

# LEXICON BASED ANALISIS DAN RANDOM FOREST TERHADAP ISU POLITIK DINASTI INDONESIA PADA APLIKASI X

Humuntal Rumapea<sup>1</sup>, Krisna Diva<sup>2</sup>, Harlen Gilbert Simanullang<sup>3</sup>

<sup>1,2,3</sup>Program Studi Teknik Informatika, Fakultas Ilmu Komputer,  
Universitas Methodist Indonesia

<sup>1</sup>hrumapea1608@gmail.com, <sup>2</sup>krisnadiva0412@gmail.com, <sup>3</sup>harlen.gilbert@gmail.com

## ABSTRACT

Dynastic politics in Indonesia remains a widely discussed issue, eliciting diverse public opinions ranging from support as a political right to criticism of democratic quality [1], with social media, particularly the X platform, serving as an important venue for public sentiment analysis. This study employs a combination of the Lexicon Based method using the InSet Lexicon and the Random Forest algorithm to analyze public sentiment on dynastic politics. The dataset consists of 1,593 tweets collected from August 1 to December 24, 2024, which underwent text preprocessing, labeling into three sentiment categories: positive, negative, and neutral, and word weighting using TF-IDF. The methodology includes splitting the data into training and testing sets with an 80:20 ratio, applying under sampling on the training data to balance class distribution, and training a Random Forest model with 100 decision trees and a maximum depth of 5 per tree, based on the entropy criterion. Evaluation results show that the model successfully classifies public sentiment with an accuracy of 89%, precision of 82%, recall of 81%, and f1-score of 81%.

**Keywords:** Dynastic Politics, Lexicon Based, Random Forest, Sentiment Analysis.

## I. PENDAHULUAN

Indonesia sebagai negara demokrasi telah menyelenggarakan pemilu sebanyak 13 kali hingga 2024. Pemilu menjadi mekanisme penting dalam mewujudkan prinsip kedaulatan rakyat yang memberikan kesempatan sama kepada setiap individu untuk berpartisipasi dalam proses politik. Namun dalam praktiknya, terdapat fenomena politik dinasti dimana anggota keluarga atau kerabat dekat pejabat mencalonkan diri atau menduduki jabatan politik tertentu. Fenomena ini memunculkan beragam pandangan di masyarakat, sebagian menganggapnya sebagai hak politik yang sah, namun sebagian lain memandangnya dapat berimplikasi pada terciptanya pemerintahan yang kurang berkualitas [1].

Platform media sosial X telah menjadi arena diskusi publik yang penting untuk berbagai isu politik, termasuk politik dinasti. Isu ini kembali menjadi sorotan seiring munculnya kasus-kasus anggota keluarga pejabat yang mencalonkan diri dalam pemilihan umum, memicu perdebatan publik yang intens mengenai dampak politik dinasti terhadap kualitas demokrasi Indonesia. Untuk memahami sentimen publik terhadap isu ini secara efektif, diperlukan pendekatan analisis sentimen yang tepat.

Analisis sentimen adalah suatu metode untuk menganalisis dan mengkaji pendapat, opini, atau komentar mengenai topik tertentu [2]. Mengingat data dari aplikasi X tidak memiliki label sentimen, dibutuhkan metode yang dapat melakukan klasifikasi sentimen secara otomatis. Metode *Lexicon Based* yang memanfaatkan kamus untuk menilai kata-kata penting dalam dokumen menjadi solusi efektif untuk menangani data tidak berlabel.

Penelitian sebelumnya [3] menunjukkan bahwa penggunaan *InSet Lexicon* untuk analisis sentimen *tweet* berbahasa Indonesia dapat mencapai akurasi hingga 79.2%.

Setelah data diberi label menggunakan *InSet Lexicon*, diperlukan model yang dapat mendeteksi sentimen pada data baru secara otomatis. Namun, klasifikasi sentimen sering menghadapi permasalahan ketidakseimbangan kelas yang dapat menyebabkan model *Machine Learning* bias dalam memprediksi kelas mayoritas. Studi terdahulu [4] menjelaskan bahwa teknik *undersampling* dapat meningkatkan performa klasifikasi secara efektif pada *dataset* yang tidak seimbang. Untuk menangani permasalahan tersebut, penelitian ini menggunakan metode *undersampling* untuk menyeimbangkan distribusi data [5].

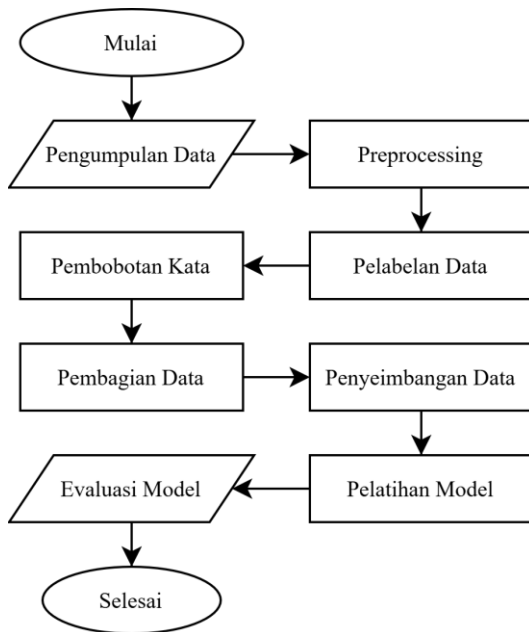
Algoritma *Random Forest* dipilih sebagai metode pemodelan karena merupakan *Ensemble Learning* yang terdiri dari kumpulan *Decision Tree* dengan voting mayoritas, memiliki ketahanan terhadap data *outlier* dan efisiensi dalam mengelola penyimpangan data [6]. Hasil penelitian [7] menunjukkan *Random Forest* dapat mencapai akurasi hingga 98% dalam analisis sentimen, sementara studi lainnya [8] mencapai akurasi 84% dengan pengaturan parameter yang optimal.

Berdasarkan permasalahan tersebut, penelitian ini bertujuan menganalisis sentimen politik dinasti di aplikasi X menggunakan kombinasi metode *Lexicon Based* dan *Random Forest* dengan teknik *undersampling*. Penelitian ini diharapkan dapat memberikan pemahaman yang komprehensif tentang persepsi masyarakat terhadap politik dinasti sekaligus berkontribusi pada pengembangan metodologi analisis sentimen data media sosial berbahasa Indonesia.



## II. METODOLOGI

Penelitian ini menganalisis sentimen masyarakat terhadap isu politik dinasti di Indonesia pada platform X menggunakan kombinasi metode *Lexicon Based* dan algoritma *Random Forest*. Metodologi penelitian terdiri dari delapan tahapan utama yang meliputi pengumpulan data, *preprocessing*, pelabelan data, pembobotan kata, pembagian data, penyeimbangan data, pelatihan model, dan evaluasi model. Pendekatan *Lexicon Based* dengan *InSet Lexicon* digunakan untuk memberikan label sentimen secara otomatis, sedangkan *Random Forest* diterapkan sebagai algoritma klasifikasi untuk membangun model prediksi sentimen. Keseluruhan tahapan penelitian ditunjukkan pada Gambar 1.



Gambar 1. Metodologi Penelitian

### Pengumpulan Data

Data yang digunakan bersumber dari platform media sosial X dan diperoleh menggunakan bahasa pemrograman *Python* dengan tools *Tweet-Harvest*. Kata kunci penelusuran yang digunakan adalah "politik dinasti" pada periode 1 Agustus hingga 24 Desember 2024. Pemilihan periode ini didasarkan pada intensitas diskusi politik dinasti yang meningkat menjelang dan setelah pemilihan umum.

### Preprocessing

Tahap *preprocessing* dilakukan untuk membersihkan dan menyiapkan data *tweet* sebelum dianalisis. Proses ini meliputi *cleansing* untuk menghilangkan karakter tidak diperlukan seperti URL dan tanda baca, *case folding* untuk mengubah huruf menjadi kecil, *tokenizing* untuk memecah kalimat menjadi kata tunggal, *stopword removal* untuk menghapus kata umum tanpa makna penting, dan *stemming* untuk mengubah kata berimbuhan menjadi kata dasar [9].

### Pelabelan Data

Tahap pelabelan data dilakukan untuk mengklasifikasikan sentimen *tweet* menggunakan kamus *InSet Lexicon* dengan menghitung skor setiap kata. *Lexicon Based* adalah suatu teknik seleksi kata penting dalam dokumen yang dilakukan berdasarkan kamus/leksikon yang sudah ada [10]. Kamus Indonesia *Sentiment Lexicon* terdiri dari 3.609 kata positif dan 6.609 kata negatif, dengan bobot polaritas berkisar antara -5 hingga +5 [11]. Penentuan label sentimen menggunakan aturan sebagai berikut:

$$Label \begin{cases} \text{Positif if Skor} > 0 \\ \text{Netral if Skor} = 0 \\ \text{Negatif if Skor} < 0 \end{cases} \quad (1)$$

Keterangan:

*Label* = merupakan klasifikasi sentimen dari sebuah kalimat, yang ditentukan berdasarkan nilai dari dimana jika lebih besar dari 0 positif, jika bernilai 0 netral, dan jika lebih kecil dari 0 negatif.

### Pembobotan Kata

Pembobotan kata merupakan tahapan untuk menentukan nilai bobot setiap kata dalam *tweet*. Pada penelitian ini tahap pembobotan kata dilakukan dengan menggunakan metode *Term Frequency-Inverse Document Frequency* (TF-IDF) yang merupakan sebuah metode transformasi kata menjadi nilai numerik dalam bentuk vektor untuk menghitung bobot kata pada dokumen atau korpus, dimana nilai bobot tersebut berfungsi untuk menentukan seberapa penting kata [12]. Rumus TF-IDF dapat dilihat pada persamaan berikut:

$$TF - IDF(t, d) = TF(t, d) \times IDF(t) \quad (2)$$

Keterangan:

$TF(t, d)$  = Frekuensi kemunculan *term* t dalam dokumen d

$f(t, d)$  = Banyaknya *term* t dalam dokumen

$IDF(t)$  = Frekuensi kemunculan *term* t dalam keseluruhan dokumen

$N$  = Total keseluruhan dokumen yang ada

$df(t)$  = Banyaknya dokumen yang memuat *term* t

$TF - IDF(t, d)$  = Hasil perkalian antara nilai  $TF(t, d)$  dengan  $IDF(t)$

Hasil TF-IDF dinormalisasi menggunakan normalisasi L2 untuk meningkatkan performa model *Machine Learning* dan memastikan konsistensi skala vektor [13]. Rumus normalisasi L2 yang digunakan adalah sebagai berikut:

$$TF - IDF(t, d)_{norm} = \frac{TF - IDF(t, d)}{\sqrt{\sum_{t \in d} TF - IDF(t, d)^2}} \quad (3)$$

Keterangan:

$TF - IDF(t, d)$  = Bobot *term* t dalam dokumen d sebelum normalisasi.

$TF - IDF(t, d)_{norm}$  = Bobot *term* t dalam dokumen d setelah normalisasi.

**Pembagian Data**

Setelah tahap pembobotan selesai, *dataset* dibagi ke dalam dua bagian yaitu data latih dan data uji. Proses pembagian dilakukan dengan komposisi 80% data digunakan sebagai data latih dan 20% sisanya sebagai data uji.

**Penyeimbangan Data**

Tahap selanjutnya adalah penyeimbangan pada data latih menggunakan teknik *undersampling*. Hal ini dilakukan karena terdapat ketidakseimbangan distribusi antar kelas sentimen yang dapat menyebabkan model *Machine Learning* cenderung bias terhadap kelas mayoritas dan kurang baik dalam memprediksi kelas minoritas. *undersampling* bekerja dengan cara menghapus sampel dari kelas mayoritas secara acak hingga jumlahnya setara dengan kelas minoritas [14].

**Pelatihan Model**

Pada tahap ini dilakukan pembangunan model menggunakan algoritma *Random Forest* yang merupakan metode *Ensemble Learning* yang menggabungkan hasil dari beberapa model [15]. Model yang akan dibentuk terdiri dari 100 *Decision Tree* dengan kedalaman maksimal 5 level dan dilatih menggunakan data latih yang telah diseimbangkan melalui proses *undersampling*. Tahapan pembentukan *Random Forest* adalah sebagai berikut [16]:

1. Membentuk *bootstrapped dataset* dengan mengambil sampel data secara acak sebanyak jumlah data yang diperlukan menggunakan teknik *replacement* pada data latih.
2. Menyeleksi fitur secara acak tanpa *replacement* pada *bootstrapped dataset* untuk menghitung nilai *Entropy* dan *Information Gain*. Perhitungan kedua nilai tersebut dapat dilakukan menggunakan rumus berikut:

$$Entropy(S) = -\sum_{i=1}^n p_i * \log_2 p_i \tag{4}$$

$$Gain(S, A) = Entropy(S) - \sum_{i=1}^n \frac{|S_i|}{|S|} * Entropy(S_i) \tag{5}$$

Keterangan:

*Entropy(S)* = Tingkat ketidakpastian dalam *dataset S*

*n* = Jumlah total kelas dalam *dataset*

*p<sub>i</sub>* = Proporsi data pada kelas *i*

*Gain(S, A)* = Pengurangan *entropy* setelah *dataset S* dibagi oleh atribut *A*

*n* = Jumlah *subset* yang dihasilkan oleh atribut *A*

*|S<sub>i</sub>|* = Jumlah data dalam *subset S<sub>i</sub>*

*|S|* = Jumlah total data dalam *dataset S*

3. Memakai *threshold* dan fitur dengan *Information Gain* tertinggi untuk dijadikan *rule* pada *node* di setiap level *Decision Tree*.
4. Mengulangi tahap 1-3 hingga *n Decision Tree* terbentuk sesuai parameter yang ditentukan.
5. Melakukan proses voting terhadap data uji oleh seluruh *n Decision Tree* yang telah dilatih untuk

menentukan hasil prediksi akhir.

**Evaluasi Model**

Setelah proses pelatihan model selesai dilakukan, evaluasi performa model kemudian dijalankan menggunakan data uji dengan *Confusion Matrix*. *Confusion Matrix* merupakan suatu matriks yang menampilkan jumlah data pengujian yang berhasil diklasifikasikan dengan tepat dan tidak tepat [17]. Ilustrasi tabel *Confusion Matrix* dapat dilihat pada Tabel 1.

**Tabel 1.** *Confusion Matrix*

Kelas Sebenarnya	Kelas Prediksi		
	Negatif	Netral	Positif
Negatif	TNeg	FNegNet	FNegPos
Netral	FNetNeg	Tnet	FNegPos
Positif	FPosNeg	FPosNet	TPos

Dari tabel di atas maka akan dilakukan evaluasi performa model dengan menghitung beberapa metrik evaluasi sebagai berikut:

1. *Accuracy*: Rasio dari prediksi yang benar baik positif maupun negatif terhadap keseluruhan data yang dapat dihitung menggunakan rumus berikut:

$$Accuracy = \frac{TPos + TNeg + Tnet}{Total\ Sample} \tag{6}$$

2. *Precision*: Rasio prediksi benar positif terhadap keseluruhan hasil prediksi positif yang dapat dihitung menggunakan rumus berikut:

$$Precision_c = \frac{TP_c}{TP_c + \sum_{j \neq c} FP_{jc}} \tag{7}$$

3. *Recall*: Rasio dari prediksi benar positif terhadap keseluruhan data yang benar positif yang dapat dihitung menggunakan rumus berikut:

$$Recall_c = \frac{TP_c}{TP_c + \sum_{j \neq c} FN_{jc}} \tag{8}$$

4. *F1-Score*: Rata-rata dari *precision* dan *recall* yang dapat dihitung menggunakan rumus berikut:

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

**III. HASIL DAN PEMBAHASAN**

**Data Yang Dikumpulkan**

Data yang berhasil dikumpulkan sebanyak 1.593 *tweet* dalam bahasa Indonesia yang membahas isu politik dinasti pada periode 1 Agustus hingga 24 Desember 2024. Data *tweet* yang dikumpulkan dapat dilihat pada Tabel 2.



**Tabel 2.** Data *Tweet* Politik Dinasti

No	Tweet
1	@PKSejahtera @AnisbyarwatiPKS Ayo dukung dinasti Jokowi
2	@NasDem Kongres dukung politik dinasti?
3	Waspada Masalah utamanya bukan dua tokoh ini Hikmah Kasus Putusan MK 60 70 itu kita bisa bersatu padu seluruh elemen masyarakat utk menentang Politik Dinasti Mulyono melaju di pilkada yg mau membatalkan Keputusan MK 60 70 dan merubah Keputusan MK dgn RUU Pilkada yg baru.. <a href="https://t.co/HNK8qI5iWo">https://t.co/HNK8qI5iWo</a>
4	@DaddyMinusSugar @DokterTifa @jokowi @UGMYogyakarta Banyak yang bilang walapun mereka tidak pro Jokowi mereka enggan percaya dengan dokter ini. Saya pun tak suka politik dinasti tapi saya tak mau mengotori jejak digital saya dengan turut menyebarkan kebohongan. Ada hisabnya loh menanti kita kelak.
...	...
1.593	@kompascom @msaid_didu Dan akan selamanya keluarga besar @jokowi tersandera dengan apa yang sudah dilakukan oleh @kaesangp ini jika tidak segera ada klarifikasi. Dan suatu saat kejadian ini akan menjadi senjata ampuh bagi lawan politik untuk mendiskreditkan dinasti Jokowi.

**Hasil Preprocessing**

*Preprocessing* dilakukan pada 1.593 data *tweet* melalui lima tahapan pembersihan data. Contoh hasil *preprocessing* dapat dilihat di Tabel 3.

**Tabel 3.** Hasil *Preprocessing*

Sebelum	Sesudah
@kompascom @msaid_didu Dan akan selamanya keluarga besar @jokowi tersandera dengan apa yang sudah dilakukan oleh @kaesangp ini jika tidak segera ada klarifikasi. Dan suatu saat kejadian ini akan menjadi senjata ampuh bagi lawan politik untuk mendiskreditkan dinasti Jokowi.	['akan', 'lama', 'keluarga', 'besar', 'sandra', 'apa', 'sudah', 'laku', 'tidak', 'segera', 'klarifikasi', 'suatu', 'jadi', 'ini', 'akan', 'jadi', 'senjata', 'ampuh', 'lawan', 'politik', 'diskredit', 'dinasti', 'jokowi']

**Hasil Pelabelan Data**

Pelabelan data dilakukan dengan menjumlahkan polaritas kata yang cocok di *InSet Lexicon*, lalu klasifikasinya berdasarkan Persamaan 1. Hasil distribusi label sentimen dapat diamati pada Tabel 4.

**Tabel 4.** Hasil Pelabelan Data

No	Tweet	Skor	Label
1	['ayo', 'dukung', 'dinasti', 'jokowi']	5	Positif
2	['kongres', 'politik', 'dinasti', 'dukung']	0	Netral

3	['waspada', 'utama', 'bukan', 'tokoh', 'hikmah', 'kasus', 'putus', 'mk', 'kita', 'satu', 'padu', 'seluruh', 'elemen', 'masyarakat', 'utk', 'tentang', 'politik', 'dinasti', 'mulyono', 'laju', 'pilkada', 'yg', 'mau', 'batal', 'putus', 'mk', 'rubah', 'putus', 'mk', 'dgn', 'ruu', 'pilkada', 'yg', 'baru']	-26	Negatif
4	['waspada', 'utama', 'bukan', 'tokoh', 'hikmah', 'kasus', 'putus', 'mk', 'kita', 'satu', 'padu', 'seluruh', 'elemen', 'masyarakat', 'utk', 'tentang', 'politik', 'dinasti', 'mulyono', 'laju', 'pilkada', 'yg', 'mau', 'batal', 'putus', 'mk', 'rubah', 'putus', 'mk', 'dgn', 'ruu', 'pilkada', 'yg', 'baru']	-35	Negatif
...	...	...	...
1.593	['akan', 'lama', 'keluarga', 'besar', 'sandra', 'apa', 'sudah', 'laku', 'tidak', 'segera', 'klarifikasi', 'suatu', 'jadi', 'ini', 'akan', 'jadi', 'senjata', 'ampuh', 'lawan', 'politik', 'diskredit', 'dinasti', 'jokowi']	-36	Negatif

Berdasarkan hasil pelabelan data pada Tabel 4, distribusi tiap label sentimen dan persentasenya dapat diamati pada Tabel 5.

**Tabel 5.** Distribusi Label Sentimen

Label	Jumlah	Presentase
Negatif	1.079	67,73%
Netral	221	18,39%
Positif	293	13,87%

**Hasil Pembobotan Kata**

Pembobotan data dilakukan menggunakan TF-IDF untuk mengkonversi kata menjadi representasi numerik berdasarkan frekuensi kemunculan dan keunikannya. Hasil pembobotan dapat diamati pada Tabel 6.

**Tabel 6.** Distribusi Label Sentimen

No	Term	TF			IDF	TF-IDF		
		D1	D2	D1		D1	D2	D1
1	abrik	0	0	0	3,741	0	0	0
2	adil	0	0	0	3,741	0	0	0
3	aja	0	0	1	3,048	0	0	0,419
...	...	...	...	...	...	...	...	...
5.937	warga	0	0	0	3,741	0	0	0



**Hasil Pembagian Data**

*Dataset* dibagi ke dalam dua bagian yaitu 80% digunakan untuk data latih dan 20% untuk data uji. Visualisasi hasil pembagian data dapat diamati pada Gambar 2.

<b>Distribusi Data Latih:</b>	
Label negatif:	863 (67.74%)
Label netral:	177 (13.89%)
Label positif:	234 (18.37%)
<b>Distribusi Data Uji:</b>	
Label negatif:	216 (67.71%)
Label netral:	44 (13.79%)
Label positif:	59 (18.50%)

**Gambar 2.** Hasil Pembagian Data

Berdasarkan Gambar 2, pembagian *dataset* dengan rasio 80:20 menghasilkan distribusi kelas sentimen yang proporsional, di mana label negatif masih mendominasi, sehingga data latih dan data uji tetap konsisten dan representatif untuk pelatihan serta evaluasi model.

**Hasil Penyeimbangan Data**

Berdasarkan teknik *undersampling*, diperoleh hasil penyeimbangan distribusi data latih untuk setiap kelas sentimen seperti yang ditunjukkan pada Gambar 3.

<b>Distribusi Data Latih Setelah Undersampling:</b>	
Label negatif:	177
Label netral:	177
Label positif:	177
<b>Persentase Label:</b>	
Label negatif:	33.33%
Label netral:	33.33%
Label positif:	33.33%

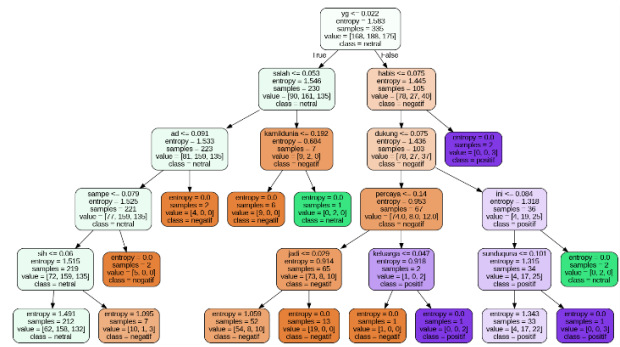
**Gambar 3.** Hasil Penyeimbangan Data

Gambar 3 menunjukkan hasil penyeimbangan data latih menggunakan teknik *undersampling*, sehingga jumlah data pada setiap kelas sentimen menjadi sama dan distribusi kelas lebih seimbang.

**Pelatihan Model**

Berdasarkan proses pelatihan *Random Forest* yang telah dijelaskan, maka diperoleh 100 *Decision Tree* dengan kedalaman maksimum 5. Model dilatih menggunakan library *scikit-learn* yang menggunakan kriteria *entropy* untuk pemilihan fitur terbaik.

Salah satu dari 100 *Decision Tree* yang terbentuk dalam model *Random Forest* dapat dilihat pada Gambar 4.



**Gambar 4.** *Decision Tree* 1

Untuk visualisasi penuh dari keseluruhan *tree* dalam model *Random Forest*, dapat diakses melalui URL: [https://drive.google.com/drive/folders/IU-OnyJ3bFulAMmkAR913FaZ\\_ovGznaOC](https://drive.google.com/drive/folders/IU-OnyJ3bFulAMmkAR913FaZ_ovGznaOC)

**Evaluasi Model**

Proses evaluasi model dilakukan dengan menguji performa model *Random Forest* menggunakan data uji. Hasil evaluasi menggunakan *confusion matrix* yang dapat diamati pada Gambar 5.

True Labels	negatif	209	5	2
	netral	2	37	5
	positif	8	14	37
		negatif	netral	positif
		Predicted Labels		

**Gambar 5.** *Confusion Matrix*

Berdasarkan *confusion matrix* pada Gambar 5, nilai *precision*, *recall*, *f1-score*, dan *accuracy* dapat diamati pada *classification report* di Gambar 6.

	precision	recall	f1-score
negatif	0.95	0.97	0.96
netral	0.66	0.84	0.74
positif	0.84	0.63	0.72
accuracy			0.89
macro avg	0.82	0.81	0.81

**Gambar 6.** *Classification Report*

Gambar 6 menunjukkan *classification report* hasil evaluasi model *Random Forest*. Kelas negatif



mencapai *precision* 95%, *recall* 97%, dan *f1-score* 96%. Kelas netral memperoleh *precision* 66%, *recall* 84%, dan *f1-score* 74%, sedangkan kelas positif mencapai *precision* 84%, *recall* 63%, dan *f1-score* 72%. Model menghasilkan *accuracy* keseluruhan sebesar 89%.

#### IV. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan bahwa analisis sentimen terhadap isu politik dinasti di Indonesia menggunakan *InSet Lexicon* pada 1.593 *tweet* menunjukkan distribusi sentimen yang bervariasi, dengan kecenderungan ke arah negatif. Model *Random Forest* yang diimplementasikan untuk klasifikasi sentimen menunjukkan performa yang baik dengan *accuracy* sebesar 89%, *precision* 82%, *recall* dan *F1-score* masing-masing 81%.

Penerapan teknik *undersampling* membantu meningkatkan kemampuan model dalam mengenali kelas minoritas sehingga klasifikasi yang dihasilkan menjadi lebih proporsional. Hasil ini menunjukkan bahwa kombinasi metode *Lexicon Based* dan *Random Forest* dapat diterapkan secara efektif pada analisis sentimen data media sosial berbahasa Indonesia dan dapat menjadi acuan bagi penelitian selanjutnya.

#### V. REFERENSI

- [1] A. Dedi, "Politik Dinasti dalam Perspektif Demokrasi," *Jurnal MODERAT*, vol. 8, no. 1, 2022.
- [2] E. Darwisah Harahap and R. Kurniawan, "Analisis Sentimen Komentar Terhadap Kebijakan Pemerintah Mengenai Tabungan Perumahan Rakyat (TAPERA) Pada Aplikasi X Menggunakan Metode Naïve Bayes," 2024.
- [3] D. Musfiroh, U. Khaira, P. Eko, P. Utomo, and T. Suratno, "Analisis Sentimen terhadap Perkuliahan Daring di Indonesia dari Twitter Dataset Menggunakan InSet Lexicon," vol. 1, pp. 24–33, 2021.
- [4] E. Saputro and D. Rosiyadi, "Bianglala Informatika Penerapan Metode Random Over-Under Sampling Pada Algoritma Klasifikasi Penentuan Penyakit Diabetes," vol. 10, no. 1, p. 2022, 2022, [Online]. Available: <https://archive.ics.uci.edu/ml/machine-learning->
- [5] D. R. Manalu, M. C. L. Tobing, and M. Yohanna, "Analisis Sentimen Twitter Terhadap Wacana Penundaan Pemilu Dengan Metode Support Vector Machine," vol. 6, no. 2, pp. 149–156, 2024.
- [6] M. F. Y. Herjanto and C. Carudin, "Analisis Sentimen Ulasan Pengguna Aplikasi Sirekap pada Play Store Menggunakan Algoritma Random Forest Classifier," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 12, no. 2, Apr. 2024, doi: 10.23960/jitet.v12i2.4192.
- [7] S. Nanda, D. Mualfah, and D. A. Fitri, "Analisis Sentimen Kepuasan Pengguna Terhadap Layanan Streaming Mola Menggunakan Algoritma Random Forest," 2022.
- [8] F. A. Larasati, D. E. Ratnawati, and B. T. Hanggara, "Analisis Sentimen Ulasan Aplikasi Dana dengan Metode Random Forest," 2022. [Online]. Available: <http://j-ptiik.ub.ac.id>
- [9] S. Agung, "Implementasi Text Mining untuk Analisis Review pada Aplikasi Crowdfunding LX dan ST Menggunakan Metode Sentiment Analysis Satria Agung Satria Agung," vol. 2, no. 1, pp. 124–130, 2024, doi: 10.35870/ljit.v2i1.2245.
- [10] P. Amira Sumitro *et al.*, "Analisis Sentimen Terhadap Vaksin Covid-19 di Indonesia pada Twitter Menggunakan Metode Lexicon Based," 2021, [Online]. Available: <https://developer.twitter.com>
- [11] F. Koto and G. Y. Rahmangtyas, "Inset lexicon: Evaluation of a word list for Indonesian sentiment analysis in microblogs," in *Proceedings of the 2017 International Conference on Asian Language Processing, IALP 2017*, Institute of Electrical and Electronics Engineers Inc., Jul. 2017, pp. 391–394. doi: 10.1109/IALP.2017.8300625.
- [12] E. Rifut Nur Mustaqim, U. Pagalay, and C. Crysdiyan, "Prediksi Tingkat Kepercayaan Masyarakat terhadap Pilpres 2024 Menggunakan TF-IDF dan BoW Menggunakan Metode SVM," 2024.
- [13] R. F. Ramadhan, S. Hadi Wijoyo, and M. C. Saputra, "Penerapan Metode K-Means Clustering pada Ulasan Perumahan PT XYZ di Google Maps untuk Formulasi Strategi Bisnis dengan Analisis SWOT," 2023. [Online]. Available: <http://j-ptiik.ub.ac.id>
- [14] H. Hairani, K. E. Saputro, and S. Fadli, "K-means-SMOTE for handling class imbalance in the classification of diabetes with C4.5, SVM, and naive Bayes," *Jurnal Teknologi dan Sistem Komputer*, vol. 8, no. 2, pp. 89–93, Apr. 2020, doi: 10.14710/jtsiskom.8.2.2020.89-93.
- [15] Ary Prandika Siregar, Dwi Priyadi Purba, Jojo Putri Pasaribu, and Khairul Reza Bakara, "Implementasi Algoritma Random Forest Dalam Klasifikasi Diagnosis Penyakit Stroke," *Jurnal Penelitian Rumpun Ilmu Teknik*, vol. 2, no. 4, pp. 155–164, Nov. 2023, doi: 10.55606/juprit.v2i4.3039.
- [16] M. E. Bastian, B. Rahayudi, and D. E. Ratnawati, "Prediksi Trend Harga Saham Jangka Pendek berdasarkan Fitur Technical Analysis dengan menggunakan Algoritma Random Forest," 2021. [Online]. Available: <http://j-ptiik.ub.ac.id>
- [17] D. Normawati and S. A. Prayogi, "Implementasi Naïve Bayes Classifier Dan Confusion Matrix Pada Analisis Sentimen Berbasis Teks Pada Twitter," 2021.

