ZONAsi Jurnal Sistem Informasi
ZONAsi VOL. 5 NO. 2 MEI 2023
Page: 323 - 333
ISSN: 2656-7407 (Online)
2656-7393 (Print)

KLASIFIKASI SENTIMEN VAKSIN COVID-19 MENGGUNAKAN *K-NEAREST NEIGHBOR* BERDASARKAN *WORD EMBEDDINGS FASTTEXT* PADA *TWITTER*

Afri Naldi[1], Surya Agustian[2]
[1,2,3]Uiversitas Islam Negeri Sultan Syarif Kasim Riau
([1]Program Studi Teknik Informatika Fakultas Sains dan Teknologi UIN Suska Riau)
([1]Panam, Jl. HR. Soebrantas No. 155, Km. 15, Tampan, Pekanbaru, Riau 28293)
e-mail: [1]**11651103444@students.uin-suska.ac.id**, [2]surya.agustian@uin-suska.ac.id

Abstrak

Pada akhir 2019 muncul penyakit semacam flu pertama di kota Wuhan, yang menginfeksi paru-paru dan sistem pernapasan manusia. Diduga penyakit tersebut diduga berasal dari kelelawar. WHO memberi nama penyakit ini dengan nama Covid-19, ketika virus ini tersebar ke seluruh dunia sehingga menyebabkan pandemi. Pemerintah Indonesia mengambil tindakan vaksinasi untuk mengatasi virus ini, namun mendapat respon pro dan kontra sebagai bentuk sentimen dari masyarakat. Penelitian ini membahas klasifikasi sentimen terhadap kebijkan vaksin covid-19 menggunakan algoritma K-Nearest Neighbor berdasarkan word embeddings Fasttext pada media sosial twitter. Data diperoleh dengan cara crawling menggunakan Twitter API. Pelabelan data dilakukan secara crowdsourcing dengan majority voting. Dataset terdiri atas 8000 data training, 778 data development dan 400 data testing. Hasil pengujian setelah berbagai eksperimen pencarian model optimal dari feature selection dan feature engineering, mendapatkan hasil nilai akurasi 69% dan f1-score 60%. Hasil ini cukup kompetitif dibandingkan beberapa metode machine learning lainnya dengan dataset yang sama.

Kata kunci: *K-Nearest Neighbor, Fasttext, Klasifikasi Sentimen, Vaksin Covid-19.*

Abstract

At the end of 2019 a flu-like disease first appeared in the city of Wuhan, which infected human lungs and respiratory system. It is suspected that the disease originated from bats. WHO gave the name of this disease with Covid-19, when this virus wide spread throughout the world, and causing a pandemic. Indonesian government released a vaccination policy to deal with this virus, and received pro-cons responses as a form of sentiment from the citizen. This study discusses the sentiment classification task towards the Covid-19 vaccine policy, using the K-Nearest Neighbor algorithm based on Fasttext word embeddings, on Twitter social media. Data was obtained by crawling with Twitter API. Data labeling was resolved by crowdsourcing with majority voting. The dataset consists of 8000 training data, 778 development data and 400 testing data. The test result, after various experiments to find the optimal model from feature selection and feature engineering, yields an accuracy value of 69% and an f1-score of 60%. These results are quite competitive compared to several other machine learning methods with the same dataset.

1. PENDAHULUAN

Pada akhir tahun 2019 pada tanggal 31 Desember, terdapat beberapa laporan mengenai suatu penyakit yang tidak diketahui etiologinya di Cina tepatnya di Kota Wuhan provinsi Hubei. Penyakit ini memiliki gejala seperti demam, batuk kering, dispnea, dan infeksi pada paru-paru, lalu semua kasus yang terkait dengan penyakit ini terdapat di kota Wuhan tepatnya di pasar makanan laut yang menjual berbagai jenis hewan hidup seperti kelelawar, ular, dan unggas [1].

Secara resmi World Health Organization (WHO) menyebut penyakit ini dengan nama Covid-19 pada tanggal 11 Februari 2020. Lalu sampai pada tanggal 30 Januari 2020 penyakit ini terus

berkembang dengan cepat hingga banyak negara terkena atau terjangkit penyakit ini, sehingga WHO mendeklarasikan bahwa Covid-19 sebagai ancaman untuk dunia. Virus covid-19 atau bisa disebut virus corona pertama kali masuk ke Indonesia pada tanggal 2 Maret 2020 yang terjangkit pada dua orang warga Jawa Barat tepatnya di kota Depok. Semenjak saat itu telah banyak laporan bahwa masyarakat telah terkonfirmasi terkena virus corona yang melanda dunia. Kondisi ini meyebabkan efek samping tidak hanya terhadap bidang kesehatan tetapi bidang ekonomi, pendidikan, dan bidang lainnya yang terkena dampak dari virus corona yang telah melanda dunia [2].

Pemerintah kemudian mengambil kebijakan untuk meredakan tingkat penyebaran virus corona ini dengan program vaksinisasi massal. Sesuai dengan keputusan dari Menteri Kesehatan [3] nomor HK.01.07/Menkes/12758/2020 yang telah diresmikan tepatnya pada tanggal 28 Desember 2020 telah menetapkan beberapa jenis vaksin yang diperbolehkan penggunaannya yaitu Bio Farma, Sinovac, Moderna, Sinopharm, Novavax, dan Pfizer. Program vaksin yang ada di Indonesia menuai pro-kontra dari masyarakat khususnya penggunaan Sinovac yang dari China. Media sosial menjadi wadah untuk menuangkan dengan bebas pendapatnya, yang dapat berisi sentimen positif atau negatif terhadap program vaksinisasi ini. Salah satu media sosial yang digunakan yaitu Twitter [3], menjadi tempat terbanyak yang dipakai sebagai rujukan bagi pemerintah untuk memahami keinginan masyarakat.

Cukup banyak tweet yang memiliki sentimen positif seperti tweet dari akun @filoshopiee "GSHSGSHGSYSG I'M EXCITED BESOK AKU VAKSIN you guys stay safe and healthy yaa!!". Tetapi banyak pula yang negatif seperti tweet dari akun @widjaja_harta "Kenapa orang sebego itu bisa duduk di kursi DPR? Menolak vaksin? Rela bayar 5 juta per orang untuk anggota keluarganya. EGOIS, STUPID and IGNORANT.....". Sehingga dapat dikatakan bahwa respon dari penerapan vaksinisasi ini tidak hanya mendapat respon baik tetapi respon tidak baik juga terdapat didalamnya.

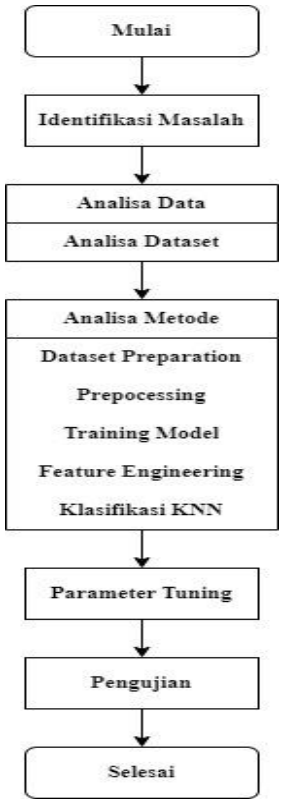
Beberapa penelitian sebelumnya telah menerapkan berbagai metode untuk mendeteksi sentimen di dalam teks, dengan hasil akurasi di bawah 70% dan F1-score di sekitar 60% atau kurang. Klasifikasi sentimen untuk kelas positif, negatif dan netral, terhadap vaksin COVID-19 dengan metode *Naïve Bayes* yang telah diteliti oleh [4], melakukan optimasi NB melalui kombinasi yang dipilih pada *teks preprocessing* dan *balancing data training*, sehingga dapat meningkatkan hasil klasifikasi dibandingkan *baseline*-nya. Metode *Support Vector Machine* dengan fitur *bag of words* pada dataset yang sama digunakan dalam [5], dan *fitur word embeddings* dikembangkan dalam [6]. Metode berbasis *deep learning* dengan *long-short term memory* (LSTM) juga diterapkan pada [7] dengan pendekatan optimasi fitur yang hampir sama untuk mengatasi data yang tidak *balance*. Sedangkan metode klasik berbasis *rule regular expression* dan *textblob* digunakan pada [8] untuk klasifikasi sentimen positif, negatif dan netral pada tweet dengan jumlah dataset yang kecil.

Klasifikasi biner untuk kelas positif dan negatif saja, lebih mudah untuk mendapatkan akurasi yang tinggi menggunakan *machine learning*, seperti pada [9], yang mendeteksi sentimen menggunakan *Naïve Bayes* dan *Decission Tree*, atau di dalam [10] untuk mendeteksi apakah terjadi serangan atau tidak pada suatu jaringan komputer.

Hasil klasifikasi non-biner seperti pada kelas positif, negatif dan netral pada sentimen twitter di atas masih membuka peluang untuk peningkatan akurasi, disebabkan skor yang relatif masih rendah, yaitu di kisaran di bawah 70% untuk akurasi, dan sekitar 60% untuk F1-score. Hal ini menunjukkan bahwa sistem masih sering gagal mengklasifikasi kelas positif, negatif maupun sebagai netral dan sebaliknya. Penelitian ini mengusulkan penerapan metode *K-Nearest Neighbor* dalam melakukan klasifikasi sentimen terhadap program vaksinisasi pada *tweet* Bahasa Indonesia. Sebagai fitur digunakan *word embeddings Fasttext* yang dioptimasi untuk meningkatkan akurasi model klasifikasi. Penilitian ini dilakukan pada dataset yang sama pada [4] [5], [6] dan [7], sehingga dapat memperbandingan performa terbaik dari *K-Nearest Neighbor* terhadap metode-metode terdahulu.

2. METODE PENELITIAN

Penelitian ini dilaksanakan secara garis besar dalam tahapan seperti pada Gambar 1 berikut, yaitu identifikasi masalah, analisa data, analisa metode, parameter tuning, dan pengujian.



Gambar 1. Metodologi Penelitian

2.1 Identifikasi Maslah

Sesuai dengan latar belakang yang sudah dijelaskan di bagian pendahuluan, masalah yang akan dipecahkan dalam penelitian ini adalah bagaimana melakukan klasifikasi sentimen untuk kelas positif, negatif dan netral dalam kasus tanggapan masyarakat terhadap kebijakan vaksinasi Covid-19 pada media sosial twitter, dengan profil dataset yang tidak berimbang di antara kelas-kelasnya. Penelitian-penelitian terdahulu baru menggunakan berbagai metode *machine learning*, dapat menghasilkan akurasi yang masih kurang baik dan masih dapat ditingkatkan lagi. Target yang dijadikan pengukuran adalah *F1-score*, untuk memantau sejauh apa sistem yang dikembangkan dapat mengklasifikasi kelas positif dan negatif di antara dataset yang lebih banyak netralnya.

2.2 Analisa Data

Pengumpulan data dengan cara crawling menggunakan Twitter API dalam pemgrograman Python. Data *tweet* dikumpulkan dari bulan Maret 2021 sampai April 2021, dengan menggunakan beberapa kata kunci seperti “Vaksinasi Indonesia”, “Vaksin”, “Vaksin Sinovac”, “Vaksin Nusantara”, “Vaksin Gotong Royong”, “Vaksin Covid Gratis”, Vaksin Sukarela”, “Vaksin Covid”, “Vaksin Corona” dan kata kunci lainnya yang masih relevan dengan kasus vaksin Covid. Total data tweet yang terkumpul di awal sebanyak 13.115 tweet.

Kemudian dilakukan pembersihan data, (*data cleaning*), semua tweet yang terduplikasi karena penggunaan kata kunci yang mirip, dihapus dan diambil 1 saja. Lalu dilakukan proses pemberian label data yaitu positif, netral, dan negatif, secara *crowd sourcing*. Label positif berisikan perasaan positif seperti kesenangan, kegembiraan, pujian, dukungan, serta saran-saran positif. Sedangkan label negatif biasanya berisikan keluhan, kekecewaan, ketidakpuasan, hinaan, hingga ujaran kebencian. Label netral berisikan data yang tidak termasuk kedalam kriteria data positif maupun negatif. Setiap tweet diberikan label oleh 3 orang, dan label yang dipilih adalah label yang mendapat suara terbanyak (*majority vote*). Dalam kasus ketiga anotator memberikan label yang berbeda, maka tweet dianggap tidak valid dan dihapus dari dataset.

Tabel 1. Contoh Pelabelan Data

Sentimen	Tweet
Positif	Mantap nih, Indonesia berarti telah menerima 59,5 juta dosis bahan baku vaksin dari Sinovac dengan kedatangan lagiâ€¦ https://t.co/vreS13Cv3y
Netral	Lebih sejuta dos vaksin Pfizer Covid-19 akan diterima kerajaan bulan hadapan, kata Menteri Sains, Teknologi dan Inoâ€¦ https://t.co/iJtldtFWsa
Negatif	Lalu kalau sudah di vaksin semua, jamin aman gitu??repot juga, kalau atasan2 di pusat sudah meremehkan, ke bawah pun iâ€¦ https://t.co/7to7zjDxv7

Dari proses pemberian label, diperoleh data final, sebanyak 9178 tweet. Data ini dibagi ke dalam 3 subset, yaitu data training, data validasi dan data testing dengan komposisi sebagaimana Tabel 2. Data *training* merupakan data latih yang digunakan pada metode yang dikembangkan, dan data *development* dipakai sebagai data test/validasi untuk mencari *model machine learning* dengan hasil yang paling optimal. Sedangkan data *test* adalah jenis data *tweet* untuk yang digunakan untuk evaluasi unjuk kerja sistem, sebagai data benchmark pengujian yang dipakai bersama.

Tabel 2. Hasil Pembagian Data

Data	Jumlah
Data Training	8000
Data Validasi	778
Data Test	400

2.3 *Prepocessing*

Tahapan ini melakukan pemrosesan teks tweet, supaya data siap untuk digunakan oleh *machine learning*. Penyesuaian teks dilakukan sesuai dengan kebutuhan input machine learning yang dipakai. Beberapa langkah-langkah yang umum dilakukan pada proses ini, yaitu *case folding*, *tokenizing*, *punctuation and stopword removal* dan juga menggunakan beberapa langkah tambahan bertujuan untuk menyesuaikan kombinasi langkah sesuai kebutuhan eksperimen. *Stemming* atau pemotongan imbuhan tidak dilakukan karena lebih banyak mengubah isi konteks sehingga dapat mengubah arah sentimen.

2.4 Training Model

Ada 2 model yang dibuat dalam penelitian ini. Model yang pertama adalah model bahasa (*Language Model*), berupa *word embeddings* yang dibentuk menggunakan metode FastText. Proses training dilakukan terhadap data train dan data validasi yang digabungkan, agar jumlah variasi kalimat lebih banyak. Output dari pelatihan ini adalah daftar word embeddings untuk setiap kata atau token yang terlihat pada saat training, yaitu kata-kata di dalam tweet pada data train dan data dev.

Sedangkan model yang kedua, adalah model k-NN yang digunakan untuk tahap klasifikasi. Sebagai input, diberikan vektor kalimat (*sentence vector*) yang dibentuk dari word vector FastText. Sentence vector adalah rata-rata dari resultan vektor kata-kata atau token di dalam suatu tweet, yang telah dinormalisasi sebelumnya dengan panjang tiap vektor kata (*norm vektor*).

Untuk membentuk model klasifier k-NN, data training digunakan untuk melatih model dan menyetel parameter terbaik (*parameter tuning*) serta penerapan rekayasa fitur (*feature engineering*). Model terbaik diperoleh berdasarkan hasil F1-score tertinggi dari setiap eksperimen kombinasi, terhadap data validasi.

Data test sama sekali tidak boleh diikutkan selama training. Sistem akan mencoba menangani klasifikasi terhadap data baru yang tidak pernah terlihat selama proses training.

2.5 *Feature Engineering*

Feature Enginering merupakan suatu proses yang dilakukan pada tahapan penelitian untuk memperoleh sebuah fitur tambahan bertujuan untuk membantu menemukan akurasi tinggi agar mendapatkan hasil yang terbaik. Pada proses *Feature Enginering* kita dapat memilih fitur atau membuat sebuah fitur yang baru supaya kinerja mesin lebih meningkat lagi.

Penelitian ini menggunakan sebuah fitur *scaling* bertujuan untuk mendapatkan hasil yang lebih bagus dan menemukan tingkat akurasi yang lebih tinggi dari kombinasi pada tahap *prepocessing* yang telah dikerjakan. Hal seperti ini dapat disebut sebagai cara untuk *improvement* melalui data. Adapun *Feature engineering* yang dipakai yaitu *minmax scaler* dan *robust scaler*.

2.5.1 *MinMax Scaler*

MinMax Scaler merupakan tahapan dari *prepocessing*, dimana melakukan transformasi fitur dengan menskalakan fitur secara individual dan bertahap dalam rentang tertentu sebagaimana persamaan (1). Hal ini bertujuan supaya rentang pada setiap fitur tidak terlalu besar. [11].

$$x_{sc} = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \qquad (1)$$

2.5.2 *Robust Scaler*

Robust Scaler merupakan tahapan teknik optimasi untuk mentransformasikan suatu nilai dengan menggunakan tahapan median dan *quartiles* untuk mendapatkan hasil yang permodelan yang lebih baik seperti pada persamaan (2). Tahap ini termasuk ke dalam proses *improve* melalui data untuk menghilangkan nilai yang janggal agar hasil dari permodelan tidak masuk kedalam *oversight* [12].

$$\|x\|_q \triangleq (\sum_{i=1}^{n} |x_i|^q)^{\frac{1}{q}} \qquad (2)$$

2.6 *K-Nearest Neighbor*

K-Nearest Neighbor (K-NN) merupakan metode klasifikasi dengan langkah-langkah yang sederhana. Metode K-NN melakukan proses klasifikasi suatu data berdasarkan sejumlah *k* data terdekat di sekitarnya [13]. Langkah-langkah metode K-NN [14] adalah:

Langkah-1: Tentukan parameter *k* (jumlah tetangga terdekat yang menjadi acuan penentuan kelas)

Langkah-2: Hitung jarak antara data baru yang akan diklasifikasikan dengan seluruh data yang ada pada data training. Jarak dihitung umumnya dengan menggunakan *Euclidean Distance* seperti persamaan (3).

Langkah-3: Buat ranking berdasarkan jarak *Euclidean* terkecil.

Langkah-4: Potong ranking sebanyak *y* tetangga terdekat

Langkah-5: Evaluasi kelas pada kelompok ranking yang terbentuk dari *y* tetangga terdekat. Kelas yang paling banyak ditetapkan sebagai kelas untuk data baru tersebut.

$$d_i = \sqrt{\sum_{i=1}^{p} (x_{2i} - x_{1i})^2} \qquad (3)$$

2.7 Pengujian

Eksperimen dilakukan untuk mengoptimasi model K-NN dan model *FastText*, untuk mendapatkan hasil *F1-score* terbaik dari berbagai kombinasi eksperimen. Hasil tersebut divalidasi dengan menggunakan data development sebagai data test pada sistem yang dikembangkan. Untuk melihat keberhasilan sistem mendeteksi kelas positif dan negatif, *confusion matrix* dapat digunakan, selain *evaluation report*, *precision*, *recall* dan *f1-score* yang dibangkitkan oleh *library scikit learn* pada Python.

3. HASIL DAN PEMBAHASAN

3.1 Data Balancing

Statistik dataset yang ada dapat dilihat pada Tabel 3 berikut ini. Terlihat bahwa untuk data train, kelas netral sangat banyak sehingga dapat menyebabkan sistem hanya dapat mengenali kelas netral saja pada saat proses *training*. Apabila ada data baru yang berisi sentimen, hasil pelatihan yang

tidak pas akan menyebabkan sistem gagal memprediksi sentimen, baik positif atau negatif. Hal ini dapat terlihat dari hasil pengujian di Tabel 4, yaitu hanya memiliki *F1-score* 35%, padahal akurasinya tinggi sebesar 84%, yang sama dengan jumlah komposisi data netral di dalam data train.

Tabel 3. Komposisi data awal untuk data train dan data

Data train (8000 data)			**Data dev (778 data)**		
Positif	**Netral**	**Negatif**	**Positif**	**Netral**	**Negatif**
463	6664	873	45	648	85
5.79%	83.30%	10.91%	5.79%	83.30%	10.91%

Beberapa cara balancing dengan mengurangi proporsi data netral, telah dilakukan, agar perbandingan antara data netral dengan data yang mengandung sentimen lebih berimbang (skema net vs *sentiment equal, net reduction*), mengikuti teknik yang dipakai pada [7] dan [16]. Namun hasil yang diperoleh dirasa masih dapat ditingkatkan, dengan *F1-score* pada saat *validasi* di kisaran 52%. Hal ini diprediksi karena jumlah *neighbor* (tetangga) yang dominan dari kelas netral saat suatu data baru diuji, padahal sentimen yang seharusnya mungkin positif atau negatif. Hipotesanya, apabila data positif atau negatif ditambah, kemungkinan dari *k* tetangga terdekat mungkin akan terpilih kelas sentimen positif atau negatif di dalam ranking *k*.

Berdasarkan hasil tersebut, perlu diupayakan penambahan proporsi data yang bersentimen ke dalam data training, apakah positif atau negatif, sehingga jumlahnya menjadi berimbang. Langkah awal yang dilakukan adalah dengan menduplikasi data positif dan negatif sampai jumlah tertentu (*all-balanced*), dengan data netral dipertahankan pada kisaran 2000 sampel. Hasil yang diperoleh menunjukkan peningkatan skor validasi melebihi 50%.

Langkah selanjutnya, porsi data netral ditambah, sampai ke jumlah final yang dipilih untuk data netral pada porsi data training adalah 5000. Kemudian, data positif diduplikasi beberapa kali, sehingga jumlah totalnya menjadi 5000. Demikian juga data negatif diduplikasi beberapa kali, sehingga jumlah totalnya pun menjadi 5000 sampel. Total data training saat ini menjadi 15.000 sampel, dengan jumlah kelas yang sama banyak. Setelah dilakukan pengujian, hasil *F1-score* menggunakan data *validasi* menunjukkan penurunan dari sebelumnya, kembali ke kisaran angka 50% atau lebih sedikit. Oleh karena itu, dari hasil empiris ini disimpulkan bahwa ada posisi tertentu dari teknik duplikasi ini yang menghasilkan hasil yang paling optimal.

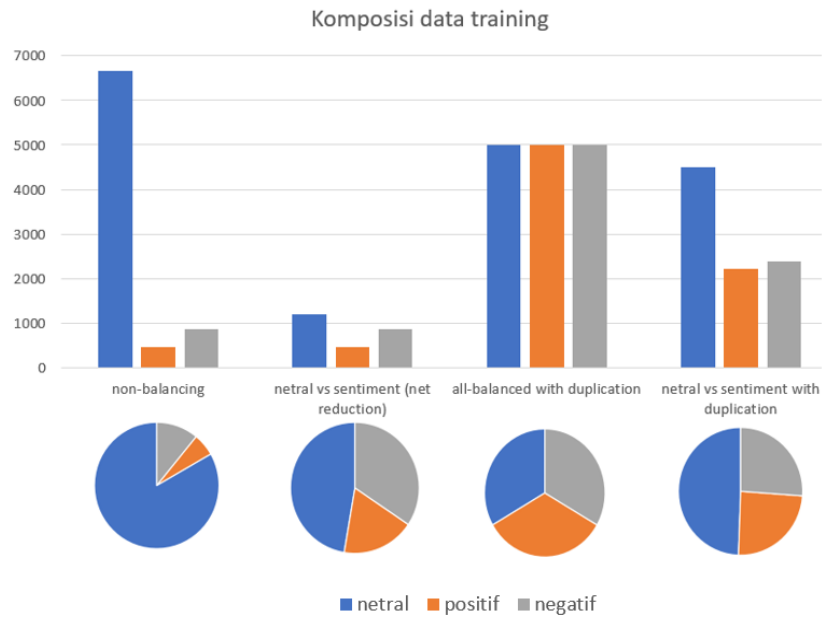
Untuk mencari titik optimal tersebut, diputuskan menggunakan skema proporsi berimbang antara kelas netral dengan gabungan kelas mengandung sentimen (net vs *sentiment equal, with duplication*) dengan penambahan sampel kelas positif dan negatif. Percobaan ditetapkan menggunakan proporsi data netral sebanyak 4500 sampel, data positif setelah duplikasi menjadi 2222, dan data negatif menjadi 2381, sehingga perbandingan antara netral:(pos:neg) menjadi 2:1:1, atau kalau digabung, perbandingan netral:(pos+neg) adalah sekitar 2:(1+1).

Tabel 4. Skema pengujian data balancing dan hasil validasi

Kelas	**Skema Balancing**			
	no-balancing	**net vs sentiment equal (net reduction)**	**all-balanced (with duplication)**	**net vs sentiment equal (with duplication)**
	14:1:2	3:1:2	1:1:1	2:1:1
netral	6664	1300	5000	4500
positif	463	463	5000	2222
negatif	873	873	5000	2381
	Hasil validasi terhadap data development			
Acc	0.84	0.77	0.66	**0.77**
F1-score	0.35	0.52	0.44	**0.53**

Pengujian baseline dilakukan untuk keempat percobaan ini, dengan komposisi kelas disajikan dalam Tabel 4 di atas. Data train yang telah mengalami tahapan *oversampling* pada pengujian ketiga, dengan rasio netral:(pos:net), yaitu 1:(1:1), memiliki tingkat performa yang lebih rendah dibandingkan rasio 2:(1:1) pada pengujian keempat. Sedangkan tanpa penyeimbangan data pada pengujian pertama, data sangat rentan terhadap *overfitting* terhadap kelas netral, dengan rasio awal sekitar 14:(1:2).

Penggunaan proporsi kelas netral yang dikurangi menjadi sebanyak data kelas positif dan negatif pada penguijan kedua, dengan rasio data menjadi 3:1:2 sebagaimana yang dilakukan pada [7] dan [16], memberikan hasil yang baik, tetapi masih lebih rendah dari cara duplikasi data. Gambar 3 berikut menunjukkan visualisasi komposisi data netral versus sentimen untuk berbagai skema *balancing* yang sudah diujikan dalam penelitian ini.



Gambar 3. Komposisi netral vs sentimen dalam skema *balancing* data

Dari hasil pengujian menggunakan metode baseline pada skema data *balancing* di atas, maka dipakai skema keempat untuk optimasi model selanjutnya, karena memiliki hasil *validasi* yang terbaik dari keempat percobaan tersebut, sebagaimana hasil dalam Tabel 4.

3.2 Kombinasi Teks *Prepocessing*

Tahapan pengolahan teks secara umum diperlukan agar teks menjadi lebih terstruktur dalam unit terkecilnya (*token*). Beberapa tahapan standar pengolahan teks yang dipakai dalam penelitian IR dan NLP, juga dapat diterapkan dalam tugas klasifikasi sentimen ini. Namun demikian, karena input berupa kalimat pendek dan tidak baku (bahasa percakapan sehari-hari), maka tidak semua tahapan pemrosesan teks dapat menghasilkan output yang baik sesuai tujuannya. Oleh karena itu, dalam menemukan hasil yang terbaik, komposisi atau variasi dari teks *preprocessing*, menjadi aspek yang menarik untuk diteliti.

Dalam penelitian ini, dicoba melihat pengaruh penerapan *case folding*, pemotongan *stop words*, dan penghilangan tanda baca (*punctuation removal*), sebagaimana pada [17] untuk menemukan model pemrosesan teks yang optimal. Mengikuti langkah pada [17], eksperimen yang dilakukan pada penelitian ini terkait pemilihan variasi teks preprocessingnya dapat dilihat pada Tabel 5 berikut.

Tabel 5. Komposisi variasi pengujian komponen Teks Preprocessing [17]

Case Folding	Stopword	Punctuation	ID Eksperimen
Tidak	Tidak	Tidak	C1
Ya	Tidak	Tidak	C2
Tidak	Ya	Tidak	C3
Tidak	Tidak	Ya	C4
Tidak	Ya	Ya	C5
Ya	Tidak	Ya	C6
Ya	Ya	Tidak	C7
Ya	Ya	Ya	C8

Semua eksperimen (C1-C8) dilakukan dengan menggunakan komposisi data yang sudah diagumentasi dengan cara duplikasi sebagaimana diterangkan pada bagian sebelumnya. Hasil pengujian terhadap data validasi, diterangkan dalam Tabel 6 berikut.

Tabel 6. Hasil pengujian komposisi teks preprocessing

ID eksperimen	akurasi	F1 score
C1	0.77	0.45
C2	0.76	0.45
C3	0.76	0.52
C4	0.79	0.52
C5	0.76	0.50
C6	**0.77**	**0.53**
C7	0.75	0.51
C8	0.76	0.52

Selanjutnya model eksperimen C6 akan dipakai untuk pengujian fitur *engineering*, yaitu optimasi terhadap besaran input k-NN.

3.3 Fitur Engineering

Optimasi selanjutnya adalah proses *scaling* (penskalaan) untuk menormalisasi tiap elemen vektor kalimat. Cara ini digunakan setelah melihat hasil eksperimen pada [17], yang menunjukkan peningkatan performa dalam tugas klasifikasi biner untuk *hate speech* dan *abusive language*. Fitur yang dipakai juga sama, yaitu *minmax* dan *robust scaler* dari *library sklearn*.

Tabel 7. Pengujian fitur engineering

Feature	Akurasi	F1-Score
NonScaler	0,77	0,53
Minmax	0,77	0,52
Robust	0,77	0,51

Dari hasil pengujian ini, ternyata untuk klasifikasi multilabel (3 kelas) tidak memiliki dampak yang positif untuk meningkatkan *F1-score* maupun akurasi. Alih-alih bertambah, justru *F1-score*-nya malah berkurang walaupun dalam jumlah yang sangat kecil (1%). Oleh karena itu, maka fitur C6 tanpa menggunakan fungsi scaling, dipilih sebagai model yang paling optimal dari k-NN, dan dipakai untuk pengujian final terhadap data test.

3.4. Pembahasan

Tahap akhir dari penelitian ini adalah pengujian terhadap data test, yang belum pernah terlihat selama proses training menghasilkan model optimal. Dari 400 sampel pada data test, sistem berhasil

mengklasifikasi kelas positif, dan negatif, sebagaimana terlihat pada *Confusion Matrix* pada Tabel 8. Akurasi yang diperoleh adalah 66% dengan *F1-score* sebesar 57%.

Tabel 8. Confusion Matrix hasil pengujian terhadap data test

		Prediksi		
		neg	net	pos
True Label	neg	**45**	54	9
	net	44	**152**	38
	pos	10	26	**22**

Hasil ini di antara penelitian lainnya, cukup kompetitif sebagaimana terlihat pada Tabel 9. Metode k-NN berada di posisi yang sama dengan *XGBoost* yang berbasis *Decision Tree*, dan *Naïve Bayes*. Namun dari segi akurasi, k-NN lebih baik dari pada *Naïve Bayes*. Sedangkan metode SVM dengan fitur TF-IDF memiliki hasil yang lebih rendah, mungkin disebabkan karena fitur TF-IDF yang kurang optimal dari segi dimensi input untuk SVM. Sedangkan LSTM memiliki hasil terendah, disebabkan karena metode berbasis *deep learning* membutuhkan data training dalam jumlah besar.

Tabel 9. Hasil Penelitian ini di antara Penelitian lainnya

Metode	Akurasi	F1-score
Naïve Bayes [4]	61%	57%
SVM + TF.IDF [5]	65%	56%
SVM + Word2Vec [6]	68%	59%
LSTM + Word2Vec [7]	66%	54%
SVM + FastText [18]	**69%**	**65%**
XGBoost [19]	66%	57%
Logistic Regression [16]	67%	60%
KNN + FastText (Penelitian ini)	**66%**	**57%**

Secara umum, metode k-NN hanya kalah dari metode *Logistic Regression* dan SVM dengan input word embeddings (*FastText dan Word2vec*). Namun, apabila fitur input k-NN tidak menggunakan *word embeddings* (misalnya menggunakan fitur *bag of words TF.IDF*), bisa jadi performanya akan jatuh menjadi lebih rendah lagi.

4. KESIMPULAN

Dari hasil penelusuran model optimal untuk k-NN, untuk kasus klasifikasi sentimen pada twitter yang banyak menggunakan bahasa non formal, maka tetap menggunakan *stopword* di dalam teks (tidak menghapusnya pada tahap *preprocessing*) dapat meningkatkan performa klasifikasi. Sedangkan menghapus tanda baca dan mengubah menjadi huruf kecil, menjadi opsi yang perlu dipilih agar hasil klasifikasi lebih baik. Penggunaan fitur *scaler* untuk normalisasi vektor input tidak diperlukan, karena tidak terbukti dapat meningkatkan hasil akurasi, bahkan menurunkan performa *F1-score*-nya.

Model k-NN yang paling optimal dari penelitian ini, mendapatkan hasil akurasi dan *F1-score* berturut-turut adalah 66% dan 57%. Hasil ini cukup kompetitif bila dibandingkan dengan metode berbasis machine learning konvensional lainnya. Sedangkan terhadap metode *deep learning*, metode k-NN memiliki performa yang jauh lebih baik, disebabkan karena metode *deep learning* membutuhkan data yang cukup besar untuk proses trainingnya, yang tidak didapatkan dari *dataset* pada kasus penelitian ini.

Saran penelitian selanjutnya yang dapat dikembangkan dari peneltian ini, menguji apakah k-NN akan dapat memiliki performa yang lebih baik bila menggunakan fitur *input bag of words* dengan TF.IDF. Hal ini perlu dilakukan bila mau menguji hipotesa yang telah disampaikan di bagian pembahasan.

Daftar Pustaka

[1] A. Makmun and S. F. Hazhiyah, "Tinjauan Terkait Pengembangan Vaksin Covid 19," *Molucca Medica*, vol. 13, pp. 52–59, 2020, doi: 10.30598/molmed.2020.v13.i2.52.

[2] B. Laurensz, and E. Sediyono, "Analisis Sentimen Masyarakat terhadap Tindakan Vaksinasi dalam Upaya Mengatasi Pandemi Covid-19, vol. 10, no. 2, pp. 118–123, 2021.

[3] Keputusan Menteri, "Keputusan Menteri Kesehatan Republik Indonesia Nomor Hk.01.07/Menkes/12757/2020 Tentang Penetapan Sasaran Pelaksanaan Vaksinasi Corona Virus Disease 2019 (Covid-19),

[4] Prima Yohana, Surya Agustian, Siska Kurnia Gusti, "Klasifikasi Sentimen Masyarakat Terhadap Kebijakan Vaksin Covid-19 pada Twitter dengan Imbalance Classes Menggunakan Naive Bayes", *SNTIKI* 14, Pekanbaru, 2022

[5] M. Rizki, *Analisis Sentimen Masyarakat Terhadap Vaksin Covid-19 Menggunakan Metode Support Vector Machine pada Media Sosial Twitter*, Tesis report, UIN Suska Riau, 2022.

[6] M. Sahbuddin and S. Agustian, "Support Vector Machine Method with Word2vec for Covid-19 Vaccine Sentiment Classification on Twitter," *Journal of Informatics and Telecommunication Engineering*, vol. 6, no. 1, pp. 288–297, Jul. 2022, doi: 10.31289/jite.v6i1.7534.

[7] M. Ihsan, B. S. Negara, and S. Agustian, "LSTM (Long Short Term Memory) for Sentiment COVID-19 Vaccine Classification on Twitter," *Digital Zone: Jurnal Teknologi Informasi dan Komunikasi*, vol. 13, no. 1, pp. 79–89, May 2022, doi: 10.31849/digitalzone.v13i1.9950.

[8] D. Hernikawati, "Kecenderungan Tanggapan Masyarakat Terhadap Vaksin Sinovac Berdasarkan Lexicon Based Sentiment Analysis," IPTEK-KOM, vol. 23, no. 1, pp. 21–31, 2021, https://doi.org/10.33164/iptekkom.23.1.2021.21%20-%2031

[9] A. Harun and D. P. Ananda, "Analysis of Public Opinion Sentiment About Covid-19 Vaccination in Indonesia Using Naïve Bayes and Decission Tree Analisa Sentimen Opini Publik Tentang Vaksinasi Covid-19 di Indonesia Menggunakan Naïve Bayes dan Decission Tree," *Indonesia Journal of Machine Learning and Computer Science*, vol. 1, no. April, pp. 58–63, 2021.

[10] M. F. Fibrianda and A. Bhawiyuga, "Analisis Perbandingan Akurasi Deteksi Serangan Pada Jaringan Komputer Dengan Metode Naïve Bayes Dan Support Vector Machine (SVM)," vol. 2, no. 9, pp. 3112–3123, 2018.

[11] V. Riandaru Prasetyo, M. Mercifia, A. Averina, L. Sunyoto, and Budiarjo, "Prediksi Rating Film Pada Website IMDB Menggunakan Metode Neural Network Film Rating Prediction on IMDB Website Using Neural Network," *Jurnal Ilmiah NERO*, vol. 7, no. 1, 2022.

[12] D. Bertsimas, J. Dunn, C. Pawlowski, and Y. D. Zhuo, "Robust Classification," *INFORMS Journal on Optimization*, vol. 1, no. 1, pp. 2–34, Jan. 2019, doi: 10.1287/ijoo.2018.0001.

[13] P. Arsi, L. N. Hidayati, and A. Nurhakim, "Komparasi Model Klasifikasi Sentimen Issue Vaksin Covid-19 Berbasis Platform Instagram," *Jurnal Media Informatika Budidarma*, vol. 6, no. 1, p. 459, Jan. 2022, doi: 10.30865/mib.v6i1.3509.

[14] W. Yunus, "Algoritma K-Nearest Neighbor Berbasis Particle Swarm Optimization Untuk Prediksi Penyakit Ginjal Kronik," *Jurnal Teknik Elektro CosPhi*, vol. 2, no. 2, pp. 2597–9329, 2018.

[15] I. Widhi Saputro and B. Wulan Sari, "Uji Performa Algoritma Naïve Bayes untuk Prediksi Masa Studi Mahasiswa Naïve Bayes Algorithm Performance Test for Student Study Prediction," *Citec Journal*, vol. 6, no. 1, 2019.

[16] Ash Shiddicky, Surya Agustian, "Analisis sentimen masyarakat terhadap kebijakan vaksinasi covid-19 pada media sosial twitter menggunakan metode logistic regression", *Jurnal Computer Science and Information Technology* (*CoSciTech*), Vol. 3 (2), pp 99-106, 2022.

[17] Afhdal Zikri, Surya Agustian, "Penerapan Support Vector Machine dan FastText untuk Mendeteksi Hate Speech dan Abusive pada Twitter", *Jurnal Media Informatika Budidarma*, Vol. 7 (1), pp 436-443, 2023

[18] Mukti M Kusairi, Surya Agustian, "SVM Method with FastText Representation Feature for Classification of Twitter Sentiments Regarding the Covid-19 Vaccination Program", *Digital Zone: Jurnal Teknologi Informasi dan Komunikasi*, Vol. 13 (2), pp 140-150, 2022

[19] Habib Hakim Sinaga, Surya Agustian, "Pebandingan Metode Decision Tree dan XGBoost untuk Klasifikasi Sentimen Vaksin Covid-19 di Twitter", *Jurnal Nasional Teknologi dan Sistem Informasi* - Vol. 8 (3), pp 107-114, 2022