

Pengembangan Aplikasi Analisis Sentimen Aplikasi PeduliLindungi Menggunakan Metode Naïve Bayes

Edsel Jayadi

Informatika, Fakultas Ilmu Komputer dan Desain Institut Teknologi dan Bisnis Kalbis
Jalan Pulomas Selatan Kav. 22, Jakarta 13210
Email: edseljayadi12@gmail.com

Abstract: This study aims to build sentiment analysis application using Naïve Bayes method to analyze public view about an application called PeduliLindungi, and measure it's accuracy by using twitter dataset. PeduliLindungi is an application developed by the government in order to track and stop Coronavirus Disease (COVID-19). The dataset used in this study is collected by using crawling method with Tweepy. Collected dataset will then go through data pre-processing and labeled by using VADER in order to separate it into positive and negative sentiments. The data will be weighted based on the frequency of its occurrence in all tweets using the TF-IDF method. The weighted data will then be classified using the Naïve Bayes method. This research used the incremental method both in model and software development. The results obtained in this study is a model with an accuracy score of 85% and an average precision, recall, and f1-score of 85%.

Keywords: Naïve Bayes, Sentiment Analysis, Incremental, TF-IDF, Twitter

Abstrak: Penelitian ini bertujuan untuk membangun aplikasi analisis sentimen menggunakan metode Naïve Bayes untuk menganalisis pandangan masyarakat terhadap sebuah aplikasi bernama PeduliLindungi, dan mengukur akurasi dengan menggunakan dataset twitter. PeduliLindungi adalah aplikasi yang dikembangkan oleh pemerintah untuk melacak dan menghentikan Coronavirus Disease (COVID-19). Dataset yang digunakan dalam penelitian ini dikumpulkan dengan menggunakan metode crawling dengan API bernama Tweepy. Dataset yang terkumpul kemudian akan melalui data pre-processing dan diberi label menggunakan VADER untuk memisahkannya menjadi sentimen positif dan negatif. Data tersebut akan dihitung bobotnya berdasarkan frekuensi kemunculannya pada semua tweet menggunakan metode TF-IDF. Data yang sudah dihitung bobotnya kemudian akan diklasifikasikan menggunakan metode Naïve Bayes. Penelitian ini menggunakan metode inkremental baik dalam pengembangan modelnya maupun perangkat lunaknya. Hasil yang diperoleh dalam penelitian ini adalah model dengan akurasi skor 85% dan rata-rata presisi, recall, dan f1-score 85%.

Kata kunci: Naïve Bayes, Analisis Sentimen, Inkremental, TF-IDF, Twitter

I. PENDAHULUAN

PeduliLindungi adalah aplikasi yang dikembangkan untuk membantu instansi pemerintah terkait dalam melakukan pelacakan untuk menghentikan penyebaran *Coronavirus Disease* (COVID-19). Aplikasi ini mengandalkan partisipasi masyarakat untuk saling membagikan data lokasinya saat bepergian agar penelusuran riwayat kontak dengan penderita COVID-19 dapat dilakukan. Pengguna aplikasi ini juga akan mendapatkan notifikasi jika

berada di keramaian atau berada di zona merah, yaitu area atau kelurahan yang sudah terdata bahwa ada orang yang terinfeksi COVID-19 positif atau ada Pasien Dalam Pengawasan [1].

Aplikasi PeduliLindungi sudah menjadi aplikasi yang wajib digunakan untuk seluruh moda transportasi, baik darat, laut, udara, dan perkeretaapian di masa pandemi Covid-19. Hal ini membuat aplikasi PeduliLindungi menjadi salah satu *trending topic* di platform media sosial bernama twitter

pada hari Selasa, 24 Oktober 2021. Banyak sekali komentar dan opini yang dikeluarkan masyarakat tentang aplikasi PeduliLindungi, baik itu tentang kepuasannya ataupun tentang kekecewaannya pada aplikasi tersebut.

Analisis sentimen adalah proses mengolah data berupa teks untuk memperoleh informasi tentang sentimen yang dapat bernilai positif atau negatif. Analisis sentimen dapat digunakan untuk mengulas data teks menjadi informasi yang berfungsi untuk membantu sebuah organisasi dalam mengambil keputusan, karena informasi yang didapatkan dapat menjadi referensi yang tepat sebelum mengambil keputusan tersebut.

Pada penelitian terdahulu yang dilakukan oleh George Kenneth Locarso, tentang analisis sentimen *review* aplikasi PeduliLindungi di Google Play Store. *Review* tersebut akan dibagi menjadi 3 kategori yaitu positif, netral, dan negatif. Dengan menerapkan metode Naïve Bayes, penelitian ini berhasil mengklasifikasikan *review* ke dalam 3 kategori tersebut dengan tingkat akurasi sebesar 83,3% [2].

Penelitian terdahulu yang dilakukan oleh Zulkifli Rais, Ferigo Taufani Tri Hakiki, dan Riska Aprianti yang menerapkan metode Naïve Bayes untuk analisis sentimen aplikasi PeduliLindungi. Sentimen akan dibagi menjadi 3 kategori yaitu positif, netral, dan negatif berdasarkan *rating* yang diberikan oleh *user* di Google Play Store. Penelitian ini berhasil mengklasifikasi sentimen kedalam 3 kategori tersebut dengan akurasi sebesar 73% [3].

Berdasarkan penelitian terdahulu tersebut, terlihat bahwa metode Naïve Bayes merupakan metode yang paling umum digunakan dalam melakukan

analisis sentimen. Maka dari itu, metode Multinomial Naïve Bayes dengan ekstraksi fitur TF-IDF akan digunakan.

II. METODE PENELITIAN

A. Metode Inkremental

Metode inkremental merupakan salah satu metode pengembangan perangkat lunak yang mampu meminimalisir ketidak sesuaian dalam proses pengembangan perangkat lunak [4]. Metode inkremental dipilih sebagai metode pengembangan perangkat lunak karena metode ini dianggap mampu mengakomodasikan perubahan secara fleksibel dan memiliki resiko yang lebih rendah terhadap proses pengembangan sistem.

B. Twitter

Twitter adalah sebuah layanan media sosial dan *microblog* daring yang didirikan oleh Jack Dorsey pada bulan Maret 2006 dan dapat diakses bebas ke masyarakat pada bulan Juli yang fungsinya untuk memberikan kenyamanan mengirim dan membaca pesan berupa teks yang dikenal oleh pengguna twitter sebagai *tweet* atau cuitan. Twitter menjadi salah satu dari sepuluh situs yang paling sering dikunjungi masyarakat Indonesia [5].

C. Python

Python adalah salah satu bahasa pemrograman komputer yang *open source* sehingga mempermudah dalam pembelajaran, mendukung pemrograman berorientasi objek, fungsional, struktur kodingnya prosedural, dan lain – lain [6]. Python dikenal sebagai bahasa yang dinamis, sehingga tidak perlu mengidentifikasi secara eksplisit [7]. Bahasa pemrograman Python dapat digunakan untuk membuat program

desktop, mobile, website, Internet of Things, Game, dan sebagainya.

D. VADER

VADER (*Valence Aware Dictionary and sEntiment Reasoner*) adalah sebuah leksikon dan juga *tool* berbentuk *rule-based sentiment analyzer*. VADER diperkenalkan pada tahun 2014 oleh C.J Hutto dan Eric Gilbert yang metode pembentukannya didasarkan pada pendekatan *human-centric*, dimana metode tersebut akan menggabungkan analisis kualitatif dan validasi empiris menggunakan kebijaksanaan dan penilaian manusia [8]. VADER dibuat secara khusus untuk memprediksi sentimen yang di ekspresikan di dalam sebuah media sosial. *Library* VADER tersedia secara *open-sourced* dan juga telah dilisensi oleh MIT [9].

E. Naïve Bayes

Algoritma Naïve Bayes merupakan salah satu algoritma yang terdapat pada teknik klasifikasi. Naïve Bayes merupakan pengklasifikasian dengan metode probabilitas dan statistik yang dikemukakan oleh ilmuwan Inggris Thomas Bayes, yaitu memprediksi peluang di masa depan berdasarkan pengalaman dimasa sebelumnya sehingga dikenal sebagai Teorema Bayes. Berikut adalah rumus Teorema Naïve Bayes [10].

$$P(H|X) = \frac{P(X|H).P(H)}{P(X)} \quad (1)$$

Dimana X adalah data dengan *class* yang belum diketahui, H adalah hipotesis data X merupakan suatu *class* spesifik, P(X) adalah probabilitas X, P(H) adalah probabilitas hipotesis H (*prior probability*), P(H|X) adalah probabilitas hipotesis H berdasar kondisi X (*posteriori probability*), dan P(X|H)

adalah probabilitas X berdasarkan kondisi pada hipotesis H.

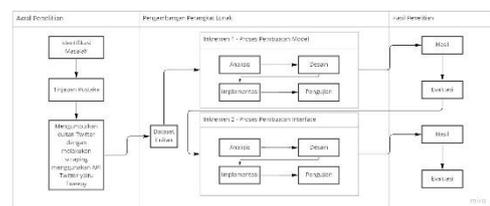
F. Multinomial Naïve Bayes

Multinomial Naïve Bayes adalah *conditional probability* yang dilakukan tanpa memperhitungkan urutan kata dan informasi yang ada dalam kalimat atau dokumen secara umum [11]. Metode Multinomial Naïve Bayes dalam melakukan perhitungan peluang sebuah kata, masuk ke dalam kategori dapat dilakukan dengan menggunakan persamaan *conditional probability*. Pada persamaan *conditional probability* untuk menghindari nilai *zero* digunakan *add-one* atau *Laplace smooting*, dengan menambahkan angka 1 untuk tiap perhitungan kata uji. Persamaan *conditional probability* dapat dilihat pada rumus berikut [12].

$$P(w_i|c_j) = \frac{\text{count}(w_i, c_j) + 1}{(\sum_{w \in V} \text{count}(w, c_j)) + |V|} \quad (2)$$

$\text{count}(w_i, c_j) + 1$ merupakan jumlah kemunculan dari kata uji yang muncul dalam kategori c_j ditambah dengan 1 untuk menghindari nilai *zero*. $\text{count}(w, c_j)$ merupakan jumlah kemunculan seluruh kata yang ada pada kategori c_j . $|V|$ merupakan jumlah seluruh kata unik yang ada pada seluruh kategori.

G. Proses Penelitian



Gambar 1 Proses Penelitian

Sesuai pada gambar diatas, proses penelitian akan dibagi menjadi tiga tahap. Tahap awal penelitian adalah tahap persiapan dan pengumpulan dataset. Pengambilan dataset dari twitter

dilakukan karena jumlah sentimen masyarakat Indonesia terhadap aplikasi PeduliLindungi pada platform tersebut cukup tinggi, dan metode pengambilannya sudah dipermudah dengan menggunakan API Tweepy.

Tahap pengembangan perangkat lunak adalah tahap dimana perangkat lunak penelitian mulai dibuat sesuai dengan metode yang sudah ditentukan yaitu metode inkremental. Metode inkremental dibagi menjadi dua bagian yaitu Inkremental I dan Inkremental II. Hasil penelitian adalah tahap dimana hasil dari pengembangan perangkat lunak akan dievaluasi.

H. Proses Pembuatan Dataset

Dataset penelitian dikumpulkan dengan cara melakukan *crawling* pada medial sosial twitter. Sebelum *crawling* dapat dilakukan, *user* harus mendaftar terlebih dahulu melalui *Twitter Developer Platform* untuk meminta izin akses twitter API bernama Tweepy. Data yang akan diambil dari twitter adalah data yang memiliki kata kunci PeduliLindungi atau #PeduliLindungi.

id	text	lang	date
1	@moonlight_ kkk kkk pedulilindungi ya	id	2022-03-23 16:40:47
2	habahahaha lagi nyesak nangis sbtbaa sindi bikin katawa harusnya pedulilindungi, sindi bilangny id	id	2022-03-23 16:21:44
3	@PeduliLindungi_ @Komedirejeh nama 190k nya klo ga salah R_ @cncmanagement kalnya id	id	2022-03-23 16:06:20
4	@Ommon9 @PeduliLindungi_ @Komedirejeh Linc he, senus ki wong Majang ki? Majang ndi? Ma Tak kiro Malangan, ibake tonggo deve id	id	2022-03-23 15:52:36
5	skrg hasil tes di pedulilindungi nunjuk hasil 10 hari sebelumnya id	id	2022-03-23 15:31:30
6	@TchmanID Dah kat event dan hosting saen? id	id	2022-03-23 15:16:14
7	@kswhtthetrik @Ommon9 @justbakindddd @Komedirejeh Bener banget, dan kebetulan dia j id	id	2022-03-23 15:15:03
8	@Ommon9 @justbakindddd @PeduliLindungi_ @Komedirejeh Pasti bapak itu orangnya friendly id	id	2022-03-23 15:06:10
9	@dazchabn_ @hribacot Freelance yg mengkat whehe id	id	2022-03-23 14:54:56

Gambar 2 Hasil *Crawling* Twitter

I. Inkremental I

Inkremental I adalah tahap dimana kebutuhan dari model analisis sentimen ditentukan. Tahap ini juga akan membuat desain alur logika pembuatan model agar model dapat dibuat secara teratur. Desain yang sudah dibuat akan diimplementasikan dalam bentuk kode program menggunakan bahasa pemrograman python.

1. Analisis

Data cuitan yang berhasil diambil dengan Tweepy pada tanggal 23 Maret 2022, akan disimpan dalam format .csv untuk kemudahan pemakaian. Data yang sudah disimpan kemudian akan dilakukan *pre-processing* untuk mengurangi *noise* sehingga data menjadi lebih akurat dan relevan.

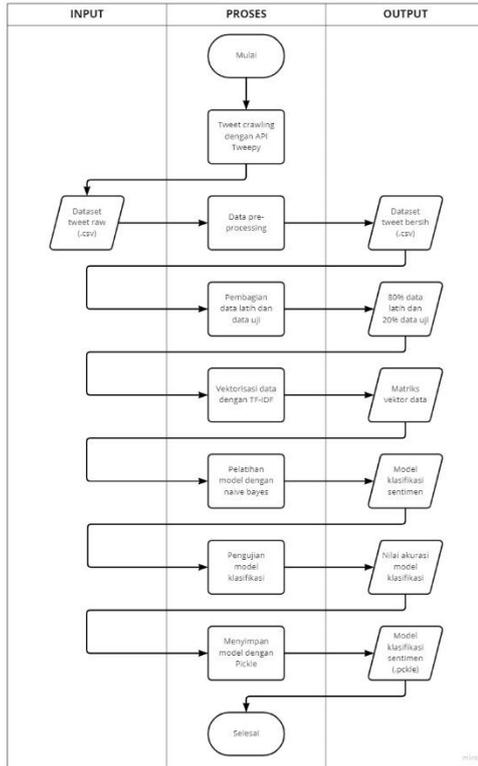
Tahap pertama dari data *pre-processing* membersihkan data dari data duplikat, tanda baca, simbol, *link*, *emoji*, RT (*Re-Tweet*), dan karakter non-ascii. Tahap kedua dari data *pre-processing* akan menerjemahkan data ke bahasa Inggris dan mengimplementasikan *contraction*, *stopwords removal*, dan *stemming*.

Pelabelan sentimen akan dilakukan dengan bantuan *library* VADER, *library* VADER hanya dapat membaca data teks dalam bahasa Inggris. Sentimen akan dibagi menjadi 3 yaitu sentimen positif, negatif, dan netral. kalimat akan dilabeli sebagai positif jika nilai *compound* lebih besar atau sama dengan 0.05, kalimat akan dilabeli sebagai negatif jika nilai *compound* lebih kecil atau sama dengan -0.05, kalimat akan dilabeli sebagai netral jika nilai *compound* memiliki hasil diantara 0.05 dan -0.05. Sentimen netral yang didapat dari VADER akan dihapus karena tidak digunakan dalam penelitian ini. Model *machine learning* analisis sentimen dibuat menggunakan algoritma klasifikasi Naïve Bayes

Tabel 1 Instrumen Penelitian

No.	Perangkat	Nama
1.	Sistem Operasi	Windows 10
2.	Model	DESKTOP-TBBU39C
3.	Processor	Intel Core i5-3470
4.	Memory	6 GB

2. Desain



Gambar 3 Desain Alur Logika Pembuatan Model

Alur dari model ini, dimulai dengan melakukan data *crawling* pada twitter dengan menggunakan API Tweepy. Input data .csv akan dilakukan *pre-processing* untuk membersihkan data dari karakter-karakter yang tidak dibutuhkan seperti simbol, *link*, angka, *username*, spasi, kata penghubung, dan tokenisasi. Dataset bersih kemudian akan dibagi menjadi dua bagian yaitu data latih dan data uji dengan *output* 80% data latih dan 20% data uji. Data lalu akan dikonversi menjadi vektor dengan menggunakan TF-IDF *vectorizer* untuk mengubah data menjadi format dataset yang dapat diproses oleh model. Model

klasifikasi yang sudah jadi akan melalui proses pengujian dimana tingkat akurasi klasifikasi sentimen positif dan negatifnya dihitung menjadi *output* nilai akurasi model

3. Implementasi

Pada tahapan ini, bahasa pemrograman yang akan digunakan dalam penelitian adalah python. Google colab akan digunakan untuk mempermudah proses pengetikan kode program. *Library* yang akan digunakan pada penelitian ini adalah tweepy, scikit-learn, string, emoji, googletrans, pycontraction, vaderSentiment, nltk, pandas, gensim, time, string, dan re.

4. Pengujian

Pada tahap ini, skenario pengujian dari tahap implementasi program yang sudah direalisasikan akan dilakukan dengan menggunakan metode *White-box*. Pengujian metode *White-box* mengecek apakah program yang sudah dibuat dapat mengeksekusi perintahnya sesuai harapan dan tanpa terjadinya *error*. Skenario pengujian dengan metode *White-box* dapat dilihat pada tabel 2.

Tabel 2 Pengujian White-box Inkremental I

Pengujian	Harapan
Fungsi Data <i>Pre-processing</i>	Data cuitan dibersihkan dari data duplikat, <i>emoji</i> , tanda baca, kata sambung, dan berubah menjadi bahasa inggris
Fungsi <i>Labelling</i> Data Menggunakan VADER	Melabeli data secara otomatis dan membagi kelas label menjadi data positif (1) dan data negatif (-1)
Metode <i>Train/Test Split</i>	Membagi dataset menjadi 80% data latih dan 20% data uji
Metode TF-IDF	Melakukan pembobotan dan ekstraksi fitur data
Pelatihan Model <i>Machine Learning</i>	Melakukan <i>training</i> pada 80% data latih dengan menggunakan metode Multinomial Naïve Bayes
Pengujian Model <i>Machine Learning</i> dengan <i>Confusion Matrix</i>	Mengevaluasi model yang sudah dibuat dengan memberikan nilai akurasi, <i>precision</i> , <i>recall</i> , dan <i>f1-score</i>

J. Inkremental II

Inkremental II adalah tahap dimana kebutuhan dari aplikasi analisis sentimen ditentukan. Tahap ini berfokus untuk membangun aplikasi *desktop* analisis sentimen agar nilai sentimen positif dan negatif terhadap aplikasi PeduliLindungi dapat divisualisasikan. *Mock-up* akan dibuat menggunakan Balsamiq Mockups 3 untuk membantu proses pembuatan aplikasi. Aplikasi *desktop* menggunakan *library* bernama PyQt5 untuk membuat tampilan dari aplikasi.

1. Analisis

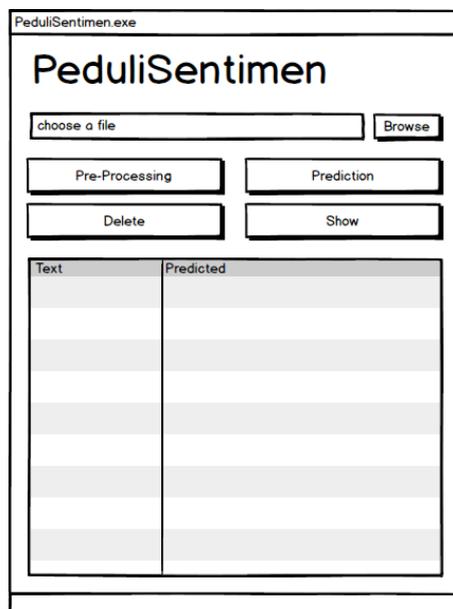
Pada tahap inkremental ii ini, kebutuhan dari tampilan program analisis sentimen aplikasi PeduliLindungi akan dipertimbangkan demi mengutamakan kemudahan penggunaan program tersebut. Berikut adalah uraian yang akan ditampilkan dari model program analisis sentimen:

Tabel 3 Uraian Tampilan Aplikasi

No.	Uraian
1.	Tampilan judul dalam bentuk label.
2.	Tombol-tombol untuk melakukan proses upload file, <i>pre-processing</i> data, memprediksi data, menampilkan data, dan menghapus data.
3.	Hasil analisis sentimen yang akan ditunjukkan dalam bentuk tabel.
4.	Pie chart yang akan muncul setelah tombol prediction di klik.

2. Desain

Pada tahap ini, perancangan dari tampilan program analisis sentimen aplikasi PeduliLindungi akan dibuat dalam bentuk *mock-up*. Hal ini dilakukan untuk membantu proses pertimbangan dari kemudahan dan kenyamanan pemakaian program tersebut. *Mock-up* akan dibuat dengan menggunakan aplikasi Balsamiq Mockups 3.



Gambar 4 *Mock-up* Tampilan PeduliSentimen

3. Implementasi

Pada tahap ini, desain yang sudah dirancang dan dibuatkan *mock-up* akan direalisasikan dengan menggunakan bahasa pemrograman python. Proses ini juga akan dibantu dengan

menggunakan *library open-source* yang bernama PyQt5. *Library* PyQt5 menyediakan aplikasi QtDesigner yang dapat digunakan untuk membuat tampilan secara praktis dengan fitur *drag and drop*. *File* tampilan yang dihasilkan kemudian dapat dikonversi menjadi format *.py*, dan digunakan untuk memberikan fungsi yang sesuai dengan rancangan yang telah ditentukan.

4. Pengujian

Pada penelitian ini, pengujian pada tampilan yang sudah dibuat akan dilakukan dengan menggunakan metode *Black-box*. Pengujian metode *Black-box* akan membuat skenario untuk menguji fungsionalitas dari tampilan program, khususnya untuk mengecek apakah *input* dan *output* dari program sudah sesuai dengan apa yang diharapkan atau belum.

Tabel 4 Pengujian Black-box Inkremental II

Pengujian	Harapan
Tombol 'Browse'	Dapat membuka jendela <i>File Dialog</i> , dan memilih <i>file</i> dengan format <i>.csv</i> untuk disimpan kedalam pandas dataframe
Tombol 'Pre-Process'	Dapat mengaplikasi data <i>pre-processing</i> terhadap dataset yang sudah tersimpan didalam pandas dataframe, dan menampilkan hasil <i>pre-processing</i> kedalam tabel.
Tombol 'Prediction'	Dapat memprediksi sentimen dari dataset yang sudah melalui data <i>pre-processing</i> , dan menampilkan hasil prediksi tersebut kedalam tabel serta memberikan visualisasi berupa <i>pie chart</i> .
Tombol 'Delete'	Dapat menghapus seluruh data yang disimpan didalam pandas dataframe, dan membersihkan tabel dan juga <i>textbox</i> yang sudah terisi.
Tombol 'Show'	Dapat menampilkan data <i>file .csv</i> yang sudah dipilih pada saat <i>browse file</i> kedalam tabel.

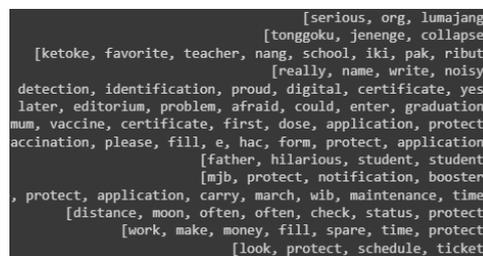
III. HASIL DAN PEMBAHASAN

A. Hasil Inkremental I

Hasil dari tahap Inkremental I adalah model klasifikasi yang dapat memprediksi sentimen masyarakat Indonesia terhadap aplikasi PeduliLindungi. Tahap awal dari Inkremental I adalah untuk mengumpulkan data berupa cuitan dari twitter, untuk digunakan sebagai data latih dan uji dengan cara memanfaatkan python *library* yang bernama Tweepy.

1. Hasil Data Pre-processing

Data yang berhasil dikumpulkan dengan *crawling* menggunakan *library* Tweepy berjumlah 949 data. Data tersebut kemudian akan melalui data *pre-processing* seperti *Regular Expression*, *Drop Duplicate*, *Contraction*, *Stopwords Removal*, *Stemming*, dan Tokenisasi. Setelah melalui *pre-processing*, data yang dimiliki telah berkurang menjadi 868 data. Contoh hasil dari data *pre-processing* dapat dilihat pada gambar 5.



Gambar 5 Hasil Data Pre-processing

2. Hasil Labelling VADER

Hasil dari data *pre-processing*, kemudian akan melalui proses *labelling* untuk memberikan referensi kepada model klasifikasi yang akan dibuat. Pelabelan dengan VADER akan dibagi menjadi tiga sentimen yaitu positif, negatif, dan netral. Cuitan yang dilabeli sebagai netral akan di *drop* karena tidak digunakan didalam penelitian. Cara penulisan sentimen yang digunakan didalam penelitian ini adalah angka 1 sebagai sentimen positif dan -1 sebagai sentimen negatif. Total data yang

dimiliki setelah cuitan dengan sentimen netral di *drop* adalah 515 data. Contoh hasil *labelling* dengan menggunakan VADER dapat dilihat pada gambar 6.

```
[ 'test', 'antigen', 'test', 'friday', 'graduation', 'negative', 'result', 'x' ] -1
[ 'checked', 'guard', 'go', 'swab', 'result', 'go', 'editorium', 'editorium' ] -1
[ 'goddam', 'protect' ] -1
[ 'take', 'care', 'certificate', 'place', 'certificate', 'already', 'app', 'si' ] 1
[ 'gsp', 'naah', 'like', 'drop', 'wheehehe' ] 1
[ 'asks', 'minimum', 'order', 'driver', 'period', 'day', 'positive', 'covid' ] 1
[ 'certificate', 'v', 'k', 'show', 'certificate', 'protect', 'must', 'print' ] 1
[ 'protect' ] 1
[ 'recorder', 'editorium', 'problem', 'afraid', 'could', 'go', 'graduation' ] -1
[ 'want', 'see', 'guy', 'fight' ] -1
[ 'board', 'plane', 'full', 'empty', 'space', 'full', 'examination', 'protector' ] -1
[ 'via', 'da', 'immediately', 'follow', 'exercise', 'admin', 'protect', 'xixixi' ] -1
[ 'never', 'checked', 'protect' ] -1
```

Gambar 6 Hasil Labelling VADER

3. Perbandingan Hasil Model

```
Durasi Testing: 0.001 detik
Akurasi: 0.825
```

	precision	recall	f1-score	support
Negatif	0.00	0.00	0.00	18
Positif	0.83	1.00	0.90	85
accuracy			0.83	103
macro avg	0.41	0.50	0.45	103
weighted avg	0.68	0.83	0.75	103

```
Confusion Matrix:
[[85  0]
 [18  0]]
```

Gambar 7 Hasil Model 515 Data

Terlihat pada gambar 7 model yang sudah dibuat memiliki akurasi sebesar 83%. Akan tetapi model tersebut memiliki hasil *classification report* dimana hasil prediksi sentimen negatif pada data *testing*, memiliki skor *precision*, *recall*, dan *f1-score* yang sangat rendah yaitu 0. Hasil dari *confussion matrix* juga menunjukkan hasil *true positive* sebanyak 85 dan *true negative* sebanyak 0.

Hal ini menunjukkan bahwa model yang dibuat sangat bias terhadap satu kelas yaitu kelas positif. Jika model ini digunakan, hasil prediksi dari model akan sangat cenderung untuk menghasilkan sentimen positif atau bahkan memprediksi seluruh sentimen sebagai positif. Kasus ini terjadi karena data yang digunakan untuk pelatihan dan pengujian memiliki kelas yang tidak seimbang, dimana 515 data yang digunakan memiliki 398 data positif dan 117 data negatif. Karena model yang dibuat akan

digunakan untuk aplikasi analisis sentimen, maka sebaiknya model dengan bias tinggi dihindari untuk dipakai kedalam aplikasi.

Masalah distribusi kelas yang tidak seimbang dapat diselesaikan dengan cara melakukan *oversampling* atau *undersampling*. *Oversampling* dapat dilakukan pada distribusi kelas data yang lebih rendah, sedangkan *undersampling* dapat dilakukan pada distribusi kelas yang lebih tinggi. Karena penelitian ini tidak mengimplementasikan *data augmentation*, metode *oversampling* tidak dapat digunakan pada penelitian ini. Maka dari itu, metode *undersampling* akan dipilih untuk menyelesaikan masalah tersebut.

```
Durasi Testing: 0.002 detik
Akurasi: 0.750
```

	precision	recall	f1-score	support
Negatif	0.62	0.87	0.72	15
Positif	0.89	0.68	0.77	25
accuracy			0.75	40
macro avg	0.76	0.77	0.75	40
weighted avg	0.79	0.75	0.75	40

```
Confusion Matrix:
[[17  8]
 [ 2 13]]
```

Gambar 8 Hasil Model 200 Data

Penggunaan metode *undersampling* pada gambar 8 diatas dilakukan dengan cara menyamakan distribusi kelas data yang berlebihan menjadi sama jumlahnya dengan kelas data yang kekurangan. Pada penelitian ini, penyeimbangan distribusi kelas data akan dilakukan dengan cara memotong data tersebut menjadi 100 data positif dan 100 data negatif.

Gambar 8 menunjukkan terjadinya penurunan terhadap *accuracy score* yang dihasilkan oleh model, tetapi skor dari *precision*, *recall*, dan *f1-score* yang dihasilkan model tersebut terlihat lebih bagus dibandingkan dengan skor pada gambar 7. Hanya saja hasil yang

diberikan oleh model gambar 8 masih belum memuaskan.

Metode *undersampling* digunakan kembali dengan pemotongan data menjadi 50 data positif dan 50 data negatif. Hasil menggunakan 50 data positif dan 50 data negatif tersebut dapat dilihat pada gambar 9.

```

Durasi Testing: 0.002 detik
Akurasi: 0.850
precision recall f1-score support
Negatif 0.89 0.80 0.84 10
Positif 0.82 0.90 0.86 10
accuracy 0.85 0.85 0.85 20
macro avg 0.85 0.85 0.85 20
weighted avg 0.85 0.85 0.85 20

Confusion Matrix:
[[9 1]
 [2 8]]
    
```

Gambar 9 Hasil Model 100 Data

Model dari gambar 9 terlihat memberikan hasil yang lebih baik dibandingkan dengan model dari gambar 8. Model dari gambar 9 memberikan hasil *accuracy score* yang lebih besar yaitu 85%. Karena model dengan 100 data memiliki hasil yang lebih baik dibandingkan model dengan 200 data, model dengan 100 data akan digunakan.

4. Hasil Pengujian *White-box*

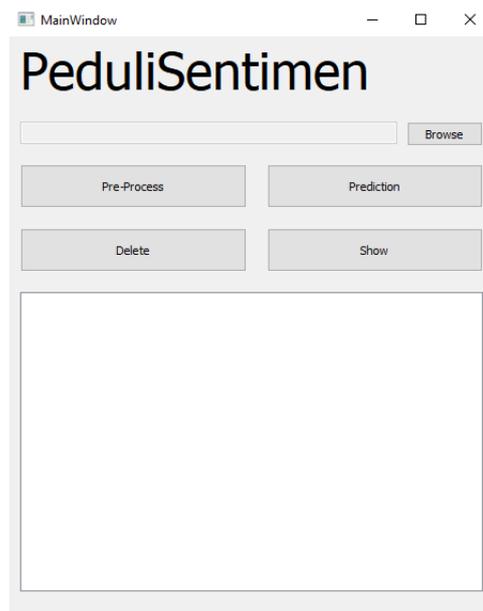
Berdasarkan hasil yang sudah ditunjukkan diatas, dapat disimpulkan bahwa pengujian yang dilakukan dengan menggunakan tabel *White-box* telah berhasil. Fungsi yang dibuat dapat melakukan tugasnya sesuai harapan, dan memberikan hasil yang diinginkan.

B. Hasil Inkremental II

Hasil dari tahap inkremental ii adalah aplikasi *desktop* untuk melakukan prediksi sentimen. Tampilan dari aplikasi *desktop* dirancang dengan menggunakan QtDesigner yang disediakan oleh *library* PyQt5.

1. Tampilan Aplikasi

Desain *mock-up* yang sudah dibuat akan dijadikan referensi dalam perancangan tampilan, agar aplikasi yang dibangun dapat digunakan sesuai dengan kebutuhan penelitian. Hasil tampilan aplikasi dapat dilihat pada gambar 10.

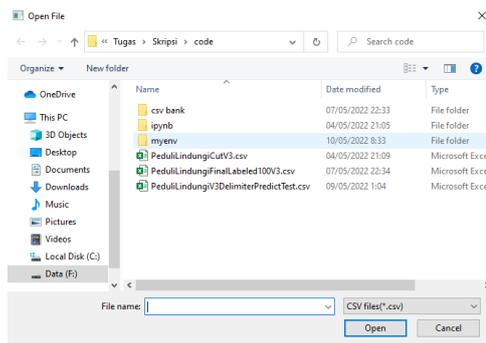


Gambar 10 Hasil Tampilan Aplikasi

Dalam penelitian ini, tampilan akan dibangun dengan 5 tombol dan 1 tabel. Tombol tersebut adalah tombol *Browse*, *Pre-Process*, *Prediction*, *Delete*, dan *Show*.

2. Tombol *Browse*

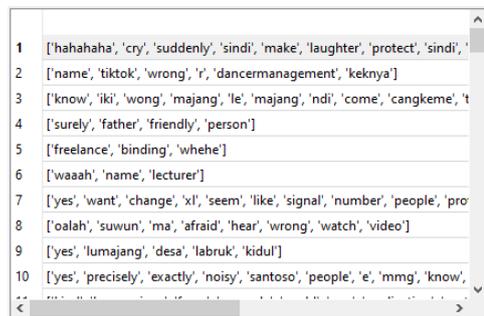
Tombol *Browse* berfungsi untuk membuka *file dialog* yang dapat memilih *file .csv* yang terkait dengan aplikasi PeduliLindungi untuk digunakan. Jendela *file dialog* dapat dilihat pada gambar 11.



Gambar 11 Jendela *File Dialog*

3. Tombol *Pre-Process*

Tombol *Pre-Process* berfungsi untuk melakukan data *pre-processing* seperti menghilangkan simbol, angka, tanda baca, *link*, *username*, dan sebagainya untuk mengurangi *noise data*. Hasil dari tombol *Pre-Process* kemudian akan ditampilkan didalam tabel yang sudah disediakan. Hasil tombol *Pre-Process* dapat dilihat pada gambar 12.



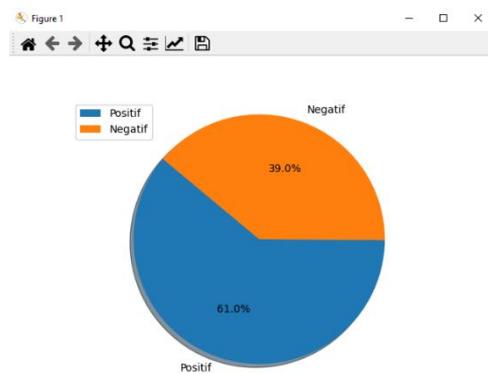
Gambar 12 Hasil Tombol *Pre-Process*

4. Tombol *Prediction*

Tombol *Prediction* berfungsi untuk memprediksi sentimen dari teks yang ada didalam dataset. Hasil dari tombol *Prediction* kemudian akan dimasukkan kedalam tabel dan juga memberikan visualisasi data dalam bentuk *pie chart*. Hasil tombol *Prediction* dan *pie chart* dapat dilihat pada gambar 13 dan 14.

	predicted
1	Positif
2	Negatif
3	Negatif
4	Positif
5	Positif
6	Positif
7	Positif
8	Negatif
9	Positif
10	Positif

Gambar 13 Hasil Tombol *Prediction*



Gambar 14 Visualisasi *Pie Chart*

5. Tombol *Delete*

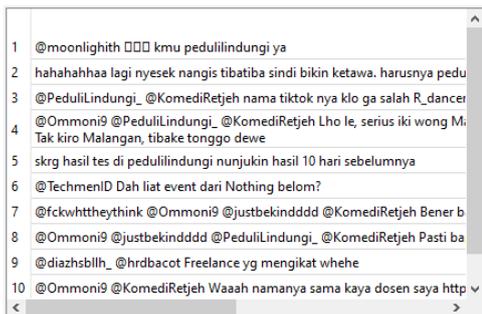
Tombol *Delete* berfungsi untuk menghapus data yang sudah ditampilkan didalam tabel, mengosongkan pandas dataframe yang menyimpan dataset, dan membersihkan *textbox* yang memperlihatkan lokasi *file .csv* yang dipilih. Hasil tombol *Delete* dapat dilihat pada gambar 15.



Gambar 15 Hasil Tombol *Delete*

6. Tombol Show

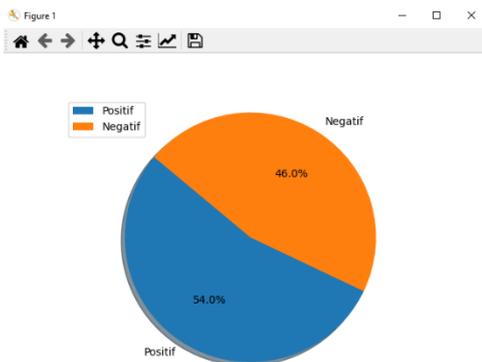
Tombol *Show* berfungsi untuk mengecek apakah isi data *file* .csv yang dipilih sudah benar atau belum. Jika isi data belum sesuai dengan yang diinginkan. Hasil dari tombol *Show* dapat dilihat pada gambar 16.



Gambar 16 Hasil Tombol *Show*

7. Hasil Pengujian Data Riil

Pengujian dilakukan dengan menggunakan 100 data riil yang dikumpulkan melalui *crawling* twitter pada tanggal 4 Mei 2022. Pada gambar 17 terlihat bahwa pengujian penelitian ini memberikan hasil sebanyak 54% sentimen positif dan 46% sentimen negatif.



Gambar 17 Hasil Pengujian Data Riil

8. Hasil Pengujian *Black-box*

Berdasarkan hasil dan evaluasi dari seluruh fitur dan fungsionalitas yang sudah dirancang dalam penelitian ini, dapat disimpulkan bahwa pengujian yang

dilakukan dengan menggunakan tabel *Black-box* telah berhasil. Tombol-tombol yang dibuat dapat menjalankan fungsinya sesuai harapan, dan memberikan hasil yang diinginkan.

IV. SIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, kesimpulan yang dapat diambil dari penelitian analisis sentimen pada aplikasi PeduliLindungi dan pengembangan aplikasi desktopnya, adalah sebagai berikut:

1. Berdasarkan hasil pengujian, pengembangan aplikasi analisis sentimen dapat mengetahui kecenderungan sentimen masyarakat terhadap aplikasi PeduliLindungi.
2. Hasil pengujian pada 100 data riil yang diambil pada tanggal 4 April 2022, diperoleh hasil presentase sentimen sebesar 54% sentimen positif dan 46% sentimen negatif.
3. Setelah melalui data *pre-processing* data cuitan berkurang hingga 515 data cuitan. 515 data cuitan tersebut terdiri dari 398 data positif dan 117 data negatif. Karena telah dilakukan *drop* pada data duplikat dan data dengan sentimen netral.
4. Penggunaan *library* VADER untuk melakukan *labelling* terhadap cuitan twitter yang berbahasa Indonesia tidak optimal. Data perlu diterjemahkan ke bahasa Inggris terlebih dahulu sebelum VADER dapat digunakan. Hasil terjemahan yang didapatkan juga kurang baik karena banyaknya slang dan bahasa daerah yang tidak dapat diterjemahkan ke bahasa Inggris.
5. Model yang dihasilkan dengan menggunakan 515 data cuitan

memiliki bias yang sangat tinggi terhadap satu kelas data. Bias terjadi karena kelas data yang dimiliki tidak seimbang, sehingga metode *undersampling* perlu diimplementasikan.

6. Pengujian model naïve bayes yang dilakukan terhadap 20% data uji memberikan perhitungan *classification report* dengan perolehan skor akurasi sebesar 85% dan skor rata-rata *precision, recall*, dan *f1-score* sebesar 85%.

DAFTAR RUJUKAN

- [1] PeduliLindungi. <https://www.pedulilindungi.id/> (accessed Mar. 03, 2022).
- [2] G. K. Locarso, "ANALISIS SENTIMEN REVIEW APLIKASI PEDULILINDUNGI PADA GOOGLE PLAY STORE MENGGUNAKAN NBC," *J. Tek. Inform. Kaputama*, vol. 6, no. 2, pp. 353–361, 2022.
- [3] Z. Rais, F. Taufani, T. Hakiki, and R. Aprianti, "Sentiment Analysis of Peduli Lindungi Application Using the Naive Bayes Method," *J. Appl. Sci. Math. Its Educ.*, vol. 11, no. 1, 2022.
- [4] M. Syarif and W. Nugraha, "Metode Incremental Dalam Membangun Aplikasi Identifikasi Gaya Belajar Untuk Meningkatkan Hasil Belajar Siswa," *Jusikom J. Sist. Komput. Musirawas*, vol. 4, no. 1, pp. 42–49, 2019, doi: 10.32767/jusikom.v4i1.441.
- [5] F. Ratnawati, "Implementasi Algoritma Naive Bayes Terhadap Analisis Sentimen Opini Film Pada Twitter," *INOVTEK Polbeng - Seri Inform.*, vol. 3, no. 1, p. 50, 2018, doi: 10.35314/isi.v3i1.335.
- [6] L. F. Narulita, "Pengaruh Proses Stemming Pada Kinerja Analisa Sentimen Pada Review Buku," *J. Has. Penelit. LPPM Untag Surabaya*, vol. 3, no. Januari, pp. 55–59, 2018.
- [7] M. Lutz, *Python Pocket reference*, 5th ed. Sebastopol, United States: O'Reilly Media, Inc., 2014.
- [8] E. Hutto, C.J. and Gilbert, "VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text," *Eighth Int. AAAI Conf. Weblogs Soc. Media*, p. 18, 2014, [Online]. Available: <https://www.aaai.org/ocs/index.php/ICWSM/ICWSM14/paper/viewPaper/8109>.
- [9] C.J. Hutto, "VADER-Sentiment-Analysis," *GitHub*, 2014. <https://github.com/cjhutto/vaderSentiment> (accessed Mar. 15, 2022).
- [10] Bustami, "Penerapan Algoritma Naive Bayes," *J. Inform.*, vol. 8, no. 1, pp. 884–898, 2014.
- [11] A. R. T. Lestari, R. S. Perdana, and M. A. Fauzi, "Analisis Sentimen Tentang Opini Pilkada DKI 2017 Pada Dokumen Twitter Berbahasa Indonesia Menggunakan Naive Bayes dan Pembobotan Emoji," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 1, no. 12, pp. 1718–1724, 2017, [Online]. Available: <http://j-ptiik.ub.ac.id>.
- [12] F. Gunawan, M. A. Fauzi, and P. P. Adikara, "Analisis Sentimen Pada Ulasan Aplikasi Mobile Menggunakan Naive Bayes dan Normalisasi Kata Berbasis Levenshtein Distance (Studi Kasus Aplikasi BCA Mobile)," *Syst. Inf. Syst. Informatics J.*, vol. 3, no. 2, pp. 1–6, 2017, doi: 10.29080/systemic.v3i2.234.