

# ANALISIS SENTIMEN PENANGANAN COVID-19 DENGAN SUPPORT VECTOR MACHINE: EVALUASI LEKSIKON DAN METODE EKSTRAKSI FITUR

*Analysis Of Covid-19 Handling Sentiment With Support Vector Machine: Lexicon Evaluation  
And Feature Extraction Methods*

**Wildan F. Abdillah\*<sup>1</sup>, Agyztia Premana<sup>2</sup>, R. M. Herdian Bhakti<sup>3</sup>**

<sup>1,2,3</sup> Prodi Teknik Informatika, Fakultas Teknik, Universitas Muhadi Setiabudi, Brebes, Indonesia  
e-mail: \*<sup>1</sup>wfa.umus@gmail.com, <sup>2</sup>a.premana@umus.ac.id, <sup>3</sup>herdian.bhakti@umus.ac.id

## **Abstrak**

Peneliti melakukan uji perbandingan metode ekstraksi fitur dan leksikon bahasa Indonesia pada analisis sentimen teks di media sosial terkait penanganan Covid-19. Set data yang digunakan berasal dari penelitian sebelumnya yang divalidasi ahli bahasa. Penelitian ini bertujuan mengukur performa kamus sentimen kata dari leksikon InSet dan *sentistrength\_id* terhadap label aktual, berikutnya untuk mengetahui pengaruh pemilihan metode ekstraksi fitur term *presence*, *BoW*, dan *TF-IDF*. Proses penelitian dimulai dengan tahap prapengolahan teks, pemilihan metode ekstraksi fitur, pelabelan kata dengan leksikon, dan klasifikasi sentimen dengan *Support Vector Machine (SVM)*. Evaluasi akhir dilakukan dengan *k-Fold cross validation* menggunakan data yang disintesis dengan algoritma *SMOTE*. Pengujian awal menunjukkan bahwa kamus sentimen kata leksikon *sentistrength\_id* menghasilkan skor akurasi sedikit lebih tinggi (64,46%) dibanding *InSet* (62,65%) saat dikombinasikan dengan *TF-IDF*. Di tahap evaluasi akhir, kamus sentimen kata *sentistrength\_id* masih menunjukkan performa yang relatif lebih baik berdasarkan rerata nilai akurasi, presisi, sensitivitas, dan *f-measure* (59,22%, 61,1%, 31,3%, 41,0%) dibanding *InSet* (58,98%, 62,2%, 26,1%, 36,6%) saat dikombinasikan dengan *TF-IDF*. Secara umum, performa kedua leksikon memang masih di bawah data dengan label aktual yang mengindikasikan kedua leksikon belum cukup kuat dalam menentukan kelas kata. Penyebabnya bisa karena jumlah data yang relatif sedikit atau karena normalisasi teks belum cukup dilakukan secara maksimal.

**Kata kunci**—analisis sentiment berbasis leksikon, support vector machine, machine learning, smote

## **Abstract**

An outline of comparison study results between extraction feature methods and lexicons on a sentiment analysis described in this paper. The dataset, consisting social media text using bahasa Indonesia related to covid-19 handling by government, validated by linguists from previous research. Classifier performance is compared using each of *InSet* and *sentistrength\_id*—lexicon against the actual label, with different extraction features method (*term presence*, *BoW*, and *TF-IDF*). The research begin with text preprocessing, feature extraction, sentiment labeling, and sentiment classification using *Support Vector Machine (SVM)*. The pilot result shows that *sentistrength\_id* word sentiment dictionary (*sentiwords\_id*) generates slight better accuration score (64,46%) compared with *InSet* (62,65%) when combined with *TF-IDF* on 5 iteration *k-Fold cross validation*. We then oversample the training set using *SMOTE* and re-evaluate the model. Final result shows that *sentistrength\_id* word sentiment dictionary still generates better results based on the average accuration, precision, recall, and *f-measure* (59,22%, 61,1%, 31,3%, 41,0%) compared with *InSet* (58,98%, 62,2%, 26,1%, 36,6%) when combined with *TF-IDF*. In general, both lexicon generate lower performances compared with actual label. We conclude that text normalization process should be optimally done first to support lexicon performance on classifying text label.

**Keywords**—lexicon-based sentiment analysis, support vector machine, machine learning, smote

---

### **Informasi Artikel:**

**Submitted:** September 2021, **Accepted:** Oktober 2021, **Published:** November 2021  
**ISSN:** 2685-4902 (media online), Website: <http://jurnal.umus.ac.id/index.php/intech>

## PENDAHULUAN

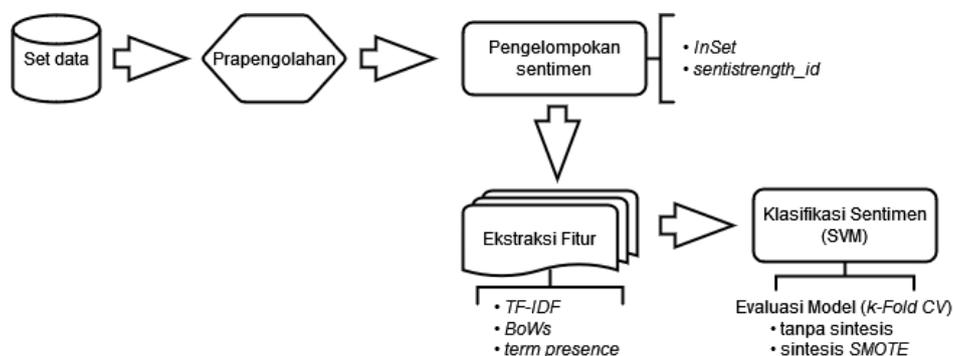
Media sosial—selain menjadi tempat bagi individu maupun organisasi untuk saling berkomunikasi—turut menjadi fungsi demokrasi bagi institusi pemerintahan, kaitannya dengan peningkatan transparansi dan partisipasi publik [1]. Dalam periode krisis kesehatan publik, pemerintah harus bekerja dengan sigap untuk mengkomunikasikan informasi secara efektif dan efisien pada masyarakat. Gagalnya komunikasi akan berakibat masyarakat menjadi takut, gamang, dan cemas di masa kritis [2]. Di penelitian sebelumnya, Prastyo et al. melakukan analisis sentimen menggunakan *Support Vector Machine (SVM)* dengan *kernel* linear dan *kernel poly* ternormalisasi pada teks di media sosial mengenai penanganan *Covid-19* oleh pemerintah Indonesia [3]. Dalam penelitian tersebut, meskipun Prastyo et al. menggunakan *sentistrength\_id* untuk menentukan kelas sentimen kata, kelas sentimen atau label yang dihasilkan tetap kemudian divalidasi ulang oleh ahli bahasa.

Analisis sentimen idealnya membutuhkan data dan sumber daya yang besar [4], terlebih jika murni dilakukan dengan pendekatan berbasis pembelajaran mesin. Sebagai alternatif, pendekatan berbasis pembelajaran mesin dapat digabungkan dengan pendekatan berbasis leksikon untuk membangun basis pengetahuan yang dibutuhkan [5]. Leksikon sentimen dan sumber leksikal dalam bahasa Inggris tersedia cukup banyak. Contohnya antara lain *SentiWordNet*, *Bing Liu lexicon*, *Sentistrength*, *LIWC*, dan *SO-CAL*. Sementara, leksikon dalam bahasa Indonesia masih terhitung sedikit jumlahnya. Contoh leksikon sentimen bahasa Indonesia yang memiliki publikasi adalah *InSet* [6] dan *sentistrength\_id*. [7].

Dalam penelitian ini, kami menguji secara empiris kamus sentimen kata dari leksikon *InSet* dan *sentistrength\_id* untuk mengetahui performa kedua leksikon bahasa Indonesia tersebut. Data teks yang akan digunakan berasal dari salah satu set data penelitian Prastyo et al. yang sudah dianotasi oleh ahli bahasa [3]. Kami juga mengobservasi pengaruh pemilihan metode ekstraksi fitur—*term presence*, *BoW*, dan *TF-IDF*—terhadap skor performa *classifier*. Hasil penelitian nantinya diukur berdasarkan skor akurasi, presisi, sensitivitas, dan *f-measure*.

## METODE PENELITIAN

Tahapan penelitian akan dimulai dari pengumpulan data, prapengolahan data, ekstraksi fitur, pengelompokan sentimen, klasifikasi sentimen, sampai dengan evaluasi model. Proses evaluasi model akan dilakukan dua kali. Pada evaluasi awal, model klasifikasi divalidasi dengan *k-Fold cross validation* sebanyak 5 iterasi. Pada evaluasi akhir, data teks akan lebih dahulu disintesis dengan algoritma *SMOTE* untuk menangani *class imbalance* sebelum divalidasi dengan *k-Fold cross validation* sebanyak 5 iterasi.



**Gambar 1. Desain Penelitian**

### Pengumpulan Data

Dalam penelitian ini, kami menggunakan salah satu set data Prastyo et al. [3], yakni data teks konten percakapan publik di media sosial *Twitter*. Data tersebut memuat cuitan untuk aspek umum mengenai penanganan *Covid-19* oleh pemerintah Indonesia. Cuitan *Twitter* tersebut diaggregasi dengan kata kunci “#COVID-19indonesia, COVID-19 di Indonesia, penanganan pemerintah terhadap COVID-19” dari tanggal 23 Maret 2020–14 Mei 2020. Peneliti memperoleh set data ini melalui pengajuan permohonan data kepada penulis jurnal secara langsung. Atas persetujuan dari penulis jurnal juga, peneliti mempublikasikan set data tersebut—bersama dengan algoritma yang digunakan dalam penelitian ini—di repositori *Github* peneliti [8]. Dari jumlah awal sebanyak 2.269, data tersebut kemudian kami pilih hanya yang memuat label positif dan negatif saja sehingga jumlahnya menjadi 1.918.

### Prapengolahan Data

Proses prapengolahan untuk penelitian ini dimulai dengan penghapusan atau normalisasi karakter *non-ASCII*, tautan (*URLs*), *mention*, konversi tagar (*hashtag*) menjadi kata, serta simbol dan angka. Proses dilanjutkan dengan pengkoreksian kata yang memuat karakter yang muncul berulang lebih dari tiga kali, penghapusan spasi yang tidak diperlukan, dan *case folding* (konversi semua karakter menjadi huruf kecil). Kemudian, proses diteruskan dengan tokenisasi untuk mengubah kata *slang* (tidak baku) menjadi bentuk formalnya dan untuk menghapus daftar *stopword*.

**Tabel 1. Proses dan Sampel Prapengolahan Data**

	Normalisasi	Sampel
1	<i>Non-ASCII</i>	Ãfâ€šÃ
2	Tautan	pic.twitter.com/pZ5O7ppWls, https://www.instagram.com/p/B-GU55- p4ag/?igshid=12cz2i3zyqvao
3	Konversi tagar	"#DiRumahAja" → "Di Rumah Aja"
4	<i>Mention</i>	@jokowi, @Pak_JK
5	Koreksi kata	Yuppp → Yup
6	<i>Case folding</i>	Pemerintah TIDAK AKAN ... → pemerintah tidak akan ...
7	Tokenisasi	pemerintah tidak akan ... → "pemerintah", "tidak", "akan", "..."
8	Koreksi kata <i>slang</i>	"skrng" → "sekarang"
9	<i>Stopword</i>	ada, adalah, asal, bahwa, belum, ...

Pengkoreksian kata *slang* dilakukan dengan memanfaatkan kamus kata *slang* bahasa Indonesia Salsabila et al. [9,10] yang dibangun berdasarkan komentar Instagram [11]. Sementara penghapusan *stopword* dilakukan dengan memanfaatkan daftar *stopword* bahasa Indonesia [12] yang merupakan modifikasi daftar *stopword* Tala [13]. Setelah semua proses dikerjakan, proses terakhir adalah pengecekan data untuk mencari dan menghapus entri duplikat, yakni baris-baris yang memuat kalimat yang sama.

### Pengelompokan Sentimen (Pelabelan Kata)

Proses pengelompokan sentimen dilakukan dengan memanfaatkan dua leksikon berbahasa Indonesia, yakni *InSet* dan *sentistrength\_id*. Kedua leksikon ini digunakan sebagai perbandingan. Dengan menggunakan data cuitan *Twitter* berbahasa Indonesia, *InSet* (*Indonesia Sentiment Lexicon*) dibangun melalui pembobotan manual pada setiap kata dan disempurnakan nilainya dengan *stemming* dan pembobotan berdasarkan sinonim kata [6]. *InSet* memuat 3.609 kata positif dan 6.609 kata negatif dengan rentang nilai antara -5 sampai dengan +5.

Sementara, *sentistrength\_id* dibangun berdasarkan model leksikon sentimen *sentistrength*, yaitu model yang menggunakan aturan dan informasi linguistik untuk mendeteksi nilai sentimen dalam teks singkat berbahasa Inggris. *Sentistrength\_id* merupakan hasil terjemahan bahasa Indonesia dari *sentistrength* dengan penambahan dan pengurangan data serta melibatkan 3 responden ahli di bidang linguistik [7]. Model leksikon ini memuat kamus sentimen, *emoticons*, ungkapan, *boosterwords* (kata penguat), kata negasi, dan kata tanya. Kecuali kata negasi dan kata tanya, semua kamus *sentistrength\_id* memiliki rentang nilai antara -5 sampai dengan +5. Sebagai catatan, kamus yang digunakan dalam penelitian ini hanya sentimen kata saja tanpa menggunakan kamus lainnya (*boosterwords*, ungkapan, negasi, dll.) dengan pertimbangan supaya uji perbandingan antara leksikon *sentistrength\_id* dan *InSet* menjadi setara.

**Tabel 2. Sampel Leksikon *InSet* dan *sentistrength\_id***

Leksikon	Sampel	Nilai
<i>InSet</i>	hai	+3
	berduka	-5
<i>sentistrength_id</i>	sakit	-4
	bagus	+4

### Ekstraksi Fitur

Setelah melalui tahap prapengolahan, teks-teks dari dua set data yang tersisa kemudian dilakukan ekstraksi fitur untuk mencari atribut yang punya pengaruh penting dalam suatu kalimat sekaligus untuk memperkecil dimensi data. Proses ekstraksi dilakukan dengan memberi pembobotan pada setiap vektor kata untuk setiap kalimat. Dalam penelitian ini, metode pemilihan fitur yang digunakan adalah *term presence*, *BoW*, dan *TF-IDF*. Ketiga metode tersebut digunakan dan diuji sebagai perbandingan untuk mengetahui performa terbaik.

### Klasifikasi Sentimen

Data-data yang sudah dikelompokkan menurut sumber leksikal *InSet* dan *sentistrength\_id* kemudian digunakan untuk melatih *classifier SVM*. Tujuannya untuk mengonversi representasi fitur ternormalisasi ke dalam vektor dalam dimensi yang sama dan dibatasi dengan *hyperplane*. Nilai yang didapatkan dari masing-masing metode ekstraksi fitur (*term presence*, *BoW*, dan *TF-IDF*) turut digunakan sebagai nilai pembobotan. Terakhir, hasil klasifikasi (prediksi nilai label) akan dianalisis dan dievaluasi menggunakan parameter skor akurasi, presisi, sensitivitas (*recall*), dan *f-measure*.

### Evaluasi Model

Proses evaluasi model akan dilakukan sebanyak dua kali. Pada evaluasi awal, model klasifikasi cukup divalidasi dengan *k-Fold cross validation* sebanyak 5 iterasi. Pada evaluasi akhir, data teks akan lebih dahulu disintesis dengan algoritma *SMOTE* sebelum divalidasi dengan *k-Fold cross validation* sebanyak 5 iterasi. Sintesis data utamanya dilakukan untuk menangani *class imbalance* serta untuk memperkuat bukti dan penatikan kesimpulan yang diperoleh dari evaluasi model.

## HASIL DAN PEMBAHASAN

### Prapengolahan Data

Peneliti menggunakan modul operasi *regular expression (Regex)* dari Python untuk mengerjakan proses prapengolahan atau normalisasi teks. Algoritma prapengolahan data yang digunakan peneliti didokumentasikan dan disimpan di repositori *Github* [8]. Sesudah semua proses normalisasi teks dikerjakan, proses terakhir adalah pengecekan data untuk menghapus

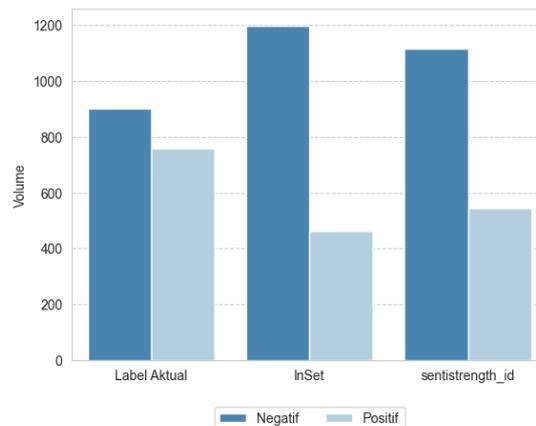
entri duplikat. Dari jumlah sampel data awal sebanyak 1.918, setelah tahap prapengolahan, jumlah data sekarang menjadi 1.660.

**Tabel 3. Sampel Teks Awal dan Teks Ternormalisasi**

Jenis	Sampel
Teks awal	Yuk Pak @jokowi #KarantinaWilayah #LockDownIndonesia sampai kapan kita melihat kenaikan kasus positif Covid-19 hari ke hari, jumlah kematian yg terus meningkat. Mari cari solusi untuk membantu segala dampak buruk lockdown bagi masyarakat. Sampai kapan #Indonesia begini Pak??
Teks ternormalisasi	yuk karantina wilayah lock down indonesia kenaikan positif covid kematian meningkat mari cari solusi membantu dampak buruk lockdown masyarakat indonesia

### Pengelompokan Sentimen (Pelabelan Kata)

Proses pengelompokan sentimen dilakukan dengan memanfaatkan dua leksikon bahasa Indonesia, yakni *InSet* dan *sentistrength\_id* yang digunakan sebagai perbandingan. Sebagai pengingat, khusus *sentistrength\_id*, untuk penelitian ini, kamus yang digunakan hanya sentimen kata saja tanpa mempertimbangkan kamus lainnya (*boosterwords*, ungkapan, negasi, dll). Algoritma yang digunakan untuk memberi nilai label (positif atau negatif) memanfaatkan modul *VADER* (*Valence Aware Dictionary and sEntiment Reasoner*) yang ada dalam *NLTK* dengan cara mengganti daftar leksikonnya [8,14].



**Gambar 2. Perbandingan Jumlah Teks Positif dan Negatif pada Pelabelan dengan Leksikon *InSet*, *sentistrength\_id*, dan Label Aktual**

Dari data teks yang dikelompokkan, leksikon *InSet* menghasilkan 1.197 label negatif dan 463 label positif. Sentimen kata *sentistrength\_id* menghasilkan 1.114 label negatif dan 546 label positif. Sedangkan, label aktual atau label yang dianotasi ahli bahasa pada set data asli, memiliki 900 label negatif dan 760 label positif.

### Ekstraksi Fitur

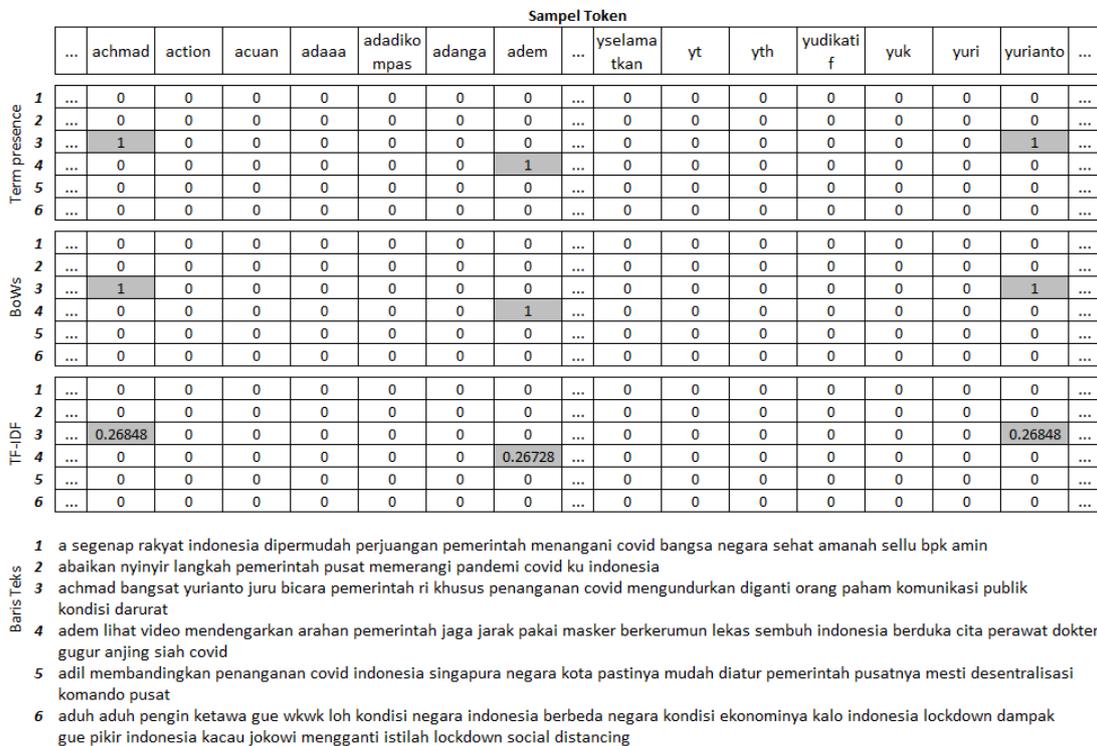
Sebelum tahap ekstraksi fitur dikerjakan, data teks dilakukan tokenisasi kembali, kemudian dibagi menjadi data latih (*training*) dan data uji (*testing*) dengan proporsi 70:30 untuk data latih dan data uji secara berturut-turut. Jadi, dari jumlah 1.660 sampel data setelah tahap prapengolahan, data dibagi menjadi 1.162 data latih dan 498 data uji. Sampel tokenisasi data latih dan data uji dapat dilihat pada Tabel 4. Nomor index menunjukkan nomor baris pada set data setelah tahap prapengolahan.

Peneliti menggunakan modul *word\_tokenize* dari *NLTK* untuk tokenisasi kata, serta modul *CountVectorizer* dan *TfidfVectorizer* dari *scikit-learn* untuk ekstraksi fitur [14,15]. Algoritma ekstraksi fitur yang digunakan didokumentasikan dan disimpan di repositori *Github* peneliti [8]. Semua pemvektor kata (*vectorizer*) yang digunakan menggunakan parameter *default*, tanpa menggunakan *max\_df* atau *min\_df*. Dengan kata lain, semua vektor atau sampel kata dalam set data turut dihitung untuk pembobotan. Sebagai catatan, fungsi *random.seed()* dari *numpy* dipakai untuk menghasilkan randomisasi yang terukur.

**Tabel 4. Sampel Tokenisasi Data Latih dan Data Uji**

Index	Sampel Teks dengan Tokenisasi	Jenis
211	['covid', 'menyebabkan', 'health', 'care', 'sy...	Data
685	['konsekuensi', 'mudah', 'kebijakan', 'lockdow...	Latih
...		(jumlah
951	['naudzubillah', 'min', 'dzalik', 'lihat', 'ke...	1.162)
858	['melonjaknya', 'covid', 'indonesia', 'bertamb...	
654	['kemunculan', 'covid', 'pemerintah', 'memikir...	Data
6	['aduh', 'seniman', 'eh', 'selebriti', 'indone...	Uji
...		(jumlah
787	['maaf', 'kebijakan', 'pemerintah', 'meresahka...	498)
1192	['penanganan', 'covid', 'indonesia', 'lanjutka...	

Hasil yang diperoleh dari masing-masing metode ekstraksi fitur, yaitu *term presence*, *BoW*, dan *TF-IDF*, lalu digunakan sebagai data latih dan data uji pada *classifier* (algoritma klasifikasi), menggantikan data latih dan data uji sebelumnya. Sampel hasil masing-masing metode ekstraksi fitur dapat dilihat pada Gambar 3.



**Gambar 3. Sampel Hasil Ekstraksi Fitur dari Setiap Metode**

### Klasifikasi Sentimen

Penelitian ini menggunakan *SVM* sebagai algoritma klasifikasi (*classifier*). Data latih dan data ujinya diperoleh dari masing-masing hasil ekstraksi fitur (*term presence*, *BoW*, dan *TF-IDF*) dan hasil pelabelan leksikon (*InSet* dan *sentistrength\_id*). Algoritma yang digunakan untuk klasifikasi sentimen memanfaatkan modul *svm*, *accuracy\_score*, *model\_selection*, dan *LabelEncoder* dari *scikit-learn* [15]. *SVM* dengan *kernel* linear dipilih sebagai *classifier* karena klasifikasi yang dilakukan dalam penelitian ini berupa klasifikasi biner atau linear. Sebagai catatan, parameter *random\_state* dalam *classifier* dipakai untuk menghasilkan randomisasi yang terukur. Demi alasan penyederhanaan, parameter regularisasi dalam *classifier* tidak dikonfigurasi, dibiarkan sesuai pengaturan bawaan ( $C=1.0$ ).

**Tabel 5. Sampel Hasil Prediksi Klasifikasi Sentimen dari Setiap Jenis Pelabelan dan Metode Ekstraksi Fitur**

	Indeks Teks Data Uji										
	654	6	507	1276	1365	1404	1136	1313	316	1465	...
L0	0	0	0	1	1	1	1	0	1	1	...
L0 + EF1	0	0	0	1	1	1	0	1	0	0	...
L0 + EF2	0	0	0	1	1	1	0	1	0	0	...
L0 + EF3	0	0	0	1	1	1	0	1	0	0	...
L1 + EF1	0	0	0	0	1	0	0	0	0	0	...
L1 + EF2	0	0	0	0	1	0	0	0	0	0	...
L1 + EF3	0	0	0	0	0	0	0	0	0	0	...
L2 + EF1	0	0	0	1	1	1	1	0	1	1	...
L2 + EF2	0	1	0	1	1	1	1	0	1	1	...
L2 + EF3	0	0	0	1	1	1	0	0	1	1	...

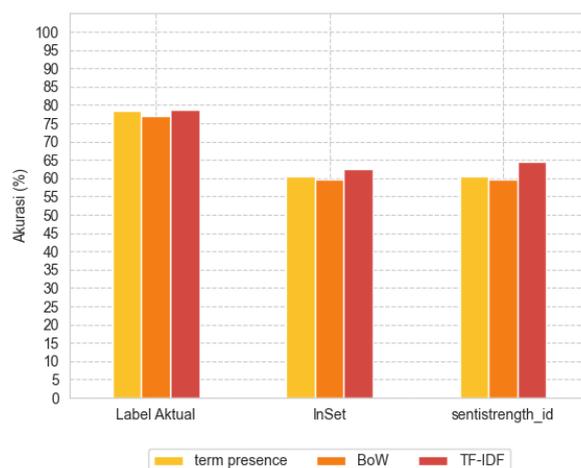
\*Keterangan: L0 = Label orisinal (Prastyo et al.), L1 = *InSet*, L2 = *sentistrength\_id*; EF1 = *term presence*, EF2 = *BoW*, EF3 = *TF-IDF*; 0 = Negatif, 1 = Positif.

Sampel hasil prediksi klasifikasi sentimen dari masing-masing jenis pelabelan dan metode ekstraksi fitur dapat dilihat pada Tabel 5. Sementara, perbandingan skor akurasi klasifikasi sentimen yang dihasilkan dapat dilihat pada Tabel 6 dan Gambar 4. Untuk setiap jenis pelabelan, hasil awal yang diperoleh menunjukkan bahwa metode ekstraksi fitur *TF-IDF* mengungguli skor akurasi metode ekstraksi fitur lainnya. Saat dikombinasikan dengan *TF-IDF*, data dengan pelabelan leksikon sentimen kata *sentistrength\_id* memiliki skor akurasi sedikit lebih tinggi (64,46%) dibanding data dengan pelabelan leksikon *InSet* (62,65%), meski masih jauh dibawah skor akurasi dari data dengan label orisinal (78,71%) dengan selisih ~14%.

**Tabel 6. Skor Akurasi Klasifikasi Sentimen Berdasarkan Jenis Label dan Metode Ekstraksi Fitur**

Ekstraksi Fitur	Akurasi Label (%)		
	Orisinal	<i>InSet</i>	<i>sentistrength_id</i>
<i>Term presence</i>	78,51	60,44	60,44
<i>BoW</i>	76,91	59,64	59,64
<i>TF-IDF</i>	78,71	62,65	64,46

Pada tahap ini, skor akurasi tertinggi setiap label didapat ketika dikombinasikan dengan *TF-IDF* dibanding metode ekstraksi fitur lainnya. Di tahap evaluasi model setelah ini, data akan disintesis untuk melihat jika terdapat misinterpretasi *false positive* dan *false negative* karena memang ada *class imbalance* pada data teks (data prapengolahan) yang digunakan dalam penelitian untuk semua jenis pelabelan.



**Gambar 4. Skor Akurasi Klasifikasi Sentimen Berdasarkan Jenis Label dan Metode Ekstraksi Fitur**

### Evaluasi Model

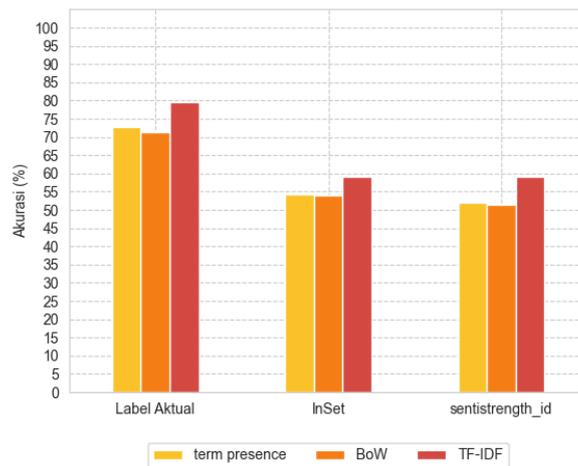
Untuk mengetahui performa model dengan lebih spesifik, model klasifikasi kemudian dievaluasi menggunakan *k-Fold cross validation*. Data yang digunakan untuk evaluasi adalah data latih (teks dan label) dari pengujian awal. Algoritma yang digunakan memanfaatkan modul *KFold* dari *scikit-learn*. Dalam penelitian ini, jumlah iterasi *k-Fold* yang digunakan adalah 5. Artinya, 1.162 data latih yang digunakan dalam pengujian awal akan dibagi lagi menjadi 80% untuk data latih evaluasi dan 20% data uji evaluasi. Untuk setiap iterasi, *k-Fold cross validation* akan mengacak kombinasi data latih dan data uji evaluasi.

Semua jenis data label yang digunakan memiliki jumlah sampel yang tidak berimbang, walaupun rasionya berbeda-beda. Oleh sebab itu, sebelum dievaluasi menggunakan *k-Fold cross validation*, data latih dari pengujian awal terlebih dahulu dilakukan sintesis data. Tujuannya untuk memperkuat bukti dan penarikan kesimpulan yang diperoleh dari evaluasi model. Dalam penelitian ini, sintesis data dilakukan dengan metode *oversampling* menggunakan algoritma *borderline-SMOTE SVM (SVM-SMOTE)* dari *package imbalanced-learn* [16].

**Tabel 7. Akurasi Rerata Klasifikasi Sentimen dengan *k-Fold CV* Berdasarkan Jenis Label dan Metode Ekstraksi Fitur**

Ekstraksi Fitur	Akurasi Label (%)		
	Orisinal	<i>InSet</i>	<i>sentistrength_id</i>
<i>Term presence</i>	72,71	54,16	51,99
<i>BoW</i>	71,45	54,04	51,45
<i>TF-IDF</i>	79,52	58,98	59,22

Demi penyederhanaan dalam visualisasi dan perbandingan, nilai akurasi yang ada pada Tabel 7 dan Gambar 5 adalah skor akurasi rata-rata dari masing-masing skor akurasi setiap iterasi *k-Fold cross validation*. Penurunan skor akurasi setelah data disintesis dan divalidasi silang adalah hal normal, semakin besar *imbalanced class* yang terjadi di data awal, semakin besar pula penurunan skor akurasi yang terjadi. Berdasarkan hasil yang diperoleh, untuk setiap jenis pelabelan, penggunaan metode ekstraksi fitur *TF-IDF* masih mengungguli skor akurasi metode ekstraksi fitur lainnya.



**Gambar 5. Akurasi Rerata Klasifikasi Sentimen dengan *k-Fold CV* Berdasarkan Jenis Label dan Metode Ekstraksi Fitur**

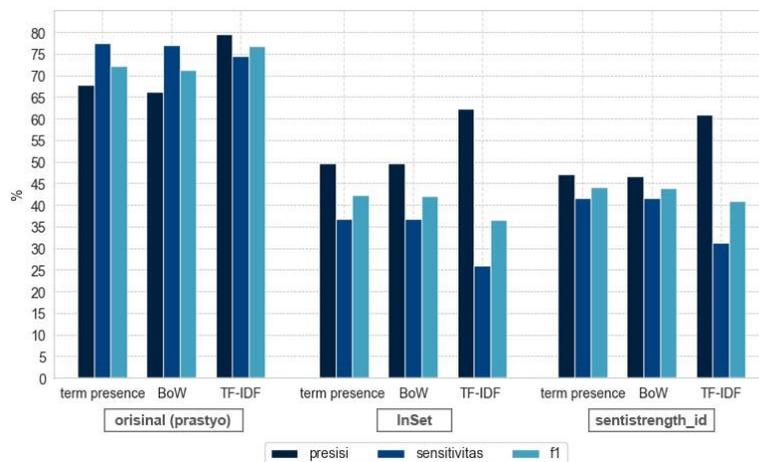
Saat dikombinasikan dengan *TF-IDF*, data dengan pelabelan leksikon sentimen kata *sentistrength\_id* memiliki skor akurasi yang sedikit lebih tinggi (59,22%) dibanding data dengan pelabelan leksikon *InSet* (58,98%), tetapi memiliki skor akurasi lebih rendah saat dikombinasikan dengan *term presence* atau *BoW*. Namun demikian, kedua leksikon ini menghasilkan skor akurasi yang masih jauh di bawah dibanding label orisinal (79,52%) dengan selisih ~20%.

**Tabel 8. Rerata Nilai Parameter Evaluasi dari 5 iterasi *k-Fold CV* pada Setiap Jenis Pelabelan dan Metode Ekstraksi Fitur**

Ekstraksi Fitur	Label Orisinal			<i>InSet</i>			<i>sentistrength_id</i>		
	%p	%s	%f	%p	%s	%f	%p	%s	%f
EF1	67,8	77,4	72,2	49,7	36,8	42,3	47,1	41,7	44,2
EF2	66,3	77,1	71,2	49,6	36,8	42,2	46,7	41,7	44,0
EF3	79,6	74,6	76,9	62,2	26,1	36,6	61,1	31,3	41,0

\*Keterangan: EF1 = *term presence*, EF2 = *BoW*, EF3 = *TF-IDF*;

%p = rerata presisi (%), %s = rerata sensitivitas (%), %f = rerata *f-measure* (%)



**Gambar 6. Rerata Nilai Parameter Evaluasi dari 5 iterasi *k-Fold CV* pada Setiap Jenis Pelabelan dan Metode Ekstraksi Fitur**

Hasil evaluasi untuk nilai presisi, sensitivitas, dan *f-measure*, sebagaimana yang dapat dilihat pada Tabel 8 dan Gambar 6, adalah nilai rata-rata dari seluruh hasil yang didapat pada setiap iterasi *k-Fold cross validation* demi penyederhanaan dalam visualisasi. Pada data dengan label orisinal, rerata nilai presisi dan *f-measure* tertinggi diperoleh saat dikombinasikan dengan *TF-IDF* (79,6% dan 76,6%). Sementara, rerata nilai sensitivitas tertinggi diperoleh saat dikombinasikan dengan *term presence* (77,4%). Pada data dengan pelabelan *InSet* dan *sentistrength\_id*, rerata *f-measure* tertinggi didapat ketika dikombinasikan dengan *term presence* (42,3% dan 44,2%). Rerata presisi tertinggi didapat ketika dikombinasikan dengan *TF-IDF* (62,2% dan 61,1%). Rerata sensitivitas tertinggi didapat ketika dikombinasikan dengan *term presence* atau *BoW* (36,8% dan 41,7%).

## KESIMPULAN

Berdasarkan hasil yang diperoleh, penggunaan kamus sentimen kata dari leksikon *sentistrength\_id* menghasilkan skor akurasi sedikit lebih tinggi dibanding leksikon *InSet* pada saat digabungkan dengan metode ekstraksi fitur *TF-IDF*. Di tahap klasifikasi sentimen, skor akurasi yang diperoleh yaitu 64,46% (*sentistrength\_id*) dan 62,65% (*InSet*). Di tahap evaluasi model, skor akurasi yang diperoleh yaitu 59,22% (*sentistrength\_id*) dan 58,98% (*InSet*). Adanya penurunan akurasi di tahap evaluasi model adalah hal yang normal. Sebab, di tahap evaluasi model, data latih disintesis terlebih dahulu dengan metode *oversampling* dan divalidasi silang. Sintesis data dilakukan karena memang terjadi *imbalanced class* pada data yang digunakan di semua jenis pelabelan.

Untuk perbandingan metode ekstraksi fitur, di tahap klasifikasi sentimen maupun evaluasi model, bukti empiris menunjukkan bahwa penggunaan metode ekstraksi fitur *TF-IDF* menghasilkan skor akurasi paling tinggi dibanding metode ekstraksi fitur lain pada semua jenis pelabelan. Memang, penggunaan *TF-IDF* menghasilkan rerata nilai presisi, sensitivitas, dan *f-measure* tertinggi pada label aktual saja (79,6%, 74,6%, 76,9%), tetapi tidak pada pelabelan dengan *InSet* maupun *sentistrength\_id*. Namun, perlu diingat bahwa label aktual adalah tolak ukur bagi jenis pelabelan lain yang diperbandingkan sehingga evaluasi perlu difokuskan pada pelabelan dengan *InSet* dan *sentistrength\_id*.

Kesimpulan akhir yang diambil dari penelitian ini adalah kedua leksikon sentimen yang digunakan belum cukup kuat dalam menentukan kelas sentimen kata. Penyebabnya ada beberapa kemungkinan, seperti jumlah sampel yang sedikit atau karena tahap prapengolahan data tidak dilakukan secara maksimal sehingga ragam kata yang tidak termuat dalam kamus sentimen kata tidak dikalkulasi dengan baik. Terkait penggunaan metode ekstraksi fitur, catatan pertama adalah penggunaan *TF-IDF* hanya akan optimal jika pelabelan kata dikerjakan dengan baik. Pada kasus data yang tidak dilakukan normalisasi kata secara optimal, *TF-IDF* justru akan membuat komputasi jadi lebih kompleks karena ragam kata berbeda, tetapi sebenarnya punya makna sama akan dihitung sebagai vektor kata yang berbeda. Jika tetap dipaksakan pada sampel teks dan pelabelan yang buruk, penggunaan *TF-IDF* mengakibatkan penurunan nilai akurasi, presisi, sensitivitas, dan *f-measure*.

## DAFTAR PUSTAKA

- [1] DePaula N, Dincelli E, Harrison TM, "Toward a typology of government social media communication: Democratic goals, symbolic acts and self-presentation", *Government Information Quarterly*, 2018, 35(1), 98–108, doi: 10.1016/j.giq.2017.10.003.
- [2] Chen Q, Min C, Zhang W, Wang G, Ma X, Evans R, "Unpacking the black box: How to promote citizen engagement through government social media during the COVID-19 crisis", *Computers in Human Behavior*, 2020, 12, doi: 10.1016/j.chb. 2020.106380.

- [3] Prastyo PH, Sumi AS, Dian AW, Permanasari AE, “Tweets Responding to the Indonesian Government’s Handling of COVID-19: Sentiment Analysis Using SVM with Normalized Poly Kernel”, *Journal of Information Systems Engineering and Business Intelligence*, 2020, 11.
- [4] Liu B, *Sentiment analysis: mining opinions, sentiments, and emotions*. New York, NY: Cambridge University Press, 2015.
- [5] Cambria E, Das D, Bandyopadhyay S, Feraco A, editors, *A Practical Guide to Sentiment Analysis*. vol. 5. Cham: Springer International Publishing, 2017. doi: 10.1007/978-3-319-55394-8.
- [6] Koto F, Rahmaningtyas GY, “Inset lexicon: Evaluation of a word list for Indonesian sentiment analysis in microblogs”, 2017 International Conference on Asian Language Processing (IALP), Singapore: IEEE, 2017, p. 391–4, doi: 10.1109/IALP.2017.391–4.
- [7] Wahid DH, Azhari SN, “Peringkasan Sentimen Esktraktif di Twitter Menggunakan Hybrid TF-IDF dan Cosine Similarity”, *Indonesian Journal of Computer Cybernetics System (IJCCS)*, 2016, 10:207, doi: 10.22146/ijccs.16625.
- [8] Abdillah WF, *sentimen-bahasa: Implementasi analisis sentimen untuk teks berbahasa Indonesia di media sosial*, 2021, [Online], tautan: <https://github.com/onpilot/sentimen-bahasa>, diakses pada: 25 Agustus 2021.
- [9] Salsabila NA, *Kamus Alay - Colloquial Indonesian Lexicon*, 2020, [Online], tautan: <https://github.com/nasalsabila/kamus-alay>, diakses pada: 26 Mei 2021.
- [10] Salsabila NA, Winatmoko YA, Septiandri AA, Jamal A, “Colloquial Indonesian Lexicon”, 2018 International Conference on Asian Language Processing (IALP), Bandung, Indonesia: IEEE; 2018, p. 226–9, doi: 10.1109/IALP.2018.8629151.
- [11] Septiandri AA, Wibisono O, “Detecting spam comments on Indonesia’s Instagram posts”, *Journal of Physics: Conference Series*, 2017, 801:012069, doi: 10.1088/1742-6596/801/1/012069.
- [12] Wahid DH, *ID-Stopwords: Stopwords collection of Bahasa Indonesia*, 2020, [Online]. tautan: <https://github.com/masdevid/ID-Stopwords>, diakses pada: 26 Mei 2021.
- [13] Tala FZ, *A Study of Stemming Effects on Information Retrieval in Bahasa Indonesia*, 2003, M. Sc. Thesis, Institute for Logic, Language and Computation (ILLC) Universiteit van Amsterdam, Amsterdam, Netherlands.
- [14] Bird S, Loper E, Klein E, *Natural Language Processing with Python*, O’Reilly Media Inc., 2009.
- [15] Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, et al, “Scikit-learn: Machine Learning in Python”, *Journal of Machine Learning Research*, 2011, 12, 2825–30.
- [16] Lemaître G, Nogueira F, Aridas CK, “Imbalanced-learn: A Python Toolbox to Tackle the Curse of Imbalanced Datasets in Machine Learning”, *Journal of Machine Learning Research*, 2017, 18(17), 1–5.