Department of Digital Business
Journal of Artificial Intelligence and Digital Business (RIGGS)
Homepage: https://journal.ilmudata.co.id/index.php/RIGGS
Vol. 4 No. 3 (2025) pp: 1486-1496
P-ISSN: 2963-9298, e-ISSN: 2963-914X

Analisa Kinerja Algoritma *Supervised Learning* pada Sentimen Ulasan Aplikasi Investasi Online Bibit

Muhamad Arief Yulianto[1], Romi Andrianto[2]
[12] Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Pamulang, Indonesia, 15417
dosen02547@unpam.ac.id

Abstrak

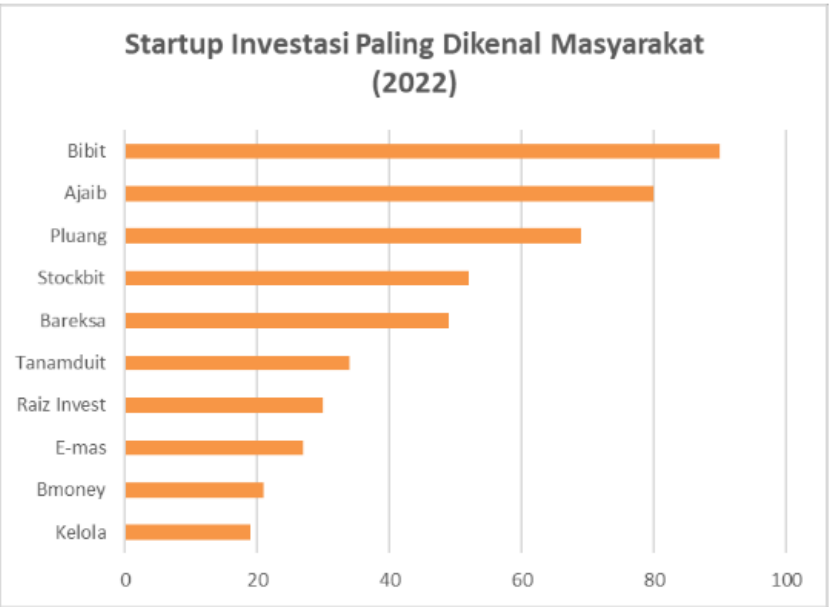
Penelitian ini bertujuan untuk melakukan analisa kinerja algoritma supervised learning dalam melakukan sentimen terhadap ulasan pengguna pada aplikasi investasi online Bibit di google playstore. Berdasarkan hasil survei yang telah dilakukan oleh DailySocial yang tertera pada databoks menunjukkan bahwa aplikasi investasi online Bibit menduduki peringkat pertama sebagai aplikasi terpopuler di kalangan masyarakat Indonesia. Data ulasan pengguna sebanyak 40141 ulasan, diperoleh dengan menggunakan pendekatan teknik web scraping menggunakan bahasa pemrograman python. Adapun langkah – langkah yang dilakukan dalam penelitian ini dimulai dari proses pengumpulan data, pelabelan data, pemrosesan awal data atau preprocessing data, ekstraksi fitur, splitting data dan diakhir dengan pemodelan menggunakan algoritma supervised learning. Model yang digunakan pada algoritma supervised learning pada penelitian ini yaitu naïve bayes, random forest, logistic regression dan decision tree. Dari keempat model tersebut dilakukan pengujian untuk dapat memperoleh nilai tingkat akurasi setiap model dalam melakukan analisa sentimen terhadap data tersebut. Berdasarkan hasil pengujian tersebut menunjukkan bahwa model logistic regression menduduki peringkat pertama yang memiliki tingkat akurasi sebesar 86,6%, disusul oleh model lainnya.

Kata kunci: *Akurasi, Analisa Sentimen, Supervised Learning, Bibit*

1. Latar Belakang

Berdasarkan laporan statistik pasar modal Indonesia pada januari 2024 yang bersumber dari Kustodian Sentral Efek Indonesia (KSEI) melaporkan bahwa adanya kenaikan jumlah investor pasar modal dari penghujung tahun 2023 sampai bulan januari 2024 menjadi sebesar 1,30% dengan dengan jumlah investor sebanyak 12,33 juta investor[1]. Di akhir tahun 2024 hingga maret 2025 menunjukkan *trend positive* pada perkembangan jumlah inverstor pasar modal dengan rata – rata pertumbuhan sebesar 1,98 % dengan jumlah investor sebanyak 15,77 juta[2]. Salah satu usaha untuk mendapatkan *profit* secara efektif bisa dengan melakukan investasi. Pasar modal merupakan salah satu jenis investasi yang selalu mengikuti perkembangan zaman dengan instrumen investasi beresiko cukup tinggi seperti saham, *futures, warrants* maupun *option* baik skala *domestic* maupun *international*[3].

Di kalangan khalayak umum, penilaian predikat sebuah aplikasi itu baik atau buruk secara umum dilihat berdasarkan jumlah pengunduh dan banyaknya bintang yang di peroleh paling tinggi pada *play store.* Namun, menurut mba Erfina dkk, menyatakan bahwa perlu adanya analisa tambahan untuk menentukan bahwa aplikasi tersebut yang berupa ulasan komentar dari pengunduh yang dapat dijadikan sebagai *variable* tambahan dalam pemberian predikat baik atau buruk terhadap suatu aplikasi[4]. Merujuk pada hasil survei yang dilakukan oleh Lembaga Jajak Pendapat (JakPat) yang menyatakan bahwa 63% responden lebih memilih aplikasi investasi yang memiliki kredibilitas yang baik dengan dasar sudah terdaftar di Otoritas Jasa Keuangan (OJK). Seirama dengan hal tersebut responden juga cenderung memilih aplikasi investasi yang mampu menawarkan fleksibilitas pembayaran dan kemudahan dalam pemaikaiannya (*user friendly).* Pada tahun 2022, lembaga survei *DailySocial* menyatakan bahwa aplikasi investasi yang paling terkenal di kalangan masyarakat Indonesia yaitu aplikasi Bibit sebanyak 90% dari total responden, diikuti dengan aplikasi – aplikasi lainnya seperti Ajaib, Pluang, Stockbit dan sebagainya, hal tersebut dapat dilihat dari gambar berikut ini : [5]



Gambar 1. *Startup* Investasi Paling Dikenal Masyarakat

Data mining memiliki kemampuan dalam hal menghasilkan sebuah pola tertentu terhadap kumpulan data tekstual yang bervolume besar. Salah satu varian dari data mining yaitu *text mining* yang mampu membentuk sebuah pola berupa klasifikasi dokumen, analisa sentiment, *information retrieval, clustering* maupun *information extraction.* Dalam hal ini, sentiment analisa memiliki kemampuan untuk mengelompokkan suatu opini ke dalam opini negative, opini netral maupun opini positive secara otomatis[6]. Melalui pendekatan analisis sentimen mampu menggambarkan perasaan seseorang yang diungkapkan melalui suatu tulisan, opini, sikap, penilaian emosi terhadap suatu topik, layanan organisasi, produk ataupun kegiatan lainnya[7], permasalahan ini juga sering disebut sebagai Aspect-Based Sentiment Analisis (ABSA)[8].

Machine learning (ML) salah satu bagian bidang ilmu komputer yang mampu menghasilkan sebuah mesin untuk dapat belajar secara mandiri melalui pembelajaran sekumpulan data dengan pendekatan algoritma yang telah dibangu menjadi sebuah model yang telah di latih sebelumnya[9]. Terdapat tiga pendekatan dalam mengaplikasikan *machine learning* yang digunakan untuk membangun sebuah pola dan pengetahuan sebuah model yaitu *supervised learning, unsupervised learning* dan *reinforcement learning*[10]. Algoritma *Supervised Learning* menghasilkan model dari hasil training data set yang dilengkapi dengan informasi *data input* dan *output (label).* Sedangkan algoritma *unsupervised learning* membangun model melalui training data tanpa label dikarenakan cara kerjanya yang bersifat deskriptif bukan prediktif[11]. Berbeda dengan *reinforcement learning* yang membenung model dengan sistem *trial* dan *error* yang mampu menghasilkan informasi untuk memperbaiki pengatahuan sebelumnya [12].

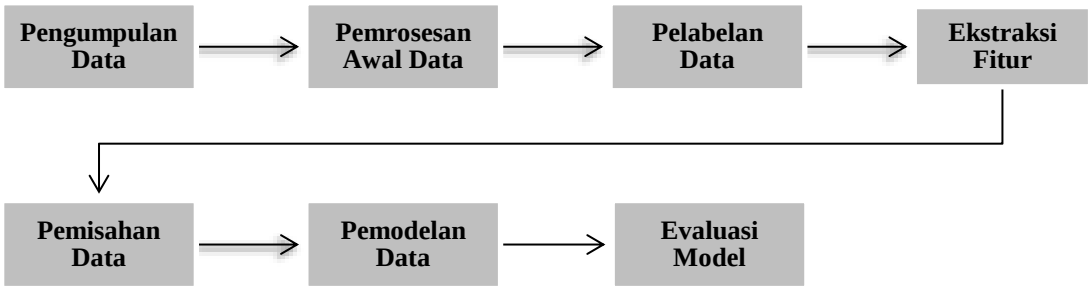
Analisa sentimen sudah diterapkan pada berbagai aspek bidang di kehidupan masyarakat oleh beberapa penelitian sebelumnya. Diantaranya dibidang pendidikan[13][4], kesehatan[7], *e-commerce*[14][8], mobil listrik[15], harga saham[16][17], aplikasi[6][18],aplikasi investasi[5][19][20]. Penelitian yang telah dilakukan oleh Sri Lestari, dkk melakukan analisa sentimen postive terhadap beberapa aplikasi saham di *google playstore* menggunakan algoritma *support vector machine (svm)* memperoleh tingkat akurasi yang berbeda pada setiap aplikasi saham dengan jumlah data latih tidak lebih dari 1200 ulasan[20]. Berbeda dengan Fath Ezzati Kavabila, dkk memanfaatkan algoritma *support vector(svm)* dalam melakukan analisa sentimen terhadap aplikasi ajaib menggunakan 2000 ulasan menghasilkan tingkat akurasi sebesar 85,75%[19]. Sedangkan Nurul Fathiah Annisa, dkk melakukan analisa sentiment aplikasi saham menggunakan algoritma *naïve bayes* menggunakan 3360 data ulasan berbagai aplikasi saham menghasilkan tingkat akurasi sebesar 79,5%[5].

Berdasarkan uraian sebelumnya menunjukkan bahwa aplikasi investasi bibit menduduki peringkat tertinggi sebagai aplikasi investasi yang paling diminati oleh kalangan masyarakat. Serta terdapat beberapa uraian penelitan yang sudah dilakukan dalam melakukan analisa sentimen aplikasi sentiment menggunakan jumlah data ulasan yang berbeda dengan pendekatan algoritma yang berbeda memperolah hasil tingkat akurasi yang berbeda pula. Oleh karena itu, tujuan pada penelitian ini yaitu untuk melakukan analisa kinerja algoritma *supervised learning* dalam melakukan analisa sentimen terhadap ulasan aplikasi bibit engan jumlah data ulasan yang dilakukan lebih besar dari data penelitian sebelumnya.

DOI: https://doi.org/ 10.31004/riggs.v4i3.2168

2. Metode Penelitian

Tahap ini menjelaskan langkah – langkah yang akan dilakukan peneliti secara sistematis untuk mencapai tujuan penelitan dalam melakukan analisa kinerja algoritma *supervised learning* dalam melakukan analisa sentimen pada ulasan aplikasi Bibit. Mulai dari proses pengumpulan data hingga mendapatkan performa akurasi algoritma *supervised learning* dalam menyelesaikan kasus tersebut. Adapun langkah – langkah detailnya dalam melakukan penelitian ini dapat dilihat melalui diagram proses berikut ini :



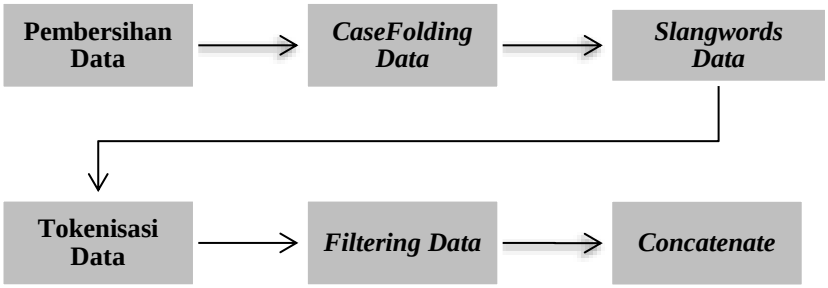
Gambar 2. Diagram Proses Penelitian

2.1 Pengumpulan Data

Pada proses pengumpulan data ini dilakukan dengan menggunakan metode *web scrapping* yang diarahkan menuju alamat situs aplikasi bibit yang sudah diupload di *google playstore.* Implementasi teknik ini menggunakan bahasa *python* dengan memanfaatkan pustaka yang ada di *python* serta mengatur kriteria data yang akan diambil meliputi id aplikasi, bahasa ulasan yang diambil, negara asal ulasan tersebut, ulasan paling relevan, serta jumlah data yang akan diambil. Pada penelitian ini, jumlah data batas maksimal data yang diambil sebanyak 40141 ulasan. Setelah data berhasil didapatkan, selanjutnya data tersebut di simpan dalam bentuk file csv yang akan dilakukan pemrosesan selanjutnya.

2.2 Pemrosesan Awal Data

Pemrosesan awal data dilakukan dengan tujuan untuk mendapatkan data yang bersih terbebas dari data duplikat, data yang tidak lengkap ataupun data yang tidak sesuai. Proses ini perlu dilakukan sebelum proses pemodelan data menggunakan algoritma *supervised learning* untuk mendapatkan data latih yang terhindar dari kesalahan. Adapun pemrosesan awal data dapat dilihat berdasarkan diagram proses di bawah ini :



Gambar 3. Diagram Pemrosesan Awal Data

a. Pembersihan Data (*Cleaning Data)*

Proses ini digunakan untuk membersihkan teks dengan beberapa langkah, seperti menghapus mention, hashtag, RT (retweet), tautan (link), angka, dan tanda baca. Selain itu, itu juga menggantikan karakter newline dengan spasi dan menghilangkan spasi ekstra di awal dan akhir teks.

b. *CaseFolding Data*

Tahap ini data akan mengalami perubahan atau konversi semua karakter dalam teks menjadi huruf kecil (*lowercase*), sehingga teks menjadi lebih seragam.

c. *Slangwords* Data

Pada proses ini dilakukan konversi pada dataset yang mengandung kata – kata *slang* menjadi kata – kata yang lebih baku dan diterima dikalangan umum.

d. Tokenisasi Data

Proses yang dilakukan pada tahap ini berfungsi untuk membagi teks menjadi daftar kata atau token. Hal ini akan berguna untuk mengurai teks menjadi komponen-komponen dasar.

DOI: https://doi.org/ 10.31004/riggs.v4i3.2168

e. *Filtering Data*

Pada tahap keempat ini memilki tujuan untuk menghapus kata-kata yang tidak bermakna yang ada pada *libray stopwords* dalam teks sehingga kata – kata yang ada pada teks hanya kata – kata yang memiliki makna lebih penting.

f. *Concatenate*

Prose penggabungan kembalai daftar token – token data menjadi sebuah kalimat yang utuh untuk dilakukan proses selanjutnya.

2.3 Pelabelan Data

Proses pemberian label data yang telah dikumpulkan dilakukan secara otomatis dengan memanfaatkan dataset kosakata sentimen positif ataupun negatif untuk kalimat berbahasa dari *github* yang sudah diupload oleh saudara Fajri Koto yang telah disimpan dalam bentuk file tsv[21]. Sebelum dilakukan proses pemberian sentimen terhadap data tersebut, langkah selanjutnya yaitu menghilangkan data - data ganda terlebih dahulu. Lalu dilanjutkan dengan melakukan perhitungan untuk mencari nilai polaritas dari kalimat tersebut dengan membuat sebuah fungsi. Dari fungsi tersebut mampu menghasilkan nilai polaritas suatu data dengan ketentuan jika nilai polaritasnya lebih besar dari nol (0) maka kalimat tersebut memiliki sentimen positif. Sebalikya jika nilai polaritasnya kurang dari nol (0) maka kalimat tersebut akan memiiliki sentimen negatif. Sedangkan selain nilai tersebut, kalimat diterjemahkam dengan sentimen netral.

2.4 Ekstraksi Fitur

Data yang telah melalui pemrosesan awal sehingga data sudah terhindar dari *noise*, maka proses selanjutnya yang akan dilakukan yaitu ekstraski fitur. Hal ini bertujuan untuk melakukan perubahan kata – kata menjadi angka yang akan digunakan sebagai data latih untuk melakukan pemodelan sentimen analisa. Adapun metode yang digunakan untuk melakukan ekstraksi fitur ini menggunakan pendekatan TF-IDF (*Term Frequency – Inverse Document Frequency)* untuk mengetahui frekuensi kata yang sering muncul.

2.5 Pemisahan Data

Data yang telah melalui pemrosesan awal sehingga data sudah terhindar dari *noise*, maka proses selanjutnya yang akan dilakukan yaitu membagi data menjadi dua bagian yaitu *data training* (latih) dan *data testing* (uji). Pembagian data ini dilakukan dengan memanfaatkan *library* dari *python*

2.6 Pemodelan Data

Pada tahap ini data akan mengalami proses pemodelan menggunakan empat (4) algoritma *supervised Learning* diantaranya *Logistic Regression (LR), Random Forest (RF), Naïve Bayes (NV)* dan *Decision Tree (DT).*

a. *Logistic Regression (LR)*

Merupakan salah satu kaidah statistik yang mampu menganalisa korelasi antara dua atau lebih variabel independen dan independen kategorikal[22].

b. *Random Forest*

Salah satu model *ensamble learniang* dari beberapa *decision tree* yang bertujuan untuk meningkatkan perfoma model dalam memprediksi. Model ini memiliki kemampuan dalam menyelesaikan permasalahan klasifikasi maupun regresi [22].

c. *Naïve Bayes*

Model yang mampu melakukan proses pengelompokkan suatu data berdasarkan nilai suatu probabilitas dari hasil yang telah diberikan terhadap beberapa kondisi menggunakan teorema Bayes. [23].

d. *Decision Tree*

Bentuk algoritma yang membangun model menyerupai struktur pohon dimana setiap simpul dalam pohon tersebut akan membentuk suatu keputusan berdasarkan fiur – fiur terkait [24].

2.7 Evaluasi Model

Pada tahap ini memiliki tujuan untuk mengetahui kinerja keempat algoritma *supervised learning* dalam melakukan analisa sentimen terhadap ulasan – ulasan pemakai aplikasi bibit berdasarkan perhitungan matrik tingkat akurasi.

3. Hasil dan Diskusi

Pada bagian ini akan diuraikan secara terperinci setiap tahapan yang akan dilakukan untuk menggapai tujuan penelitian ini dalam menganalisa algoritima *supervised learning* dalam melakukan sentimen terhadap ulasan aplikasi bibit. Adapun detaial hasil pembahahasannya adalah sebagai berikut:

3.1 Pengumpulan Data

Proses pengumpulan data dilakukan dengan menggunakan teknik *scraping* yang melibatkan beberapa *library python* yaitu *google_play_scrapper*. Untuk melakkukan teknik ini, penulis memanfaatkan fungsi bawaan library

DOI: https://doi.org/ 10.31004/riggs.v4i3.2168

tersebut yaitu *reviews_all()* dengan menambahkan beberapa parameter yang digunakan untuk melengkapi kebutuhan data yang diharapkan meliputi memasukkan id aplikasi bibit yang telah diupload *google playstore* yang akan diambil data ulasan yaitu *com.bibit.bibitid.* Beberapa parameter untuk mendukung hal ini yaitu pemilihan bahasa dan negara yang diatur pada atribut *lang* dan *country* yang diisi dengan nilai *id* untuk mendapatkan bahasa dan negara yang digunakan adalah Indonesia. Serta hanya ulasan yang relevan saja yang diambil juga diatur dalam proses ini dengan memanfaatkan parameter *sort* yang diisi dengan nilai *Sort.MOST_RELEVANT.* Serta diakhiri dengan mengatur parameter *count* untuk mendapatkan jumlah data ulasan yang akan diambil. Berdasarkan pengaturan parameter tersebut peneliti mampu mendapatkan hasil ulasan sebanyak 40141 data yang akan dilakukan untuk proses tahapan selanjutnya.

Tabel 1 Tampilan Contoh *Dataset*

No	username	score	At	content
1	Pengguna Google	1	2025-07-14 05:05:10	Ribet banget, mau upgrade bibit plus saja dipe..
2	Pengguna Google	4	2025-07-16 09:19:37	aplikasinya sangat bagus dan cocok untuk inves...
…	…	…	…	…
40140	Pengguna Google	1	2022-02-25 02:10:29	👎
40141	Pengguna Google	1	2021-09-14 17:27:48	😂

3.2 Pemrosesan Awal

Tahapan ini dilakukan untuk mempersiapkan data yang bersih dan siap untuk dilakukan pemodelan data. Sebelum dilakukan proses ini, data sudah dicheck terlebih dahulu terkait dengan duplikasi data, sehingga data yang digunakan sudah terhidar dari duplikasi. Adapun beberapa tahapan yang dilakukan dalam proses ini adalah sebagai berikut :

a. Pembersihan Data (*Cleaning Data)*

Proses ini digunakan untuk membersihkan teks agar terbebas dari karakter yang tidak diinginkan seperti tanda mention, hashtag, RT, *http,* karakter special, dengan beberapa langkah, seperti menghapus mention, hashtag, RT (retweet), tautan (link), angka, dan tanda baca, *newline* dan menghilangkan spasi ekstra di awal dan akhir teks.

Tabel 2 *Cleaning Data*

Text Awal	Text *Cleaning*
Ribet banget, mau upgrade bibit plus saja dipersulit bukan main. Sudah sesuai prosedur pun, tiba2 stuck, diklik "lanjut", malah balik lagi ke halaman awal disuruh ulang lagi prosesnya. Begitu terus sampai belasan kali. Sudah chat bantuan, malah disuruh main game dulu, karena tiap pertanyaan akan dibalas tiap 2 jam 😂	Ribet banget mau upgrade bibit plus saja dipersulit bukan main Sudah sesuai prosedur pun tiba stuck diklik lanjut malah balik lagi ke halaman awal disuruh ulang lagi prosesnya Begitu terus sampai belasan kali Sudah chat bantuan malah disuruh main game dulu karena tiap pertanyaan akan dibalas tiap jam

b. *CaseFolding Data*

Proses ini sangat bermanfaat dalam menyeragamkan data dengan cara merubah semua karakter teks menjadi huruf kecil (*lowercase)*.

Tabel 3 *CaseFolding Data*

Text Awal	Text *Cleaning*
Ribet banget, mau upgrade bibit plus saja dipersulit bukan main	ribet banget mau upgrade bibit plus saja dipersulit bukan main

DOI: https://doi.org/ 10.31004/riggs.v4i3.2168

c. *Slangwords* Data

Pada proses ini dilakukan konversi pada dataset yang mengandung kata – kata *slang* menjadi kata – kata yang lebih baku dan diterima dikalangan umum. Kumpulan data slangwords didapatkan dari github yang sudah diupload oleh akun louisowen6.

Tabel 4 *Slangwords Data*

Text Awal	Text *Cleaning*
ribet banget … malah … bantuan malah disuruh …	ribet banget ... bahkan … bantuan bahkan disuruh …

d. Tokenisasi Data

Proses penguraian teks menjadi komponen – komponen terkecil pada teks. Komponen – komponen terkceil ini dijadikan sebagai daftar kata atau token untuk proses selanjutnya.

Tabel 5 Tokenisasi Data

Text Awal	Text Tokenisasi
ribet banget, mau upgrade bibit plus saja dipersulit bukan main	[ribet, banget, mau, upgrade, bibit, plus, saja, dipersulit, bukan, main]

e. *Filtering Data*

Proses pensortiran data ini ditujukan untuk mendapatkan kumpulan kata yang memiliki makna dengan cara mencocokkan data dengan kumpulan kata tidak bermakna pada *library stopwords.* Sehingga jika data kita terdapat kata yang ada pada *library* tersebut, maka kata tersebut akan di hapus dari data kita. Sehingga data kita sudah bersih dari daftar kata yang tidak bermakna yang terkandung dalam *library* tersebut.

Tabel 6 *Filtering Data*

Text Awal	Text *Filtering*
[ribet, banget, mau, upgrade, bibit, plus, saja, dipersulit, bukan, main]	[ribet, banget, upgrade, bibit, plus, dipersulit, main]

f. *Concatenate*

Prose penggabungan kembali daftar token – token data yang sudah terbebas dari daftar kata yang tidak bermakna menjadi sebuah kalimat yang utuh untuk dilakukan proses selanjutnya. Hal ini perlu dilakukan sebelum melakukan proses pelabelan data untuk mendapatkan hasil sentimen dari data tersebut.

Tabel 7 *Concatenate Data*

Text Awal	Text Akhir
[ribet, banget, upgrade, bibit, plus, dipersulit, main]	ribet banget upgrade bibit plus dipersulit main

3.3 Pelabelan Data

Dalam proses analisis sentimen berbasis teks, pelabelan data merupakan tahap penting yang menentukan kualitas model klasifikasi yang akan dibangun. Salah satu pendekatan efisien untuk melakukan pelabelan otomatis adalah menggunakan metode sentiment scoring berbasis leksikon. Dalam konteks Bahasa Indonesia, leksikon sentimen yang dikembangkan oleh Mas Fajri Koto menjadi salah satu sumber daya yang sangat bermanfaat. Leksikon tersebut berisi kumpulan kata kata positif sebanyak 3609 sedangkan kata negatif sebanyak 6609. Tahapan selanjutnya melakukan scoring sentiment untuk melakukan pelabelan pada data ulasan dengan aturan jika sentiment score > 0 maka data ulasan masuk ke dalam kelas sentiment positif. Sebaliknya jika hasil sentimen score < 0 maka data ulasan termasuk ke dalam kelas negatif, selain nilai score tersebut maka data ulasan masuk ke dalam kelas netral. Berdasarkan proses tersebut, diperoleh hasil ulasan sentimen positif sebanyak 18920 ulasan (47,1%), sentimen negative sebanyak 11307 ulasan (28,2%) dan sentimen netral sebanyak 9914 ulasan (24,7%). Adapun

DOI: https://doi.org/ 10.31004/riggs.v4i3.2168

hasil sentimen tersebut peneliti sajikan dalam *wordcloud* yang menggambarkan masing-masing pada kelas sentimen sebagai berikut:

Gambar 4. *WordCloud* Kelas Sentimen

3.4 Ekstraksi Fitur

Proses ektraksi fitur yang digunakan dalam penelitian ini menggunakan teknik TF-IDF yaitu dengan mengubah teks menjadi fitur numerik (vektor) dengan tujuan agar data yang dimiliki dapat diproses menggunakan algoritma pada *machine learning.* Proses konversi ini hanya dilakukan terhadap kata – kata yang sering muncul pada sebuah dokumen sedangkan kata – kata yang jarang muncul atau sangan sering akan dibuang. Untuk melakukan hal ini peneliti mengatur hanya mengambil 200 fitur teratas berdasarkan skor TF-IDF serta frequensi kemunculan kata minimal berada pada 17 dokumen. Sedangkan kata – kata yang muncul lebih dari 80% akan diabaikan karena dianggap kata yang tidak memiliki makna.

Tabel 8. Tampilan Contoh Hasil TF-IDF

data	akun	aplikasi	aplikasinya	user
1	0.249826	0.0	0.217737	0.0
2	0.0	0.268055	0.0	0.508131
3	0.0	0.279063	0.0	0.0

Berdasarksan hasil tersebut menunjukkan bahwa Kata "akun" memiliki skor 0.249826, artinya cukup penting dalam pada data 1. Juga kata "aplikasinya" cukup relevan dengan skor 0.217737 pada data 1. Sedamglam kata lain seperti "aplikasidan "user" tidak muncul atau tidak penting yang ditunjukkan dengan skor 0.0. Sedangkan kata "aplikasi" dainggap penting dengan skor 0.268055) yang mungkin bisa dijadikan kata kunci pada data 2. Juga kata "user" menjadi sangat penting dengan skor 0.508131 sehingga memiliki pengaruh tinggi terhadap isi pada data 2 ini. Pada contoh data ke 3, hanya kata "aplikasi" yang punya bobot tinggi dengan skor 0.279063 sehingga fokus pembahasan kemungkinan hanya pada topik ini di data tersebut.

3.5 Pemisahan Data

Pada penelitian ini, proses pembagian data (data splitting) untuk pelatihan dan pengujian model machine learning, khususnya pada data teks yang telah direpresentasikan dalam bentuk TF-IDF pada proses sebelumnya. Dalam implementasinya, memanfaatkan fungsi dari *scikit-laern* yaitu train_test_split yang digunakan untuk membagi dataset menjadi data latih dan data uji. Adapun parameter yang digunakan dalam fungsi tersebut berupa fitur input hasil matriks TF-IDF, dan target dari masing – masing input. Sedangkan ukuran data uji yang digunakan menggunakan prosentasi 80:20 artinya 80% data akan dijadikan sebagai data pelatihan sedangkan 20% menjadi data pengujian. Selain itu juga mentapkan *seed* acak dengan nilai 42 supaya hasil pembagian data selalu sama setiap kali kode dijalankan, sehingga data yang diuji merupakan data yang belum pernah dilihat oleh model sebelumnya.

3.6 Pemodelan Data

Pada penelitian ini, pemodelan dilakukan menggunakan pendekatan supervised learning, di mana dataset yang digunakan telah dilengkapi dengan label sebagai target output. Beberapa algoritma supervised learning diterapkan untuk membandingkan performa masing-masing model terhadap data, di antaranya *Naïve Bayes, Random Forest, Logistic Regression* dan *Decision Tree*. Model-model ini dipilih karena mampu menangani data

DOI: https://doi.org/ 10.31004/riggs.v4i3.2168

kategorikal maupun numerik, serta memiliki interpretabilitas dan akurasi yang baik dalam tugas klasifikasi. Selanjutnya model ini akan dilakukan pelatihan dengan hasil pelatihan tersebut diukur dengan menggunakan metrik seperti akurasi, precision, recall, F1-score, dan confusion matrix.

a. *Naïve Bayes*

Objek model *Bernoulli Naïve Bayes* dihasilkan dengan cara import algoritma *Bernoulli Naive Bayes* dari pustaka *scikit-learn*. Model ini akan digunakan untuk memprediksi label dari data latih yang sudah dirubah formatnya dari sparse matrix hasil TF-IDF menjadi array dengan data uji. Selanjutnya model akan dilatih menggunakan data latih untuk melakukan prediksi kelas sentimen positif, negative maupun netral.

b. *Random Forest*

Membangun dan melatih model *Random Forest* dengan mengimport *RandomForestClassifier* dari pustaka *scikit-learn*. Model ini akan digunakan untuk memprediksi label dari data latih yang sudah dirubah formatnya dari sparse matrix hasil TF-IDF menjadi array dengan data uji. Secara default, model akan menggunakan 100 pohon keputusan (*n_estimators*=100) dan parameter lain *default*. Selanjutnya model akan dilatih menggunakan data latih untuk melakukan prediksi kelas sentimen positif, negative maupun netral.

c. *Logistic Regression*

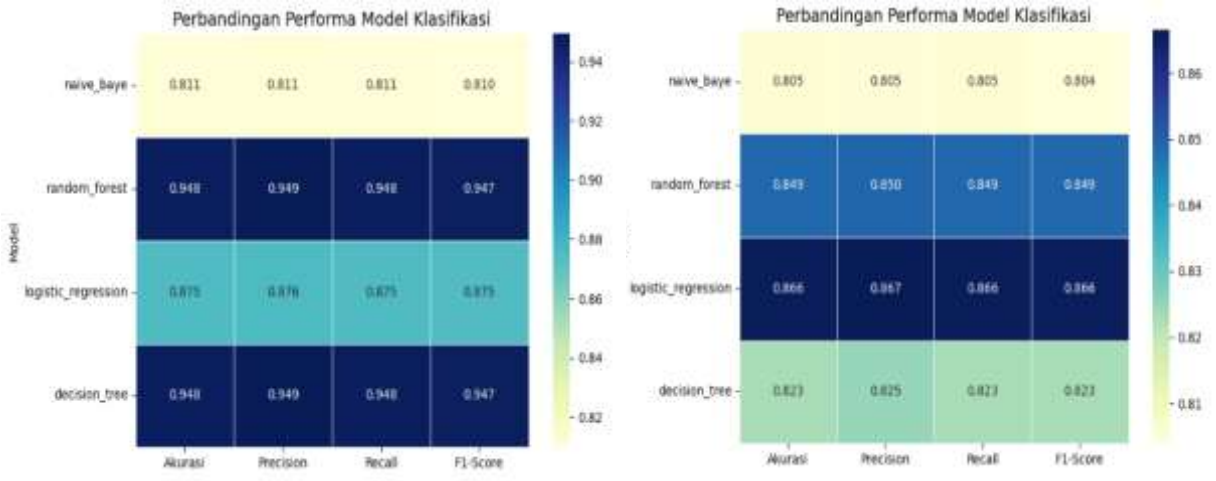
Model *Random Forest* dibangun dengan mengimport *LogisticRegression* dari pustaka *scikit-learn*. untuk klasifikasi teks dengan data teks yang direpresentasikan dalam bentuk TF-IDF. Model ini berkerja dalam mengklasifikasikan teks dengan menentukan jumlah maksimum iterasi saat proses pelatihan sebesar 2000 kali. Selain itu menambahkan parameter solver='saga' mengacu pada algoritma optimisasi yang dipakai untuk menemukan koefisien (bobot) terbaik sehingga membuatnya cepat konvergen pada dataset besar. Paremater terakhir dengan memberikan nilai random_state=42 sehingga data yang diuji merupakan data yang belum pernah dilihat oleh model sebelumnya. Selanjutnya model akan dilatih menggunakan data latih untuk melakukan prediksi kelas sentimen positif, negative maupun netral.

d. *Decision Tree*

*M*embangun dan melatih model *Random Forest* dengan mengimport *RandomForestClassifier* dari pustaka *scikit-learn* untuk klasifikasi teks dengan data teks yang direpresentasikan dalam bentuk TF-IDF. Secara *default* parameter yang digunakan yaitu criterion='gini' yang beruguna untuk mengukur pemisahan antar kelas. Selanjutnya model akan dilatih menggunakan data latih untuk melakukan prediksi kelas sentimen positif, negative maupun netral.

3.7 Evaluasi Model

Setelah dilakukan proses pemodelan data, langkah selanjutnya yaitu melakukan evaluasi terhadap model yang telah dibuat sebelumnya. Proses evaluasi model ini dilakukan dengan menggunakan data training maupun testing dengan menggunakan metrik performa seperti *accurracy*, *precision*, *recall*, dan *F1-score*. Evaluasi dilakukan terhadap data training dan data uji ini digunakan untuk mendapatkan model terbaik dalam melakukan analisa sentimen terhadap aplikasi bibit. Adapun hasil evaluasi model pelatihan dan pengujian ditampilkan dalam bentuk *heatmap* sebagai berikut :



Gambar 5 Perbandingan Evaluasi Model Data *Training* dan *Testing*

Semua model memiliki performa sangat baik pada data training. *Random Forest* dan *Decision Tree* menunjukkan skor tertinggi atau hampir sempurna, namun ada kemungkinan mengalami *overfitting*. *Logistic*

DOI: https://doi.org/ 10.31004/riggs.v4i3.2168

Regression juga cukup tinggi, dengan akurasi 87.5% sedangkan *N*aïve *Bayes* memiliki performa paling rendah, walaupun tetap cukup baik yaitu lebih besar dari 80%. Sedangkan Pada data testing, semua skor menurun, hal ini wajar karena model diuji terhadap data baru. *Random Forest* dan *Decision Tree* mengalami penurunan yang signifikan yaitu dari 0.948 ke 0.849 dan 0.823, hal ini menunjukkan overfitting pada data training. Sedangkan *Logistic Regression* lebih stabil yaitu performa training memiliki nilai dari 0.875 menjadi 0.866. Hal ini menunjukkan kemampuan generalisasi yang baik. Namun *Naïve Bayes* tetap rendah dan konsisten di sekitar 0.80 pada data *training* dan *testing.* Hal ini mengindikasikan bahwa model ini stabil namun tidak terlalu kuat. Berdasarkan hasil tersebut maka *Logistic Regression* merupakan pilihan terbaik untuk mendapatkan model yang seimbang antara akurasi, stabilitas, dan kemampuan generalisasi. Sedangkan *Random Forest* cocok jika data dilatih dan diuji pada domain serupa, tetapi perlu teknik regularisasi untuk mencegah *overfitting*. Berbeda dengan *Naive Bayes* dan *Decision Tree* memiliki performa lebih rendah, juga *Naive Bayes* cocok jika data bersifat linear dan sederhana sedangkan *Decision Tree* rentan *overfit* jika tidak dipangkas (*pruning*).

4. Kesimpulan

Penelitian ini bertujuan untuk menganalisis kinerja beberapa algoritma supervised learning dalam mengklasifikasikan sentimen pengguna terhadap aplikasi investasi online Bibit. Algoritma yang dibandingkan meliputi Naive Bayes, Decision Tree, Random Forest, dan Logistic Regression. Evaluasi dilakukan menggunakan data ulasan pengguna sebanyak 40141 yang telah melalui tahap preprocessing dan pelabelan sentimen secara otomatis. Berdasarkan hasil pengujian, diperoleh bahwa algoritma Random Forest dan Decision Tree menunjukkan performa tertinggi pada data pelatihan dengan akurasi mencapai 94,8%, namun performanya menurun pada data pengujian, yang mengindikasikan terjadinya overfitting. Sebaliknya, Logistic Regression memberikan hasil yang lebih stabil antara data latih dan data uji, dengan selisih metrik evaluasi yang relatif kecil, serta akurasi tertinggi pada data uji sebesar 86,6%. Hal ini menunjukkan bahwa Logistic Regression memiliki kemampuan generalisasi yang lebih baik terhadap data yang belum pernah dilihat sebelumnya.

Referensi

[1] KSEI, "Siaran Pers Antusiasme Antusiasme Investor Muda Berinvestasi Terus Meningkat," *Kasei.Co.Id*, pp. 1–6, 2023, [Online]. Available: https://www.ksei.co.id/files/uploads/press_releases/press_file/id-id/232_berita_pers_antusiasme_investor_muda_berinvestasi_terus_meningkat_20231031134735.pdf.

[2] T. G. Y. S. Muhammad Fikri and G. A. Cakti, "Potret Investasi di Pasar Modal 2025," *DataIndonesia.id*, vol. 1, no. 1, Jakarta, pp. 1–37, 2025.

[3] D. Argapryla, "Pengaruh Literasi Keuangan, Overconfidence Dan Persepsi Risiko Terhadap Minat Berinvestasi Reksadana Mahasiswa …," vol. 5, no. 3, pp. 494–512, 2022.

[4] A. Erfina, E. S. Basryah, A. Saepulrohman, and D. Lestari, "Analisis Sentimen Aplikasi Pembelajaran Online Di Play Store Pada Masa Pandemi Covid-19 Menggunakan Algoritma Support Vector Machine (Svm)," *Semin. Nas. Inform.*, vol. 1, no. 1, pp. 145–152, 2020, [Online]. Available: http://jurnal.upnyk.ac.id/index.php/semnasif/article/view/4094.

[5] Norwahidah Mohamad Yazid, "ANALISIS SENTIMEN APLIKASI INVESTASi," *J. Electr. Syst.*, vol. 20, no. 4s, pp. 582–589, 2024, doi: 10.52783/jes.2074.

[6] L. O. Lukmana, "Analisis Sentimen Terhadap Ulasan Pengguna Aplikasi Threads Instagram Di Playstore Menggunakan Algoritma Naive Bayes," *J. Inform. dan Tek. Elektro Terap.*, vol. 13, no. 2, 2025, doi: 10.23960/jitet.v13i2.6250.

[7] F. F. Mailoa, "Analisis sentimen data twitter menggunakan metode text mining tentang masalah obesitas di indonesia," *J. Inf. Syst. Public Heal.*, vol. 6, no. 1, p. 44, 2021, doi: 10.22146/jisph.44455.

DOI: https://doi.org/ 10.31004/riggs.v4i3.2168

[8] I. S. K. Idris, Y. A. Mustofa, and I. A. Salihi, "Analisis Sentimen Terhadap Penggunaan Aplikasi Shopee Mengunakan Algoritma Support Vector Machine (SVM)," *Jambura J. Electr. Electron. Eng.*, vol. 5, no. 1, pp. 32–35, 2023, doi: 10.37905/jjeee.v5i1.16830.

[9] C. Puspitasari, "Sekilas tentang Data Science, Data Mining, dan Machine Learning," *Binus university*, 2022. https://binus.ac.id/malang/2022/05/sekilas-tentang-data-science-data-mining-dan-machine-learning/ (accessed Jul. 08, 2025).

[10] R. S. Nurhalizah, R. Ardianto, and P. Purwono, "Analisis Supervised dan Unsupervised Learning pada Machine Learning: Systematic Literature Review," *J. Ilmu Komput. dan Inform.*, vol. 4, no. 1, pp. 61–72, 2024, doi: 10.54082/jiki.168.

[11] H. Abijono, P. Santoso, and N. L. Anggreini, "Algoritma Supervised Learning Dan Unsupervised Learning Dalam Pengolahan Data," *J. Teknol. Terap. G-Tech*, vol. 4, no. 2, pp. 315–318, 2021, doi: 10.33379/gtech.v4i2.635.

[12] M. Karimah, A. Hindasyah, and T. Taryo, "Analisa Kinerja Algoritma Random Forest Classifier dengan Mutual Information dan Skip-Gram pada Klasifikasi Jurnal INIS," *J I M P - J. Inform. Merdeka Pasuruan*, vol. 7, no. 3, p. 101, 2023, doi: 10.51213/jimp.v7i3.638.

[13] Sulistiowati, V. Nurcahyawati, Muhammad Alfa Fawwaz, Erwin Sutomo, and Tutut Wurijanto, "Deteksi sentimen ulasan pengguna e-learning menggunakan algoritma naive bayes (Studi Kasus: RuangGuru, Pahamify, Merdeka Mengajar)," *INFOTECH J. Inform. Teknol.*, vol. 5, no. 2, pp. 239–248, 2024, doi: 10.37373/infotech.v5i2.1399.

[14] Tania Puspa Rahayu Sanjaya, Ahmad Fauzi, and Anis Fitri Nur Masruriyah, "Analisis sentimen ulasan pada e-commerce shopee menggunakan algoritma naive bayes dan support vector machine," *INFOTECH J. Inform. Teknol.*, vol. 4, no. 1, pp. 16–26, 2023, doi: 10.37373/infotech.v4i1.422.

[15] Luthfi Krisna Bayu and T. Wurijanto, "Analisis sentimen mobil listrik menggunakan metode Naïve Bayes Classifier," *INFOTECH J. Inform. Teknol.*, vol. 5, no. 2, pp. 328–335, 2024, doi: 10.37373/infotech.v5i2.1465.

[16] Christophorus, B. Saputra, and D. P. Koesrindartoto, "Pemanfaatan Analisis Sentimen dalam Prediksi Harga Saham: Studi pada Investor Retail Indonesia," vol. 21, no. 1, pp. 1–17, 2024, [Online]. Available: http://ejournal.atmajaya.ac.id/index.php/JM.

[17] A. P. Soedarbe, M. R. Ma'arif, A. W. Murdiyanto, and M. A. A. Al Badawi, "Analisis Sentimen Pergerakan Harga Saham Sebuah Perusahaan di Media Sosial Twitter," *Teknomatika J. Inform. dan Komput.*, vol. 14, no. 2, pp. 64–68, 2021, doi: 10.30989/teknomatika.v14i2.1129.

[18] N. Meilani, Mhd. Furqan, and Suhardi, "Analisis sentimen pengguna aplikasi BSI mobile akibat ransomware menggunakan algoritma support vector machine," *INFOTECH J. Inform. Teknol.*, vol. 5, no. 1, pp. 42–51, 2024, doi: 10.37373/infotech.v5i1.1102.

[19] F. E. Kavabilla, T. Widiharih, and B. Warsito, "Analisis Sentimen Pada Ulasan Aplikasi Investasi Online Ajaib Pada Google Play Menggunakan Metode Support Vector Machine Dan Maximum Entropy," *J. Gaussian*, vol. 11, no. 4, pp. 542–553, 2023, doi: 10.14710/j.gauss.11.4.542-553.

[20] S. Lestari and S. Saepudin, "Support Vector Machine: Analisis Sentimen Aplikasi Saham di Google Play Store," *JUSIFO (Jurnal Sist. Informasi)*, vol. 7, no. 2, pp. 81–90,

DOI: https://doi.org/ 10.31004/riggs.v4i3.2168

2021, doi: 10.19109/jusifo.v7i2.9825.

[21] F. Koto and G. Y. Rahmaningtyas, "Inset lexicon: Evaluation of a word list for Indonesian sentiment analysis in microblogs," *Proc. 2017 Int. Conf. Asian Lang. Process. IALP 2017*, vol. 2018-January, no. December, pp. 391–394, 2017, doi: 10.1109/IALP.2017.8300625.

[22] W. O. Simanjuntak, A. Bijaksana, P. Negara, and R. Septriana, "Perbandingan Algoritma Logistic Regression dan Random Foret (Studi Kasus : Klasifikasi Emosi Tweet) Comparison Of Logistic Regression And Random Forest Algorithms (Case Study: Tweet Emotion Classification)," *J. Apl. dan Ris. Inform.*, vol. 02, no. 1, pp. 160–164, 2023, doi: 10.26418/juara.v2i1.69682.

[23] D. H. Depari, Y. Widiastiwi, and M. M. Santoni, "Perbandingan Model Decision Tree, Naive Bayes dan Random Forest untuk Prediksi Klasifikasi Penyakit Jantung," *Inform. J. Ilmu Komput.*, vol. 18, no. 3, p. 239, 2022, doi: 10.52958/iftk.v18i3.4694.

[24] A. Danny *et al.*, "Perbandingan Algoritma Supervised Learning Terhadap Klasifikasi Citra Bunga Untuk Mengukur Akurasi Masing-Masing Algoritma," *Semin. Nas. Inform. Bela Negara*, vol. 3, pp. 2747–0563, 2023.

DOI: https://doi.org/ 10.31004/riggs.v4i3.2168