

KOMPARASI ALGORITMA SUPPORT VECTOR MACHINE DAN RANDOM FOREST UNTUK ANALISIS SENTIMEN METAVERSE

Putri Kumala Sari¹, Ryan Randy Suryono²

^{1,2}Program Studi Sistem Informasi, Fakultas Teknik dan Ilmu Komputer
Universitas Teknokrat Indonesia
ryan@teknokrat.ac.id

ABSTRAK

Fenomena metaverse menggambarkan transformasi signifikan dalam interaksi manusia dengan dunia digital. Saat ini, pemerintah dan industri berupaya memahami arah pengembangan metaverse untuk mewujudkan digitalisasi dalam berbagai sektor. Penelitian ini menggunakan data media sosial X untuk melakukan analisis sentimen publik terhadap metaverse. Metode klasifikasi teks dibangun menggunakan model algoritma *Support Vector Machine* dan *Random Forest*. Dengan melakukan komparasi kedua model tersebut dan menerapkan metode optimasi SMOTE maka eksperimen ini menghasilkan akurasi yang tinggi. Hasilnya menunjukkan bahwa kedua model menghasilkan akurasi sebesar 91% untuk model algoritma Random Forest dan 90% untuk model algoritma Support Vector Machine. Dengan demikian dapat disimpulkan bahwa model algoritma Random Forest lebih baik dari Support Vector Machine. Selain itu penerapan SMOTE terbukti meningkatkan kemampuan untuk mengenali sentimen positif pada keduanya, meskipun terjadi *trade-off* antara *recall* dan *precision*. *Trade-off* terjadi karena adanya keterbatasan dalam model klasifikasi yang membuat peningkatan dalam satu metrik akan mengakibatkan penurunan dalam metrik lainnya, karena algoritma harus mengambil keputusan yang mengoptimalkan keduanya secara bersamaan.

Keyword : *metaverse, analisis sentimen, support vector machine, random forest, SMOTE.*

1. PENDAHULUAN

Pesatnya pertumbuhan media sosial dan platform online telah menghasilkan banyak data, termasuk opini dan sentimen masyarakat. Media sosial yang digunakan dalam penelitian ini adalah Twitter. Berdasarkan data yang dirilis oleh We Are Social, jumlah pengguna Twitter global mencapai 666,2 juta pada Oktober 2023, sehingga menempatkan platform tersebut peringkat ke-12 berdasarkan jumlah pengguna. Twitter juga sering digunakan sebagai sumber data untuk analisis tren dan opini publik [1].

Melalui media sosial twitter yang saat ini telah berubah nama menjadi X sejak awal Juli 2023. Kita dapat mengakses data dan informasi yang dapat digunakan sebagai materi penelitian analisis sentimen. Setelah pengumuman Mark Zuckerberg mengenai rencananya untuk mengubah nama perusahaan induknya dari Facebook menjadi Meta Platforms Inc atau Meta Exactly, kita melihat bahwa topik *metaverse* sedang menjadi sorotan utama dalam berita dan percakapan publik [2]. Pada tahun 1992, konsep metaverse pertama kali muncul. Metaverse merupakan sebuah dunia virtual yang mencakup ruang dan objek digital dimana pengguna dapat berinteraksi secara *real time* dengan lingkungan yang dibuat oleh komputer [3].

Metaverse juga dapat dianggap sebagai representasi virtual dari dunia nyata, di mana pengguna dapat berinteraksi, bekerja, berkolaborasi. Metaverse juga telah menjadi topik yang semakin populer dalam industri teknologi, terutama dengan

perkembangan teknologi *Virtual Reality* (VR) dan *Augmented Reality* (AR) yang semakin canggih [4].

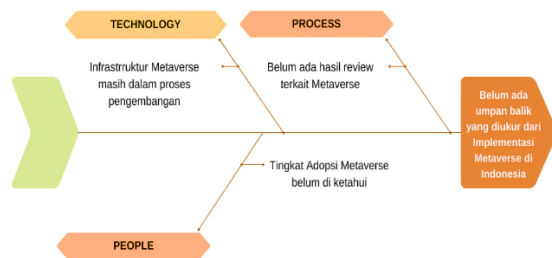
Penelitian sebelumnya telah menggunakan algoritma Support Vector Machine dan metode pemberian bobot TF-IDF untuk menganalisis upaya pemberantasan korupsi di Indonesia. Hasil studi menunjukkan bahwa dari 1890 dan 3846 data term yang telah diuji, nilai akurasi 82%, presisi 90%, recall 88%, dan skor F1 89% [5]. Sebuah penelitian lain yang membahas opini terkait lockdown di Jakarta juga menggunakan algoritma Support Vector Machine dan pembobotan TF-IDF untuk menganalisis sentimen. Data yang digunakan diambil dari platform media sosial X, dan hasil penelitian menunjukkan tingkat akurasi sebesar 74%, presisi 75%, recall 92%, dan F1-score 83% [6].

Dalam sebuah penelitian sebelumnya, dilakukan analisis mengenai sentimen terhadap layanan Indihome dari Twitter dengan menggunakan algoritma SVM. Data yang digunakan berjumlah 280 tweet, dan hasil evaluasi menggunakan Confusion Matrix menunjukkan akurasi sebesar 87%, presisi 86%, recall 95%, dan F1-score 90% [7]. Pada penelitian sebelumnya, diterapkan metode Modified Balanced Random Forest dengan menggunakan Mutual Information yang menghasilkan F1-score sebesar 75% dan tingkat akurasi 79%. Penelitian ini melibatkan langkah stemming dalam proses preprocessing data. Pada dataset lain yang memuat ulasan film dalam bahasa Inggris yang tidak seimbang, Modified Balanced Random Forest digunakan untuk

meningkatkan nilai F1-score dari Random Forest sebesar 27% [8].

Penelitian ini bertujuan mengevaluasi sentimen masyarakat terhadap perkembangan teknologi metaverse, dengan melakukan komparasi kinerja algoritma klasifikasi *Support Vector Machine (SVM)* dan *Random Forest*. Melalui penerapan metode optimasi SMOTE untuk mendapatkan kinerja yang optimal dari kedua algoritma tersebut, penelitian ini juga berusaha menentukan algoritma klasifikasi yang paling efektif. Hasilnya diharapkan dapat memberikan pemahaman yang lebih mendalam terkait metaverse serta menjadi kontribusi bagi penelitian teknologi dan inovasi di Indonesia.

Setelah mengetahui latar belakang, penulis mengidentifikasi dan menjelaskan penyebab masalah. Proses analisis dan identifikasi tersebut diilustrasikan melalui diagram *fishbone* yang dapat dilihat pada Gambar 1 dibawah ini.



Gambar 1. Diagram *Fishbone*

Gambar 1 menjelaskan belum adanya umpan balik yang diukur dari implementasi *metaverse* di Indonesia menjadi kendala dalam memahami dampak yang ditimbulkannya. Kurangnya data dan informasi mengenai efek positif dan negatif dari penggunaan *metaverse* menghambat penilaian terhadap nilai serta manfaatnya bagi masyarakat. Faktor – faktor penyebab yang menyebabkan hal ini meliputi tingkat adopsi metaverse yang belum diketahui, di mana masyarakat Indonesia belum sepenuhnya mengadopsi *metaverse* sebagai bagian dari kehidupan digital mereka karena kurangnya pemahaman dan kesadaran mengenai potensi penggunaannya. Selain itu, belum ada hasil review yang memadai terkait implementasinya.

2. TINJAUAN PUSTAKA

2.1 *Text Mining*

Text mining adalah proses yang berfokus pada pencarian pengetahuan dari data yang tersimpan dalam bentuk dokumen atau teks. Tujuan dari proses ini adalah untuk mengekstraksi dan mengidentifikasi informasi yang berguna. Untuk memenuhi kebutuhan analisis yang akan dilakukan, data mentah yang dihasilkan dari proses ini harus disiapkan terlebih dahulu.[9].

2.2 Analisis Sentimen

Analisis sentimen merupakan proses untuk menganalisis dan mengekstrak sentimen atau opini dari teks atau data yang dikumpulkan dari berbagai sumber, seperti, ulasan produk, atau komentar pengguna dan media sosial. Tujuan analisis sentimen yaitu untuk memahami pandangan, perasaan, atau sikap yang terkandung dalam teks tersebut, apakah itu positif, negatif, atau netral [10].

2.3 Twitter

Twitter merupakan media sosial yang memungkinkan pengguna untuk berbagi pemikiran, informasi, dan opini dalam pesan singkat yang disebut “*tweets*”. Serta pengguna dapat mengirim dan menerima *tweets*, dan berinteraksi dengan pengguna lain melalui fitur like, komentar dan *retweet* [11].

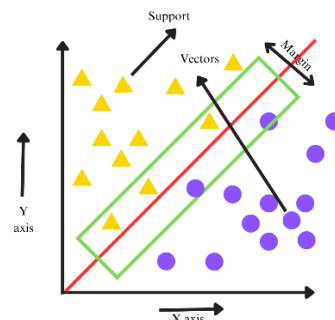
2.4 Klasifikasi

Klasifikasi merupakan teknik yang digunakan untuk memprediksi sentimen atau polaritas teks berdasarkan karakteristiknya. Algoritma klasifikasi yang digunakan penelitian ini yaitu *Support Vector Machine(SVM)* dan *Random Forest*.

2.4.1 *Support Vector Machine (SVM)*

Support Vector Machine (SVM) adalah sebuah algoritma yang bermanfaat untuk mengelompokkan data, baik yang memiliki pola linear maupun non-linear. Algoritma ini berhasil menangani masalah yang berkaitan dengan pola non-linear karena mampu memanfaatkan konsep kernel untuk memproyeksikan data ke dalam ruang dimensi yang lebih besar. Jenis objek yang berbeda dapat dibagi ke dalam kelompok-kelompok yang berbeda dengan menggunakan SVM. Model *supervised*, pendekatan klasifikasi SVM menggunakan sampel data yang telah diklasifikasikan sebelumnya untuk memprediksi kelas yang mungkin ada dengan menggunakan sampel data yang sudah ada [12].

Ini karena kernel linear memiliki fungsi yang paling sederhana dan dapat digunakan untuk klasifikasi data yang tidak terklasifikasi dengan linear [13]. Ilustrasi SVM digambarkan oleh Gambar 2.



Gambar 2. Algoritma *Support Vector Machine*

Rumus SVM dijelaskan pada persamaan 1 berikut ini :

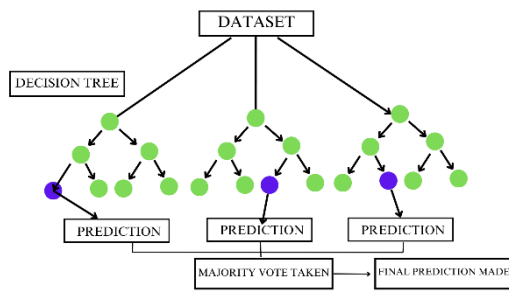
$$f(x) = \text{sign}(\sum_{i=1}^n a_i y_i K(x_i, x) + b) \quad (1)$$

Keterangan :

- (x) : data yang akan diklasifikasikan
- x_i : data latih
- y_i : label kelas
- a_i : bobot
- $K(x_i, x)$: kernel yang menghitung jarak antara x_i dan x
- b = bias

2.4.2 Random Forest

Random Forest merupakan algoritma yang membentuk pohon-pohon klasifikasi dan regresi, di mana nodenya dipisahkan menggunakan algoritma yang dioptimalkan sebagai fungsi untuk meminimalkan *squared-error loss* [14]. Random Forest dibuat dengan memilih atribut secara acak dan menerapkan metode CART untuk membentuk pohon keputusan. Pohon – pohon tersebut kemudian membentuk kumpulan pohon yang disebut sebagai forest. Algoritma Random Forest direpresentasikan pada Gambar 3.



Gambar 3. Algoritma Random Forest.

Rumus random forest dapat dilihat pada persamaan 2.

$$F(x) = \frac{1}{J} \sum_{j=1}^J h_j(x) \quad (2)$$

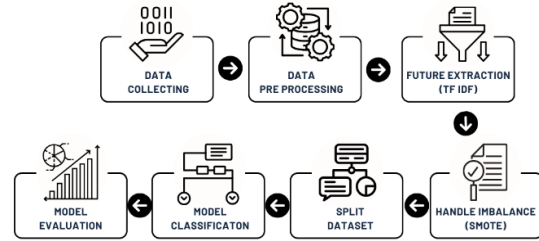
Keterangan :

- $F(x)$: output dari Random Forest
- J : jumlah pohon dalam ensemble
- $h_j(x)$: output dari pohon ke – (j)

3 METODE PENELITIAN

Alur penelitian di bagi menjadi 6 tahapan yaitu diawali dengan tahap *Data Collection*, kemudian *preprocessing*, *future extraction* menggunakan metode TF-IDF, selanjutnya ketidakseimbangan data atau *Handle Imbalance* diatasi dengan teknik SMOTE, lalu dibagi menjadi data uji dan data latih untuk tahap klasifikasi model berikutnya menggunakan algoritma *Support Vector Machine* (SVM) dan *Random Forest*. Tahap terakhir yaitu Evaluasi model menggunakan metrik *accuracy*, *precision*, *recall*, dan *F1-score* untuk

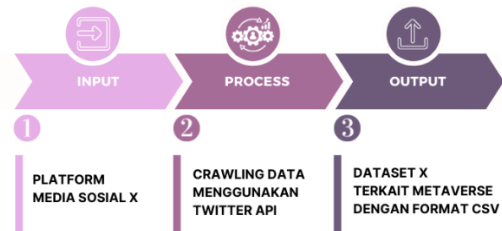
membandingkan performa kedua algoritma dalam menganalisis sentimen pada platform X terkait metaverse. Gambar 4 merupakan ilustrasi dari alur penelitian ini.



Gambar 4. Alur penelitian

3.1 Data Collecting

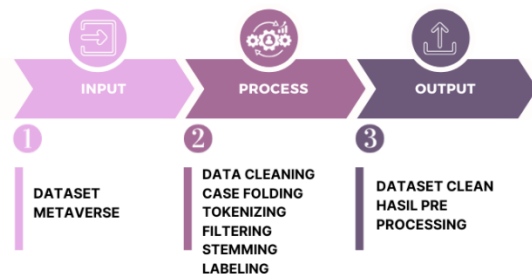
Dataset diambil melalui platform media sosial Twitter menggunakan metode crawling data dengan memanfaatkan bahasa pemrograman Python. Menggunakan library Harvest untuk mengumpulkan *tweets* yang mengandung kata kunci “metaverse”. Jumlah data yang diambil dalam rentang waktu Januari hingga April 2023 adalah 6348 *tweets*. Dataset berisi teks berbahasa Indonesia. Selanjutnya data tersebut disimpan dalam format csv.



Gambar 5. Alur pengumpulan data

3.2 Data Preprocessing

Preprocessing merupakan proses mengubah data tak terstruktur menjadi data terstruktur sesuai dengan format yang dibutuhkan [13], tujuan dari proses ini adalah untuk memastikan data mudah dibaca [14]. Gambar 6 adalah alur *preprocessing* data.



Gambar 6. Alur preprocessing data

Beberapa Teknik *preprocessing* yang diterapkan dalam studi penelitian ini antara lain :

3.2.1 Data Cleaning

Data Cleaning merupakan prosedur untuk menghilangkan tanda baca seperti titik, koma, tanda seru, tanda tanya dan kurung. Serta data duplikat dan simbol seperti nama pengguna, simbol *re-tweet* dan URL [15]. Dataset yang terdiri dari 6348 *tweets* setelah dilakukan proses *cleaning*, data nya berkurang menjadi 5857 *tweets*.

3.2.2 Case Folding

Case Folding digunakan untuk mengubah huruf dalam dokumen teks menjadi huruf kecil, [16], dengan tujuan untuk membuat karakter dalam data menjadi seragam. Seluruh informasi yang mengandung karakter selain huruf dan angka akan dihilangkan [17].

3.2.3. Tokenizing

Tokenizing merupakan proses memecah teks atau data yang berbentuk kalimat atau dokumen menjadi bagian kecil yang disebut "token".

3.2.4. Filtering (Stopword Removal)

"Stopword" mengacu pada kata-kata yang memiliki frekuensi tinggi dalam suatu dokumen, baik dari segi kompleksitas waktu maupun ruang, namun memiliki nilai informatif yang sangat rendah [17]. Menurut [18], contoh dari daftar *stopword* Bahasa Indonesia meliputi "yang", "di", "untuk", "dan", "ke", "dari", dan sebagainya.

3.2.5. Stemming

Stemming merupakan tahapan untuk memproses kalimat dari sebuah kata menjadi bentuk dasarnya.

Hasil *preprocessing* dapat dilihat pada Tabel 1 berikut ini.

Tabel 1. Hasil *preprocessing*

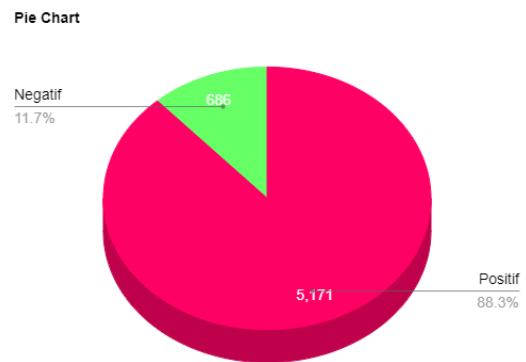
Tahapan	Hasil <i>preprocessing</i>
<i>Dataset</i>	negara maju di Asia nih punya visi yang jelas buat ngembangin metaverse!
<i>Cleaning</i>	negara maju di Asia nih punya visi jelas buat metaverse
<i>Case Folding</i>	negara maju di asia nih punya visi jelas buat metaverse
<i>Tokenizing</i>	['negara', 'maju', 'di', 'asia', 'nih', 'punya', 'visi', 'jelas', 'buat', 'metaverse']
<i>Filtering</i>	['negara', 'maju', 'asia', 'nih', 'punya', 'visi', 'jelas', 'buat', 'metaverse']
<i>Stemming</i>	negara maju asia nih punya visi jelas buat metaverse

3.2.6 Labeling

Dalam analisis sentimen, labeling mengacu pada tindakan memberi kategori sentimen tertentu pada teks, seperti "positif", dan "negatif" Proses

dapat dilakukan oleh manusia atau secara otomatis menggunakan mesin. Pada tahap labeling untuk mengenali sentimen positif atau negatif dalam analisis teks, mesin menggunakan berbagai teknik pemrosesan bahasa alami. Pertama, mesin melakukan pra-pemrosesan teks untuk membersihkan data dari karakter yang tidak penting dan memecah teks menjadi token-token atau kata-kata terpisah. Selanjutnya, mesin menggunakan metode analisis kata kunci untuk mencari kata-kata yang terkait dengan sentimen positif atau negatif, seperti "mengagumkan" untuk sentimen positif atau "menghancurkan" untuk sentimen negatif.

Dari hasil labeling tersebut terdapat 5171 *tweets* yang teridentifikasi sebagai sentimen positif dan 686 *tweets* sebagai sentimen negatif. Persentase dapat dilihat pada Gambar 7.



Gambar 7. Persentase nilai sentiment

3.3 Future Extraction (TF – IDF)

Menurut [19] *Feature Extraction* atau Ekstraksi fitur diartikan sebagai salah satu teknik yang sering digunakan untuk mengurangi kompleksitas data di mana data yang besar diubah menjadi fitur – fitur yang lebih sederhana namun tetap memiliki nilai informasi Penambahan fitur ini dapat secara signifikan meningkatkan kinerja model. Untuk menentukan rasio berapa kali suatu kata muncul dalam suatu dokumen dibandingkan dengan jumlah kata secara keseluruhan dalam dokumen, term frekuensi (TF) dan inversi dokumen frekuensi (IDF) digunakan. [20].

- Frekuensi Term (TF) :

$$TF(t,d) = \frac{\text{Jumlah kemunculan term } t \text{ dalam dokumen } d}{\text{Total jumlah term dalam dokumen } d} \quad (3)$$

TF mengukur seberapa sering suatu term muncul dalam suatu dokumen.

- Invers Dokumen Frekuensi (IDF)

$$IDF(t,D) = \log\left(\frac{\text{Total jumlah dokumen dalam } D}{\text{Jumlah dokumen dalam } D \text{ yang mengandung term } t} + 1\right) \quad (4)$$

IDF mengevaluasi seberapa umum atau jarang suatu term muncul di seluruh dokumen.

- TF-IDF

$$TF-IDF(t,d,D) = TF(t,d) \times IDF(t,D) \quad (5)$$

TF-IDF adalah hasil dari perkalian antara nilai TF dan nilai ID. Dengan menggabungkan informasi

tentang frekuensi term dalam dokumen (TF) dan distribusi term di seluruh himpunan dokumen (IDF), TF-IDF memberikan skor yang memberikan bobot yang lebih baik.

3.4 Handle Imbalance (SMOTE)

Synthetic Minority Over-sampling Technique (SMOTE), menangani ketidakseimbangan kelas. Metode ini mensintesis sampel baru dari kelas minoritas untuk menyeimbangkan kumpulan data. Data pelatihan termasuk contoh minoritas baru-baru ini, dan data tambahan digunakan untuk melatih pengklasifikasi [20]. Akurasi hasil model akan dipengaruhi oleh perbedaan jumlah setiap kelas.

3.5 Split Dataset

Dalam tahap *split dataset* pemisahan data menjadi data latih dan data uji sangat penting untuk memastikan model *machine learning* yang dibangun dapat bekerja dengan baik. Rasio umum yang digunakan adalah 80% untuk data latih dan 20% untuk data uji. Berbagai penelitian, termasuk studi terbaru pada tahun 2022. Penelitian ini menegaskan bahwa model yang dibangun dengan rasio 80%-20% cenderung memberikan hasil yang lebih baik, karena memberikan lebih banyak data latih untuk melatih model. Dengan lebih banyak data latih, model dapat mempelajari pola dan fitur data dengan lebih baik, yang pada akhirnya meningkatkan kemampuannya untuk memberikan prediksi yang lebih akurat pada data uji [21].

3.6 Model Classification (Klasifikasi Model)

Penerapan model klasifikasi melibatkan penggunaan algoritma *Support Vector Machine* (SVM) dan *Random Forest*. Proses klasifikasi ini menghasilkan tingkat akurasi yang nantinya akan dikomparasikan.

3.7 Model Evaluation (Evaluasi Model)

Evaluasi model untuk menentukan kinerja proses yang dilakukan dengan *confusion matrix*. *Confusion Matrix* merupakan suatu cara untuk menilai sejauh mana kinerja suatu model pembelajaran mesin dalam meklasifikasikan data. Matriks ini membentuk suatu table 2x2 yang memberikan informasi tentang kebenaran dan ketidakbenaran prediksi yang dihasilkan oleh model tersebut. Tabel 2 berisi keterangan untuk setiap elemen dalam *Confusion Matrix* :

Tabel 2. *Confusion Matrix*

<i>Actual</i>	<i>Prediction</i>	
	<i>Positive</i>	<i>Negative</i>
<i>Positive</i>	<i>True Positive</i>	<i>False Negative</i>
<i>Negative</i>	<i>False Positive</i>	<i>True Negative</i>

Keterangan :

- *True Positive (TP)* : Banyaknya data yang berhasil diidentifikasi dengan akurat sebagai positif oleh model.

- *False Positive (FP)* : Banyaknya data yang model salah mengklasifikasikan sebagai positif.
- *True Negative (TN)* : Banyaknya data yang berhasil diidentifikasi dengan akurat sebagai positif oleh model.
- *False Negative (FN)* : Banyaknya data yang model salah mengklasifikasikan sebagai negatif.

Akurasi sistem dapat dihitung menggunakan rumus persamaan 6 berikut.

$$accuracy = \frac{TP+TN}{TP+FP+TN+FN} \tag{6}$$

Nilai *precision*, *recall*, dan *F1-score* dapat dihitung dengan menggunakan rumus dalam persamaan (7), (8), (9). Nilai *precision* dan *recall* diperoleh dari perhitungan rata – rata *precision* dan *recall*. Nilai *precision* adalah rasio antara TP dan total data positif yang diprediksi, dan nilai *recall* adalah rasio antara TP dan total data aktual. Nilai *F1-score* yang lebih tinggi menunjukkan nilai *precision* dan *recall* yang lebih tinggi.

$$precision = \frac{TP}{TP+FP} \tag{7}$$

$$recall = \frac{TP}{TP+FN} \tag{8}$$

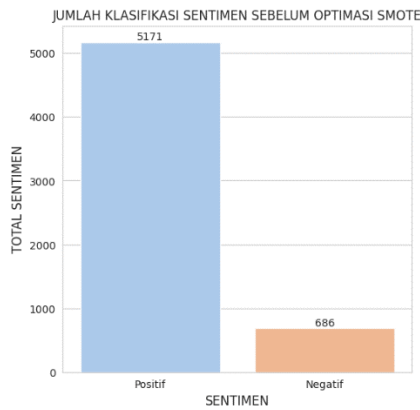
$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision+Recall} \tag{9}$$

Dari rumus yang telah di definisikan diatas, maka nilai dapat diketahui sebagai acuan perbandingan model yang telah dibuat.

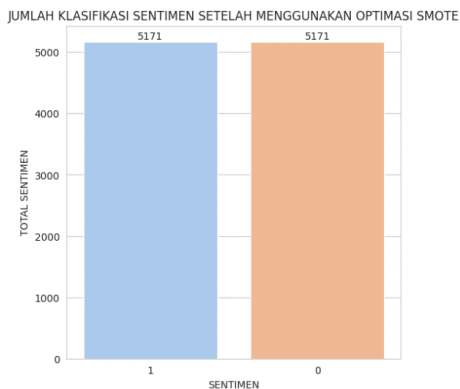
4 HASIL DAN PEMBAHASAN

4.1 Dataset

Dataset diperoleh dari pengguna pada media sosial X, melibatkan proses *crawling* dengan jumlah data mencapai 6348 *tweets*. Setelah itu, dilakukan tahap *preprocessing* teks pada dataset, diikuti dengan pelabelan atau pembuatan kelas. Dalam penelitian ini terdeteksi ketidakseimbangan kelas sehingga diterapkan metode *SMOTE* untuk menangani perbedaan jumlah antar kelas. Setelah melalui tahap *preprocessing*, jumlah dataset menjadi 5857 *tweets*. Perbandingan antara data sebelum penerapan *SMOTE* dan setelahnya dapat dilihat pada Gambar 8 dan Gambar 9.



Gambar 8. Jumlah klasifikasi sentimen sebelum menggunakan SMOTE.

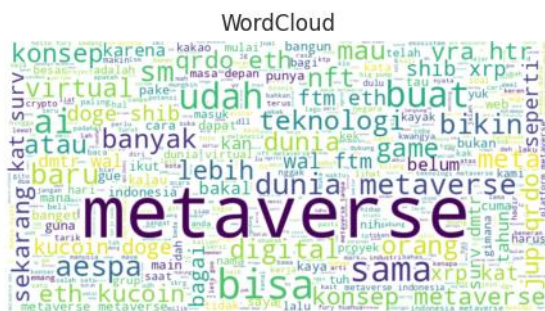


Gambar 9. Jumlah klasifikasi sentimen setelah menggunakan SMOTE.

Berdasarkan ilustrasi di atas dapat dipahami bahwa penerapan SMOTE meningkatkan data sintesis pada sentimen negatif sebesar 4.485 data.

4.2 Tahap Visualisasi Word Cloud

Tujuan penggunaan Word Cloud adalah untuk memvisualisasikan dataset secara grafis, sehingga mempermudah identifikasi data yang paling umum muncul dalam dokumen. Proses pembuatan Word Cloud menggunakan perpustakaan matplotlib dalam bahasa pemrograman Python. Hasil visualisasi Word Cloud dapat dilihat pada Gambar 10. Dari visualisasi pada Gambar 10, terlihat bahwa kata dengan ukuran yang lebih besar menunjukkan kata-kata yang sering muncul, seperti "metaverse".



Gambar 10. Hasil Visualisasi Word Cloud

4.1 Tahap Pengujian

Setelah melewati serangkaian tahap, data kemudian siap untuk dianalisis. Dalam penelitian ini komparasi dilakukan dengan dua metode, yaitu *Support Vector Machine* dan *Random Forest*. 80% dari data digunakan sebagai data *training* sementara sisanya 20% digunakan sebagai data *testing*. Dalam tahap pengujian yang akan dikomparasikan yaitu nilai *accuracy*, *F1-score*, *precision*, *recall*, dan *Confusion Matrix*.

4.3.1 Nilai Accuracy, Recall, Precision, F1-Score.

Hasil perbandingan nilai *accuracy Support Vector Machine* dan *Random Forest* dapat dilihat pada Tabel 3.

Metode	Accuracy score	
	With SMOTE	Without SMOTE
SVM	0.90	0.90
Random Forest	0.91	0.91

Tabel 3. Hasil Nilai Accuracy.

Tabel 2 menunjukkan bahwa pada metode Support Vector Machine (SVM), SMOTE tidak berpengaruh signifikan terhadap akurasi yang tetap 90% baik dengan atau tanpa SMOTE. Hal ini juga terjadi pada metode *Random Forest* dengan nilai *accuracy* sebesar 91%. Oleh karena itu, dari sisi *accuracy Random Forest* dengan penerapan SMOTE dapat dianggap sebagai model yang lebih unggul dalam mengatasi ketidakseimbangan kelas pada dataset sentimen terkait metaverse ini.

Dalam penelitian ini selain menggunakan nilai *accuracy* juga diterapkan *F1-score*, *precision*, dan *recall* sebagai parameter perbandingan. Hasilnya dapat dilihat pada Tabel 4 dan Tabel 5.

- Algoritma *Support Vector Machine*

Report	With SMOTE		Without SMOTE	
	Positif	Negatif	Positif	Negatif
Recall	0.95	0.52	0.96	0.46
precision	0.94	0.60	0.93	0.64
F1 score	0.94	0.56	0.95	0.53

Tabel 4. Hasil algoritma SVM

Hasil uji model *Support Vector Machine* menunjukkan bahwa dengan penerapan SMOTE, terdapat peningkatan *recall* untuk sentimen positif, yang mengindikasikan kemampuan model untuk lebih baik dalam mengenali kasus sentimen positif. Namun perlu dicatat bahwa peningkatan ini disertai dengan penurunan *precision*. Disisi lain, *recall* untuk sentimen negatif tetap relatif stabil sedangkan *precision* mengalami penurunan.

- Algoritma *Random Forest*

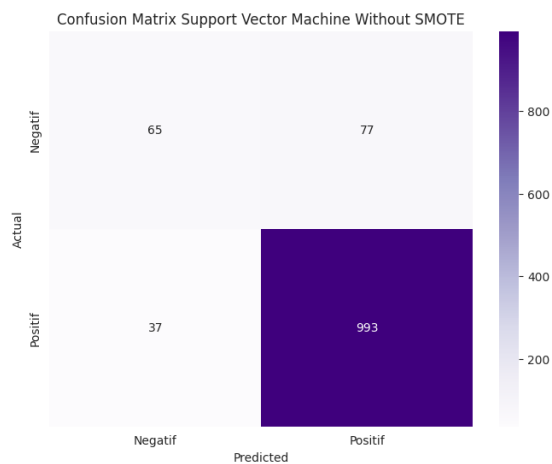
Report	With SMOTE		Without SMOTE	
	Positif	Negatif	Positif	Negatif
recall	0.97	0.42	0.99	0.29
precision	0.92	0.68	0.91	0.80
F1 score	0.95	0.50	0.95	0.42

Tabel 5. Hasil algoritma *Random Forest*

Hasil uji model *Random Forest* menunjukkan bahwa terjadi penurunan pada *recall* untuk sentimen positif namun terjadi peningkatan pada sentiment negatif. Dari segi *precision*, model dengan SMOTE menunjukkan presisi yang lebih tinggi untuk sentimen negatif. Secara keseluruhan, hasil *F1 score* mencerminkan adanya *trade-off* antara *recall* dan *precision* dengan penerapan SMOTE, menunjukkan peningkatan keseimbangan antara keduanya terutama untuk sentimen positif, namun dengan peningkatan risiko *false negatives*.

4.3.2 Hasil *Confusion Matrix*.

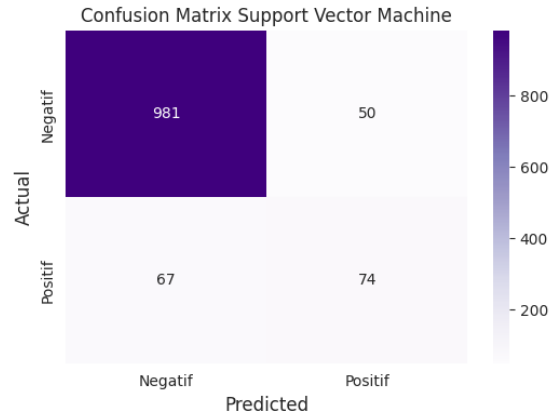
Selain itu penelitian ini juga melakukan komparasi *confusion matrix* pada kedua algoritma ini, kita dapat memahami secara mendalam sejauh mana kinerja model dalam mengklasifikasikan sentiment untuk mendapatkan gambaran komprehensif mengenai seberapa baik model ini dapat mengidentifikasi sentimen positif dan negatif.



Gambar 11. Hasil *Confusion Matrix* Algoritma *Support Vector Machine* sebelum optimasi SMOTE.

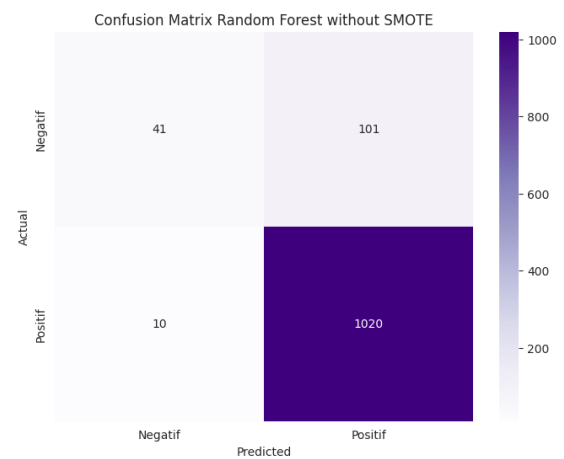
Hasil *Confusion matrix* sebelum menggunakan SMOTE pada algoritma SVM menunjukkan kecenderungan model kurang akurat dalam mengenali sentimen positif terkait metaverse. Dengan 65 *True Positives* dan 77 *False Negatives*, model lebih sukses mengidentifikasi sentimen negatif (993 *True Negatives*). Hasil ini dapat menjadi perhatian, terutama jika pengenalan

sentimen positif berkaitan dengan metaverse penting untuk analisis yang lebih mendalam terkait tren atau adopsi positif di dalamnya. Peningkatan kemampuan model dalam mengenali sentimen positif mungkin diperlukan untuk hasil analisis yang lebih akurat dan informatif.



Gambar 12. Hasil *Confusion Matrix* Algoritma *Support Vector Machine* setelah optimasi SMOTE.

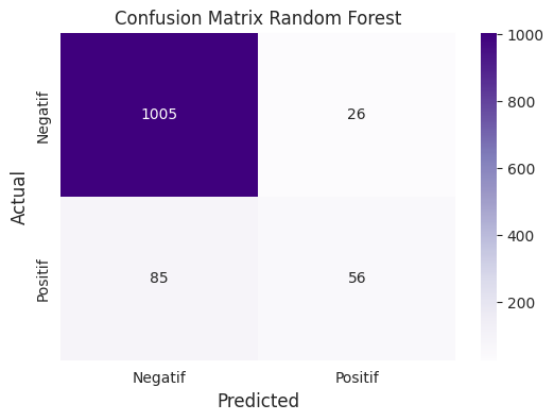
Setelah menggunakan SMOTE pada algoritma SVM, terjadi peningkatan yang signifikan dalam kemampuan model mengenali sentimen positif terkait metaverse, terlihat dari kenaikan *True Positives* (TP) menjadi 981 dan penurunan *False Negatives* (FN) menjadi 50. Namun, ada peningkatan *False Positives* (FP) yang juga signifikan (67), yang dapat memperlihatkan adanya kesalahan dalam mengidentifikasi sentimen positif.



Gambar 13. *Confusion Matrix* Algoritma *Random Forest* sebelum optimasi SMOTE

Analisis menunjukkan bahwa model *Random Forest* sebelum SMOTE memiliki kecenderungan lebih baik dalam mengenali sentimen negatif (1020 *True Negatives*), tetapi mengalami kesulitan dalam mengidentifikasi sentimen positif, ditunjukkan oleh jumlah *False Negatives* yang tinggi (101). Dampaknya terhadap analisis sentimen terkait

metaverse adalah model ini mungkin kurang responsif terhadap sentimen positif, yang dapat mengakibatkan pengabaian terhadap pandangan positif atau tren adopsi yang terjadi dalam konteks metaverse. Perbaikan dalam mengenali sentimen positif dapat meningkatkan keakuratan analisis terkait dengan aspek positif di dalamnya.



Gambar 14. Confusion Matrix Algoritma Random Forest setelah optimasi SMOTE

Penerapan SMOTE pada Random Forest memberikan peningkatan yang signifikan dalam kemampuan model mengenali sentimen positif, terlihat dari peningkatan jumlah True Positives (1005) dan penurunan False Negatives (26). Meskipun demikian, model masih mengalami kesulitan dalam mengenali sentimen negatif, yang tercermin dari jumlah False Positives (85) dan jumlah True Negatives yang relatif rendah (56).

Hasil pembahasan menunjukkan bahwa eksperimen ini berhasil membandingkan kinerja algoritma SVM dan Random Forest dalam mengklasifikasikan sentimen terhadap metaverse. Penerapan SMOTE meningkatkan keseimbangan dataset dan memberikan keunggulan pada Random Forest dengan akurasi 91% dibandingkan dengan 90% pada SVM. Meskipun Random Forest menghadapi trade-off antara recall dan precision, hasilnya menunjukkan keseimbangan yang lebih baik dalam mengklasifikasikan sentimen positif. Meski demikian, keduanya masih menghadapi kesulitan dalam mengenali sentimen negatif.

5. KESIMPULAN DAN SARAN

Secara keseluruhan, penelitian ini memberikan pemahaman tentang komparasi kinerja algoritma Support Vector Machine (SVM) dan Random Forest dalam analisis sentimen metaverse. Dalam eksperimen ini, Random Forest dengan penerapan metode SMOTE telah terbukti lebih unggul dengan mencapai akurasi 91%, sedangkan SVM mencapai 90%. Peningkatan keseimbangan dataset, khususnya dalam mengenali sentimen positif, adalah dampak

positif dari penerapan SMOTE. Trade-off antara recall dan precision terjadi pada kedua algoritma, namun Random Forest menunjukkan keseimbangan yang lebih baik. Hasil eksperimen ini berkontribusi terhadap metode penanganan ketidakseimbangan kelas, penyesuaian parameter, dan penambahan fitur untuk meningkatkan kemampuan model, terutama dalam mengidentifikasi sentimen negatif. Penelitian selanjutnya dapat dikembangkan dengan mengeksplorasi metode penanganan ketidakseimbangan kelas lainnya dan melibatkan pendekatan berbasis deep learning untuk memberikan akurasi dan hasil klasifikasi yang lebih baik.

DAFTAR PUSTAKA

- [1] N. Thakur, "MonkeyPox2022Tweets: A Large-Scale Twitter Dataset on the 2022 Monkeypox Outbreak, Findings from Analysis of Tweets, and Open Research Questions," *Infect. Dis. Rep.*, vol. 14, no. 6, pp. 855–883, 2022, doi: 10.3390/idr14060087.
- [2] J. López Díez, "Metaverse: Year One. Mark Zuckerberg's video keynote on Meta (October 2021) in the context of previous and prospective studies on metaverses," *Pensar la Publicidad. Rev. Int. Investig. Public.*, vol. 15, no. 2, pp. 299–303, 2021.
- [3] A. Al-adaileh, M. Al-kfairy, and M. Tubishat, "Pr ep rin t n ot er r Pr ep rin t n ot er r ed," 2023.
- [4] S. Sumayah, F. Sembiring, and W. Jatmiko, "Analysis of Sentiment of Indonesian Community on Metaverse Using Support Vector Machine Algorithm," *J. Tek. Inform.*, vol. 4, no. 1, pp. 143–150, 2023, doi: 10.52436/1.jutif.2023.4.1.417.
- [5] D. Darwis, E. S. Pratiwi, and A. F. O. Pasaribu, "Penerapan Algoritma Svm Untuk Analisis Sentimen Pada Data Twitter Komisi Pemberantasan Korupsi Republik Indonesia," *Edu tic - Sci. J. Informatics Educ.*, vol. 7, no. 1, pp. 1–11, 2020, doi: 10.21107/edutic.v7i1.8779.
- [6] A. R. Isnain, A. I. Sakti, D. Alita, and N. S. Marga, "Sentimen Analisis Publik Terhadap Kebijakan Lockdown Pemerintah Jakarta Menggunakan Algoritma Svm," *J. Data Min. dan Sist. Inf.*, vol. 2, no. 1, p. 31, 2021, doi: 10.33365/jdmsi.v2i1.1021.
- [7] R. Tineges, A. Triayudi, and I. D. Sholihati, "Analisis Sentimen Terhadap Layanan Indihome Berdasarkan Twitter Dengan Metode Klasifikasi Support Vector Machine (SVM)," *J. Media Inform. Budidarma*, vol. 4, no. 3, p. 650, 2020, doi: 10.30865/mib.v4i3.2181.
- [8] S. Khairunnisa, A. Adiwijaya, and S. Al Faraby, "Pengaruh Text Preprocessing terhadap Analisis Sentimen Komentar

- Masyarakat pada Media Sosial Twitter (Studi Kasus Pandemi COVID-19)," *J. Media Inform. Budidarma*, vol. 5, no. 2, p. 406, 2021, doi: 10.30865/mib.v5i2.2835.
- [9] F. F. Rachman and S. Pramana, "Analisis Sentimen Pro dan Kontra Masyarakat Indonesia tentang Vaksin COVID-19 pada Media Sosial Twitter," *Heal. Inf. Manag. J.*, vol. 8, no. 2, pp. 100–109, 2020, [Online]. Available: <https://inohim.esaunggul.ac.id/index.php/INO/article/view/223/175>
- [10] G. Gezici and B. Yanıkoğlu, "Sentiment Analysis in Turkish," no. February, pp. 255–271, 2018, doi: 10.1007/978-3-319-90165-7_12.
- [11] K. Jahanbin, M. Jokar, and V. Rahmanian, "Using twitter and web news mining to predict the monkeypox outbreak," *Asian Pacific Journal of Tropical Medicine*, vol. 15, no. 5. Wolters Kluwer Medknow Publications, pp. 236–238, May 01, 2022. doi: 10.4103/1995-7645.346083.
- [12] A. M. Mostafa, M. Aljasir, M. Alruily, and A. Alsayat, "applied sciences Innovative Forward Fusion Feature Selection Algorithm for Sentiment Analysis Using Supervised Classification," 2023.
- [13] J. Lasama, A. P. E. P, A. Prasetiadi, F. Teknologi, I. Teknologi, and T. Purwokerto, "Prediksi Tsunami Pada Gempa Menggunakan Random Forest Classifier," 2019.
- [14] P. Mega, N. Dharmapatni, N. Luh, and P. Merawati, "Jurnal Bumigora Information Technology (BITe) Penerapan Algoritma Support Vector Machine Dalam Sentimen Analisis Terkait Kenaikan Tarif BPJS Kesehatan Jurnal Bumigora Information Technology (BITe) Jurnal Bumigora Information Technology (BITe) Jurnal," vol. 2, no. 2, pp. 105–112, 2020, doi: 10.30812/bite.v2i2.904.
- [15] U. B. Jaya, P. Studi, T. Informatika, F. I. Komputer, and U. B. Jaya, "Perbandingan Algoritma Naïve Bayes dan Support Vector Machine(SVM)," vol. 3, no. 2, pp. 178–194, 2019.
- [16] K. M. Hana, "Multi-label Classification of Indonesian Hate Speech on Twitter Using Support Vector Machines," no. August 2020, 2021, doi: 10.1109/ICoDSA50139.2020.9212992.
- [17] A. W. Pradana and M. Hayaty, "The effect of stemming and removal of stopwords on the accuracy of sentiment analysis on indonesian-language texts," vol. 4, no. 3, 2019.
- [18] "A Study of Feature Extraction techniques for Sentiment Analysis," pp. 1–12.
- [19] Mega, P., Dharmapatni, N., Luh, N., & Merawati, P. (2020). Penerapan Algoritma Support Vector Machine dalam Sentimen Analisis Terkait Kenaikan Tarif BPJS Kesehatan. *Jurnal Bumigora Information Technology*, 2(2), 105–112. <https://doi.org/10.30812/bite.v2i2.904>
- [20] A. Y. Kuntoro, T. Asra, and E. Bayu, "Gojek and Grab User Sentiment Analysis on Google Play Using Naive Bayes Algorithm And Support Vector Machine Based Smote Technique Gojek and Grab User Sentiment Analysis on Google Play Using Naive Bayes Algorithm And Support Vector Machine Based Smote Tec", doi: 10.1088/1742-6596/1641/1/012102.
- [21] K. S. Reddy, "A comparative study of different ratios of training and testing data for machine learning algorithms," in *International Journal of Computer Science and Information Security*, vol. 20, no. 1, pp. 1-7, 2022.