

SENTIMEN ANALISIS COVID-19 DENGAN METODE PROBABILISTIC NEURAL NETWORK DAN TF-IDF

Ina Najiyah¹, Ifani Haryanti²

¹Universitas Adhirajasa Reswara Sanjaya
e-mail: inajiyah@ars.ac.id

²Universitas Adhirajasa Reswara Sanjaya
e-mail: ifani@ars.ac.id

Abstrak

Penelitian ini bertujuan melakukan *sentiment analysis* tentang corona virus pada kegiatan sehari-hari yang diunggah di facebook, Twitter dan Instagram dengan *output* yaitu 3 *class*: positif, *negative* dan netral. Metode yang dipilih adalah metode klasifikasi *Probabilistic Neural network*. Sebelum melakukan klasifikasi, *praprocessing* pada penelitian ini meliputi tokenisasi, normalisasi, menghilangkan *emoticon*, *Convert Negasi*, *Stopword removal* serta TF-IDF. *dataset* yang digunakan berjumlah 1177 *dataset* dengan pembagiannya yaitu 560 *dataset* positif, 355 *dataset* *negative* dan 262 *dataset* netral. Program dirancang menggunakan Bahasa pemrograman python dengan beberapa library seperti keras, tensorflow dan pandas. User interface dibuat berbasis android. Akurasi yang didapatkan pada pelatihan menggunakan *Probabilistic Neural network* sebesar 89%. Hasil pengujian adalah penelitian ini mampu melakukan *sentiment analysis* dengan kesalahan sebesar 11% dilihat dari *confussion matrix*.

Kata Kunci: analisis sentimen, covid19, *neural network*.

Abstract

This study aims to conduct a sentiment analysis about the corona virus in daily activities uploaded on Facebook, Twitter and Instagram with the output of 3 classes: positive, negative and neutral. The method chosen is the Probabilistic Neural network classification method. Prior to classification, preprocessing in this study includes tokenization, normalization, removing emoticons, Negation Convert, Stopword removal and TF-IDF. The dataset used is 1177 datasets with the division of 560 positive datasets, 355 negative datasets and 262 neutral datasets. The program is designed using the Python programming language with several libraries such as Keras, Tensorflow and Pandas. The user interface is based on android. The accuracy obtained in training using a Probabilistic Neural network is 89%. The test result is that this study is able to perform sentiment analysis with an error of 11% seen from the confusion matrix.

Keywords: Sentimen Analysis, Covid19, Neural network.

1. Pendahuluan

Corona COVID-19 merupakan sebuah virus yang muncul pada tahun 2019 dan bermula di china (Amelia, 2020). Penyebaran virus ini dianggap sangat cepat dan melebar luas ke seluruh Negara salah satunya Indonesia (Kasmad, 2020). Munculnya virus covid 19 ini mendapat banyak tanggapan dan opini yang dikemukakan oleh masyarakat Indonesia pada kehidupan sehari-hari khususnya pada platform digital atau media social seperti facebook, twitter dan instagram. Tanggapan

dan opini yang dikemukakan melalui social media tersebut dapat menjadi pengaruh kepada masyarakat yang membacanya sehingga perlu dilakukan penyaringan opini tersebut agar tidak salah menerima informasi dan tidak salah menilai apakah opini tersebut bersifat *negative*, *positive* atau netral salah satu caranya adalah melakukan *sentiment analysis*. Sentimen analisis adalah riset komputasional dari opini *sentiment* dan emosi yang diekspresikan secara tekstual (Zulfa & Winarko, 2017). *sentiment analysis* pada data twitter pernah dilakukan

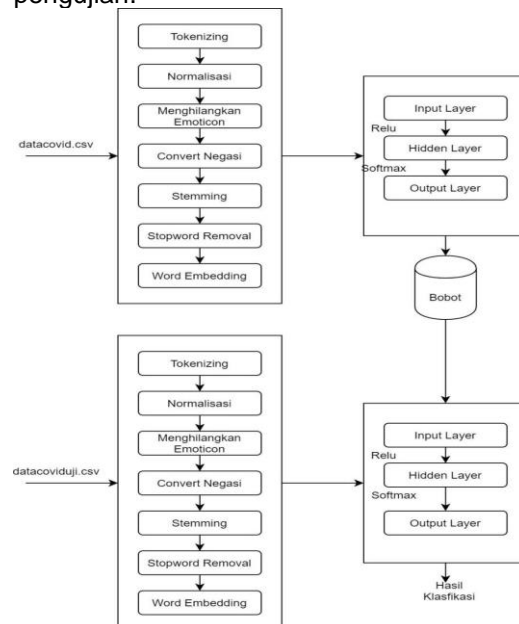
menggunakan metode RNN(Nemes & Kiss, 2021), penelitian ini menghasilkan akurasi sebesar 75% dari 1100 *dataset*. Hal ini dikarenakan proses yang dilakukan hanya *Tokenizing* dan filtering sehingga masih banyak data atau teks yang tidak di clean secara baik.

Selain itu pernah juga dilakukan menggunakan metode naïve bayes dengan *dataset* yaitu tentang pembelajaran daring dengan akurasi 72% (Samsir et al., 2021). Hal ini disebabkan karena fuzzy logic cara kerjanya lebih bagus terhadap data berupa angka dan *dataset* yang kecil. Dari penelitian sebelumnya, akurasi masih terbilang cukup tetapi kurang baik sehingga perlu diuji menggunakan metode lain yang menghasilkan akurasi lebih baik. *dataset* yang dipakai pada penelitian sebelumnya pun terbilang masih sedikit. *Neural network* merupakan metode yang sering digunakan dalam *dataset* gambar maupun teks dan menghasilkan akurasi baik seperti pada kasus analisis *sentiment* mengenai sarana dan transportasi mudik tahun 2019 (Pertiwi, 2019), selain itu penelitian lain menggunakan metode ini untuk analisis sentimen usaha waralaba berdasarkan data twitter (Mardiana et al., 2019), selain itu pernah juga dilakukan untuk Analisis Sentimen Pada Bpjs Kesehatan (Faturahman et al., 2020), penelitian lain menggunakan metode ini untuk Analisis Sentimen Terhadap *Review* Aplikasi E-commerce (Khatami et al., 2020), dan pernah juga dilakukan recurrent *neural network* (rnn) dengan long short term memory (Istm) untuk analisis sentimen data instagram(Cahyadi et al., 2020). Pada penelitian ini *sentiment analysis* dilakukan menggunakan metode *Probabilistic Neural network* dan metode TF-idf. TF-idf ditambahkan guna dapat melakukan representasi teks sehingga proses klasifikasi lebih akurat.

TF-idf dapat merepresentasi teks lebih baik dibandingkan sckit learn dikarenakan cara kerjanya tepat untuk data yang berupa teks, Data yang digunakan pada penelitian ini adalah data opini mengenai covid 19 yang ada pada facebook, twitter dan instagram. Data yang digunakan merupakan data public yaitu yang diambil dari kaggle, dan uci repository. Tujuan dari penelitian ini yaitu untuk melakukan *sentiment analysis* dan mengklasifikasikan opini mengenai covid19 menjadi tiga *class* yaitu *class* positif, *negative* dan netral..

2. Metode Penelitian

Sistem yang akan dibangun dibagi menjadi dua tahapan utama. Pertama terdapat proses pelatihan dan proses pengujian.



Gambar 1 Gambaran Alur Sistem

3. Hasil dan Pembahasan

3.1 Analisis Data Masukan

Berdasarkan analisis masalah yang telah dibahas sebelumnya, maka untuk menyelesaikan permasalahan tersebut, tahap awal yang perlu dilakukan adalah mengumpulkan data-data untuk dijadikan *dataset*. Data yang digunakan diambil dari social media seperti Twitter Facebook dan Instagram. Data yang didapatkan berupa kalimat yang berisi status, caption serta komentar orang terhadap pandemic Covid 19 bisa berupa deskripsi hingga ungkapan emosi. Pada kalimat tersebut berisi kata, angka, tanda baca hingga *emoticon*. Untuk polaritas yang mendominasi hanyalah positif dan negatif, karena data yang memiliki polaritas netral sangat sedikit sehingga polaritas yang digunakan hanya positif dan *negative* saja. Berikut merupakan contoh *dataset* yang ditunjukkan pada Tabel 1 .

Dari beberapa contoh pada Tabel 1 dapat disimpulkan bahwa setiap status, caption serta komentar berbeda beda. Hal tersebut bisa berdampak pada tingkat akurasi. Seperti pada contoh ulasan ke 4 penggunaan tanda baca tidak sesuai penempatan yang sebenarnya. Pada proses klasifikasi teka, tanda baca tidak digunakan dalam proses sehingga biasanya dihapus. Penggunaan *emoticon* seperti pada contoh

ulasan pertama dan kedua di atas, sangat berpengaruh terhadap penentuan polaritas. Contoh pada ulasan pertama penggunaan *emoticon* kurang tepat penggunaannya sehingga *emoticon* tersebut akan dihilangkan atau diabaikan. Agar sistem dapat mengenali setiap kata maka diperlukan pemisahan tiap kata pada sebuah kalimat. Sehingga kata-kata yang telah terpisah akan digunakan untuk fitur. Hasil pemisahan kata tidak bisa langsung digunakan untuk proses klasifikasi, setiap penulisan akan diperiksa dan diubah ke kata yang benar. Atau jika menemukan kata yang tidak baku akan diubah ke dalam kata baku. Agar sistem dapat mengenali setiap kata negatif maka diperlukan penggabungan kata pada sebuah kalimat. Pada contoh ulasan no 2 kata “tidak” dan kata “selamat” akan digabungkan sehingga kata tersebut bermakna negatif pada sistem. Contoh kata kata negasi adalah “tidak”, “jangan”, “bukan” dan lain sebagainya. Pada proses klasifikasi pun tidak seluruhnya kata digunakan pada proses klasifikasi, hanya beberapa kata yang bermakna yang sebaiknya diproses, agar fitur yang dihasilkan tidak terlalu banyak. Pada contoh ulasan ketiga di atas terdapat kata umum seperti “ini” dan “jadi” yang biasanya muncul dalam jumlah besar dan dianggap tidak memiliki makna, sehingga kata tersebut akan dihapus. Setelah menyisakan kata-kata yang memiliki makna, maka setiap kata yang berimbuhan seperti pada ulasan kedua dan keempat di atas akan diubah ke dalam bentuk kata dasar, agar tidak menghasilkan dimensi data yang terlalu besar. Oleh karena itu diperlukan tahap *preprocessing* untuk mengubah *dataset* tersebut agar siap digunakan untuk *inputan* pada system yang dibangun. Analisis data masukan dilakukan dengan tujuan untuk mendapatkan nilai-nilai dari data masukan untuk kemudian digunakan pada proses klasifikasi dengan metode *Probabilistic Neural network* dan TF-idf. Data masukan yang digunakan pada penelitian ini adalah data text yang diambil dari social media mengenai covid19. Pada penelitian ini, fitur yang digunakan adalah text Bahasa Indonesia. Text atau kalimat tersebut nantinya akan dilakukan *Tokenizing*, *stemming*, dan representasi text sebelum

dilakukan klasifikasi. Berikut merupakan jumlah *dataset* yang diambil dari social media dengan 3 *class* yaitu *class* positif, negatif dan netral.

Tabel 1. *Dataset*

No	Isi Kalimat	Polaritas
1	Seneng banget gue dengan covid19 ini jadi makin banyak waktu sama keluarga, gak sibuk dengan urusan masing-masing XD	Positif
2	Coba kalian bayangin, ibu melahirkan disuruh rapid dulu, keburu anaknya tidak selamat	Negatif
3	Angka Covid tembus 1jt.... yaaampuuun parah banget	Negatif

3.2. Proses Pelatihan

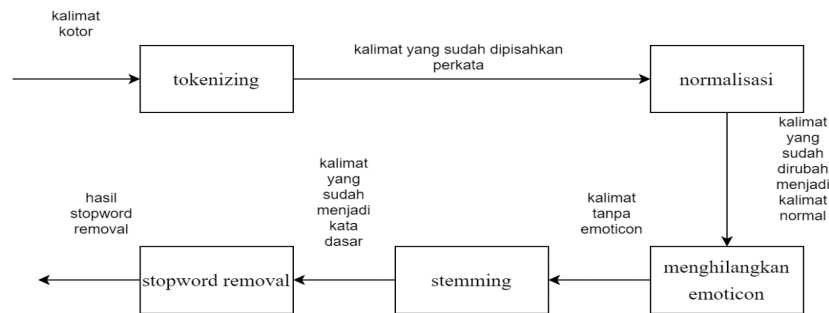
Pra proses Data

Setelah mendapatkan data latih yang bersumber dari twitter, facebook dan Instagram yang berisi tema covid19. Maka data latih tersebut akan dijalankan guna membangun program analysis *sentiment* yang menghasilkan tiga kelas yaitu positif, negatif dan netral. Adapun diantaranya data latih sebagai berikut:

Tabel 2. Tabel Data Latih

No	Isi Kalimat	Polaritas
1	Seneng banget gue dengan covid19 ini jadi makin banyak waktu sama keluarga, gak sibuk dengan urusan masing-masing XD	Positif
2	Coba kalian bayangin, ibu melahirkan disuruh rapid dulu, keburu anaknya tidak selamat	Negatif

Preprocessing dilakukan untuk mengurangi noise dan menyeragamkan data supaya memudahkan proses pelatihan dan proses pengujian. Adapun tahapan *preprocessing* yang akan dilakukan yaitu *Tokenizing*, normalisasi, menghilangkan *emoticon*, *stemming*, dan *Stopword removal*. Gambaran umum tahap *preprocessing* dapat dilihat pada Gambar 2.

Gambar 2 Tahapan *Preprocessing*

1. Tokenizing

Tokenisasi adalah proses untuk membagi teks yang dapat berupa kalimat, paragraf atau dokumen, menjadi token-token/bagian-bagian tertentu. Proses pemisahan kalimat menjadi kata ini biasanya ditentukan dari sebuah delimiter. Delimiter atau pemisah dari setiap kata ini yakni spasi. *Tokenizing* dilakukan guna memudahkan proses klasifikasi dengan memisahkan kalimat menjadi per kata.

Berdasarkan data latih pada Tabel 2 hasil proses *Tokenizing* dapat dilihat pada Tabel 3 berikut.

Tabel 3. Hasil *Tokenizing*

No	Isi Kalimat	Hasil <i>Tokenizing</i>
1	Seneng banget gue dengan covid19 ini jadi makin banyak waktu sama keluarga, gak sibuk dengan urusan masing-masing XD	Seneng banget gue dengan covid19 ini jadi makin banyak waktu sama keluarga, gak sibuk dengan urusan masing-masing XD
2	Coba kalian bayangin, ibu melahirkan disuruh rapid dulu, keburu anaknya tidak selamat	Coba kalian bayangin, ibu melahirkan disuruh rapid dulu, keburu anaknya tidak selamat

2. Normalization

Proses ini merupakan proses dimana mengubah kata yang tidak sesuai dengan EYD menjadi sesuai dengan EYD. Bertujuan untuk menormalkan kalimat sehingga kalimat gaul menjadi normal (Adiyasa, sehingga bahasa gaul tersebut dapat dikenali sebagai bahasa yang sesuai dengan KBBI harus dilakukan untuk normalisasi kalimat adalah:

- Meregangkan tanda baca (*punctuation*) dan symbol selain alphabet, lalu mengubah kata tidak baku menjadi baku. Contoh tahap ini sebagai berikut:

Tabel 4. Hasil Normalisasi

No	Isi Kalimat	Hasil Normalisasi
1	Seneng banget gue dengan covid19 ini jadi makin banyak waktu sama keluarga gak sibuk dengan urusan masing-masing XD	Seneng banget saya dengan covid19 ini jadi semakin banyak waktu dengan keluarga tidak sibuk dengan urusan masing masing XD
2	Coba kalian bayangin, ibu melahirkan disuruh rapid dulu, keburu anaknya tidak selamat	Coba kalian bayangkan ibu melahirkan disuruh rapid dulu, keburu anaknya tidak selamat

- b. Mengubah menjadi huruf kecil semua normalisasi kata .

Tabel 5. Hasil Normalisasi

No	Isi Kalimat	Hasil Normalisasi
1	Senang banget saya dengan covid19 ini jadi semakin banyak waktu dengan keluarga tidak sibuk dengan urusan masing masing XD	senang banget saya dengan covid19 ini jadi semakin banyak waktu dengan keluarga tidak sibuk dengan urusan masing masing xd
2	Coba kalian bayangkan ibu melahirkan disuruh rapid dulu, keburu anaknya tidak selamat	coba kalian bayangkan ibu melahirkan disuruh rapid dulu, keburu anaknya tidak selamat

3. Menghilangkan *emoticon*.

Ketika sedang menulis status (tweet) seseorang kadang salah atau kurang tepat dalam penggunaan *emoticon*, entah disengaja atau tidak banyak yang melakukannya. Contohnya: Mereka hanya bisa memfitnah karena tidak bisa ketemu fakta buruk :), kata opini fitnah tapi *emoticonnya* senyum :), dengan begitu *emoticon* akan mengganggu dalam proses *sentiment analysis* tweet, jadi dalam proses ini *emoticon* dihapus atau diabaikan. Beberapa *emoticon*, *feeling* and *sentiment* dapat dilihat pada Gambar 3.

Emoticon	Feeling	Sentiment
:) :-)	Happy	Positive
:(:-)	Sad	Negative
:D :-D	Very Happy!	Positive
D: D=	Very Sad	Negative
* * * * *	Fascinated	Positive

Gambar 3. Emotikon

Hasil proses menghilangkan *emoticon* seperti berikut:

Tabel 6. Hasil Menghilangkan *Emoticon*

No	Isi Kalimat	Hasil Menghilangkan Emotikon
1	Senang banget saya dengan covid19 ini jadi semakin banyak waktu dengan keluarga tidak sibuk dengan urusan masing masing XD	senang banget saya dengan covid19 ini jadi semakin banyak waktu dengan keluarga tidak sibuk dengan urusan masing masing
2	Coba kalian bayangkan ibu melahirkan disuruh rapid dulu, keburu anaknya tidak selamat	coba kalian bayangkan ibu melahirkan disuruh rapid dulu, keburu anaknya tidak selamat

4. *Stemming*

Tahap *preprocessing* selanjutnya yang dilakukan adalah proses *stemming*, setiap token kata yang berimbuhan akan diubah menjadi bentuk kata dasar, sehingga meminimalisir dimensi kata yang terbentuk. Pada penelitian ini, untuk metode *stemming* Bahasa Indonesia yang digunakan adalah metode Arifin Setiono. Metode tersebut dipilih karena dalam penelitian Dian Novitasari mendapatkan kesimpulan bahwa proses *stemming* dokumen teks berbahasa Indonesia menggunakan metode Arifin & Setiono memiliki presisi lebih tinggi sebesar 95% dibandingkan dengan *stemming* menggunakan metode Porter yang menghasilkan sebesar 90% [40]. Dalam metode tersebut mengasumsikan bahwa setiap kata masukan memiliki dua awalan (prefiks) dan tiga akhiran (sufiks). Berikut detail dari asumsi dari metode Arifin Setiono:

[AW1] + [AW2] + KD + [AK3] + [AK2] + [AK1]

Dimana AW merupakan awalan, KD merupakan kata dasar dan AK merupakan akhiran. Tahapan yang dilakukan pada metode tersebut adalah sebagai berikut [41]:

- a. Melakukan pemeriksaan pada kamus kata dasar, jika terdapat kata yang sesuai pada kamus kata dasar maka proses cukup samapai disini, jika tidak akan dilanjutkan dengan mempersiapkan variabel awalan[0], awalan[1], akhiran[0], akhiran[1], akhiran[2] untuk menampung imbuhan yang telah terpisah dari kata dasar.
- b. Pemotongan dilakukan secara berurutan sebagai berikut:
 - 1) AW I, hasil disimpan pada awalan[0].
 - 2) AW II, hasil disimpan pada awalan[1].
 - 3) AK I, hasil disimpan pada akhiran[0].
 - 4) AK II, hasil disimpan pada akhiran[1].
 - 5) AK III, hasil disimpan pada akhiran[2].

Dalam setiap proses pemotongan di atas selalu dilakukan pemeriksaan dengan kamus kata dasar. Hal tersebut dilakukan untuk mengetahui apakah hasil pemotongan tersebut sudah ada dalam bentuk kata dasar. Apabila sudah dalam bentuk kata dasar, maka proses dinyatakan selesai dan tidak perlu melanjutkan proses pemotongan selanjutnya.

- c. Namun jika sampai tahap dua belum menemukan kata dasar, maka akan dilakukan proses kombinasi. Kata dasar yang telah dihasilkan akan dikombinasikan dengan imbuhan-imbuhan dalam 12 konfigurasi berikut:
 - a. KD
 - b. KD + AK III
 - c. KD + AK III + AK II
 - d. KD + AK III + AK II + AK I
 - e. AW I + AW II + KD
 - f. AW I + AW II + KD + AK III
 - g. AW I + AW II + KD + AK III + AK II
 - h. AW I + AW II + KD + AK III + AK II + AK I
 - i. AW II + KD
 - j. AW II + KD + AK III
 - k. AW II + KD + AK III + AK II
 - l. AW II + KD + AK III + AK II + AK I

Pemeriksaan dalam 12 kombinasi di atas selalu dilakukan pemeriksaan dengan kamus kata dasar. Hal tersebut dilakukan untuk mengetahui apakah hasil pemotongan tersebut sudah ada dalam bentuk kata dasar.

Apabila sudah dalam bentuk kata dasar, maka proses dinyatakan selesai, namun jika sampai pada tahap akhir tidak ditemukan kata dasar yang tepat maka akan dikembalikan pada kata semula.

Dalam kasus ini terdapat token kata "tempatnya", kata tersebut akan diubah menjadi kata dasar. Tahapan awal yang dilakukan adalah melakukan pemeriksaan dengan kamus kata dasar, namun karena kata "tempatnya" tidak ada dalam kamus kata dasar maka masuk pada tahap pemotongan.

Tahap pemotongan dilakukan dengan 5 urutan, setiap tahapan memiliki kamusnya masing-masing untuk menentukan imbuhan yang tepat pada kata tersebut. Berikut dibawah ini daftar imbuhan di setiap tahapan yang bisa dilihat pada Tabel 7

Tabel 7. Hasil *Stemming*

No	Isi Kalimat	Hasil <i>Stemming</i>
1	senang banget saya dengan covid19 ini jadi semakin banyak waktu dengan keluarga tidak sibuk dengan urusan masing masing	senang banget saya dengan covid19 ini jadi semakin banyak waktu dengan keluarga tidak sibuk dengan urusan masing masing
2	coba kalian bayangkan ibu melahirkan disuruh rapid dulu, keburu anaknya tidak selamat	coba kalian bayang ibu lahir suruh rapid dulu, keburu anak tidak selamat

5. Convert Negasi

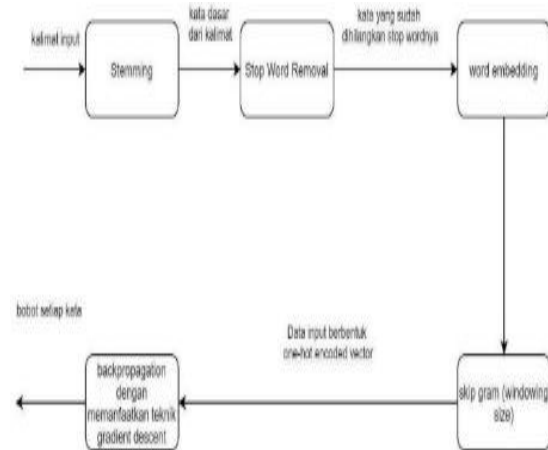
Tahapan ini adalah proses konversi kata kata negatif pada sebuah postingan mengenai covid19. Kata negasi dapat merubah makna *sentiment* suatu dokumen, sehingga kata negasi akan digabungkan dengan kata selanjutnya. Contoh kata negasi “tidak”, “bukan”, “jangan” dan sebagainya. Berikut merupakan proses *Convert Negasi*.

Tabel 8. Hasil *Convert Negasi*

No	Isi Kalimat	Hasil
1	senang banget saya dengan covid19 ini jadi semakin banyak waktu dengan keluarga tidak sibuk dengan urusan masing masing	senang banget saya dengan covid19 ini jadi semakin banyak waktu dengan keluarga tidaksibuk dengan urusan masing masing
2	coba kalian bayang ibu lahir suruh rapid dulu, keburu anak tidak selamat	coba kalian bayang ibu lahir suruh rapid dulu, keburu anak tidakselamat

6. Stopword removal

Setelah melewati tahap *stemming*, tahapan selanjutnya hasil dari proses *stemming* akan melalui proses *Stopword removal*. Proses *Stopword removal* pada penelitian ini adalah penghilangan kata yang terdapat list *stopword* seperti pada Lampiran A1. Kata “*stopword*” biasanya merujuk pada kata-kata yang paling umum seperti kata “dan”, “dengan”, “di”, dan lain-lain. Proses *Stopword removal* dapat dilihat pada Gambar 4.

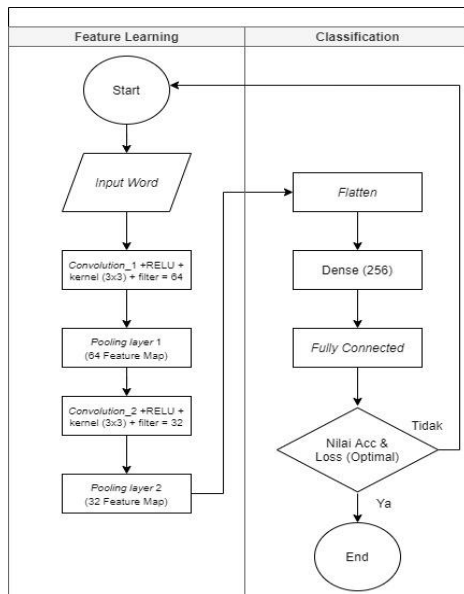


Gambar 4. Proses *Stopword removal*

Setelah melalui proses *Stopword removal* kalimat tidak mengandung kata *stopword* yang dapat membuat dimensi kamus kata menjadi besar. Hasil dari proses *Stopword removal* dapat dilihat pada Tabel 9.

Tabel 9. Hasil *Stopword removal*

No	Isi Kalimat	Hasil <i>Stopword removal</i>
1	senang banget saya dengan covid19 ini jadi semakin banyak waktu dengan keluarga tidaksibuk dengan urusan masing masing	senang banget saya dengan covid19 semakin banyak waktu keluarga tidaksibuk urusan masing masing
2	coba kalian bayang ibu lahir suruh rapid dulu, keburu anak tidakselamat	coba bayang ibu lahir suruh rapid dulu, keburu anak tidakselamat



Gambar 5. FlowChart Model

- 1) Inisialisasi nilai awal pada filter pada lapisan konvolusi dan bobot pada lapisan *fully connected* dengan nilai acak, dan bisa dengan nilai awal 0.
- 2) Melakukan proses konvolusi text masukan sesuai dengan filter pada lapisan konvolusi. Proses konvolusi dilakukan sesuai dengan persamaan.
- 3) *Feature maps* yang didapatkan akan dikurangi ukurannya untuk mengurangi kompleksitas perhitungan pada lapisan selanjutnya. Proses ini dilakukan pada 40 lapisan subsampling. Proses *subsampling* dengan menggunakan *max pooling*, atau meloloskan nilai tertinggi dari *Feature maps* yang ada dalam sebuah jendela *subsampling*.
- 4) Hasil dari lapisan *subsampling* merupakan *Feature maps* yang telah direduksi ukurannya, jika pada struktur lapisan PNN yang digunakan terdapat lapisan konvolusi setelah lapisan subsampling, maka tahapan selanjutnya adalah sama dengan tahap 1-3, jika tidak maka lanjutkan ke tahap 5.
- 5) *Feature maps* yang didapat dari lapisan *subsampling* terakhir merupakan *Feature maps* yang akan digunakan pada lapisan *fully connected* sebagai fitur untuk melakukan klasifikasi. *Feature maps* yang berupa matriks akan diuraikan menjadi *vector* yang panjang.

Proses ini disebut *vectorization and concatenation* yang dinotasikan pada persamaan (3.24). Fitur yang masuk ke dalam lapisan *fully connected* (f) merupakan hasil proses vektorisasi ((x)) dari hasil *subsampling* pada lapisan sebelumnya (S_{p-1}), proses ini menggabungkan seluruh n buah *Feature maps*.

- 6) Selanjutnya adalah proses perhitungan klasifikasi target dari fitur yang masuk ke dalam lapisan *fully connected*. Nilai klasifikasi kelas ((i)) ini dilakukan dengan melakukan perhitungan menggunakan persamaan Perhitungan pada persamaan ini menggunakan fitur dari lapisan *subsampling* sebelumnya ((j)) yang dikalikan dengan bobot yang terkoneksi ((i,j)) dan ditambahkan dengan bias ((i)).
$$\hat{y}(i) = \sigma \left(\sum_{j=1}^n W(i,j)f(j) + b(i) \right)$$
- 7) Untuk mengetahui seberapa baik proses pembelajaran telah dilakukan, maka nilai Loss dihitung dengan persamaan.

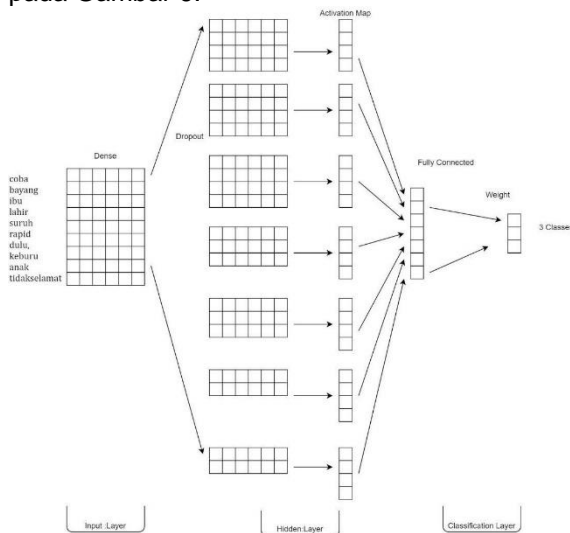
Teks dari setiap kalimat tentang covid terlebih dahulu ditransformasikan ke dalam representasi vektor kata menggunakan TF-idf kemudian dimasukkan ke *input layer*. Maksimum panjang sekuens dari *input* adalah 1000, sehingga *input* akan berupa matriks berukuran 1000x300. Adapun gambaran *inputan* pada proses PNN sebagai berikut:

Tabel 12. Proses PNN

Kata	Tahap TF-idf	Inputan ke PNN
seneng banget saya dengan covid19 ini jadi semakin banyak waktu dengan keluarga tidaksibuk dengan urusan masing masing	Maksimum panjang sekuens dari <i>input</i> adalah 1000	matriks berukuran 1000 x 300

a) Menentukan Model Network.

Probabilistic Neural network bekerja dengan cara memberikan *layer*. *layers* tersebut berupa *layer input* yang berisi *word* yang telah dilakukan *praprocess*. pada *layer input* akan ditentukan jumlah *dense* atau *node*. setelah *word* diinput di *layer*, maka terdapat *dropout* yang berfungsi menghilangkan kata yang tidak sesuai dan hanya kata yang sesuai yang dimasukan ke *layer* berikutnya yang disebut *hidden layer*. setelah difilter oleh beberapa *layers*, maka digunakan *Activation maps* untuk mengeluarkan bobot dari setiap *layer* atau kata. bobot tersebut akan menghasilkan tiga *class* yaitu *class* positif, negatif dan netral. berikut merupakan rancangan model *network* pada Gambar 6.



Gambar 6. Model Network

b) Perhitungan *Forward Backpropagation* menggunakan *Softmax Activation*

Tabel 13. *Forward Backpropagation*

class		Softmax
Negatif	$=((FC11*W1)+(FC21*W3)+(FC31*W5)+(FC41*W7)+(FC51*W9)+(FC61*W11))+(BFC*1)$ $=((0,11*0,1)+(0,14*0,3)+(0,14*0,4)+(0,09*0,2)+(0,17*0,7)+(0,14*0,19))+(0,35*1)$	$= 1.86376754 / 3.76024842 = 0.49565011$
Positif	$=((0,11*0,3)+(0,14*0,4)+(0,14*0,3)+(0,09*0,1)+(0,17*0,8)+(0,14*0,1))+(0,35*1)$	$= 1.89648088 / 3.76024842 = 0.50434989$
Netral	$=((0,11*0,4)+(0,14*0,1)+(0,14*0,3)+(0,09*0,1)+(0,17*0,8)+(0,14*0,1))+(0,35*1)$	$= 1.99648088 / 3.76024842 = 0.59234977$

Positif	$=((FC11*W2)+(FC21*W4)+(FC31*W6)+(FC41*W8)+(FC51*W10)+(FC61*W12))+(BFC*1)$ $=((0,11*0,3)+(0,14*0,4)+(0,14*0,3)+(0,09*0,1)+(0,17*0,8)+(0,14*0,1))+(0,35*1)$	$= 1.89648088 / 3.76024842 = 0.50434989$
Netral	$=((FC11*W2)+(FC21*W4)+(FC31*W6)+(FC41*W8)+(FC51*W10)+(FC61*W12))+(BFC*1)$ $=((0,11*0,4)+(0,14*0,1)+(0,14*0,3)+(0,09*0,1)+(0,17*0,8)+(0,14*0,1))+(0,35*1)$	$= 1.99648088 / 3.76024842 = 0.59234977$

8. Pengujian

Perbandingan dengan Metode Klasifikasi yang Lain Uji coba berikut dilakukan untuk membandingkan unjuk kerja desain PNN (sebagai desain PNN terbaik yang sudah terbukti) dengan metode klasifikasi yang lain, yaitu *k-Neural network* (*k-NN*) dan SVM. Untuk *k-NN*, *dataset* text mode diuji coba dengan menggunakan parameter *k* sebesar 5. Sedangkan pada metode SVM, *dataset* diuji coba menggunakan linear kernel. Pengujian dilakukan untuk Rika Rokhana: *Probabilistic Neural network* untuk ... ISSN 2301 - 4156 65 JNTETI, Vol. 8, No. 1, Februari 2019 mengklasifikasi text dalam kelas positif, negatif dan netral. Data masukan untuk *classifier k-NN* dan SVM adalah bobot nilai pada *dataset*. Hasil pengujian disajikan dalam Tabel 14.

Tabel 14. Perbandingan dengan Metode Lain

No	Jumlah <i>dataset</i>	Metode	Akurasi
1	400	SVM	70%
2	1100	KNN	76%
3	1177	PNN	86%

9. Pengujian Performansi

Tahap pengujian nilai akurasi dari metode yang digunakan pada penelitian ini dilakukan menggunakan metode hold-out dengan menggunakan 50 data sebagai data latih. Proses klasifikasi ini menggunakan kombinasi nilai parameter yaitu parameter cost (C). Adapun masing-masing nilai parameter yang digunakan diantaranya adalah C = (0.1,0.5,0.75,1,10,100). Berikut adalah hasil dari pengujian performansi menggunakan parameter yang telah disebutkan.

Confussion matrix dan hasil pengujian dengan parameter cost = 0.1 dapat dilihat pada Tabel 15

Tabel 15. *Confussion matrix* pengujian parameter C = 0.1

Aktual	Kelas Hasil Klasifikasi		
	Positif	Negatif	Netral
Positif	15	19	5
Negatif	4	10	2
Netral	3	3	13

$$Akurasi = \frac{TP + TN}{TP + FP + FN + TN}$$

$$Akurasi = \frac{15 + 19 + 5}{8 + 4 + 5 + 4 + 10 + 3 + 3 + 13}$$

$$Akurasi = \frac{39}{50} = 78\%$$

Tabel 16. hasil akurasi pengujian C = 0.1

No	C	Nilai Akurasi PNN
1	0.1	78%

Berdasarkan pengujian pertama, didapatkan nilai akurasi sebesar 78%.

4. Kesimpulan

Berdasarkan penelitian dan hasil pengujian yang telah dilakukan, dapat ditarik beberapa kesimpulan sebagai berikut :

1. Adanya penelitian ini berhasil melakukan *sentiment analysis* sehingga opini mengenai covid19 di social media pada kehidupan sehari-hari menjadi lebih jelas.

2. Penelitian ini menghasilkan akurasi sebesar 86% dimana akurasi ini lebih baik dibandingkan dengan penelitian sebelumnya yang melakukan penelitian dengan 76%. Hal ini dikarenakan *dataset* yang lebih banyak dan *praprocess* yang lebih lengkap.

Referensi

- Amelia, N. (2020). *No Title*. Unair News. <http://news.unair.ac.id/2020/01/31/mengenal-perkembangan-jenis-baru-virus-corona-yang-infeksi-ribuan-orang/>
- Cahyadi, R., Damayanti, A., & Aryadani, D. (2020). *Recurrent neural network (rnn) dengan long short term memory (lstm) untuk analisis sentimen data instagram*. 5, 1–9.
- Faturohman, F., Irawan, B., Setianingsih, C., Elektro, F. T., & Telkom, U. (2020). *ANALISIS SENTIMEN PADA BPJS KESEHATAN MENGGUNAKAN RECURRENT NEURAL NETWORK SENTIMENT ANALYSIS ON BPJS KESEHATAN USING RECURRENT NEURAL*. 7(2), 4545–4552.
- Kasmad. (2020). *Covid 19 dan Penyebarannya*.
- Khatami, F. A., Irawan, B., Si, S., & Setianingsih, C. (2020). *ANALISIS SENTIMEN TERHADAP REVIEW APLIKASI LAYANAN E-COMMERCE MENGGUNAKAN METODE CONVOLUTIONAL NEURAL NETWORK SENTIMENT ANALYSIS OF E-COMMERCE APPLICATION REVIEWS USING THE CONVOLUTIONAL NEURAL NETWORK METHOD*. 7(2), 4559–4566.
- Mardiana, T., Syahreva, H., Informasi, S., Bina, U., & Informatika, S. (2019). *KOMPARASI METODE KLASIFIKASI PADA ANALISIS SENTIMEN USAHA*. 15(2), 267–274. <https://doi.org/10.33480/pilar.v15i2.752>
- Nemes, L., & Kiss, A. (2021). *Social media sentiment analysis based on COVID-19*. *Journal of Information and Telecommunication*, 5(1), 1–15. <https://doi.org/10.1080/24751839.2020.1790793>
- Pertiwi, Wi. M. (2019). *ANALISIS SENTIMEN OPINI PUBLIK MENGENAI SARANA DAN TRANSPORTASI MUDIK TAHUN*

2019 PADA TWITTER
MENGUNAKAN ALGORITMA NAÏVE
BAYES, NEURAL NETWORK, KNN
DAN SVM. 14(1), 27–32.

Samsir, Ambiyar, Verawardina, U., Edi, F., & Watrianthos, R. (2021). Analisis Sentimen Pembelajaran Daring Pada Twitter di Masa Pandemi COVID-19 Menggunakan Metode Naïve Bayes. *Jurnal Media Informatika Budidarma*, 5, 157–163.

<https://doi.org/10.30865/mib.v5i1.2604>

Zulfa, I., & Winarko, E. (2017). Sentimen Analisis Tweet Berbahasa Indonesia Dengan Deep Belief Network. *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)*, 11(2), 187. <https://doi.org/10.22146/ijccs.24716>