Jurnal Ilmiah
Teknologi Sistem
Informasi

Jitsi

Vol. 6 (2025) No. 2

https://jurnal-itsi.org/index.php/jitsi; E-ISSN: 2722-4600; ISSN: 2722-4619
DOI: 10.62527/jitsi.6.2.422

Sentimen Analisis Review Film Pada IMDb Menggunakan Algoritma Logistic Regression

Muhammad Azka Faridi[#], Fauzia Tuzzahra[#], Adnan Al-Qadri[#], Rhaudy Nahavira[#], Putri Amelia Az-Zahrah[#], Cindi Apriliani[#], Abdiansah[#]

[#] *Teknik Informatika, Fakultas Ilmu Komputer, Universitas Sriwijaya, Indonesia*
E-mail: azkafaridi[at]gmail.com, fauziatuzzahra99[at]gmail.com, alqadriadnan2004[at]gmail.com, raudiajaa07[at]gmail.com, putriazzahrah379[at]gmail.com, apriliacindi43[at]gmail.com, abdiansah[at]unsri.ac.id

A B S T R A C T S

Sentiment Analysis is a subfield of Machine Learning that focuses on analyzing opinions expressed in textual data. IMDb, as a widely used platform, provides a space for movie enthusiasts worldwide to share their thoughts and reviews. User feedback can serve as a benchmark for evaluating a film's success. This research aims to classify reviews into positive and negative categories using the Logistic Regression algorithm combined with Grid Search and Active Learning methods. The classification results show that the highest accuracy achieved is 90.90% using Logistic Regression with the combination of Grid Search and Active Learning. Meanwhile, Logistic Regression with Active Learning alone achieved an accuracy of 90.58%, Logistic Regression with Grid Search reached 90.17%, and the basic Logistic Regression model achieved an accuracy of 89.81%.

Manuscript received Mar 29, 2025; revised May 27, 2025. accepted Jun 11, 2025 Date of publication Jun 30, 2025. International Journal, JITSI : Jurnal Ilmiah Teknologi Sistem Informasi licensed under a Creative Commons Attribution-Share Alike 4.0 International License

A B S T R A K

Sentiment analysis merupakan salah satu bidang dalam machine learning yang berfokus pada analisis opini dalam bentuk teks. IMDb, sebagai platform yang telah lama digunakan, menyediakan informasi sekaligus menjadi wadah bagi para pecinta film di seluruh dunia untuk berbagi pendapat. Respons yang diberikan oleh pengguna dapat menjadi tolak ukur keberhasilan suatu film. Penelitian ini bertujuan untuk mengklasifikasikan opini menjadi dua kategori, yaitu positif dan negatif, dengan menggunakan algoritma Logistic Regression yang dikombinasikan dengan metode Grid Search dan Active Learning. Hasil klasifikasi menunjukkan bahwa akurasi tertinggi yang diperoleh adalah sebesar 90,90% pada Logistic Regression dengan kombinasi metode Grid Search dan Active Learning. Sementara itu, Logistic Regression dengan metode Active Learning menghasilkan akurasi sebesar 90,58%, dan dengan metode Grid Search sebesar 90,17%. Adapun Logistic Regression tanpa metode optimasi menghasilkan akurasi sebesar 89,81%.

Keywords / Kata Kunci — ***Sentiment Analysis; Logistic Regression; Grid Search; Active Learning***

CORRESPONDING AUTHOR

Muhammad Azka Faridi
Teknik Informatika, Fakultas Ilmu Komputer, Universitas Sriwijaya, Indonesia
Email: azkafaridi[at]gmail.com

1. PENDAHULUAN

Di era digital saat ini, opini dan ulasan yang diberikan oleh pengguna di berbagai platform online memiliki peran yang penting dalam membentuk pandangan publik terhadap suatu produk atau layanan. Salah satu platform

yang sering digunakan untuk memberikan ulasan terkait film adalah *IMDb*, yang bisa menjadi referensi bagi para penikmat film dalam menilai kualitas suatu tayangan film berdasarkan opini, baik yang bersifat positif maupun negatif. Ulasan dari pengguna tidak hanya memengaruhi keputusan individu dalam memilih film, tetapi juga menjadi salah satu indikator keberhasilan sebuah produksi film di industri hiburan. Oleh karena itu, analisis sentimen semakin banyak diterapkan untuk memahami pola opini masyarakat sehingga dapat dimanfaatkan dalam memprediksi potensi penjualan serta membantu investor dalam proses pengambilan keputusan [1].

Analisis sentimen merupakan teknik dalam pemrosesan bahasa alami (*Natural Language Processing*) yang bertujuan untuk mengklasifikasikan suatu kalimat atau dokumen, apakah bersifat positif atau negatif [2]. Selain itu, analisis sentimen juga dapat digunakan untuk mengenali serta memahami emosi yang terkandung dalam sebuah teks [3]. Analisis sentimen melibatkan penambangan teks untuk mengevaluasi, memproses, dan mengekstrak data teks. Proses ini dilanjutkan ke tahap prapemrosesan data, yang mencakup *case folding*, *tokenization*, penghapusan data duplikat dan *stopword*, normalisasi, *stemming*, serta klasifikasi sentimen [4].

Metode analisis sentimen berbasis *machine learning* telah banyak diterapkan sebagai pendekatan utama untuk mengklasifikasikan serta memahami pola sentimen dalam data teks. Salah satu algoritma *machine learning* yang sering digunakan adalah *Logistic Regression*. Algoritma ini memiliki keunggulan dalam menghasilkan prediksi probabilitas yang akurat serta dapat mengklasifikasikan data ke dalam kelas positif, negatif, atau netral. Selain itu, *Logistic Regression* sering digunakan dalam analisis sentimen karena kemampuannya dalam memberikan interpretasi yang jelas terhadap hubungan antar variabel [5].

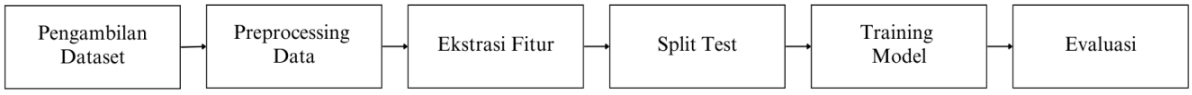
Penelitian ini juga menerapkan metode *Active Learning* dan *Grid Search*. *Grid Search* digunakan untuk mencari kombinasi *hyperparameter* terbaik dengan menguji berbagai kemungkinan nilai melalui proses *trial and error* hingga menemukan hasil yang optimal [6]. Sedangkan *Active Learning* digunakan untuk memilih sampel dengan tingkat ketidakpastian tertinggi dalam data uji, lalu menambahkannya ke dalam data latih secara bertahap, sehingga dapat mengurangi kebutuhan pelabelan tanpa mengorbankan akurasi [7].

Beberapa penelitian sebelumnya telah membuktikan bahwa *Logistic Regression* efektif dalam analisis sentimen. Misalnya, penelitian oleh Mola et al. menunjukkan bahwa metode ini mampu bersaing dengan beragam teknik berbasis *machine learning* dalam klasifikasi sentimen dan bahkan mengungguli beberapa di antaranya [8][9]. Selain itu, penelitian oleh Hagi dan Rarasati juga menunjukkan bahwa algoritma *Logistic Regression* memiliki kinerja yang baik dalam analisis sentimen pada aplikasi Sirekap [10].

Sejumlah penelitian yang sudah disampaikan di atas menunjukkan bahwa algoritma *Logistic Regression* memiliki efektivitas yang baik dalam analisis sentimen. Namun, optimisasi model masih diperlukan untuk meningkatkan kinerjanya, terutama dalam menangani data teks berskala besar. Oleh karena itu, penelitian ini bertujuan untuk mengklasifikasikan sentimen ulasan film pada *dataset IMDb 50K* dengan menggunakan algoritma *Logistic Regression* yang dioptimalkan dengan metode *Grid Search* dan *Active Learning*. Model yang dihasilkan akan dievaluasi menggunakan metrik evaluasi seperti akurasi, presisi, *recall*, dan *F1-score* guna menilai efektivitasnya dalam klasifikasi sentimen. Hasil penelitian ini diharapkan dapat memberikan wawasan lebih lanjut mengenai peran teknik optimasi dalam meningkatkan performa *Logistic Regression* dalam analisis sentimen

2. METODOLOGI PENELITIAN

Penelitian ini menerapkan pendekatan kuantitatif yang berfokus pada analisis sentimen dengan memanfaatkan algoritma *Logistic Regression*. Dalam upaya untuk meningkatkan kinerja model, dilakukan berbagai teknik pengolahan data serta optimasi menggunakan *Grid Search* dan *Active Learning*. Alur penelitian ini dapat dilihat pada Gambar 1.



GAMBAR 1. Alur Pemodelan

2.1. Pengambilan Data

Data yang digunakan dalam penelitian ini adalah kumpulan ulasan film pada situs *IMDb* yang telah dihimpun oleh Andrew Maas [11]. *Dataset* ini terdiri dari 50.000 ulasan film, yang terbagi menjadi 25.000 ulasan sentimen positif dan 25.000 ulasan sentimen negatif. *Dataset* disimpan dalam format CSV untuk mempermudah proses analisis sentimen pada tahap berikutnya.

2.2. Preprocessing Data

Tahap *preprocessing* dilakukan untuk membersihkan dan membuang data yang tidak berguna sebelum masuk ke dalam proses analisis lebih lanjut [12]. Langkah-langkah dalam proses ini meliputi *tokenization*, penghapusan karakter khusus, normalisasi teks, *lemmatization*, serta penghapusan *stopwords* dan duplikasi data. Proses ini merupakan langkah mendasar dalam tahap *preprocessing*.

2.3. Pembagian Data

Pada tahap ini, *dataset* akan dibagi menjadi dua bagian, yaitu data latih dan data uji. Data latih digunakan untuk membangun model klasifikasi *Logistic Regression*, sedangkan data uji digunakan untuk mengevaluasi kinerja model yang telah dilatih.

2.4. Ekstrasi Fitur

Ekstraksi fitur merupakan proses penting dalam klasifikasi teks yang bertujuan untuk mengubah teks yang tidak terstruktur menjadi representasi numerik dengan bobot tertentu sehingga dapat digunakan oleh algoritma *machine learning* [13]. Salah satu metode yang umum digunakan dalam ekstraksi fitur adalah *Term Frequency-Inverse Document Frequency* (*TF-IDF*). Metode ini digunakan untuk mengukur seberapa sering sebuah kata muncul dalam suatu dokumen serta kemunculannya di berbagai dokumen lain, sehingga dapat menentukan relevansi kata dalam suatu teks [14].

Untuk *mendapatkan* nilai *TF-IDF* pada Persamaan (4), terlebih dahulu perlu menghitung *term frequency* (*TF*) dengan Persamaan (1), *document frequency* (*DF*) dengan Persamaan (2), dan *inverse document frequency* (*IDF*) dengan Persamaan (3).

$$\Delta\, TF(t,d) = \frac{Number\ of\ times\ term\ t\ appears\ in\ document\ d}{Total\ number\ of\ terms\ in\ document\ d} \quad (1)$$

$$\Delta\, DF(t) = \frac{Number\ of\ documents\ that\ contain\ the\ term\ t}{Total\ number\ of\ documents\ in\ the\ corpus} \quad (2)$$

$$\Delta\, IDF(t) = log\, \frac{N}{DF(t)} \quad (3)$$

$$\Delta TF - IDF(t,d) = TF(t,d) \times IDF(t) \quad (4)$$

2.5. Klasifikasi Logistic Regression

Pada tahap klasifikasi, peneliti menggunakan algoritma *Logistic Regression* sebagai metode klasifikasi. Algoritma *Logistic Regression* digunakan untuk menganalisis hubungan antara variabel dependen dan satu atau lebih variabel independen bertipe *kategorikal* atau kontinu dalam kejadian dikotomis [15]. Dalam analisis sentimen, algoritma ini digunakan untuk menentukan hubungan antara dokumen sebagai variabel independen dan sentimen positif atau negatif sebagai variabel dependen, sehingga dokumen dapat diklasifikasikan sebagai sentimen positif atau negatif [16]. Persamaan *Logistic Regression* dapat dilihat pada Persamaan (5).

$$\Delta\, P(y) = \sigma(z)^y \times (1 - \sigma(z))^{1-y} \quad (5)$$

Selain itu, algoritma ini memanfaatkan fungsi *sigmoid* untuk memastikan bahwa output selalu berada dalam rentang 0 hingga 1, sehingga dapat diinterpretasikan sebagai probabilitas. Persamaan fungsi *sigmoid* dapat dilihat pada Persamaan (6).

$$\Delta\, \sigma(z) = \frac{1}{1+e^{-z}} \quad (6)$$

2.6. Optimasi Menggunakan Grid Search dan Active Learning

Untuk meningkatkan performa model, peneliti berencana melakukan optimasi menggunakan *Grid Search* dan *Active Learning*. *Grid Search* digunakan untuk mencari kombinasi terbaik dari *hyperparameter* yang memengaruhi kinerja model [17]. Kombinasi *hyperparameter* yang diperoleh setelah mengimplementasikan *Grid Search* akan diterapkan pada model. Kombinasi *hyper5parameter* dapat dilihat pada Tabel 1.Sementara itu, *Active Learning* diterapkan untuk meningkatkan efisiensi pemilihan data latih sehingga model dapat belajar dari sampel yang paling informatif.

TABEL 1. Kombinasi Grid Hyperparameter Pada Algorima Logistic Regression

Hyperparameter	Nilai
C	0.001, 0.1, 1, 10, 100
Penalty	‘l1’, ‘l2’
Solver	‘lbfgs’, ‘newton-cg’, ‘liblinear’, ‘sag’, ‘saga’

TABEL 2. Confussion Matrix

Actual Class	Predict Class	
	Positive	Negative
Positive	True Positive	False Negative
Negative	False Positive	True Negative

2.7. Evaluasi

Tahap akhir dalam proses *ini* adalah evaluasi kinerja model dengan menggunakan *Confusion Matrix*, seperti yang ditunjukkan pada Tabel 2, serta metrik seperti akurasi, presisi, *recall*, dan *F1-score* yang umum digunakan dalam klasifikasi teks [18]. Evaluasi ini bertujuan untuk mengukur efektivitas *Logistic Regression* dalam menganalisis sentimen serta menilai dampak dari optimasi yang telah dilakukan.

Precision dan *recall* adalah dua metrik evaluasi yang sering digunakan dalam klasifikasi teks. *Precision* menghitung rasio prediksi benar positif dibandingkan dengan keseluruhan hasil yang diprediksi positif. Sedangkan *recall* menghitung rasio prediksi benar positif dibandingkan dengan keseluruhan data yang seharusnya benar positif. Rumus untuk menghitung kedua metrik ini dapat dilihat pada Persamaan (7) dan (8).

$$\Delta\, Precision = \frac{TP}{TP+FP} \quad (7)$$

$$\Delta\, Recall = \frac{TP}{TP+FN} \quad (8)$$

Untuk mengatasi ketidakseimbangan saat *recall* tinggi namun *precision* rendah (dan sebaliknya), digunakan metrik *F1-score*. *F1-score* merupakan rata-rata harmonik dari *precision* dan *recall* yang telah dibobotkan. Rumus *F1-score* dapat dilihat pada Persamaan (9).

$$\Delta\, F1 - score = \frac{Recall \times Precision}{Recall + Precision} \quad (9)$$

Accuracy merupakan rasio antara jumlah prediksi benar (positif dan negatif) terhadap keseluruhan data. *Accuracy* dapat digunakan sebagai acuan kinerja algoritma jika jumlah kesalahan negatif dan kesalahan positif relatif *seimbang*. Namun, apabila tidak seimbang, maka disarankan menggunakan *F1-score* sebagai acuan utama [19]. Rumus *accuracy* dapat dilihat pada Persamaan (10).

$$\Delta\, Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (10)$$

3. Hasil Dan Pembahasan

3.1. Persiapan

Penelitian ini dimulai dengan mempersiapkan perangkat dan *dataset* sebagai objek utama. Perangkat yang digunakan dalam penelitian ini berupa laptop dengan prosesor Intel Core i3 1115G4, RAM 20GB, serta sistem operasi Windows 11 64-bit. Proses klasifikasi dilakukan menggunakan bahasa pemrograman *Python* dengan bantuan *library scikit-learn*. *Dataset* yang digunakan dalam penelitian ini terdiri dari 50.000 ulasan dari situs *IMDb*, dengan contoh *dataset* yang ditampilkan pada Tabel 3.

TABEL 3. Contoh Dataset

Review	Sentiment
One of the other reviewers has mentioned that ...	positive
A wonderful little production. The...	positive
I thought this was a wonderful way to spend ti..	positive
Basically there's a family where a little boy ...	negative
Petter Mattei's "Love in the Time of Money" is...	positive

3.2. Preprocessing Data

Dalam tahap ini, data teks akan diproses menjadi struktur kalimat yang lebih ringkas tanpa mengubah arti aslinya. Berikut contoh teks yang akan diproses dalam tahap *preprocessing: "The Karen Carpenter Story shows a little more about singer Karen Carpenter's complex life. Though it fails in giving accurate facts, and details.

Cynthia Gibb (portrays Karen) was not a fine election. She is a good actress, but plays a very naive and sort of dumb Karen Carpenter. I think that the role needed a stronger character. Someone with a stronger personality.

Louise Fletcher role as Agnes Carpenter is terrific, she does a great job as Karen's mother.

It has great songs, which could have been included in a soundtrack album. Unfortunately they weren't, though this movie was on the top of the ratings in USA and other several countries".* Hasil *preprocessing* dapat dilihat pada Tabel 4.

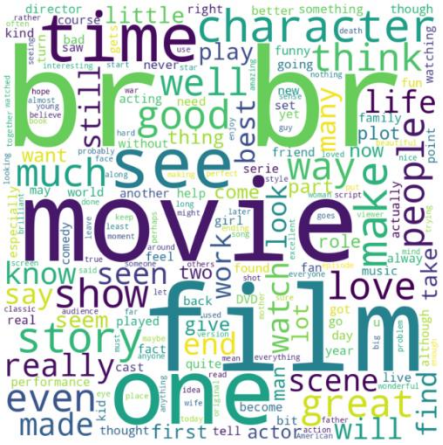
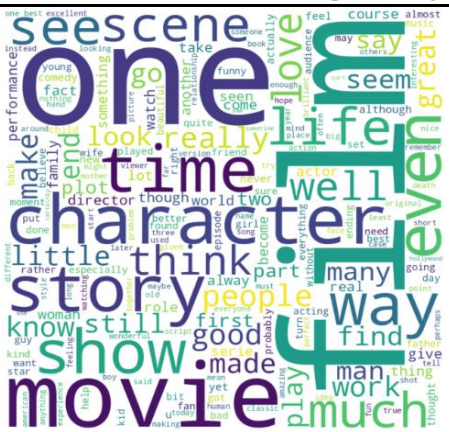
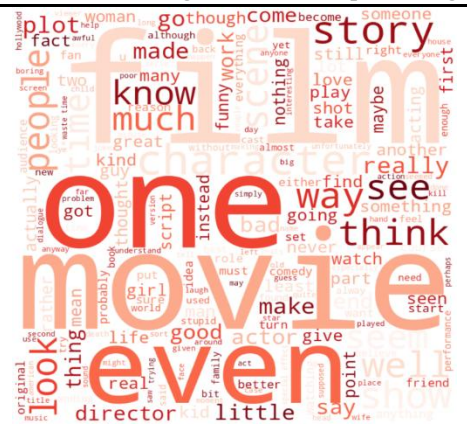
Dapat diamati bahwa proses *preprocessing* data mampu mengurangi ukuran data yang awalnya besar menjadi lebih ringkas dan bermakna. Sebelum dilakukan *preprocessing*, jumlah karakter dalam data tercatat sebanyak 594. Setelah melalui tahap *cleaning* dan *case folding*, jumlah karakter berkurang menjadi 553. Selanjutnya, pada tahap *stopword removal*, data berhasil dipangkas hingga menyisakan 438 karakter. Proses terakhir, yaitu *stemming*, lebih lanjut mereduksi jumlah karakter menjadi 426. Dengan demikian, *preprocessing* data—khususnya *lemmatization*—telah mengoptimalkan data dari 594 karakter menjadi 426 karakter yang siap digunakan dalam proses klasifikasi. Selain itu, terdapat perubahan pada kata-kata yang sering muncul dalam sentimen positif

maupun negatif setelah dilakukan *preprocessing*. Perubahan ini dapat dilihat pada Tabel 5, yang menampilkan kata-kata paling sering muncul dalam bentuk visualisasi menggunakan *library WordCloud*.

TABEL 4. Hasil Preprocessing

Tahap	Output
Cleaning dan Case Folding	the karen carpenter story shows a little more about singer karen carpenters complex life though it fails in giving accurate facts and details cynthia gibb portrays karen was not a fine election she is a good actress but plays a very naive and sort of dumb karen carpenter i think that the role needed a stronger character someone with a stronger personality louise fletcher role as agnes carpenter is terrific she does a great job as karens mother it has great songs which could have been included in a soundtrack album unfortunately they werent though this movie was on the top of the ratings in usa and other several countries
Stop Removal	karen carpenter story shows little singer karen carpenters complex life though fails giving accurate facts details cynthia gibb portrays karen fine election good actress plays naive sort dumb karen carpenter think role needed stronger character someone stronger personality louise fletcher role agnes carpenter terrific great job karens mother great songs could included soundtrack album unfortunately werent though movie top ratings usa several countries
Lemmatization	karen carpenter story show little singer karen carpenter complex life though fails giving accurate fact detail cynthia gibb portrays karen fine election good actress play naive sort dumb karen carpenter think role needed stronger character someone stronger personality louise fletcher role agnes carpenter terrific great job karen mother great song could included soundtrack album unfortunately werent though movie top rating usa several country

TABEL 5. WorldCloud Dataset

WorldCloud Positif Sebelum Preprocessing	WorldCloud Negatif Sebelum Preprocessing
	
WorldCloud Positif Setelah Preprocessing	**WorldCloud Negatif Setelah Preprocessing**
	

3.3. Pembagian Data

Setelah melalui tahap *preprocessing*, data dibagi menjadi dua bagian, yaitu data latih dan data uji. Dalam penelitian ini, *dataset* dibagi menjadi 70% untuk data latih dan 30% untuk data uji. Pembagian ini bertujuan agar model tidak hanya memahami pola yang terdapat dalam data latih, tetapi juga mampu menggeneralisasi dengan baik terhadap data baru yang belum pernah diproses [20].

3.4. Klasifikasi

Pada tahap ini, dilakukan eksperimen klasifikasi teks pada *dataset* menggunakan algoritma *Logistic Regression*. Empat metode yang digunakan adalah *Logistic Regression* standar, *Logistic Regression* dengan optimasi *Grid Search*, *Logistic Regression* dengan pendekatan *Active Learning*, serta kombinasi *Logistic Regression* dengan *Grid Search* dan *Active Learning*. Untuk model *Logistic Regression* dasar, digunakan parameter bawaan dari *library scikit-learn*, sedangkan pada metode dengan *Grid Search*, dilakukan pencarian *hyperparameter* optimal. Beberapa *hyperparameter* terbaik untuk model diperoleh dan rinciannya dapat dilihat pada Tabel 6.

TABEL 6. Kombinasi Grid Hyperparameter Pada Algorima Logistic Regression

Hyperparameter	Nilai
C	10
Penalty	‘l2’
Solver	‘liblinear’

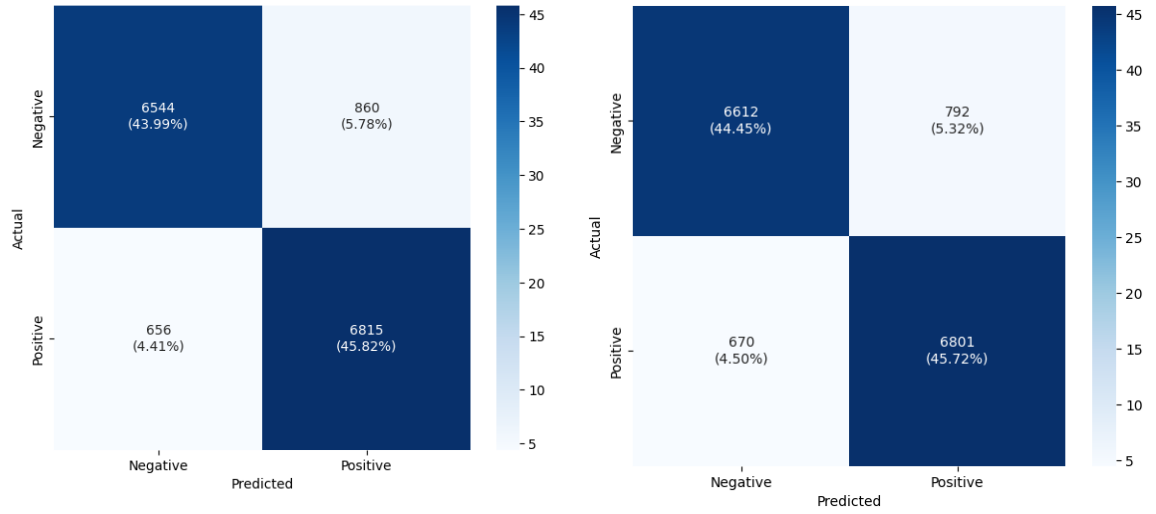
Metode *Active Learning* diterapkan dalam 250 iterasi untuk meningkatkan efisiensi dan efektivitas proses pembelajaran. Sementara itu, kombinasi *Grid Search* dan *Active Learning* digunakan untuk mendapatkan performa model yang lebih optimal.

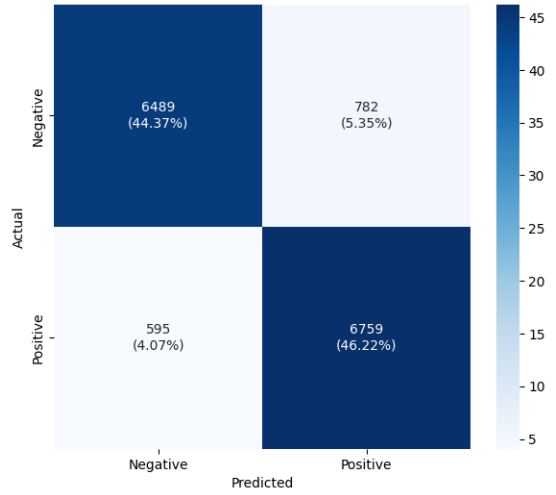
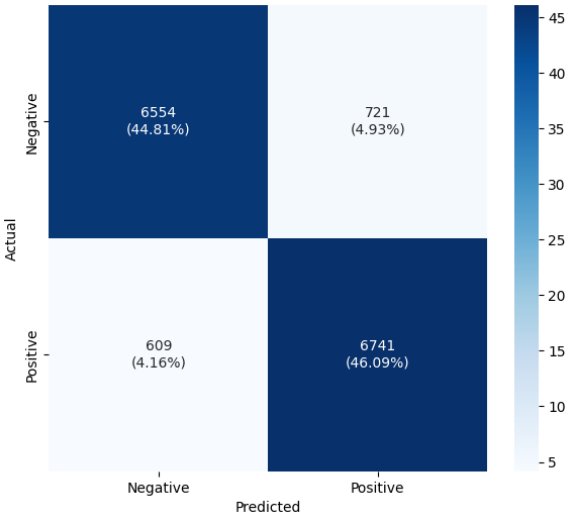
TABEL 7. Classification Report

Classification Report	Logistic Regression	Logistic Regression + Grid Search	Logistic Regression + Active Learning	Logistic Regression + Grid Search + Active Learning
Accuracy	0.89808	0.90171	0.90585	0.90906
Precision	0.89842	0.90184	0.90615	0.90918
Recall	0.89802	0.90168	0.90577	0.90902
F1-Score	0.89805	0.90171	0.90581	0.90904

TABEL 8. Confussion Matrix Model

Logistic Regression	Logistic Regression + Grid Search



Logistic Regression + Active Learning	**Logistic Regression + Grid Search + Active Learning**
	

Berdasarkan hasil yang ditampilkan pada Tabel 7, algoritma *Logistic Regression* standar memiliki akurasi terendah dibandingkan dengan metode lainnya. Peningkatan akurasi terjadi pada *Logistic Regression* dengan metode *Grid Search*, diikuti oleh *Logistic Regression* dengan metode *Active Learning*. Sementara itu, *Logistic Regression* dengan kombinasi *Grid Search* dan *Active Learning* menghasilkan akurasi tertinggi. Evaluasi performa model dilakukan berdasarkan nilai *accuracy*, *precision*, *recall*, dan *F1-score* yang diperoleh melalui *Confusion Matrix*. Setiap metode memiliki *Confusion Matrix* dengan nilai yang berbeda sesuai dengan tingkat keakuratan model. Untuk mempermudah visualisasi, eksperimen ini menggunakan *library* Seaborn dalam menampilkan diagram *Confusion Matrix*. Perbandingan *Confusion Matrix* dapat dilihat pada Tabel 8.

Dapat dilihat bahwa nilai dari *False Positive* berada di atas 5%, sementara *False Negative* berada di bawah 5%. Untuk *True Positive*, nilainya di atas 45%, sedangkan *True Negative* berada di bawah 45%. Namun, pada model *Logistic Regression* yang dikombinasikan dengan *Grid Search* dan *Active Learning*, baik nilai *False Positive* maupun *False Negative* berada di bawah 5%, sementara *True Positive* tetap di atas 45% dan *True Negative* di bawah 45%.

4. KESIMPULAN

Kesimpulan dari eksperimen ini menunjukkan bahwa klasifikasi menggunakan *Logistic Regression* dengan berbagai metode optimasi menghasilkan perbedaan akurasi yang tidak terlalu signifikan. Hasil klasifikasi menunjukkan bahwa akurasi tertinggi yang diperoleh adalah 90,90% pada *Logistic Regression* dengan kombinasi metode *Grid Search* dan *Active Learning*. Sementara itu, *Logistic Regression* dengan metode *Active Learning* menghasilkan akurasi sebesar 90,58%, dan *Logistic Regression* dengan *Grid Search* memiliki akurasi sebesar 90,17%. Adapun *Logistic Regression* tanpa optimasi memperoleh akurasi sebesar 89,81%.

Untuk penelitian selanjutnya, disarankan untuk mengeksplorasi algoritma lain seperti *Support Vector Machine*, *Random Forest*, atau model *deep learning* guna meningkatkan dan membandingkan performa dalam klasifikasi sentimen. Selain itu, penerapan teknik optimasi *hyperparameter* yang lebih canggih seperti *Bayesian Optimization* dapat membantu dalam mencari konfigurasi *hyperparameter* terbaik secara lebih efisien. Dalam hal ekstraksi fitur, disarankan memanfaatkan metode yang lebih modern seperti *word embeddings* (misalnya *Word2Vec*, *GloVe*, atau *BERT*) agar representasi teks menjadi lebih komprehensif dan mampu meningkatkan akurasi model secara signifikan

REFERENSI

[1] V. K. Singh, R. Piryani, A. Uddin, dan P. Waila, "Sentiment analysis of movie reviews: A new feature-based heuristic for aspect-level sentiment classification," 2013 International Multi-Conference on Automation, Computing, Communication, Control and Compressed Sensing (iMac4s), 2013, pp. 712-717.

[2] E. Haddi, X. Liu, dan Y. Shi, "The Role of Text Pre-processing in Sentiment Analysis," Procedia Computer Science, vol. 17, pp. 26–32, Dec. 2013.

[3] F. Mailoa, "Analisis sentimen data Twitter menggunakan metode text mining tentang masalah obesitas di Indonesia," Journal of Information Systems for Public Health, vol. 6, no. 1, pp. 44–51, 2021.

[4] D. Rifaldi, A. Fadlil, dan Herman, "Teknik Preprocessing Pada Text Mining Menggunakan Data Tweet 'Mental Health'," Decode: Jurnal Pendidikan Teknologi Informasi, vol. 3, pp. 161-171, Apr. 2023.

[5] S. A. Bahtiar, C. Dewa, and A. Luthfi, "Comparison of Naïve Bayes and Logistic Regression in Sentiment Analysis on Marketplace Reviews Using Rating-Based Labeling", journalisi, vol. 5, no. 3, pp. 915-927, Aug. 2023.

[6] I. Muhamad Malik Matin, "A Hyperparameter Tuning Using GridsearchCV on Random Forest for Malware Detection", JURNAL MULTIMEDIA NETWORKING INFORMATICS, vol. 9, no. 1, pp. 43–50, May 2023.

[7] A. I. Schein and L. H. Ungar, "Active learning for logistic regression: an evaluation," Machine Learning, vol. 68, no. 3, pp. 235–265, Oct. 2007.

[8] S. A. S. Mola, Y. C. Luttu, and D. N. Rumlaklak, "Perbandingan Metode Machine Learning dalam Analisis Sentimen Komentar Pengguna Aplikasi InDriver pada Dataset Tidak Seimbang," Jurnal Sistem Informasi Bisnis, vol. 14, no. 3, pp. 247-255, Aug. 2024.

[9] G. Aliman, N. Arago, and C. Dela Cruz, "Sentiment Analysis using Logistic Regression," Journal of Computational Innovations and Engineering Applications, vol. 7, no. 1, pp. 35–40, 2022.

[10] A. Hagi and D. B. Rarasati, "Sentiment Analysis of Sirekap Application Review Using Logistic Regression Algorithm," Jurnal Informatika, vol. 11, no. 2, pp. 55–64, Oct. 2024.

[11] A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts, "Learning word vectors for sentiment analysis," Proc. 49th Annu. Meet. Assoc. Comput. Linguist. Hum. Lang. Technol., pp. 142–150, 2011.

[12] R. Wahyudi and G. Kusumawardana, "Analisis Sentimen pada Aplikasi Grab di Google Play Store Menggunakan Support Vector Machine," Jurnal Informatika, vol. 8, pp. 200–207, Sep. 2021.

[13] I. Subagyo, L. D. Yulianto, W. Permadi, A. W. Dewantara, dan A. D. Hartanto, "Sentiment Analisis Review Film di IMDB Menggunakan Algoritma SVM," Jurnal Sistem Informasi dan Teknologi Informasi, vol. 8, no. 1, pp. 47–56, 2019.

[14] K. Lubis, T. AriBangsa, dan A. Yudertha, "Analisis Sentimen Opini Masyarakat terhadap Pindahnya Ibu Kota Indonesia dengan Menggunakan Klasifikasi Naïve Bayes," Jurnal Teknoinfo, vol. 18, no. 1, hlm. 226–238, Jan. 2024.

[15] E. R. Lidinillah, T. Rohana, and A. R. Juwita, "Analisis sentimen twitter terhadap steam menggunakan algoritma logistic regression dan support vector machine", tekno, vol. 10, no. 2, pp. 154-164, Jul. 2023.

[16] I. Rahmawati, T. Rika Fitriani, A. No'eman, dan A. Y. P. Yusuf, "Analisis Sentimen Menggunakan Algoritma Logistic Regression Pada Penerbangan Lion Air berdasarkan Ulasan Platform Online", JRITI, vol. 1, no. 1, hlm. 11–16, Agu 2023.

[17] G. M. Zakir, "Optimalisasi Hyperparameter pada Model Deteksi Transaksi Mencurigakan Menggunakan Grid-Search," e-Proceeding of Engineering, vol. 11, no. 6, pp. 6727-6732, Dec. 2024.

[18] D. D. Nur Cahyo, "Sentiment Analysis for IMDb Movie Review Using Support Vector Machine (SVM) Method", Inf. J. Ilm. Bid. Teknol. Inf. dan Komun., vol. 8, no. 2, pp. 90-95, Mar. 2023.

[19] R. Mardianto, Stefanie Quinevera, and S. Rochimah, "Perbandingan Metode Random Forest, Convolutional Neural Network, dan Support Vector Machine Untuk Klasifikasi Jenis Mangga", J. Appl. Comput. Sci. Technol., vol. 5, no. 1, pp. 63 - 71, May 2024.

[20] RIDWAN INDRANSYAH, Yulison Herry Chrisnanto, and Puspita Nurul Sabrina, "KLASIFIKASI SENTIMEN PERGELARAN MOTOGP DI INDONESIA MENGGUNAKAN ALGORITMA CORRELATED NAÏVE BAYES CLASIFIER", infotech, vol. 8, no. 2, pp. 60–66, Oct. 2022.