

Prediksi *Retweet* Berdasarkan *User-Based* dan *Content-Based* Menggunakan Metode *Ensemble Stacking*

1st Muhammad Rizqi Akbar
Fakultas Informatika
Universitas Telkom
Bandung, Indonesia
rizqiakbar@students.telkomuniver
sity.ac.id

2nd Jondri
Fakultas Informatika
Universitas Telkom
Bandung, Indonesia
jondri@telkomuniversity.ac.id

3rd Indwiarti
Fakultas Informatika
Universitas Telkom
Bandung, Indonesia
indwiarti@telkomuniversity.ac.id

Abstrak-Twitter merupakan salah satu social media yang sangat populer dan mudah digunakan untuk mendapatkan informasi secara cepat. Fitur *Retweet* merupakan salah satu alasan mengapa penyebaran informasi tersebut dapat tersebar dengan cepat. *Retweet* terjadi jika seorang *follower* men-*tweet* ulang *tweet* dari *followee*-nya. Pada penelitian ini dilakukan pemodelan untuk prediksi *retweet* berdasarkan *feature user-based* dan *content-based* dengan menggunakan metode *Ensemble Stacking* melalui proses *K-fold Cross Validation*. *Ensemble Stacking* ini dibentuk dengan 3 *base-learner* yaitu *Random Forest*, *Gradient Boosting*, dan *Support Vector Machine*(SVM). Sedangkan *meta-learner* yang digunakan adalah *Support Vector Machine*(SVM) Pemodelan ini menunjukkan hasil terbaik ketika sudah dilakukan *Imbalanced Class Handling* menggunakan Teknik *SMOTE* dan *K-fold Cross Validation* dengan $k=10$. Hasil *F1-score* menunjukkan 86.46%. Dengan hasil demikian, bisa disimpulkan bahwa pemodelan yang dibentuk mampu meningkatkan hasil prediksi dari *base-learner*nya.

Kata kunci-twitter, *retweet*, *ensemble stacking*, *k-fold cross validation*, *oversampling*

Abstract-Twitter is one of the most popular and easy-to-use social media to get information quickly. The Retweet feature is one of the reasons why information can spread quickly. A retweet occurs when followers retweet his followee tweet's. Therefore, most tweets spread from followees to their followers. In this research, modeling is done for retweet prediction based on user-based and content-based features using the Ensemble Stacking method through the K-fold Cross Validation process. In this case, This Stacking Ensemble is formed with 3 base-learners namely Random Forest, Gradient Boosting, and Support Vector Machine (SVM). While the meta-learner used is Support Vector Machine (SVM) This modeling shows the best results when it has been done Imbalanced Class Handling using the SMOTE Technique and K-fold Cross Validation with $k = 10$. F1-score results show 86.46%. With such results, it can be concluded that the modeling formed is able to improve the prediction results of the base-learner.

Keywords- twitter, retweet, ensemble stacking, k-fold cross validation, oversampling

I. PENDAHULUAN

A. Latar Belakang

Twitter merupakan sebuah sosial media yang sangat mudah dijadikan sarana untuk berbagi informasi. Pada Twitter, setiap pengguna dibolehkan untuk membuat profil Twitter, untuk menulis pesan, maupun untuk berbagi informasi dengan pengguna lain [1]. Melalui *tweet*, informasi bisa disebarkan ke publik secara real-time. Setiap *tweet* juga bisa disisipkan berupa foto, video, maupun tautan website. Selain itu, Twitter juga mempunyai fitur *Retweet* yang berfungsi untuk memposting ulang *tweet* pengguna lain ataupun *tweet* milik sendiri untuk dibagikan ke *followers* pengguna. Hal inilah yang membuat *Retweet* menjadi kunci mekanisme dari sebuah difusi informasi mengapa informasi bisa menyebar [2].

Dari banyaknya *tweet* yang dibagikan di Twitter, ada *tweet* yang mendapatkan *retweet* dan ada juga yang tidak mendapatkan *retweet*. Hal tersebut bisa menjadi potensi untuk dijadikan sebuah penelitian untuk keperluan penyelesaian masalah. Contohnya bisa di berbagai hal seperti pada pemasaran yang membutuhkan *engagement* yang besar, analisis bisnis, hingga membantu dalam memutuskan sebuah prediksi. Sehingga permasalahan pada bidang tersebut bisa terbantu [2].

Maka dari itu diperlukan sebuah pemodelan yang bisa digunakan untuk memprediksi apakah sebuah *tweet* mendapatkan *retweet* atau tidak, yang mana hasilnya berupa skor akurasi pada prediksi *retweet*. Prediksi *retweet* memiliki banyak fitur yang bisa digunakan untuk membangun model prediksi *retweet* tersebut. Fitur yang bisa digunakan antara lain fitur *User-based*, *Time-based*, dan *Content-based*. Dimana *User-based* berupa informasi terkait profil pengguna. Lalu, *Time-based* berupa fitur yang berkaitan dengan waktu sebuah *tweet* tersebut dipublikasikan. Hasil hipotesa yang disebutkan oleh Thi Bich Ngoc Hoang dan Josiane Mothe, pada “waktu senggang”, sebuah *tweet* lebih mudah mendapatkan *retweet*. Sedangkan *Content-based* berupa informasi yang berkaitan dengan isi pada sebuah *tweet* [3].

Pada penelitian yang sudah dilakukan oleh Ishita Daga, dkk. melakukan pencarian *accuracy* dari prediksi *Retweets* dan *Likes* pada Twitter dengan *Text*

Embedding TF-IDF dan Doc2Vec dengan menggunakan beberapa metode klasifikasi untuk pembandingnya. Akurasi pada prediksi *Retweets* dengan metode *Logistic Regression* menghasilkan akurasi 70.1%, dengan metode *SVM* menghasilkan akurasi 62.9%, dengan metode *Random Forest* menghasilkan akurasi 49.8%, dengan metode *Neural Network* menghasilkan akurasi 39.1%, dan menggunakan metode *Multinomial Naïve Bayes* 75.9%[4]. Sehingga dapat disimpulkan bahwa Text Embedding dan jenis metode klasifikasi sangat mempengaruhi akurasi. Sehingga itu diperlukan pemodelan dengan metode lain untuk mendapatkan akurasi lebih baik.

Pada penelitian ini, penulis berfokus pada pemodelan untuk pencarian akurasi prediksi *Retweets* dengan menggunakan fitur User-based dan Content-based dengan menggunakan Metode *Ensemble Stacking*. *Ensemble Stacking* itu sendiri merupakan gabungan dari beberapa metode klasifikasi Machine Learning. Pada *base-learner* ini menggunakan 3 buah metode klasifikasi, yaitu *Random Forest*, *Gradient Boost*, dan *Support Vector Machine*(SVM). Sedangkan pada *meta-learner* digunakan metode klasifikasi *Support Vector Machine*(SVM). Sehingga diharapkan dengan menggunakan penumpukan metode klasifikasi(*Stacking*) bisa mendapatkan akurasi yang lebih baik dibandingkan hanya menggunakan satu metode klasifikasi.

B. Topik dan Batasannya

Berdasarkan latar belakang yang sudah disebutkan, maka dapat ditentukan topik masalahnya adalah prediksi retweet menggunakan fitur User-based dan Content-based menggunakan metode *Ensemble Stacking* yang terdiri dari 3 *base-learner* dan 1 *meta-learner*. Sedangkan untuk Batasannya adalah data yang digunakan adalah data tweet yang diambil dengan kata kunci “kominfo ” dalam rentang waktu April hingga Juli 2022.

C. Tujuan

Penelitian ini mempunyai tujuan untuk membangun sistem yang dapat memprediksi *retweet* dengan fitur *user-based* dan *content-based* menggunakan metode *Ensemble Stacking*.

D. Organisasi Tulisan

Pada tulisan selanjutnya akan dijelaskan tentang studi terkait tentang penelitian dan hasil penelitian tersebut. Lalu, pada bagian selanjutnya menjelaskan tentang urutan sistem yang akan dibangun untuk melakukan penelitian. Pada bagian evaluasi menjelaskan tentang hasil dari penelitian dan analisis hasil penelitian dengan melakukan beberapa skenario yang berbeda. Pada bagian terakhir berisi kesimpulan dari penelitian dan saran untuk penelitian selanjutnya.

II. KAJIAN TEORI

Pada penelitian yang dilakukan oleh Thi Bich Ngoc Hoang dan Josiane Mothe, penelitian ini tidak hanya menjelaskan tentang tujuan penelitian dalam melakukan sebuah *tweet* akan mendapatkan *retweet* atau tidak, tapi dalam penulisannya juga dijelaskan bermacam-macam fitur yang bisa dijadikan referensi untuk pembuatan dataset yang akan digunakan untuk penelitian. Penelitian tersebut menggunakan metode klasifikasi *Random Forest* untuk melakukan prediksi. Hasilnya menunjukkan peningkatan sekitar 5% F-Measure dengan model yang peneliti buat [3].

Selanjutnya, pada penelitian yang dilakukan oleh Padmapani P. Tribhuvan, membahas tentang Arsitektur sistem untuk penambangan dan ringkasan berbasis fitur opini menggunakan *Stacking Ensemble*. Dari dataset yang berisi dari 4096 review laptop dan menggunakan *Ensemble Stacking* yang terdiri dari 3 *base-learner* berupa *Naïve Bayes*(NB), *K Nearest Neighbor*(KNN), dan *Support Vector Machine*(SVM). Hasilnya NB mendapatkan akurasi 60%, KNN mendapatkan akurasi 9.6%, SVM mendapatkan akurasi tertinggi yaitu 72%. Sementara itu, untuk *Meta-learner* menggunakan SVM. Hasil *Stacking* mendapatkan peningkatan akurasi menjadi 92%[5].

Sedangkan pada penelitian yang dilakukan oleh Yuli Sopiati, dkk. dengan mengusung metode modifikasi *Ensemble Stacking* dan dengan tujuan memprediksi *personality* pengguna twitter, hasilnya mendapatkan akurasi yang sangat tinggi senilai 99.62%. Akurasi tersebut memperoleh sedikit peningkatan dari hasil menggunakan metode *stacking* normal dengan nilai akurasi 99.24%. Dengan hasil tersebut, memungkinkan suatu *Ensemble Stacking* bisa menggunakan N-tingkatan untuk memperoleh akurasi yang lebih baik [6].

Pada penelitian yang dilakukan oleh Siddhartha Mootha, dkk. peneliti melakukan prediksi pembeli online menggunakan *Ensemble Stacking*. Pada model yang dibuat, peneliti menggunakan 20 jenis metode klasifikasi sebagai *base-learner*. Beberapa *base-learner* diantaranya mendapatkan akurasi yang baik, seperti metode *SVM*(88%), *Gradient Boosting*(90%), dan *Random Forest*(90%). Sedangkan hasil akhir *Ensemble Stacking* dengan menggunakan *meta-learner Artificial neural network* (ANN). Akurasi yang didapatkan mengalami peningkatan yang cukup signifikan dengan tingkat akurasinya adalah 94%. Peneliti menyarankan untuk mengidentifikasi fitur-fitur yang paling relevan dalam dataset untuk lebih meningkatkan sistem [7].

A. Twitter

Twitter adalah salah satu sosial media yang paling populer dan bisa digunakan secara luas oleh orang-orang di dunia. Twitter juga membolehkan pengguna untuk membuat akun profil, berkirim pesan, dan berbagi informasi dengan pengguna lain [1]. Pada Twitter, pengguna bisa

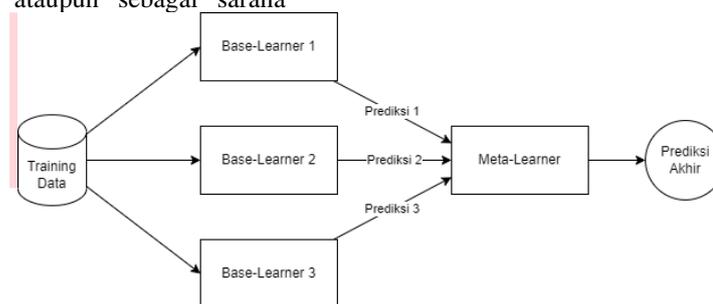
mengirimkan tweet dengan batasan maksimal 280 karakter. Selain itu, setiap tweet juga bisa disisipkan berupa gambar, video, maupun URL sebuah situs.

Difusi informasi pada Twitter bisa menyebar sangat cepat. Salah satunya dengan adanya Retweet. Secara Struktur, retweet setara dengan implementasi penerusan email pada surat elektronik dimana pengguna memposting tweet yang diposting oleh orang lain. Retweet membawa orang baru kedalam sebuah utas tertentu dan mengundang mereka untuk terlibat secara langsung. Hal tersebut bisa dipahami bahwa retweet merupakan bentuk difusi informasi ataupun sebagai sarana

untuk penyebaran informasi ke area yang lebih luas.

B. Ensemble Stacking

Metode Ensemble pada umumnya digunakan untuk perbaikan metode untuk klasifikasi. Metode ini adalah metode yang menggabungkan beberapa teknik pada Machine Learning menjadi satu model prediksi. Ada 3 metode untuk membuat ensemble, yaitu *bagging*, *boosting*, dan *stacking*. Pada penelitian kali ini akan menggunakan *Ensemble Stacking* [8].



GAMBAR 1.
ARSITEKTUR ENSEMBLE STACKING (SUMBER: TOWARDSAI.NET)

Ensemble Stacking adalah sebuah algoritma pembelajaran yang disusun dengan mengadopsi struktur 2 lapis. Lapis pertama disebutnya *base-learner* sedangkan lapis kedua disebut *meta-learner*. Ide utama dari algoritma ini adalah untuk melatih dataset dengan beberapa pembelajar utama terlebih dahulu (*base-learner*). Kemudian, hasil prediksi yang diperoleh oleh setiap pengklasifikasi *base-learner* digunakan sebagai input dari *meta-learner* untuk melakukan pelatihan lagi. Akhirnya, hasil pelatihan *meta-learner* adalah hasil prediksi akhir[9].

C. Penggunaan Fitur

Dalam penulisan ini diperlukan fitur untuk dijadikan sebuah dataset untuk dijadikan data train dan data tes untuk mendapatkan hasil prediksi *retweet*. Pada setiap *tweet* bisa merepresentasikan fitur *User-based*, *Time-based*, dan *Content-based*[3]. Namun, pada penulisan ini, Fitur yang digunakan adalah *User-based* dan *Content-based*:

1. User-based

Seseorang pengguna twitter yang banyak berinteraksi dengan pengguna lain, pasti suatu saat mendapat interaksi balasan yang sesuai dengan apa yang pengguna

tersebut lakukan. Hal itu menjadi cukup penting diperhatikan untuk tujuan penulisan ini. User-based ini sendiri meliputi:

- total_tweet* : total tweet yang diunggah oleh pengguna
- no_follower* : jumlah akun yang mengikuti pengguna
- no_following* : jumlah akun yang diikuti pengguna
- age_of_account* : umur akun sejak hari dibuat dalam satuan hari
- no_of_favorite* : jumlah tweet yang diberi *like* oleh pengguna
- avg_tweet_per_day* : jumlah rata-rata unggahan tweet per hari
- avg_fav_per_day* : jumlah rata-rata tweet yang diberi *like* per hari
- username_len* : panjang karakter username
- verified* : apakah akun memiliki status verified

2. Content-based

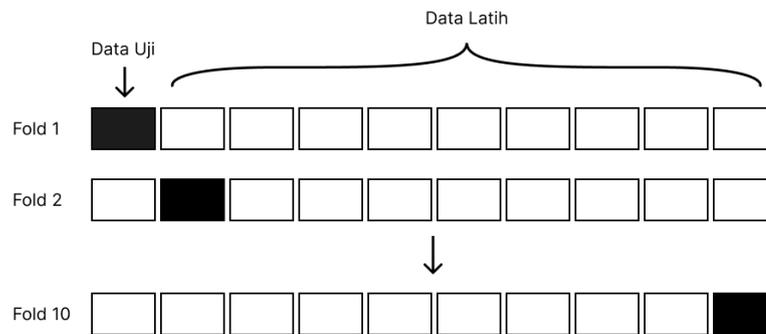
Fitur Content-based yang digunakan terdiri dari:

- a. *len_of_text* : panjang karakter dari sebuah tweet
- b. *opt_length* : Panjang karakter terdiri antara 70-100 karakter
- c. *contain_picture* : tweet mengandung gambar
- d. *contain_video* : tweet mengandung video
- e. *contain_upper* : tweet mengandung huruf kapital
- f. *contain_hashtag* : tweet mengandung hashtag
- g. *con_user_mentioned* : tweet menyebutkan username

- h. *retweeted* : mendapatkan retweet
- i. *sentiment* : tweet mengandung sentiment positif, netral, atau negatif

D. K-fold Cross Validation

Cross Validation adalah sebuah teknik melakukan validasi pada model untuk menilai bagaimana suatu hasil dari analisis akan menggeneralisasikan kumpulan data yang independen. Salah satu dari teknik *Cross Validation* adalah *k-fold cross validation* yang mana teknik ini akan memecah dataset menjadi sejumlah k dengan ukuran tiap lipatan(fold) yang sama besarnya [10].



GAMBAR 2. ILUSTRASI K-FOLD CROSS VALIDATION

Seperti pada GAMBAR 2, visualisasi tersebut merupakan contoh penggunaan *k-fold cross validation* dengan nilai $k=10$. Saat iterasi pertama(fold-1), subset 1 akan dijadikan data uji, sedangkan 9 subset lainnya dijadikan data latih. Lalu masuk ke iterasi kedua(fold-2), subset 2 bergantian menjadi data uji, sedangkan 9 subset lainnya dijadikan data uji. Iterasi tersebut dilakukan berulang kali sebanyak nilai k yang telah diatur.

Setiap *fold* akan mendapatkan nilai prediksi masing-masing, sehingga terdapat 10 hasil prediksi yang mana nantinya 10 hasil prediksi tersebut diambil nilai rata-ratanya, jadi didapatkan hasil prediksi akhir.

E. Performansi sistem

Pengukuran permormansi sangat penting guna untuk mengevaluasi kinerja algoritma yang telah dibangun. Pada performansi sistem ini, akan menggunakan Binary Classification Metrics. Dari macam-macam Binary Classification Metrics yang tersedia. Pada sistem yang akan dibangun pada penelitian ini adalah Confusion Metrix. Langkah-langkah ini didefinisikan oleh True Positive (TP), True Negative (TN), False Positive (FP) dan False Negative (FN) [11].

TABEL 1. CONFUSION MATRIX

	Actual Values	
	Positif	Negatif

Predicted Values	Positif	True Positive (TP)	False Positif (FP)
	Negatif	False Negative (FN)	True Negative(TN)

1. Presisi adalah perbandingan jumlah sampel yang diprediksi berada di kelas yang benar dengan jumlah sampel yang diprediksi oleh sistem klasifikasi

$$presisi = \frac{TP}{TP + FP} \quad (1)$$

$$F1 - score \quad (3)$$

$$= \frac{2 \times presisi \times recall}{presisi + recall}$$

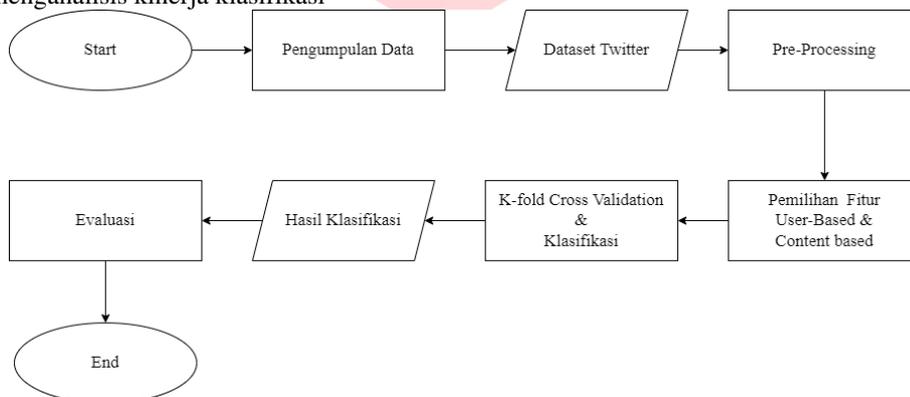
2. Recall adalah rasio jumlah sampel yang diprediksi benar dengan jumlah yang seharusnya diprediksi.

$$recall = \frac{TP}{TP + FN} \quad (2)$$

3. F1-Score adalah pengukuran untuk menganalisis kinerja klasifikasi

III. METODE

Sistem yang akan dibuat adalah Prediksi Retweet berdasarkan fitur User-based dan Content-based menggunakan metode Ensemble Stacking. Berikut adalah flowchart dari perancangan sistem yang akan dibangun.



GAMBAR 3. PERANCANGAN SISTEM

A. Pengumpulan Data

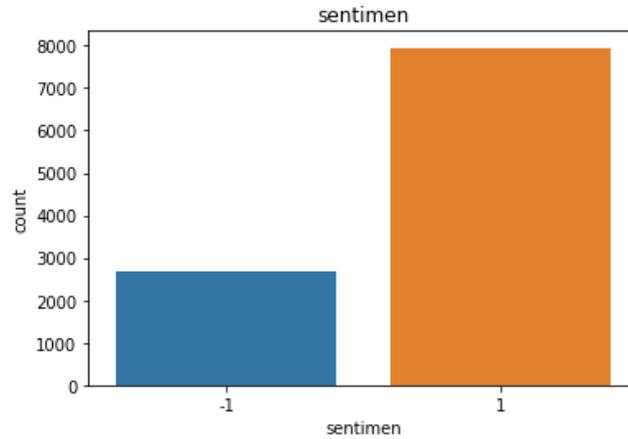
Pengumpulan data dilakukan dengan teknik *scraping* twitter menggunakan *snsrape*. Hasil dari *scraping* ini mendapatkan 20.000 tweets Bahasa Indonesia dengan kata kunci pencarian “kominfo” dengan rentang waktu bulan April - Juli 2022[12].

B. Pre-processing

1. *Text Cleaning*, digunakan untuk membersihkan isi *tweet* dari kata hashtag, tanda “@”, dan link pada *tweet*. Sehingga dari hasil *Text Cleaning* tersebut bisa digunakan untuk pencarian *sentiment* dari *tweet*.

2. *Language Translation*, pada proses ini dilakukan penerjemahan isi *tweet* yang sudah dilakukan *Text Cleaning* dari Bahasa Indonesia ke Bahasa Inggris.

3. *Polarity Detection*, proses ini berfungsi untuk mencari nilai *polarity* pada sebuah *tweet* yang sudah berbahasa Inggris dengan bantuan library *TextBlob* [13]. Alasan *Language Translation* tersebut agar mendapatkan nilai *polarity* yang lebih optimal. Selanjutnya nilai *polarity* tersebut dijadikan acuan untuk penentuan *tweet* tersebut merupakan memiliki sentimen positif, netral, atau negative.



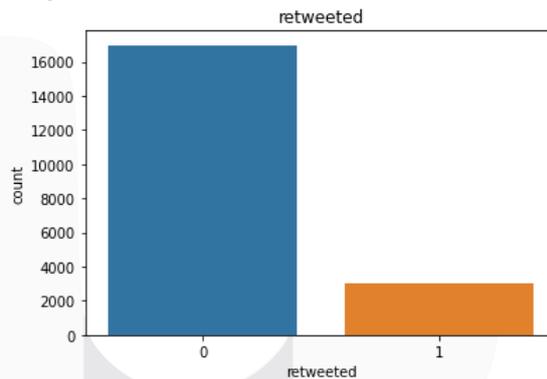
GAMBAR 4. VISUALISASI SENTIMEN TWEET

Hasil sentimen pada GAMBAR 4, angka -1 mewakili tweet dengan sentiment negative dan 1 mewakili tweet dengan sentiment positif.

4. *Text Extraction*, pada proses ini dilakukan ekstraksi teks dari tweet untuk mendeteksi adanya *hashtag*, huruf

kapital, *user mentioned*, dan Panjang tweet.

5. *Labeling*, tahap ini dilakukan untuk membuat fitur yang tidak didapatkan pada dataset. Seperti fitur pada "retweeted" untuk menentukan apakah pada *tweet* tersebut mendapatkan *retweet* atau tidak.



GAMBAR 5. DISTRIBUSI KELAS RETWEETED

Hasil visualisasi menampilkan 16979 *tweet* tidak mendapatkan *retweet* dan 3021 mendapatkan *retweet*. Data 0 merupakan data *tweet* yang tidak mendapatkan *retweet* dan data 1 adalah data *tweet* yang mendapatkan *retweet*.

6. *Data Normalization*, dengan menggunakan *min-max normalization* maka fitur yang masih bertipe data numerik akan dilakukan *scaling* menjadi dari 0-1. Sehingga skala

numerik pada fitur tidak besar dan mampu mempercepat pada proses klasifikasi.

C. Pemilihan Fitur

Pada tahap ini dilakukan penghapusan fitur-fitur yang kurang relevan untuk digunakan pada proses klasifikasi. Sehingga fitur akhir yang didapatkan sebagai berikut:

TABEL 2.
FITUR YANG DIPILIH

Fitur	Nama fitur	Tipe data
F1	total_tweet	Numerik
F2	no_follower	Numerik
F3	no_following	Numerik
F4	age_of_account	Numerik
F5	no_of_favorite	Numerik
F6	avg_tweet_per_day	Numerik
F7	avg_fav_per_day	Numerik
F8	username_len	Numerik
F9	verified	Boolean
F10	len_of_text	Numerik
F11	opt_length	Boolean
F12	contain_picture	Boolean
F13	contain_video	Boolean
F14	contain_upper	Boolean
F15	contain_hashtag	Boolean
F16	contain_user_mentioned	Boolean
F17	sen_negatif	Boolean
F18	sen_netral	Boolean
F19	sen_positif	Boolean
F20	retweeted	Boolean

Pada

TABEL 2 menampilkan 20 jenis fitur yang terdiri dari User-based dan Content-based yang mana digunakan untuk prediksi *retweet* pada penelitian ini.

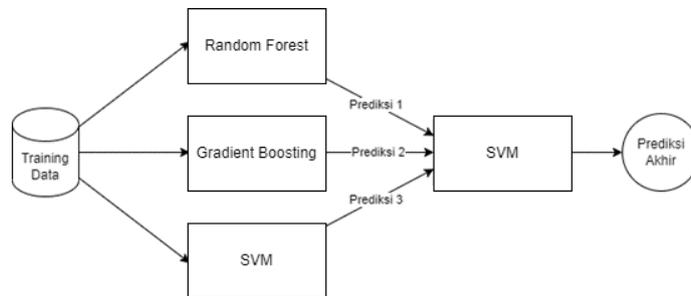
D. K-fold Cross Validation

Pada tahap ini digunakan K-fold Cross validation dengan nilai $k=3$, $k=5$, dan $k=10$ sehingga dari 20000 data akan dibagi dengan jumlah fold sesuai nilai k yang sudah ditentukan.

E. Klasifikasi Ensemble Stacking

Pemilihan metode-metode ini dipilih berdasarkan hasil dari penelitian oleh Siddartha Mootha, dkk. Pada penelitian yang dilakukan, metode tersebut

mendapatkan nilai prediksi yang tinggi. Sehingga memungkinkan metode yang dipilih bisa lebih maksimal hasilnya dibandingkan dengan metode yang digunakan lainnya[7].



GAMBAR 6. METODE ENSEMBLE STACKING YANG DIGUNAKAN

Pada GAMBAR 6. METODE ENSEMBLE STACKING YANG DIGUNAKAN, tingkat 1 Ensemble Stacking yang disusun, terdapat tiga metode klasifikasi untuk bagian Base-Learner ini, yaitu: *Random Forest*, *Gradient Boosting*, dan *Support Vector Machine(SVM)*. Pada tingkat 2, sebagai meta learner dipilih metode klasifikasi *Support Vector Machine(SVM)*. Meta Learner ini akan dilatih dengan hasil dari prediksi dari tiga metode klasifikasi pada *Base-Learner* sebelumnya.

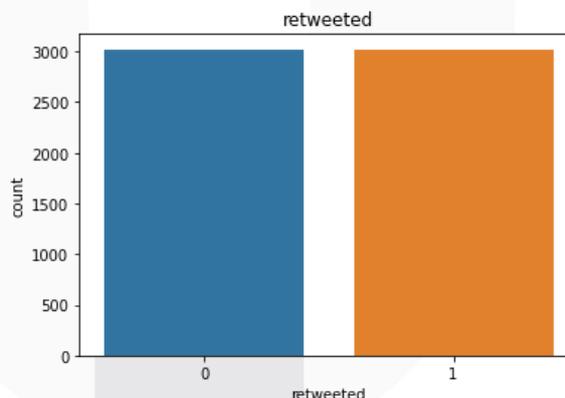
F. Skenario Pengujian

1. Skenario Pengujian 1

Pada pengujian ini, dataset yang digunakan adalah dataset yang masih dalam keadaan *imbalanced Class*.

2. Skenario Pengujian 2

Pada pengujian ini, dataset yang digunakan adalah dataset yang sudah dilakukan *Imbalanced Class Handling* menggunakan *RandomUnderSampler*.

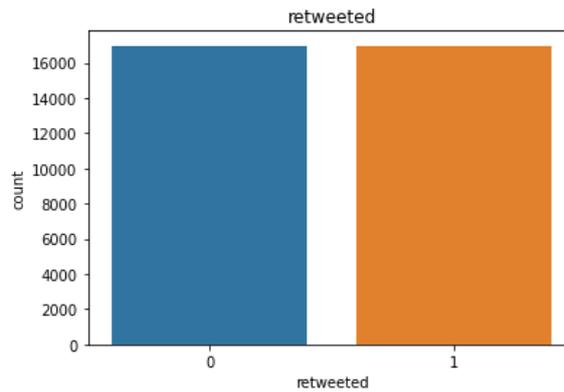


GAMBAR 7. SETELAH DILAKUKAN UNDERSAMPLING DENGAN *RANDOMUNDERSAMPLER*

Sehingga bisa didapatkan kelas yang seimbang antara kelas yang mendapatkan *retweet* dan kelas yang tidak mendapatkan *retweet* dengan total masing-masing kelas 3021 data tweet.

3. Skenario Pengujian 3

Dikarenakan distribusi pada kelas *retweeted* tidak seimbang maka diperlukan Resampling untuk menyamakan jumlah kelas pada *retweeted*.



GAMBAR 8. SETELAH DILAKUKAN OVERSAMPLING DENGAN SMOTE

Pada pengujian ini digunakan Oversampling menggunakan metode SMOTE(Synthetic Minority Oversampling Technique) sehingga kelas yang tidak mendapatkan retweet menjadi sejumlah 16979 tweet [14].

Skenario 1

Pada pengujian pertama, hasil yang didapatkan untuk melakukan prediksi *retweet* menggunakan metode Ensemble Stacking ini mendapatkan rata-rata tertinggi F1-Score pada saat K-fold Cross Validation diatur dengan nilai $k=5$. Hasil F1-Score tersebut menunjukkan nilai rata-ratanya sebesar 40.72%. Sedangkan nilai F1-Score terendah didapatkan pada saat $k=10$.

Pada performansi *Imbalanced class* ini dapat dikatakan bahwa Recall menjadi pengaruh utama mengapa F1-Score menjadi rendah juga. Hal ini dipengaruhi oleh *imbalanced class* yang mendapatkan jumlah False Negative yang relatif tinggi pada confusion matrix. sehingga membuat nilai Recall rendah.

IV. HASIL DAN PEMBAHASAN

Bagian ini berisi Hasil Pengujian dan Analisis Hasil Pengujian. Pengujian dan analisis yang dilakukan selaras dengan tujuan penulisan seperti yang dinyatakan dalam Pendahuluan sebelumnya.

A. Hasil Pengujian

TABEL 3. HASIL ENSEMBLE STACKING IMBALANCED CLASS

Jumlah fold	Presisi	Recall	F1-Score
k=3	86.38%	26.75%	40.65%
k=5	92.27%	26.18%	40.72%
k=10	91.23%	25.76%	39.72%

Pada TABEL 4. HASIL KLASIFIKASI IMBALANCED CLASS

TABEL 4, dapat dilihat perbandingan dari hasil klasifikasi dengan base-learner yang digunakan dan klasifikasi Ensemble Stacking. Hasil akurasi tertinggi didapatkan ketika hanya menggunakan Random Forest dengan F1-Score tertinggi pada $k=10$

yang nilainya 47.60%. Sedangkan hasil F1-Score pada tiap jumlah k-fold yang berbeda tetap belum bisa melampaui nilai akurasi F1-Score pada Random Forest.

TABEL 4. HASIL KLASIFIKASI IMBALANCED CLASS

Metode	F1-Score		
	k=3	k=5	k=10
Random Forest	46.21%	46.33%	47.60%
Gradient Boosting	43.26%	43.39%	43.30%
SVM	22.15%	22.95%	23.21%
Stacking (SVM Meta-learner)	40.65%	40.72%	39.72%

B. Skenario 2

Pada pengujian kedua, dimana dataset sudah ditangani dengan Undersampling menggunakan *RandomUnderSampler*, hasil akurasi pada pengujian mengalami peningkatan. F1-Score tertinggi menunjukkan pada 75.72% dengan nilai fold $k=10$.

Sedangkan untuk F1-score pada semua fold mengalami kenaikan yang sangat signifikan.

Pada performansi skenario 2 ini, nilai Recall mendapatkan peningkatan dikarenakan sudah

dilakukan Resampling, sehingga data yang sudah seimbang itu mampu mengurangi nilai False Negatif (prediksi tidak mendapatkan *retweet* padahal kenyataanya mendapat *retweet*). Maka dari itu nilai

Recall mengalami peningkatan dibanding skenario pertama.

TABEL 5.
HASIL ENSEMBLE STACKING DENGAN UNDERSAMPLING

Jumlah fold	Presisi	Recall	F1-Score
k=3	75.20%	72.33%	73.72%
k=5	74.46%	74.55%	74.46%
k=10	74.98%	76.69%	75.72%

Untuk hasil keseluruhan pada pengujian kedua ini, metode dengan Ensemble Stacking ini mendapatkan TABEL 6. HASIL KLASIFIKASI UNDERSAMPLING menampilkan bahwa semua pengujian tiap k-fold

F1-Score tertinggi pada k=10 dengan nilai 75.72%. mendapatkan peningkatan dibandingkan hanya menggunakan base-learnernya.

TABEL 6.
HASIL KLASIFIKASI UNDERSAMPLING

Metode	F1-Score		
	k=3	k=5	k=10
Random Forest	85.34%	85.99%	86.26%
Gradient Boosting	80.95%	81.15%	80.91%
SVM	73.58%	73.78%	73.66%
Stacking (SVM Meta-learner)	85.35%	86.22%	86.46%

C. Skenario 3

Pada pengujian ketiga, dilakukanlah Oversampling dengan SMOTE pada dataset. Sehingga jumlah dataset sekarang berjumlah 33.958. Sama seperti skenario kedua, dikarenakan

data sudah seimbang dan nilai False Negatif berkurang. Maka Recall pada Skenario 3 ini juga meningkat. Dari hasil penambahan dataset yang cukup signifikan dibandingkan dari skenario 2 ini, metode Ensemble Stacking ini mendapatkan nilai rata-rata F1-score tertinggi 86.46% pada k=10.

TABEL 7.
HASIL ENSEMBLE STACKING DENGAN OVERSAMPLING

Metode	F1-Score		
	k=3	k=5	k=10
Random Forest	69.87%	70.52%	71.53%
Gradient Boosting	72.91%	73.92%	75.05%
SVM	68.42%	69.08%	69.83%
Stacking (SVM Meta-learner)	73.72%	74.46%	75.72%

Hasil klasifikasi keseluruhan pada Oversampling ini, Ensemble Stacking mendapatkan peningkatan nilai

F1-Score dibandingkan dengan 2 skenario sebelumnya baik yang *imbalanced class* ataupun yang sudah ditangani *imbalanced class* dengan *Undersampling*.

TABEL 8.
HASIL KLASIFIKASI OVERSAMPLING

Jumlah fold	Presisi	Recall	F1-Score
k=3	84.14%	86.74%	85.35%
k=5	85.49%	87.20%	86.22%
k=10	85.90%	87.28%	86.46%

Metode ini mendapatkan nilai F1-Score rendah pada skenario 1 atau ketika belum dilakukan *Imbalanced Class Handling*. Pada keadaan ini, performansi

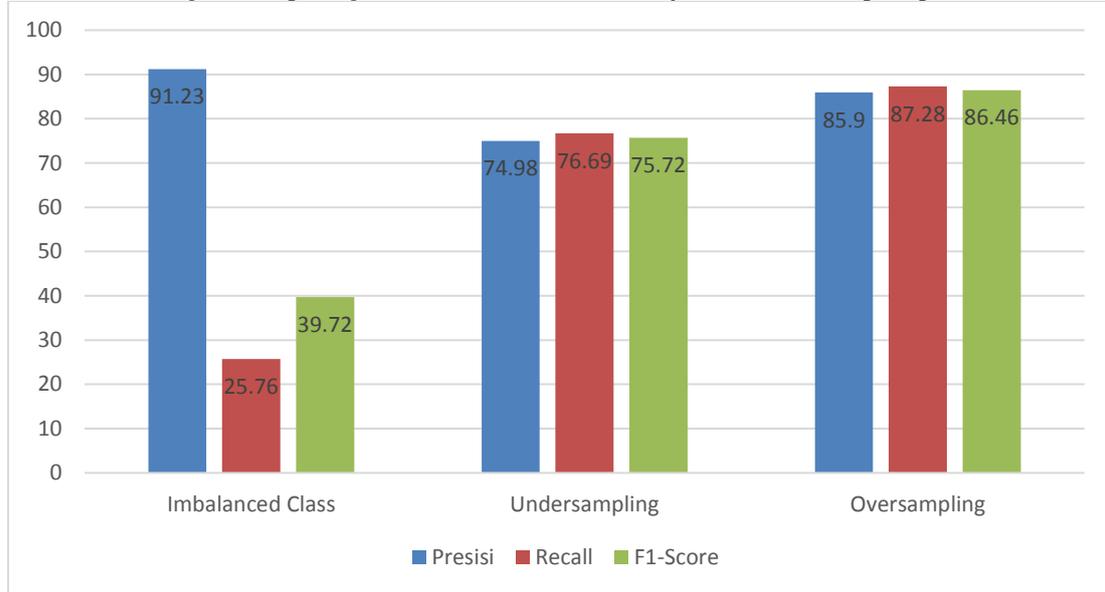
B. Analisis Hasil Pengujian

Dari hasil penelitian prediksi *retweet* menggunakan metode Ensemble Stacking yang terdiri dari 3 base-learner dan 1 meta-learner ini.

Recall mendapatkan nilai yang cukup rendah. Hal ini bermula jika data class *retweeted* tidak seimbang yang menyebabkan nilai False Negatif (prediksi tidak mendapatkan *retweet* padahal kenyataannya mendapat *retweet*) yang tinggi.

Sedangkan pada skenario kedua dan ketiga Recall mulai mengalami peningkatan setelah

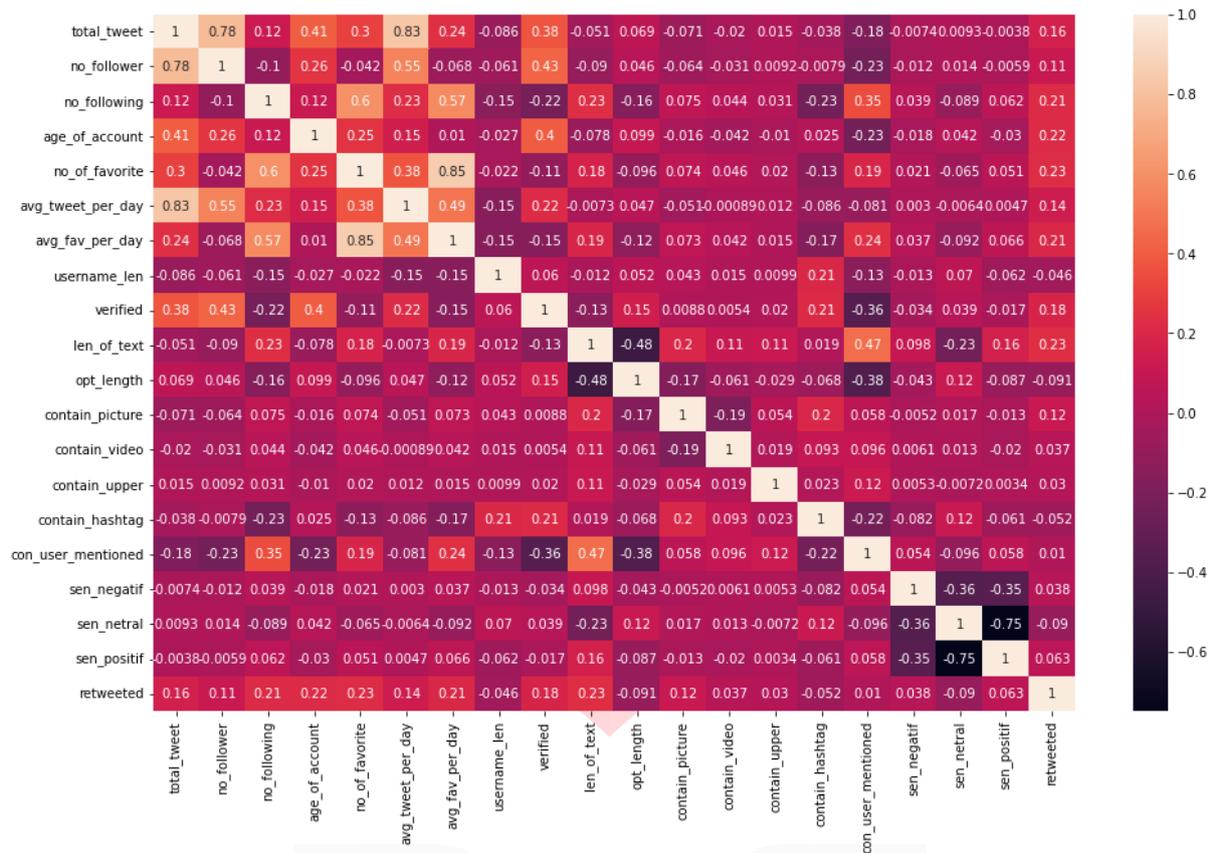
dilakukan *Resampling*. Maka dari itu nilai pada F1-Score juga meningkat. Namun pada skenario kedua dan ketiga mendapatkan nilai F1-Score berbeda. Nilai F1-Score tertinggi pada skenario kedua mendapatkan presentase 75.72% dengan k=10. Sedangkan ketika sudah dilakukan *Oversampling* dengan teknik SMOTE ditambah dengan penggunaan *K-fold Cross Validation* dengan nilai k=10, hasil F1-Score menunjukkan 86.46% seperti pada GAMBAR 9.



GAMBAR 9. PERBANDINGAN NILAI PERFORMANSI DENGAN K=10

Selain itu, perlu juga ditambahkan bahwa korelasi pada tiap feature yang berkorelasi dengan feature target kelas ataupun berkorelasi dengan

feature lainnya. Hal ini bisa digunakan untuk mengetahui feature mana yang berpengaruh pada sebuah tweet akan mendapatkan retweet atau tidak.



GAMBAR 10. KORELASI FEATURE

Setiap korelasi feature memiliki nilai yang berbeda-beda. Semakin tinggi nilai dari korelasi terhadap target class *Retweeted*, maka semakin mudah untuk memprediksi nilai *Retweeted*. Pada GAMBAR 10 menunjukkan beberapa korelasi dengan nilai yang tinggi. Feature yang memiliki nilai korelasi tinggi terhadap kelas *Retweeted* antara lain adalah “total_tweet”, “no_follower”, “no_following”, “age_of_account”, “no_of_favorite”, “avg_tweet_per_day”, “avg_fav_per_day”, “verified”, “len_of_tweet”, “contain_picture”, dan “sen_positif”. Feature tersebutlah yang mempengaruhi peningkatan nilai prediksi apakah suatu tweet akan mendapatkan retweet atau tidaknya.

V. KESIMPULAN

Pada penelitian yang sudah dilakukan, dapat disimpulkan bahwa pada prediksi retweet berdasarkan user-based dan content-based menggunakan metode Ensemble Stacking ini akan mendapatkan peningkatan performansi ketika sudah dilakukan resampling, dikarenakan dengan adanya resampling ini, Ensemble Stacking berhasil meningkatkan nilai F1-Score. Hasil terbaik didapatkan adalah pada skenario 3 dengan menerapkan Oversampling menggunakan SMOTE dengan hasil F1-Score 86.46% menggunakan *K-fold*

Cross Validation dengan *k=10*. Maka dari itu pentingnya penggunaan *Imbalanced Class Handling* untuk mengurangi nilai False Negatif pada *Confusion Matrix* agar nilai Recall dan F1-Score meningkat.

Untuk penelitian selanjutnya, dapat dikembangkan metode Ensemble Stacking ini dengan kombinasi metode-metode klasifikasi lainnya, bisa juga dengan menambahkan jumlah base-learner menjadi lebih banyak. Selain itu penulis juga menyarankan untuk melakukan eksplorasi metode Ensemble Stacking ini dengan memperbanyak jumlah tumpukan(Stack). Selain itu, perlu dilakukan *selection feature* mendapatkan feature terbaik untuk dilakukan klasifikasi.

REFERENSI

[1] S. N. Firdaus, C. Ding, and A. Sadeghian, “Retweet: A popular information diffusion mechanism – A survey paper,” *Online Soc. Networks Media*, vol. 6, pp. 26–40, Jun. 2018, doi: 10.1016/j.osnem.2018.04.001.

[2] S. N. Firdaus, C. Ding, and A. Sadeghian, “Retweet prediction considering user’s difference as an author and retweeter,” in *2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, Aug. 2016, pp. 852–859. doi: 10.1109/ASONAM.2016.7752337.

[3] T. B. N. Hoang and J. Mothe, “Predicting information diffusion on Twitter – Analysis

- of predictive features,” *J. Comput. Sci.*, vol. 28, pp. 257–264, Sep. 2018, doi: 10.1016/j.jocs.2017.10.010.
- [4] I. Daga, A. Gupta, R. Vardhan, and P. Mukherjee, “Prediction of Likes and Retweets Using Text Information Retrieval,” *Procedia Comput. Sci.*, vol. 168, pp. 123–128, 2020, doi: 10.1016/j.procs.2020.02.273.
- [5] P. P. Tribhuvan, S. G. Bhirud, and R. R. Deshmukh, “STACKING ENSEMBLE MODEL FOR POLARITY CLASSIFICATION IN FEATURE BASED OPINION MINING,” *Indian J. Comput. Sci. Eng.*, vol. 9, no. 3, pp. 91–95, Jun. 2018, doi: 10.21817/indjcse/2018/v9i3/180903004.
- [6] Y. Sopianti, E. R. Kaburuan, and A. A. Suryani, “Personality prediction using Indonesian twitter data with modified stacking method,” *Int. J. Adv. Sci. Technol.*, vol. 29, no. 5, pp. 2525–2534, 2020.
- [7] S. Mootha, S. Sridhar, and M. S. K. Devi, “A Stacking Ensemble of Multi Layer Perceptrons to Predict Online Shoppers’ Purchasing Intention,” *2020 3rd Int. Semin. Res. Inf. Technol. Intell. Syst. ISRITI 2020*, pp. 721–726, 2020, doi: 10.1109/ISRITI51436.2020.9315447.
- [8] K. Leartantulak and Y. Kitjaidure, “Music Genre Classification of audio signals Using Particle Swarm Optimization and Stacking Ensemble,” in *2019 7th International Electrical Engineering Congress (iEECON)*, Mar. 2019, pp. 1–4. doi: 10.1109/iEECON45304.2019.8938995.
- [9] Y. Xiong, M. Ye, and C. Wu, “Cancer Classification with a Cost-Sensitive Naive Bayes Stacking Ensemble,” *Comput. Math. Methods Med.*, vol. 2021, pp. 1–12, Apr. 2021, doi: 10.1155/2021/5556992.
- [10] F. Tempola, M. Muhammad, and A. Khairan, “Perbandingan Klasifikasi Antara KNN dan Naive Bayes pada Penentuan Status Gunung Berapi dengan K-Fold Cross Validation,” *J. Teknol. Inf. dan Ilmu Komput.*, vol. 5, no. 5, p. 577, Oct. 2018, doi: 10.25126/jtiik.201855983.
- [11] G. Canbek, S. Sagioglu, T. T. Temizel, and N. Baykal, “Binary classification performance measures/metrics: A comprehensive visualized roadmap to gain new insights,” in *2017 International Conference on Computer Science and Engineering (UBMK)*, Oct. 2017, pp. 821–826. doi: 10.1109/UBMK.2017.8093539.
- [12] , JustAnotherArchivist, “GitHub - {JustAnotherArchivist}/snsrape: A social networking service scraper in {Python}.” github.com, Jun. 2022.
- [13] R. Cobos, F. Jurado, and A. Blazquez-Herranz, “A Content Analysis System That Supports Sentiment Analysis for Subjectivity and Polarity Detection in Online Courses,” *IEEE Rev. Iberoam. Tecnol. del Aprendiz.*, vol. 14, no. 4, pp. 177–187, Nov. 2019, doi: 10.1109/RITA.2019.2952298.
- [14] T. E. Tallo and A. Musdholifah, “The Implementation of Genetic Algorithm in Smote (Synthetic Minority Oversampling Technique) for Handling Imbalanced Dataset Problem,” in *2018 4th International Conference on Science and Technology (ICST)*, Aug. 2018, pp. 1–4. doi: 10.1109/ICSTC.2018.8528591.