SVM Vaksin

by Layla Layla

Submission date: 19-Jul-2021 01:36PM (UTC+0700)
Submission ID: 1621473104
File name: Jurnal_17410100190_pdf_-_Copy.pdf (646.5K)
Word count: 4150
Character count: 26017

Analisis Sentimen Tanggapan Masyarakat Terhadap Vaksin Covid-19 Menggunakan Algoritma *Support Vector Machine* (SVM)

Layla Qodary Zalyhaty [1] **Vivine Nurcahyawati** [2] **Erwin Sutomo** [3]
Program Studi/Jurusan Sistem Informasi
Universitas Dinamika Surabaya
Jl. Raya Kedung Baruk 98 Surabaya, 60298
Email : 1)17410100190@dinamika.ac.id, 2)vivine@dinamika.ac.id , 3)sutomo@dinamika.ac.id

Abstract: *The COVID-19 outbreak has had a significant impact on health and economic sectors. The Government has been working to solve the problems, one of which is by procuring the COVID-19 vaccine. The provision of COVID-19 vaccine aims to reduce the transmission of coronavirus, lower the rate of pain and death, achieve herd immunity. But in the midst of the birth of the COVID-19 vaccine, there are pros and cons in the community. Some support vaccines, and others doubt the effectiveness and efficacy of the COVID-19 vaccine. Some of them even rejected vaccines even though the government gave vaccines for free. The public gave their responses and opinions in various media. One of the media that is always updated about various developments is online news. Online news contains responses and public opinion both negative and positive related to a topic. In order for vaccination to run optimally, the government needs to consider various inputs, among others, by looking at how the response and public opinion to the vaccination discourse,so that the government can evaluate and determine the next strategy related to education and socialization about the COVID-19 vaccine to the community. The result of evaluation F1 score is 88.37%, accuracy score 82.76%, precision score 79.17%, and recall 100%. Pie chart shows the percentage result of positive sentiment worth 70% and negative worth 30% of the overall data.*

Keywords: *Sentiment Analysis, COVID-19 vaccine, online news, Support Vector Machine.*

Presiden Indonesia mengumumkan wabah COVID-19 pertama di Indonesia pada bulan Maret 2020. Wabah ini berdampak sangat besar pada sektor Kesehatan dan Perekonomian Indonesia. Meskipun pusat penyebaran virus tersebut dimulai pada akhir tahun 2019 lalu berasal dari kota Wuhan, China kini virus tersebut telah tersebar menjangkit seluruh masyarakat dunia dengan jumlah kasus sebanyak lebih dari 121 juta kasus dan jumlah kematian sebanyak lebih dari 2,68 juta jiwa pertanggal 18 Maret 2021 (WHO, 2021).

Menyikapi hal tersebut, pemerintah Indonesia telah berupaya secara maksimal mengatasi tantangan yang ada, salah satunya adalah dengan pengadaan vaksin COVID-19 untuk masyarakat Indonesia. Vaksin mengandung agen yang menyerupai virus atau bakteri yang sudah mati atau dilemahkan. Vaksin ini merangsang sistem imun di dalam tubuh untuk mengenalinya sebagai agen asing, menghancurkannya. Misalkan akhirnya tetap sakit, maka sakitnya tidak akan terlalu berat (Febriyani, 2021).

Tujuan diberikannya vaksin adalah untuk mengurangi penularan virus corona, menurunkan angka kematian dan mencapai kekebalan kelompok, supaya masyarakat bisa tetap produktif.

Ditengah kemunculan vaksin COVID-19, muncul pro dan kontra di masyarakat. Beberapa orang mendukung vaksin, sementara ada juga yang meragukan keampuhan vaksin, dan bahkan ada yang menolak meskipun vaksin diberikan secara gratis (Putri, 2020). Berdasarkan hasil survei yang dilakukan oleh Saiful Mujani *Research and Consulting* (SMRC), pada tanggal 28 Februari 2021 – 8 Maret 2021 yang mencakup seluruh provinsi dengan melibatkan 1220 responden yang dipilih secara acak, masyarakat daerah DKI Jakarta paling banyak menolak vaksinasi COVID-19 yaitu sebanyak 33 persen, penolakan terbesar kedua di daerah Jawa Timur dengan nilai 32 persen, lalu Banten 31 persen, sementara penolakan terendah berada di Jawa Tengah yaitu 20 persen (Wibowo, 2021).

Masyarakat yang telah maupun yang belum divaksin memberikan respon dan opininya di berbagai media. Salah satu media yang selalu

update tentang berbagai perkembangan adalah berita online. Media berita online menyediakan informasi yang *up to date* tentang berbagai peristiwa berhubungan dengan kehidupan sehari-hari seperti Pendidikan, olahraga, politik, dan teknologi (A Soedomo, 2005).

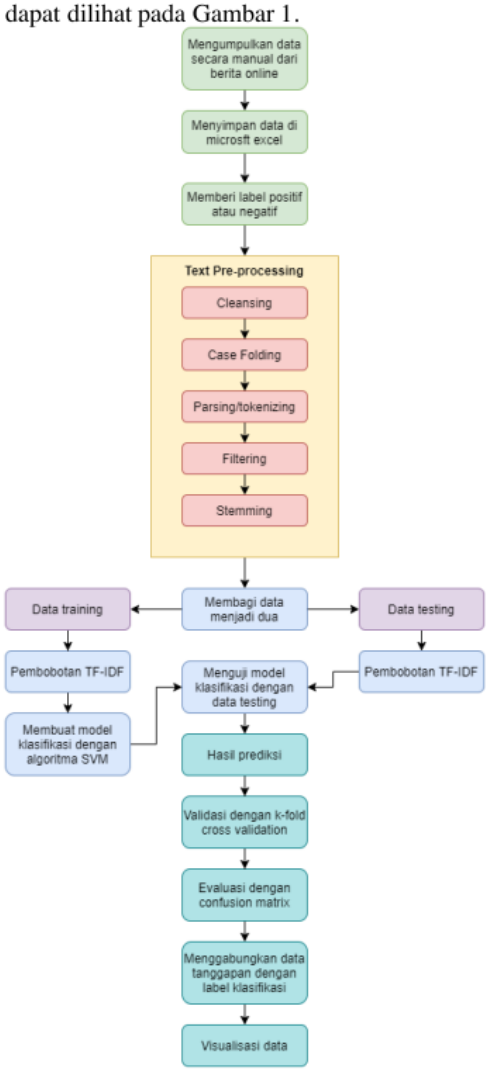
Dengan adanya kebebasan berpendapat bisa menimbulkan berbagai jenis opini yang cenderung negatif maupun positif. Supaya vaksinasi dapat berjalan maksimal, pemerintah perlu mempertimbangkan berbagai masukan, diataranya adalah dengan melihat bagaimana respon dan opini masyarakat terhadap wacana vaksinasi tersebut. Opini masyarakat tentang vaksin COVID-19 pada media berita online perlu dikaji dalam pemrosesan teks. Dalam menyaring opini-opini masyarakat perlu dilakukan proses analisis sentimen untuk mengklasifikasikan opini menjadi kelas positif atau kelas negatif. Hasil klasifikasi tersebut bisa digunakan untuk membantu pemerintah untuk mengetahui tanggapan ataupun kekhawatiran masyarakat terhadap vaksin COVID-19, sehingga pemerintah bisa melakukan evaluasi dan menentukan strategi selanjutnya terkait edukasi maupun sosialisasi tentang vaksin COVID-19 kepada masyarakat.

Dalam melakukan analisis sentimen diperlukan algoritma untuk menunjang klasifikasi sentimen. Pada penelitian ini menggunakan *Support vector machine* (SVM) untuk mengklasifikasikan respon & opini masyarakat Indonesia terhadap vaksin COVID-19 kedalam sentimen positif & negatif, dengan sumber data dari media berita online. Data yang diambil dari media berita online dipisahkan menjadi data *training* (untuk membuat model klasifikasi dengan algoritma SVM) dan data *testing* (untuk menguji model klasifikasi yang telah dibuat)

Berdasarkan penelitian-penelitian analisis sentimen sebelumnya, algoritma SVM menghasilkan akurasi yang cukup tinggi dalam melakukan klasifikasi sentimen. Penelitian yang dilakukan oleh Nurrun Muchammad Shiddieqy Hadna, Paulus Insap Santosa, dan Wing Wahyu Winarno ditemukan bahwa algoritma SVM yang dikolaborasikan dengan ekstraksi fitur TF-IDF menghasilkan akurasi sebesar 81.5%, lebih unggul daripada algoritma *Naïve Bayes* yang menghasilkan akurasi 80.8% dalam Studi Literatur Tentang Perbandingan Metode Untuk Proses Analisis Sentimen di Twitter.

METODE PENELITIAN

Dalam penelitian ini menggunakan *software Microsoft excel* dan bahasan pemrograman *Python* dengan media *Google Collaboratory*. Metode penelitian terdiri dari Tahap Awal, Analisis Data dan Tahap Akhir. Tahap Awal berisi studi literatur dan pengumpulan data. Selanjutnya Analisis Data dapat dilihat pada Gambar 1.



Gambar 1 Diagram Alir Analisis Data

Tahap Analisis Data dimulai dari pengumpulan data tanggapan masyarakat secara manual dari berita online dan menyimpannya ke dalam *Microsoft excel* dalam format .csv. Setelah itu proses pelabelan data dengan memberikan label positif dan negatif dengan panduan *Load Dictionary* untuk menetukan kata positif dan kata negatif.

Setelah data memiliki label, masukkan data melalui *google collaboratory* kedalam *python* dengan menggunakan *library pandas*

untuk *read* data dalam bentuk .csv. Setelah data berhasil dimasukkan, tahap selanjutnya adalah tahap *text pre-processing* yang gunanya untuk membersihkan data. *Text pre-processing* terdiri dari *cleansing, case folding, parsing/tokenizing, filtering* dan *stemming*. Dalam *python text pre-processing* menggunakan *library string, re, nltk* dan Sastrawi. Selanjutnya data dibagi menjadi dua bagian yaitu data *training* dan data *testing* dengan perbandingan 90:10, pembagian data menggunakan *library sklearn*.

Setelah itu data *training* dan data *testing* dilanjutkan ke tahap pembobotan TF-IDF yang gunanya untuk mengubah data teks menjadi data numerik supaya bisa dilakukan perhitungan dan juga untuk menghitung bobot setiap kata, semakin besar bobot suatu kata maka kata tersebut dianggap penting. Pembobotan TF-IDF juga berguna untuk penyaringan data karena kata yang memiliki bobot akan diproses untuk tahap selanjutnya, sedangkan kata yang bernilai 0 maka otomatis tidak akan diproses maupun ditampilkan. Dalam *python* proses pembobotan TF-IDF menggunakan *library TfidfVectorizer*.

Langkah selanjutnya adalah membuat model klasifikasi menggunakan algoritma SVM dengan menggunakan data *training*. Dalam *python* proses pembuatan model klasifikasi algoritma SVM menggunakan fungsi *LinearSVC* dari *library sklearn*. Setelah model terbentuk, saatnya untuk menguji model klasifikasi dengan data *testing*. Data *testing* yang dimasukkan ke dalam model klasifikasi hanya bagian kolom tanggapan saja tanpa menggunakan kolom label karena model klasifikasi digunakan untuk memprediksi label dari data *testing*.

Setelah hasil prediksi diketahui, dilakukan uji validasi model klasifikasi menggunakan *k-fold cross validation*. Caranya adalah dengan mengulangi sampel data uji beberapa kali dengan sampel yang berbeda-beda pada setiap pengujian. Pada *python* untuk proses *cross validation* dan *confusion matrix* menggunakan *library sklearn*.

Tahapan selanjutnya adalah mengevaluasi hasil klasifikasi dengan menggunakan *confusion matrix* untuk mengetahui nilai *true positive, true negative, false positive,* dan *false negative*. Hasil dari *confusion matrix* digunakan untukmenghitung nilai akurasi, presisi, *recall* dan F1.

Tahap selanjutnya adalah menyatukan data tanggapan(kolom tanggapan) dengan label hasil prediksi karena *output* dari hasil klasifikasi hanya menampilkan label. Setelah kolom tanggapan dan label disatukan, tahap terakhir adalah memvisualisasikan data dalam bentuk *WordCloud* dan diagram *pie*. *WordCloud* digunakan untuk menampilkan masing-masing kata positif dan negatif yang paling sering digunakan. Sedangkan diagram pie untuk menampilkan persentase sentimen positif dan negatif dari keseluruhan data hasil prediksi. Pada *python* untuk memvisualisasikan *WordCloud* menggunakan *library WordCloud* dan diagram *pie* menggunakan *library matplotlib*.

Pelabelan

Tahap pelabelan ini dilakukan secara manual dalam *Microsoft excel* dengan bantuan 2(dua) sukarelawan yang masing-masing melabeli sebanyak 50% data. Pelabelan dilakukan pada keseluruhan data. Label yang diberikan berupa positif atau negatif pada masing-masing tanggapan masyarakat pada keseluruhan data dengan panduan *Load Dictionary* untuk menentukan kata positif dan kata negatif.

Text Preprocessing

Text *Pre-processing* gunanya untuk membersihkan data, menghilangkan dan mengatasi data yang kotor, mengatasi informasi yang hilang, menghapus kata yang tidak digunakan dan mengubahnya kata menjadi kata dasar (Wardani, 2019). *Text pre-processing* terdiri dari *cleansing, case folding, parsing/tokenizing, filtering* dan *stemming*. Tahap ini bertujuan untuk mengoptimalkan hasil perhitungan kedepannya. Berikut ini adalah langkah-langkah yang harus dilakukan dalam pengolahan teks (Pravina, Cholissodin, & Adikara, 2019) :

Cleansing

Proses penghapusan tanda baca seperti .,"!?#@%(), menghapuskan URL atau *link*, dan menghapuskan angka (Pravina, Cholissodin, & Adikara, 2019). Proses ini dilakukan dengan menggunakan *library string* dan *re* dan juga fungsi *remove punctuation* di *python*.

Case folding

Proses *case folding* menggunakan menggunakan *library re* dan fungsi lower() untuk menyeragamkan seluruh huruf menjadi huruf kecil/*lowercase* (Pravina, Cholissodin, & Adikara, 2019).

Tokenizing

Proses *tokenizing* menggunakan *library nltk* untuk misahkan atau membagi teks yang awalnya berupa kalimat menjadi token atau *term* (Ulfah & Anam, 2020).

Stopword Removal atau Filtering

Proses ini gunanya untuk menghapus kata-kata yang tidak memiliki makna seperti *stoplist/stopword,* dengan menggunakan fungsi *StopWordRemover* dari *library* Sastrawi di *pyhton* (Safitri, 2020).

Stemming

Proses mengubah kata dalam dokumen menjadi kata dasar (*root word*), menghapus prefiks (imbuhan awalan) dan sufiks(imbuhan akhiran) (Pravina, Cholissodin, & Adikara, 2019). Proses ini menggunakan fungsi yang sudah disediakan oleh *python* yaitu *StemmerFactory* dari *library* Sastrawi.

Pembagian Data

Total keseluruhan data yang telah dibersihkan sejumlah 283 tanggapan, kemudian dibagi menjadi dua, data *training* sebanyak 90% yaitu 254 dan data *testing* 10% yaitu 29. Pembagian data pada *python* menggunakan *library sklearn*.

Pembobotan TF-IDF

Pembobotan *Term Frequency-Inverse Document Frequency* TF-IDF yang gunanya untuk mengubah data teks menjadi data numerik supaya bisa dilakukan perhitungan dan juga untuk menghitung bobot setiap kata, semakin besar bobot suatu kata maka kata tersebut dianggap penting. Pembobotan TF-IDF juga berguna untuk penyaringan data karena kata yang memiliki bobot akan diproses untuk tahap selanjutnya, sedangkan kata yang bernilai 0 maka otomatis tidak akan diproses ke tahap selanjutnya. Nilai TF-IDF dapat ditemukan menggunakan rumus berikut (Pravina, Cholissodin, & Adikara, 2019):

1. Menentukan nilai TF bisa menggunakan TF biner. Jika suatu kata atau term terdapat dalam sebuah dokumen maka diberi nilai 1 (satu), jika tidak ada maka diberi nilai 0 (nol).
2. Menghitung nilai *Inverse Document Frequency* (IDF) untuk melihat seberapa penting kata dalam sebuah dokumen. IDF dirumuskan sebagai berikut :

$$IDF = log\frac{D}{df} \quad (1)$$

3. Setelah itu menghitung TF IDF dengan menggabungkan perhitungan TF dengan IDF sebagai berikut :

$$W_{ij} = tf \text{ x } IDF \quad (2)$$

Keterangan :
Wij = bobot kata dalam setiap dokumen
tf = jumlah kemunculan kata dalam dokumen
D = jumlah total seluruh dokumen

Klasifikasi Support Vector Machine

Algoritma *Support Vector Machine* termasuk kedalam pembelajaran mesin (*supervised learning*) yang bisa memprediksi atau mengklasifikasi kelas berdasarkan model prediksi yang dibuat dari data *training* untuk mendapatkan pola yang nantinya bisa digunakan untuk memprediksi pada proses pelabelan (Pravina, Cholissodin, & Adikara, 2019).

Tahapan algoritma SVM adalah :

1. Meminimalkan nilai margin dengan rumus :

$$W_i.x_i + b = 0 \quad (3)$$

2. Jika nilai *wi* berada di kelas +1 maka masuk kelas positif

$$w_i.x_i + b \geq +1 \quad (4)$$

3. Jika nilai *wi* berada di kelas -1 maka masuk kelas negatif

$$w_i.x_i + b \geq -1 \quad (5)$$

4. Kernel yang umum digunakan adalah kernel Linear dengan rumus :

$$K(x,y) = x.y \quad (6)$$

Keterangan :
$i = 1,2,3,\ldots$
x = data ke-i
w = bobot

Validasi K-fold Cross Validation

Uji validasi dengan *k-fold cross validation* caranya adalah dengan mengulangi sampel data uji beberapa kali dengan sampel yang berbeda-beda pada setiap pengujian. Misalkan dengan 10-*fold cross validation* maka menggunakan sampel data testing sebanyak 10% (dihasilkan dari total 100% dibagi 10 *fold*) sebanyak 10 kali dengan sampel yang berbeda-beda. Prosedur ini merupakan cara umum untuk menguji validasi model klasifikasi (Wardani, 2019).

Dalam evaluasi klasifikasi, ada empat kemungkinan hasil klasifikasi data. Jika data aslinya positif dan diprediksi positif masuk kedalam *true positive*, dan jika data aslinya

positif diprediksi negatif masuk kedalam *false negative*. Jika data aslinya negatif diprediksi negatif, itu dihitung sebagai *true negative*, dan jika data aslinya negatif diprediksi positif termasuk *false positive* (Fawcett, 2006).

Tabel 1 *Confusion Matrix*

Class		Aktual	
		Positive	*Negative*
Prediksi	*Positive*	*True Positive* (TP)	*False Positive* (FP)
	Negative	*False Negative* (*FN*)	*True Negative* (TN)

Berdasarkan *matrix confusion*, maka dapat dihitung nilai *accuracy*, *precision*, dan *recall*. *Accuracy* menggambarkan hasil klasifikasi sentimen positif dan negatif yang benar dari keseluruhan data (Hadna, Santosa, & Winarto, 2016). Dirumuskan sebagai berikut:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (6)$$

Precision menggambarkan persentase klasifikasi sentimen positif yang benar-benar bernilai positif (Hadna, Santosa, & Winarto, 2016), dirumuskan :

$$Precision = \frac{TP}{TP+FP} \quad (6)$$

Recall menggambarkan persentase tanggapan yang klasifikasikan positif dari seluruh tanggapan yang bernilai positif (Hadna, Santosa, & Winarto, 2016) dirumuskan sebagai berikut:

$$Recall = \frac{TP}{TP+FN} \quad (8)$$

F1 adalah rata rata dari perbandingan presisi dan *recall*. F1 baik untuk digunakan sebagai acuan performansi algoritmajika jumlah *False Negative* dan *False Positive* berbeda jauh. F1 dirumuskan dengan :

$$F1 = \frac{2(Recall\ x\ Presisi)}{Recall + Presisi} \quad (9)$$

Visualisasi Data dengan WordCloud

WordCloud akan memperlihatkan kata yang paling sering muncul/digunakan pada data yang telah dianalisis sebelumnya. Visualisasi dengan *WordCloud* ditampilkan berdasarkan sentimen negatif dan positif. Dalam *python* untuk memvisualisasikan *WordCloud* menggunakan *library WordCloud*. Untuk menghasilkan *WordCloud* digunakan data hasil dari proses klasifikasi SVM bagian kolom tanggapan masyarakat. Ukuran kata yang semakin besar menandakan kata tersebut sering muncul/digunakan dalam teks tersebut.

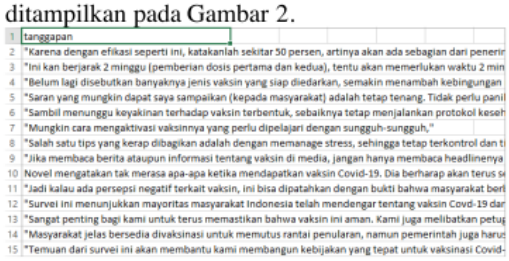
Visualisasi Data dengan Diagram Pie

Diagram *pie* menunjukkan persentase hasil tanggapan positif dan negatif terhadap vaksin COVID-19 menggunakan *python*. Data yang digunakan untuk membuat diagram adalah data hasil dari proses klasifikasi SVM karena sudah memiliki label positif dan negatif. Untuk membuat diagram dibutuhkan nilai dari jumlah keseluruhan data dan jumlah tanggapan yang dinilai positif maupun negatif, bisa didapatkan dari menghitung jumlah tanggapan positif dan negatif. Lalu data tersebut diproses dengan menggunakan *library matplotlib* yang ada di *python* untuk membuat diagram *pie*.

HASIL DAN PEMBAHASAN

Pengambilan Data

Pengambilan data dilakukan secara manual dengan mencari berita online dengan topik tanggapan masyarakat terhadap vaksin COVID-19 dari tahun 2020-2021. Total data yang ditemukan sejumlah 283 data. Data yang telah ditemukan kemudian disimpan dalam *Microsoft Excel*. Hasil data yang telah dikumpulkan ditampilkan pada Gambar 2.



Gambar 2 Tanggapan Masyarakat

Pelabelan Data

Tahap pelabelan ini dilakukan secara manual dalam *Microsoft excel* dengan bantuan 2(dua) sukarelawan yang masing-masing melabeli sebanyak 50% data. Pelabelan dilakukan pada keseluruhan data. Label yang diberikan berupa positif atau negatif pada masing-masing tanggapan masyarakat pada keseluruhan data dengan panduan *Load Dictionary* untuk menentukan kata positif dan kata negatif. Hasil pelabelan data ditampilkan pada Gambar 3.

	tanggapan	label
1	"Karena dengan efikasi seperti ini, katakanlah sekitar 50 pe	negatif
2	"Ini kan berjarak 2 minggu (pemberian dosis pertama dan k	positif
3	"Belum lagi disebutkan banyaknya jenis vaksin yang siap di	negatif
4	"Saran yang mungkin dapat saya sampaikan (kepada masya	positif
5	"Sambil menunggu keyakinan terhadap vaksin terbentuk, s	positif
6	"Mungkin cara mengaktivasi vaksinnya yang perlu dipelajar	positif
7	"Salah satu tips yang kerap dibagikan adalah dengan mema	positif
8	"Jika membaca berita ataupun informasi tentang vaksin di i	positif
9	Novel mengatakan tak merasa apa-apa ketika mendapatkar	positif

Gambar 3 Hasil Pelabelan Data

Text Preprocessing

Tahap ini melakukan pengolahan terhadap keseluruhan data. Tahap ini dimulai dari *cleansing, case folding, parsing/tokenizing, filtering dan stemming*. Sebelum melakukan *text pre-processing* data di-inputkan terlebih dahulu melalui *python* menggunakan *library pandas*. Setelah itu tampilkan isi dari data yang telah diinputkan untuk mengecek keberhasilan input data. Jika data telah berhasil di-inputkan maka data bisa ditampilkan seperti pada Gambar 4.

	tanggapan	label
0	"Karena dengan efikasi seperti ini, katakanlah...	negatif
1	"Ini kan berjarak 2 minggu (pemberian dosis pe...	positif
2	"Belum lagi disebutkan banyaknya jenis vaksin ...	negatif
3	"Saran yang mungkin dapat saya sampaikan (kepa...	positif
4	"Sambil menunggu keyakinan terhadap vaksin ter...	positif

Gambar 4 Data Sebelum *Text Pre-processing*

Sebelum dilakukan *text preprocessing*, kolom label diganti menjadi angka 1 untuk positif dan 0 untuk negatif supaya memudahkan proses TF-IDF nantinya. Selanjutnya tahap pertama dalam *text pre-processing* adalah *cleansing*. *Cleansing* bertujuan untuk menghapus tanda baca yang tidak dibutuhkan untuk analisis sentimen. Pada *python* proses *cleansing* dilakukan menggunakan *library string* dan *re* untuk menghapus tanda baca. Tampilan data setelah melalui tahap *cleansing* dapat dilihat pada Gambar 5.

	tanggapan	label
0	Karena dengan efikasi seperti ini katakanlah...	0
1	Ini kan berjarak minggu pemberian dosis per...	1
2	Belum lagi disebutkan banyaknya jenis vaksin ...	0
3	Saran yang mungkin dapat saya sampaikan kepa...	1
4	Sambil menunggu keyakinan terhadap vaksin ter...	1

Gambar 5 Setelah Melalui Tahap *Cleansing*

Setelah melalui tahap *cleansing*, langkah selanjutnya adalah *case folding* yang bertujuan untuk menyeragamkan seluruh huruf menjadi huruf kecil (*lowercase*). Pada *python* proses *case folding* dilakukan menggunakan fungsi lower() untuk mengubah semua huruf menjadi huruf kecil. Data yang telah melalui tahap *case folding* ditampilkan pada Gambar 6.

	tanggapan	label
0	karena dengan efikasi seperti ini katakanlah...	0
1	ini kan berjarak minggu pemberian dosis per...	1
2	belum lagi disebutkan banyaknya jenis vaksin ...	0
3	saran yang mungkin dapat saya sampaikan kepa...	1
4	sambil menunggu keyakinan terhadap vaksin ter...	1

Gambar 6 Setelah Melalui Tahap *Case Folding*

Selanjutnya adalah tahap *parsing/tokenizing* yaitu mengubah kalimat menjadi beberapa kata atau token. Pada *python* proses *parsing/tokenizing* dilakukan menggunakan *library nltk* dengan fungsi *tokenize* untuk mengubah kalimat menjadi token. Data yang telah melalui proses *parsing/tokenizing* ditampilkan pada Gambar 7.

	tanggapan	label
0	[karena, dengan, efikasi, seperti, ini, kataka...	0
1	[ini, kan, berjarak, minggu, pemberian, dosis,...	1
2	[belum, lagi, disebutkan, banyaknya, jenis, va...	0
3	[saran, yang, mungkin, dapat, saya, sampaikan,...	1
4	[sambil, menunggu, keyakinan, terhadap, vaksin...	1

Gambar 7 Hasil Tahap *Tokenizing*

Tahap berikutnya adalah tahap *filtering* yang bertujuan untuk menghilangkan kata yang tidak bermakna (*stopword/stoplist*) seperti kata hubung atau kata ganti. Pada *python* proses *filtering* dilakukan menggunakan *library* nltk untuk men-*download* kamus *stopwords* dan *library* Sastrawi dengan fungsi *StopWordRemover* untuk menghapus kata hubung atau kata ganti. Data yang telah melalui tahap *filtering* ditunjukkan pada Gambar 8.

	tanggapan	label
0	efikasi persen penerima vaksin memiliki proteksi	0
1	berjarak minggu pemberian dosis waktu minggu p...	1
2	banyaknya jenis vaksin diedarkan menambah kebi...	0
3	saran masyarakat tenang panik sabar bijaksana ...	1
4	menunggu keyakinan vaksin terbentuk menjalanka...	1

Gambar 8 Hasil *Stopword Removal*

Tahap terakhir dalam *text pre-processing* adalah *stemming*, yaitu proses pengubahan kata menjadi kata dasar dengan

menghilangkan imbuhan. Pada *python* proses *stemming* dilakukan menggunakan *library* Sastrawi dengan fungsi Stemmer untuk mengubah kata menjadi kata dasar. Data yang telah melalui tahap stemming ditampilkan pada Gambar 9.

	tanggapan	label
0	efikasi persen terima vaksin milik proteksi	0
1	jarak minggu beri dosis waktu minggu sunti wak...	1
2	banyak jenis vaksin edar tambah bingung masyar...	0
3	saran masyarakat tenang panik sabar bijaksana ...	1
4	tunggu yakin vaksin bentuk jalan protokol sehat	1

Gambar 9 Hasil Tahap *Stemming*

Total keseluruhan data yang telah dibersihkan sejumlah 283 tanggapan, kemudian dibagi menjadi dua, data *training* sebanyak 90% yaitu 254 dan data *testing* 10% yaitu 29 data *testing*. Pembagian data pada *python* menggunakan *library sklearn*. Hasil pembagian data *training* dan data *testing* ditampilkan pada Gambar 10 dan Gambar 11.

```
(195    antre tanah abang lansia animo sungguh dukung ...
 235    obat vaksin bebas risiko astrazeneca manfaat r...
 14     vaksin covid pakai masker cuci tangan jaga jar...
 103     gratis terap efek samping takut ragu aman vaksin
 22     uji praklinik aman lanjut fase uji klinik bada...
                          ...
 263    bijak masyarakat malas vaksinasi gratis malas ...
 168    nego negara produsen vaksin mudah mudah mei no...
 36     tambah isu vaksin cina masyarakat kualitas vak...
 96     baik sehat sih tuju aja rencana perintah dampa...
 26     putus vaksin gratis komitmen presiden jokowi d...
Name: tanggapan, Length: 254, dtype: object,
```

Gambar 10 Data *Training*

```
1      jarak minggu beri dosis waktu minggu sunti wak...
228    harap upaya tingkat percaya masyarakat program...
183    prinsip vaksin masyarakat izin badan pom aspek...
151    masyarakat non lansia usia tahun damping lansi...
126    timbang situasi kondisi kait sedia vaksin peng...
180    paham masyarakat keliru kait bebas traveling v...
61     kondisi alamiah psikologis umum kelompok muda ...
49     sosialisasi tangan covid kait vaksinasi protok...
249    senang berita datang vaksin covid bal kalo gra...
172    vaksinasi ganti protokol sehat traveling takut...
170          teliti lindung virus tahan vaksinasi penuh
Name: tanggapan, dtype: object,
```

Gambar 11 Data *Training*

Pembobotan Term Frequency-Inverse Document Frequency

Data yang telah dibagi menjadi data *training* dan data *testing* dilanjutkan proses pembobotan TF-IDF. TF merupakan banyaknya kata yang muncul pada satu dokumen, sedangkan IDF adalah banyaknya kata yang muncul pada seluruh dokumen. TF-IDF didapatkan dari mengalikan nilai TF dengan IDF. Pengolahan TF-IDF di *python* mengunakan *library* *TfidfVectorizer*. Fungsi dari tahap ini adalah mengubah data teks menjadi data numerik supaya bisa dilakukan perhitungan dan juga untuk menghitung bobot setiap kata, semakin besar bobot suatu kata maka kata tersebut dianggap penting. Pembobotan TF-IDF juga berguna untuk penyaringan data karena kata yang memiliki bobot akan diproses untuk tahap selanjutnya, sedangkan kata yang bernilai 0 maka otomatis tidak akan diproses maupun ditampilkan. Hasil pembobotan TF-IDF dapat dilihat pada Gambar 12.

```
  (0, 923)      0.08405592590509782
  (0, 502)      0.18109085213970988
  (0, 212)      0.3499258239258101
  (0, 802)      0.38617022723412187
  (0, 45)       0.4149390286058649
  (0, 457)      0.2768716549997068
  (0, 1)        0.36575843782222295
  (0, 824)      0.36575843782222295
  (0, 53)       0.4149390286058649
  (1, 637)      0.3743539125524612
  (1, 917)      0.31569963840205295
  (1, 490)      0.27807380744487803
  (1, 66)       0.31569963840205295
  (1, 709)      0.6080574744618541
  (1, 94)       0.28561342148149105
```

Gambar 12 Hasil Pembobotan TF-IDF

Klasifikasi Support Vector Machine

Pada tahap ini, algoritma SVM digunakan untuk membuat model klasifikasi. Algoritma SVM merupakan jenis pembelajaran mesin yang bisa memprediksi atau mengklasifikasi kelas berdasarkan model prediksi yang dibuat dari data *training* untuk mendapatkan pola yang nantinya bisa digunakan untuk memprediksi pada proses pelabelan. Model klasifikasi algoritma SVM menggunakan data *training* ditampilkan pada Gambar 13.

```
  (0, 145)        0.21114364134752145
  (0, 198)        0.45794904756864613
  (0, 369)        0.44061050050165573
  (0, 550)        0.5450871252650836
  (0, 588)        0.3087756782697215
  (0, 858)        0.38296776014776035
  (0, 868)        0.11152004512143725
  (1, 17)         0.2295121383707279
  (1, 24)         0.19964439439434314
  (1, 56)         0.24698364857805316
  (1, 91)         0.4590242767414558
  (1, 121)        0.20750063868476737
  (1, 145)        0.09567099367593011
  (1, 416)        0.24698364857805316
  (1, 423)        0.2295121383707279
  (1, 438)        0.18217288418701788
```

Gambar 13 Model Klasifikasi Dengan Algoritma SVM

Model klasifikasi yang sudah terbentuk sudah dapat digunakan untuk memprediksi label dari data testing. Setelah proses pembobotan TF-IDF selesai, data testing dimasukkan ke dalam model klasifikasi. Data testing yang dimasukkan kedalam model klasifikasi hanyalah bagian kolom tanggapannya saja tanpa kolom label karena bagian labelnya yang akan di prediksi oleh model klasifikasi. Hasil klasifikasi dari algoritma SVM dapat dilihat pada Gambar 14.

	tanggapan	label
0	kaji efektivitas vaksin cipta kebal individu t...	positif
1	perintah overclaim vaksin olah sedia solusi sa...	positif
2	jarak minggu beri dosis waktu minggu sunti wak...	positif
3	amelia dapat program vaksinasi covid hasil edu...	positif
4	aneh situasi darurat kian buruk perintah pakai...	positif
...	...	...
80	vaksinasi takut efek samping takut daya tubuh ...	negatif
81	tes swab rapid rasa apa gejala vaksin	positif
82	senang berita datang vaksin covid bal kalo gra...	positif
83	harap upaya tingkat percaya masyarakat program...	positif
84	buruh jadi uji coba vaksin vaksin halal aman	positif

Gambar 14 Hasil Klasifikasi Algoritma SVM

Validasi K-fold Cross Validation

Hasil pengujian dari model yang telah ditetapkan perlu dilakukan validasi untuk mengetahui seberapa baik model yang telah dibuat dan menggambarkan tingkat akurasi analisis yang dilakukan. Uji validasi dengan *k-fold cross validation* caranya adalah dengan mengulangi sampel data uji beberapa kali dengan sampel yang berbeda-beda pada setiap pengujian. Misalkan dengan 5-*fold cross validation* maka menggunakan sampel data testing sebanyak 20% (dihasilkan dari total 100% dibagi 5 *fold*) sebanyak 5 kali dengan sampel yang berbeda-beda. Validasi model klasifikasi dilakukan menggunakan 5-*fold cross validation* ditampilkan pada Tabel 2.

Tabel 2 5-*fold cross validation*

Fold ke	*Cross Validation Score*
1	72.55%
2	74.51%
3	72.55%
4	66.67%
5	84%
Rata-rata	74.05%

Evaluasi Confusion Matrix

Evaluasi dengan *confusion matrix* untuk mengetahui hasil klasifikasi *true positive, true negative, false positive dan false negative*. Hasil *confusión matrix* ditampilkan pada Tabel 3.

Tabel 3 *Confusion Matrix*

Class		Aktual	
		Positive	*Negative*
Prediksi	*Positive*	*True Positive* (TP) 19	*False Positive* (FP) 5
	Negative	*False Negative* (*FN*) *0*	*True Negative* (TN) 5

Akurasi digunakan untuk mengetahui berapa persen hasil klasifikasi sentimen positif dan negatif yang benar dari keseluruhan data. Akurasi digunakan untuk menjawab "berapa persen tanggapan yang benar diprediksi positif dan benar diprediksi negatif dari keseluruhan tanggapan?". Presisi digunakan untuk mengetahui berapa persen klasifikasi sentimen positif yang benar-benar bernilai positif. Presisi digunakan untuk menjawab "berapa persen tanggapan yang benar-benar positif dari keseluruhan tanggapan yang diprediksi positif?". Sedangkan *Recall* untuk mengetahui berapa persen tanggapan yang diprediksi positif dari seluruh tanggapan positif. *Recall* dapat digunakan untuk menjawab "berapa persen tanggapan yang diprediksi positif dibandingkan keseluruhan tanggapan yang aslinya positif?". F1

menunjukkan rata-rata perbandingan dari presisi dan *recall* yang di bobotkan. Nilai F1 100% menunjukkan nilai *presisi-recall* yang sempurna. Hasil dari evaluasi ditampilkan pada Tabel 4.

Tabel 4 Hasil Evaluasi

Validation	*Score*
F1	88.37%
Accuracy	82.76%
Precision	79.17%
Recall	100%

Visualisasi Data dengan WordCloud

Berdasarkan Gambar 15, diketahui bahwa kata negatif lah yang sering muncul yaitu kata *"vaksinasi", "percaya" dan "sehat".*

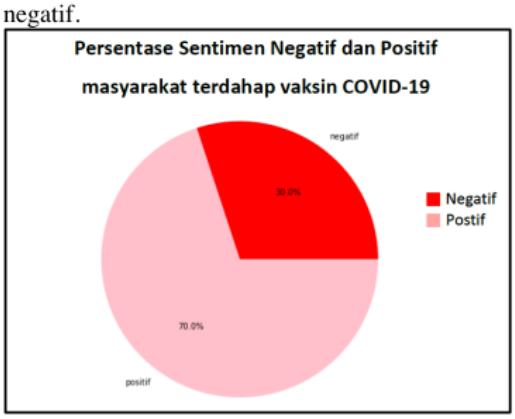
Gambar 15 *WordCloud* Positif

Berdasarkan Gambar 16, diketahui bahwa kata positif lah yang sering muncul yaitu kata "bayar", "tolak" dan "khawatir".

Gambar 16 *WordCloud* Negatif

Visualisasi Data dengan Diagram Pie

Pada Gambar 17, menunjukkan persentase sentimen masyarakat berdasarkan klasifikasi SVM. Persentase keseluruhan tanggapan masyarakat terhadap Vaksin Covid-19 adalah sebanyak 70% positif dan sebanyak 30% negatif.



Gambar 17 Visualisasi Diagram Pie

KESIMPULAN

Berdasarkan penelitian analisis sentimen tanggapan masyarakat terhadap vaksin COVID-19 menggunakan algoritma SVM yang telah dilakukan, disimpulkan:

1. Dari total 283 data tanggapan masyarakat terhadap vaksin COVID-19 dengan perbandingan 90:10 data *training* sejumlah 254 dan data *testing* sejumlah 29 menghasilkan persentase 70% untuk sentimen positif dan 30% untuk sentimen negatif.
2. Hasil validasi dan evaluasi yang rendah bisa disebabkan karena data masih kurang bersih, terdapat bahasa gaul yang tidak dikenali oleh kamus yang digunakan, keliru dalam melabelkan data di awal dan juga data *training* yang terlalu sedikit.
3. Menggunakan kamus *stopwords* dari *library nltk* kurang tepat dikarenakan masi terdapat beberapa kata hubung yang tidak terhapus.
4. Hasil validasi menunjukkan F1 (perbandingan rata-rata dari presisi dan *recall* yang di bobotkan) senilai 88.37%, Akurasi (tanggapan yang benar diprediksi positif dan benar diprediksi negatif dari keseluruhan tanggapan) sebesar 82.76%, hasil presisi (tanggapan yang benar-benar positif dari keseluruhan tanggapan yang diprediksi positif) senilai 79.17%, dan hasil *recall* (tanggapan yang diprediksi positif dari seluruh tanggapan positif) sebesar 100%.

5. Visualisasi *WordCloud* menunjukkan kata positif yang paling sering muncul pada tanggapan masyarakat adalah ”vaksinasi”, “sehat” dan “aman”, sedangkan kata negatif yang paling sering muncul adalah “bayar”, “tolak” dan “khawatir”.

DAFTAR PUSTAKA

A Soedomo, H. (2005). In *Pendidikan (Suatu Pengantar)*. Surakarta: UNS Press.

Fawcett, T. (2006). In *An Introduction to ROC Analysis. Pattern Recognition Letters* (pp. 861–874).

Febriyani. (2021). *Mengenal Vaksin Covid-19*. (Online). (https://ciputrahospital.com/mengenal-vaksin-covid-19/, diakses 20 Maret 2021).

Hadna, N. M., Santosa, P. I., & Winarto, W. W. (2016). Studi Literatur Tentang Perbandingan Metode Untuk Proses Analisis Sentimen di Twitter. *Seminar Nasional Teknologi Informasi dan Komunikasi 2016*.

Putri, G. S. (2020). *Keraguan pada Vaksin Covid-19, Bagaimana Masyarakat Harus Bersikap?*. (Online). (https://www.kompas.com/sains/read/2020/12/23/160000023/keraguan-pada-vaksin-covid-19-bagaimana-masyarakat-harus-bersikap?page=all, diakses 23 Maret 2021).

Pravina, A. M., Cholissodin, I., & Adikara, P. P. (2019). Analisis Sentimen Tentang Opini Maskapai Penerbangan pada Dokumen Twitter Menggunakan Algoritme Support Vector Machine (SVM). *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*.

Safitri, R. N. (2020). *Analisis Sentimen Review Pelanggan Hotel Menggunakan Metode K-Nearest Neighbor (K-NN) (Studi Kasus: Hotels.com Booking.com, Agoda.com)*. Surabaya: Universitas Dinamika.

Wibowo, K. S. (2021). *Survei SMRC : Warga DKI Jakarta Tertinggi Menolak Vaksinasi Covid-19*. (Online) (https://nasional.tempo.co/read/1445189/survei-smrc-warga-dki-jakarta-tertinggi-menolak-vaksinasi-covid-19/full?view=ok, diakses 23 Maret 2021).

Ulfah, A. N., & Anam, M. K. (2020). Analisis Sentimen Hate Speech Pada Portal Berita Online Menggunakan Support Vector Machine (SVM). *Jurnal Teknik Informatika dan Sistem Informasi*, 1-10.

Wardani, F. K. (2019). *Analisis Sentimen Untuk Pemeringkatan Popularitas Situs Belanja Online di Indonesia Menggunakan Metode Naive Bayes (Studi Kasus Data Sekunder)*. Surabaya: Institut Bisnis dan Informatika STIKOM Surabaya.

WHO. (2021). *Weekly Operational Update on COVID-19*. (Online). (https://www.who.int/publications/m/item/weekly-operational-update-on-covid-16-march-2021, diakses 18 Maret 2021).

SVM Vaksin

ORIGINALITY REPORT

7%	7%	0%	4%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Forum Perpustakaan Perguruan Tinggi Indonesia Jawa Timur Student Paper	4%
2	ejournal.bsi.ac.id Internet Source	2%
3	dspace.uii.ac.id Internet Source	2%

Exclude quotes	Off	Exclude matches	< 2%
Exclude bibliography	On		