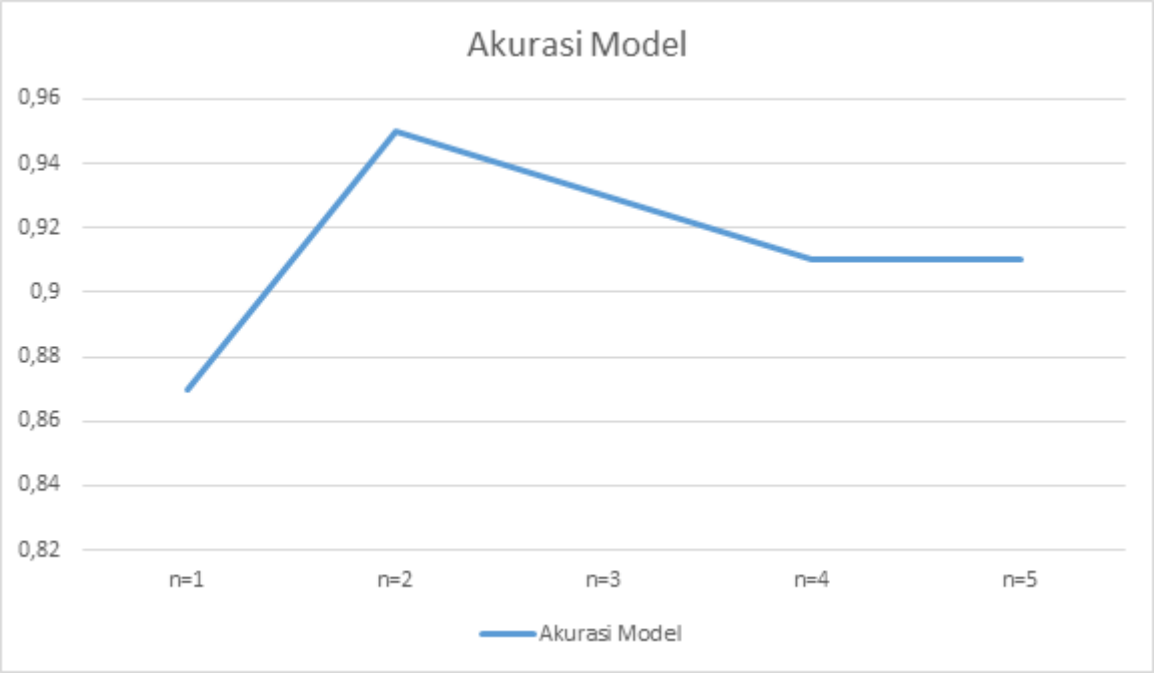
4.5 Evaluasi Model

Pada evaluasi model setelah model *machine learning* menggunakan algoritma Naive Bayes selesai melakukan proses *training*, dilakukan proses *testing* menggunakan data *testing* yang telah dibagi pada proses sebelumnya untuk mengetahui akurasi yang dihasilkan. Perhitungan akurasi ini bertujuan untuk mengetahui berapa persen surel yang benar diprediksi sebagai *spam* dari keseluruhan surel yang ada pada data *testing*. Berikut merupakan hasil akurasi yang dihasilkan oleh model terhadap data *testing* yang ditunjukkan pada Tabel 4. 2.

Tabel 4. 2 Hasil Akurasi Model

N-gram	Akurasi
1	0,87
2	0,95
3	0,93
4	0,91
5	0,91

Berdasarkan Tabel 4. 2 hasil akurasi yang dihasilkan oleh model *machine learning* memiliki nilai yang baik dari n= 1 hingga n=5. Tabel 4. 2 menunjukkan bahwa perpaduan algoritma Naïve Bayes dan ekstraksi fitur n-gram dengan nilai n=2 memiliki akurasi tertinggi yang bernilai 0,95. Hasil akurasi juga ditampilkan dalam bentuk grafik yang ditunjukkan pada Gambar 4. 16.



Gambar 4. 16 Grafik akurasi model

Dari Gambar 4. 16 model dengan nilai n=1 bernilai 0,87 kemudian mengalami kenaikan ketika nilai n=2 yang bernilai 0,95 lalu akurasi model terjadi penurunan pada nilai n=3 hingga n=5 masing-masing nilai n=3 bernilai 0,93, nilai n=4 bernilai 0,91, nilai n=5 bernilai 0,91.

Dari prediksi yang dihasilkan oleh model langkah selanjutnya adalah melakukan analisis menggunakan *Confusion matrix*. Pada *Confusion matrix* terdapat beberapa informasi yang dapat digunakan untuk mengukur performa dan mengevaluasi dari sebuah model yang telah dibuat. Perhitungan pada penelitian ini meliputi menghitung nilai presisi dan *recall*. Berikut merupakan hasil dari *confusion matrix* dari model dengan nilai n=1 hingga n=5.

Tabel 4. 3 Hasil *Confusion matrix* n=1

	Predicted Class	
True Class	***Spam***	***Non Spam***
Spam	62	0
Non Spam	15	47

spam dengan benar sebanyak 62 surel dari seluruh jumlah data *testing* dan tidak ada kesalahan dalam memprediksi spam.

Tabel 4. 4 Hasil *Confusion matrix* n=2

	Predicted Class	
True Class	***Spam***	***Non Spam***
Spam	61	1
Non Spam	5	57

Pada Tabel Tabel 4. 4 menunjukan model yang telah dibangun dapat memprediksi spam dengan benar sebanyak 61 surel dari seluruh jumlah data testing dan salah dalam memprediksi spam berjumlah 1 surel.

Tabel 4. 5 Hasil *Confusion matrix* n=3

	Predicted Class	
True Class	***Spam***	***Non Spam***
Spam	58	4
Non Spam	4	58

Pada Tabel Tabel 4. 5 menunjukan model yang telah dibangun dapat memprediksi spam dengan benar sebanyak 58 surel dari seluruh jumlah data testing dan salah dalam memprediksi spam berjumlah 4 surel.

Tabel 4. 6 Hasil *Confusion matrix* n=4

	Predicted Class	
True Class	***Spam***	***Non Spam***
Spam	58	4
Non Spam	6	56

Pada Tabel Tabel 4. 6 menunjukan model yang telah dibangun dapat memprediksi spam dengan benar sebanyak 58 surel dari seluruh jumlah data testing dan salah dalam memprediksi spam berjumlah 4 surel.

Tabel 4. 7 Hasil *Confusion matrix* n=5

	Predicted Class	
True Class	***Spam***	***Non Spam***
Spam	58	4
Non Spam	6	56

Pada Tabel 4. 7 menunjukan model yang telah dibangun dapat memprediksi spam dengan benar sebanyak 58 surel dari seluruh jumlah data testing dan salah dalam memprediksi spam berjumlah 4 surel.

Berdasarkan hasil *confusion matrix* yang ditunjukkan pada Tabel 4. 3 hingga Tabel 4. 7 didapatkan nilai *true positive, false positive, true negative dan false negative.* Dari nilai yang didapat dilakukan perhitungan menggunakan rumus *precision* dengan persamaan (2.2), *recall* dengan persamaan (2.3), dan *f-score* dengan persamaan (2.4) yang terdapat pada bab sebelumnya untuk mengevaluasi model. Hasil nilai presisi dan *recall* memiliki nilai sebesar 0-1 Semakin tinggi nilainya maka semakin baik. Perhitungan nilai *precision* bertujuan untuk mengetahui presentase surel yang benar spam dari kesuluruhan surel yang diprediksi spam oleh model yang dihasilkan dan perhitungan nilai *recall* bertujuan untuk mengetahui berapa persen surel yang diprediksi *spam* dibandingkan dengan keseluruhan surel yang sebenarnya *spam.* Berikut merupakan hasil perhitungan nilai *precision*,*recall*, dan *f-score* dapat dilihat pada Tabel 4. 8.

Tabel 4. 8 Hasil Evaluasi Model

N-gram	*Precision*	*Recall*
1	0,8	1
2	0,92	0,98
3	0,93	0,93
4	0,9	0,93
5	0,9	0,93

Dari hasil yang ditunjukan pada Tabel 4. 8 dapat dilihat nilai *precision* pada model *machine learning* dengan ekstraksi fitur n-gram yang bernilai n =3 memiliki nilai yang paling tinggi yaitu 0,93. Untuk *recall* nilai tertinggi didapatkan pada model dengan nilai n= 1.

Setelah evaluasi model, percobaan kembali dilakukan menggunakan data surel yang diambil secara random dan ditemukan kesalahan pada model 3-gram hingga 5-gram dalam melakukan prediksi yang seharusnya surel *non-spam* namun model memprediksinya sebagai *spam.* Berikut merupakan kesalahan model dalam salah memprediksi dapat dilihat pada Gambar 4. 17.

```
Wa'alaikumsalaam.  Terlampir ada dua macam dokumen: dokumen yang telah saya tanda tangani
dokumen yang saya beri tanda hasil reviu. Jika revisi sempat dilakukan sebelum submit dokumen,
lembar pengesahan bisa dipotong (misal pakai split pdf) dari dokumen yang telah tertandatangan
Oh ya mas, boleh tolong dataset dan kode
-kodenya dibagikan ke saya melalui Google Drive? Thanks.

...

Prediksi model unigram =  Non Spam
Prediksi model bigram =  Non Spam
Prediksi model trigram =  Spam
Prediksi model fourgram =  Spam
Prediksi model fivegram =  Spam
```

Gambar 4. 17 Data Salah Prediksi

4.6 User interface

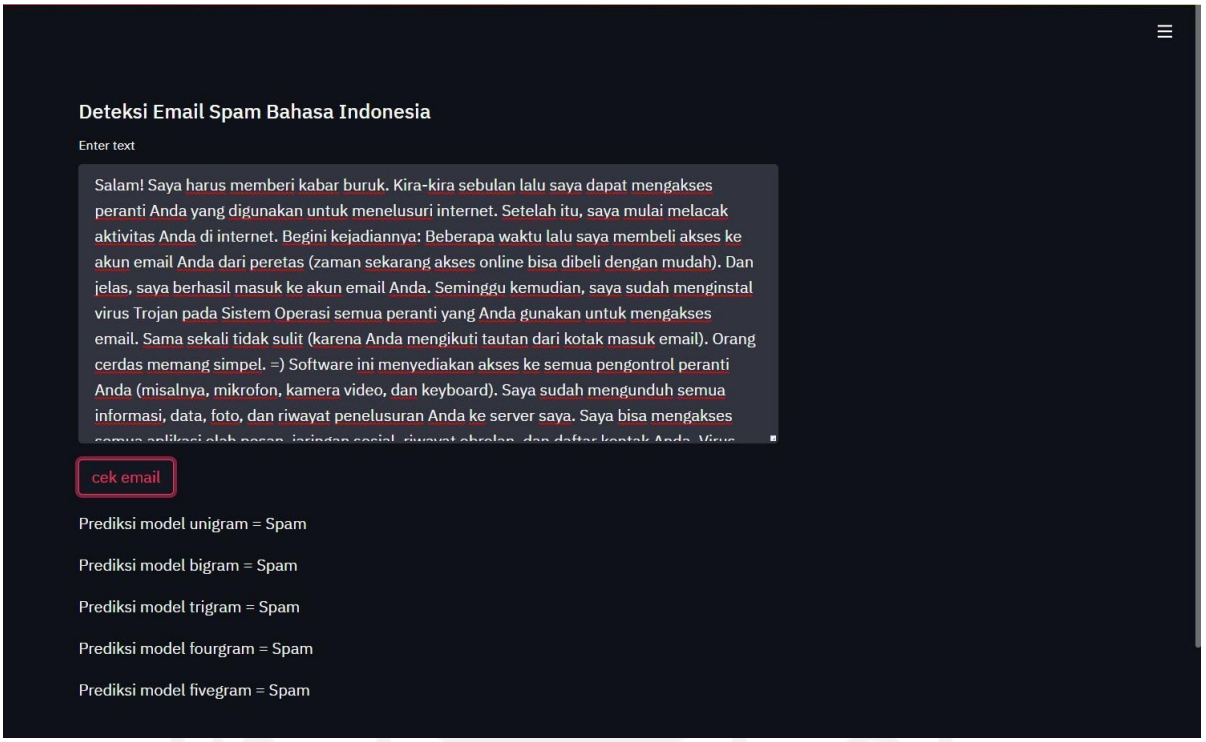
Pembuatan *user interface* dilakukan menggunakan bahasa pemrograman Python dan dengan bantuan library *open-source* streamlit. Berikut merupakan keseluruhan code program dalam pembuatan *user interface* ditunjukan pada Gambar 4. 18.

```
unigram_vectorizer = pickle.load(open('C:/Users/ASUS/Downloads/vector
izer_unigram.sav', 'rb'))
print(unigram_vectorizer)
bigram_vectorizer = pickle.load(open('C:/Users/ASUS/Downloads/vectori
zer_bigram.sav', 'rb'))
trigram_vectorizer = pickle.load(open('C:/Users/ASUS/Downloads/vector
izer_trigram.sav', 'rb'))
fourgram_vectorizer = pickle.load(open('C:/Users/ASUS/Downloads/vecto
rizer_fourgram.sav', 'rb'))
fivegram_vectorizer = pickle.load(open('C:/Users/ASUS/Downloads/vecto
rizer_fivegram.sav', 'rb'))

#load Model
```

```
unigram_model = pickle.load(open('C:/Users/ASUS/Downloads/model_unigr
am (1).sav', 'rb'))
bigram_model = pickle.load(open('C:/Users/ASUS/Downloads/model_bigram
 (1).sav', 'rb'))
trigram_model = pickle.load(open('C:/Users/ASUS/Downloads/model_trigr
am (1).sav', 'rb'))
fourgram_model = pickle.load(open('C:/Users/ASUS/Downloads/model_four
gram (1).sav', 'rb'))
fivegram_model = pickle.load(open('C:/Users/ASUS/Downloads/model_five
gram (1).sav', 'rb'))

label ={1:"Spam", 0:"Non Spam" }
string = st.text_area('Enter text', height=275)
if st.button('cek email'):
    text_data= [preprocessing(string)]

#Vectorize Unigram
    unigram_vector = unigram_vectorizer.transform(text_data)
#Vectorize Bigram
    bigram_vector = bigram_vectorizer.transform(text_data)
#Vectorize Trigram
    trigram_vector = trigram_vectorizer.transform(text_data)
#Vectorize Fourgram
    fourgram_vector = fourgram_vectorizer.transform(text_data)
#Vectorize Fivegram
    fivegram_vector = fivegram_vectorizer.transform(text_data)

    st.write("Prediksi model unigram = ", label[unigram_model.predict
(unigram_vector)[0]])
    st.write("Prediksi model bigram = ", label[bigram_model.predict(b
igram_vector)[0]])
    st.write("Prediksi model trigram = ", label[trigram_model.predict
(trigram_vector)[0]])
    st.write("Prediksi model fourgram = ", label[fourgram_model.predi
ct(fourgram_vector)[0]])
    st.write("Prediksi model fivegram = ", label[fivegram_model.predi
ct(fivegram_vector)[0]])
```

Gambar 4. 18 Kode Program *User Interface*

Setelah melakukan penulisan kode diatas akan didapatkan hasil berupa interface yang dapat dilihat pada Gambar 4. 19 dan digunakan untuk melakukan prediksi terhadap surel baru termasuk jenis surel *spam / non-spam.*



Gambar 4. 19 Hasil *User Interface*

BAB V
KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan terdapat beberapa hasil yang dapat ditarik kesimpulan, yaitu:

a. Algoritma Naïve Bayes terbukti mampu mengidentifikasi surel *spam* dan *non-spam* berbahasa Indonesia dibuktikan dengan nilai akurasi, presisi, dan *recall* yang baik pada model yang dihasilkan pada penelitian ini.
b. Melalui perbandingan yang dilakukan, perpaduan algoritma Naïve Bayes dan ekstraksi fitur n-gram dengan nilai n=2/bigram menghasilkan nilai akurasi tertinggi yaitu 95%.

5.2 Saran

Dari hasil penelitian yang telah dilakukan masih memiliki banyak kekurangan, dengan demikian untuk penelitian selanjutnya dapat dikembangkan, berikut beberapa saran dari penulis, yaitu:

a. Menggunakan metode algoritma klasifikasi lain sehingga dapat dibandingkan antara algoritma mana yang memiliki performa paling baik.
b. Menambahkan data untuk *training* sehingga model yang dihasilkan dapat semakin baik.
c. Menambahkan fitur klasifikasi yang dapat mengklasifikasikan berdasarkan tipe surel (iklan, *malware*, *phising*, dll.)

DAFTAR PUSTAKA

Abdulhamid, S. M., Shuaib, M., Osho, O., Ismaila, I., & K. Alhassan, J. (2018). Comparative Analysis of Classification Algorithms for Email Spam Detection. *International Journal of Computer Network and Information Security*, *10*(1), 60–67. https://doi.org/10.5815/ijcnis.2018.01.07

Al-Alwani, A., & Beseiso, M. (2013). Arabic Spam filtering using Bayesian Model. *International Journal of Computer Applications*, *79*(7), 11–14. https://doi.org/10.5120/13752-1582

Andriansyah, M., Oswari, T., & Prijanto, B. (2016). *Crowdsourcing : Konsep Sumber Daya Kerumunan dalam Abad Partisipasi Komunitas Internet*. 1–6.

Chandra, W. N., Indrawan, G., & Sukajaya, I. N. (2016). Spam Filtering Dengan Metode Pos Tagger Dan Klasifikasi Naïve Bayes. *Jurnal Ilmiah Teknologi Informasi Asia*, *10*(1), 47–55.

Diale, M., Celik, T., & Van Der Walt, C. (2019). Unsupervised feature learning for spam email filtering. *Computers and Electrical Engineering*, *74*, 89–104. https://doi.org/10.1016/j.compeleceng.2019.01.004

Febriyanti, A. (2018). *Analisis Sentimen Persepsi Pengguna Jne Menggunakan Algoritma Naïve Bayes Classifier*. *16522259*.

Hayuningtyas, R. Y. (2017). Aplikasi Filtering of Spam Email Menggunakan Naïve Bayes. *IJCIT (Indonesian Journal on Computer and Information Technology)*, *2*(1), 53–60.

Hengki, M., & Wahyudi, M. (2020). Klasifikasi Algoritma Naïve Bayes dan SVM Berbasis PSO Dalam Memprediksi Spam Email Pada Hotline-Sapto. *Paradigma - Jurnal Komputer dan Informatika*, *22*(1), 61–67. https://doi.org/10.31294/p.v22i1.7842

Imron, A. (2019). *ANALISIS SENTIMEN TERHADAP TEMPAT WISATA DI KABUPATEN REMBANG MENGGUNAKAN METODE NAIVE BAYES CLASSIFIER*.

Juang, D. (2016). Analisis Spam dengan Menggunakan Naïve Bayes. *Jurnal Teknovasi*, *03*(1998), 51–57.

Naem, A. A., Ghali, N. I., & Saleh, A. A. (2018). Antlion optimization and boosting classifier for spam email detection. *Future Computing and Informatics Journal*, *3*(2), 436–442. https://doi.org/10.1016/j.fcij.2018.11.006

Pratiwi, S. N. D., & Ulama, B. S. S. (2016). Klasifikasi Email Spam dengan Menggunakan Metode Support Vector Machine dan k-Nearest Neighbor. *Jurnal Sains dan Seni ITS*, *5*(2),

344–349.

Sudibyo, A., Asra, T., & Rifai, B. (2018). Klasifikasi Seleksi Atribut Pada Serangan Spam. *Jurnal PILAR Nusa Mandiri*, *14*(2), 145–150. https://doi.org/https://doi.org/10.33480/pilar.v14i2.31

Sugianto, S. A., Liliana, L., & Rostianingsih, S. (2013). Pembuatan Aplikasi Predictive Text Menggunakan Metode N-gram-based. *Jurnal Infra*, *1*(2), 1–6.

Vernanda, Y., Kristanda, M. B., & Hansun, S. (2020). Indonesian language email spam detection using n-gram and naïve bayes algorithm. *Bulletin of Electrical Engineering and Informatics*, *9*(5), 2012–2019. https://doi.org/10.11591/eei.v9i5.2444

LAMPIRAN