Vol. 5, No. 3, Desember 2024, hal. 626-634

Jurnal Computer Science and Information Technology (CoSciTech)

p-ISSN: 2723-567X
e-ISSN: 2723-5661

http://ejurnal.umri.ac.id/index.php/coscitech/index

# Pengaruh Agregasi Data pada Klasifikasi Sentimen untuk Dataset Terbatas Menggunakan SGD Classifier

**Fauzan Ray T[1], Surya Agustian*[2], Febi Yanto[3], Pizaini[4]**
Email: [1]11950110499@students.uin-suska.ac.id, [2]surya.agustian@uin-suska.ac.id, [3]febiyanto@uin-suska.ac.id, [4]pizaini@uin-suska.ac.id

[12345]Program Studi Teknin Informatika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Sultan Syarif Kasim

Diterima: 20 November 2024 | Direvisi: 20 Desember 2024 | Disetujui: 22 Desember 2024


**Abstrak**

Media sosial, khususnya Twitter atau X, adalah sumber data yang kaya untuk analisis sentimen. Namun, keterbatasan dataset menjadi tantangan utama dalam pemanfaatan machine learning, terutama untuk menghasilkan analisis sentimen yang cepat dan akurat. Penelitian ini menerapkan teknik agregasi data untuk memperluas dataset pelatihan serta menguji berbagai tahapan preprocessing, seperti cleaning, case folding, normalisasi, stemming, dan metode berbasis leksikon (lexicon-based). Metode klasifikasi yang digunakan adalah Stochastic Gradient Descent Classifier dengan representasi teks menggunakan model bahasa Fast Text untuk menghasilkan embedding kata. Lexicon-based preprocessing, khususnya untuk penanganan emoji dan emoticon, menunjukkan pengaruh signifikan saat data ditambahkan, karena mampu menangkap emosi dan konteks tambahan yang sering diabaikan dalam analisis teks konvensional. Hasil eksperimen menunjukkan bahwa penambahan data dan optimasi preprocessing meningkatkan F1 Score dari baseline 40% menjadi 52,13%, melampaui organizer yang mencapai 51,28%. Temuan ini menekankan pentingnya agregasi data, optimasi preprocessing, dan parameter tuning menggunakan grid search dalam meningkatkan kinerja model pada klasifikasi sentimen teks dengan dataset terbatas.

**Kata kunci**: *twitter, analisis sentimen, Agregasi data, Stochastic gradient descent, preprocessing*

## *Effect of Data Aggregation on Sentiment Classification for Limited Datasets Using SGD Classifier*

**Abtract**

*Social media, especially Twitter or X, is a rich source of data for sentiment analysis. However, dataset limitation is a major challenge in utilizing machine learning, especially to produce fast and accurate sentiment analysis. This research applies data aggregation techniques to expand the training dataset and tests various preprocessing steps, such as cleaning, case folding, normalization, stemming, and lexicon-based methods. The classification method used is Stochastic Gradient Descent Classifier with text representation using Fast Text language model to generate word embedding. Lexicon-based preprocessing, particularly for emoji and emoticon handling, shows significant impact when data is added, as it is able to capture additional emotion and context that is often overlooked in conventional text analysis. Experimental results show that data addition and preprocessing optimization improved F1 Score from a baseline of 40% to 52.13%, surpassing the organizer which reached 51.28%. These findings emphasize the importance of data aggregation, preprocessing optimization, and parameter tuning using grid search in improving model performance on text sentiment classification with limited datasets.*

**Keywords**: *twitter, sentiment analysis, data agregation, Stochastic gradient descent, preprocessing*

## 1. PENDAHULUAN

Dalam beberapa tahun terakhir, media sosial telah berkembang menjadi salah satu sumber data yang sangat kaya untuk analisis sentimen, terutama Twitter. Platform ini memberikan kesempatan bagi pengguna untuk mengungkapkan opini, pandangan, dan emosi mereka secara real-time dalam bentuk tweet yang singkat dan mudah diproses secara otomatis. Analisis sentimen bertujuan untuk mengidentifikasi dan mengklasifikasikan emosi atau opini yang terkandung dalam teks, hal ini telah menjadi area penelitian yang signifikan dalam bidang pemrosesan bahasa alami (Natural Language Processing). Pemanfaatan data twitter untuk analisis sentimen menarik perhatian berbagai bidang, mulai dari bisnis hingga politik, karena mampu mengungkapkan tren sosial dan opini publik secara cepat dan efektif. ketepatan dan kecepatan diperlukan untuk memenuhi kebutuhan pihak terkait untuk menganalisis sentimen terhadap berbagai masalah untuk menentukan dampak positif atau negatif dari sentimen tersebut. Sebagai contoh, dalam pemilihan pemimpin negara, tim pemenangan akan menggunakan media sosial untuk menilai persepsi dan popularitas calon yang diusung [1].

Analisis sentimen, sebagai salah satu teknik dalam pengolahan bahasa alami, bertujuan untuk mengidentifikasi dan mengkategorikan opini atau sentimen dari teks, sehingga dapat digunakan untuk menentukan apakah suatu teks memiliki sentimen positif, negatif, atau netral. Studi analisis sentimen lebih banyak berkonsentrasi pada studi kasus atau subjek daripada mengkategorikan sentimen berdasarkan kelas, seperti sentimen positif, negatif, dan netral [2]. Tantangan baru dalam klasifikasi sentimen yaitu terkait tentang keterbatasan sumber daya atau data yang digunakan, ada beberapa alasan mengapa sumber data mungkin terbatas salah satunya karena biaya yang terkait dengan pengumpulan data, jumlah sumber daya yang diperlukan untuk menilai dan mengevaluasi data, keterbatasan akses, atau masalah dalam mengumpulkan data yang relevan [3]. Penggunaan data eksternal atau teknik penambahan data menjadi upaya untuk mengatasi keterbatasan dataset dalam pelatihan model klasifikasi sentimen. Metode ini memungkinkan model untuk belajar dari kumpulan data yang lebih luas dan beragam, yang meningkatkan kinerja model ketika menggeneralisasi pola pada data uji. Metode ini meningkatkan kemampuan model untuk mempelajari lebih banyak informasi dengan memperluas cakupan data yang dapat dipelajari. Hal ini memungkinkan menghasilkan prediksi yang lebih akurat dan dapat diandalkan karena mengidentifikasi variasi dalam bahasa, konteks sosial, dan pola sentimen yang mungkin tidak terwakili dalam dataset awal.

Pada penelitian ini juga dilakukan pengujian menggunakan sistem shared task, seperti yang diterapkan dalam kompetisi HASOC 2023 [4], bertujuan untuk menarik minat para peneliti dan praktisi dalam mengembangkan metode klasifikasi sentimen mereka masing-masing. Sistem ini memungkinkan berbagai peniliti untuk berpartisipasi dalam kompetisi terbuka, di mana mereka menguji model dan teknik yang dioptimalkan menggunakan dataset yang sama. Salah satu kontribusi utama dari penelitian ini adalah penyediaan dataset tweet yang dapat dijadikan benchmark dalam klasifikasi sentimen. Dengan adanya leaderboard, metode yang diajukan oleh setiap peserta dapat dibandingkan secara langsung berdasarkan metrik evaluasi yang sama, seperti F1-score. Hal ini tidak hanya mendorong inovasi dalam pengembangan model yang lebih baik, tetapi juga memastikan adanya standar yang dapat digunakan untuk mengukur performa metode yang dioptimalkan.

Penelitian ini merupakan pengembangan lebih lanjut dari penelitian yang dilakukan pada shared task klasifikasi sentimen menggunakan dataset Kaesang. Dalam shared task ini, data yang terkait dengan topik "Kaesang" digunakan untuk pelatihan dan pengujian. Dengan mengadopsi metode baru dan membandingkan hasilnya melalui sistem leaderboard, penelitian ini berupaya untuk meningkatkan performa model klasifikasi sentimen, khususnya dalam mengatasi keterbatasan dataset dan meningkatkan F1-score sebagai ukuran utama performa model [5]. Huu-Thanh Duong juga melakukan penelitian serupa dengan meringkas teknik preprocessing untuk menormalkan data dan teknik penambahan data untuk menghasilkan data pelatihan baru dari data pelatihan asli yang terbatas[6]. Pada penelitian sebelumnya berjudul "Sentimen Analisis Aplikasi *E-Commerce* Berdasarkan Ulasan Pengguna Menggunakan Algoritma *Stochastic Gradient Descent*". Dalam penelitian ini, algoritma *Stochastic Gradient Descent* (SGD) digunakan sebagai metode klasifikasi dalam text mining untuk menganalisis yaitu apakah ulasan tersebut mengandung sentimen positif atau negatif. Hasilnya menunjukkan bahwa Tokopedia memiliki nilai *accuracy* sebesar 84%, dengan nilai *precision* 87% dan nilai *recall* 90%. Sementara itu, Shopee memiliki nilai *accuracy* sebesar 66%, dengan nilai *precision* 65% dan nilai *recall* 66%. Dengan demikian, penelitian ini berhasil mengklasifikasikan ulasan pengguna pada kedua platform e-commerce berdasarkan sentimen menggunakan metode *Stochastic Gradient Descent* [7] .

Melalui pendekatan ini, diharapkan metode yang diusulkan dapat memberikan hasil klasifikasi yang akurat meskipun jumlah data pelatihan terbatas. Selain itu, harapan lainnya adalah agar metode ini dapat diterapkan pada kasus klasifikasi sentimen lain dengan data yang telah dikumpulkan atau dilabeli sebelumnya, bahkan jika topik yang dianalisis mengalami perubahan. Dengan demikian, penelitian ini diharapkan dapat memberikan kontribusi dalam menghasilkan model yang lebih fleksibel dalam mengatasi data, seperti data yang tidak lengkap, bising (*noise*), atau berbeda dari data pelatihan yang asli, tanpa mengalami penurunan kinerja yang signifikan untuk berbagai aplikasi analisis sentimen

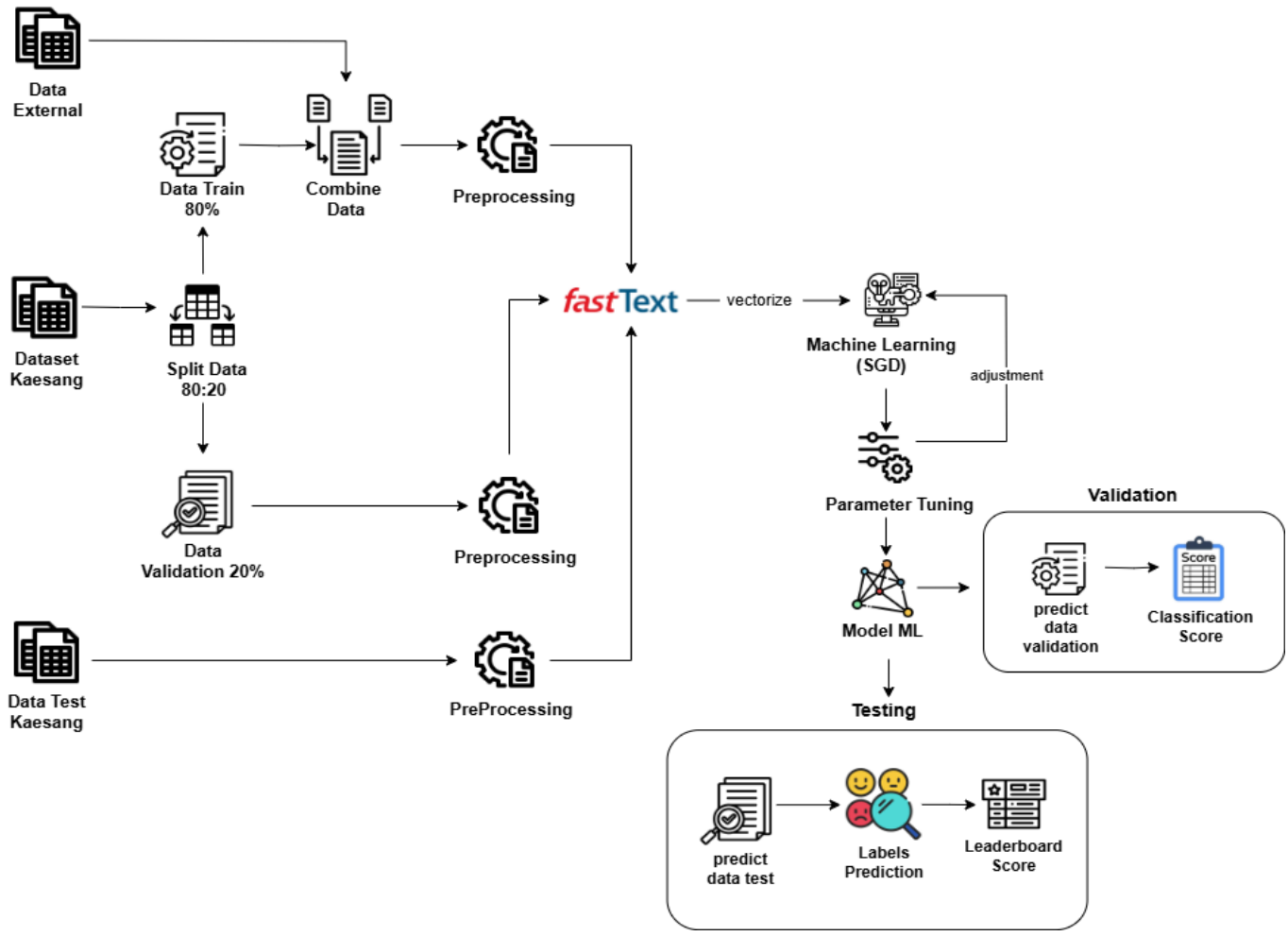
## 2. METODE PENELITIAN

Penelitian ini menggunakan metodologi yang terdiri dari beberapa tahap terstruktur untuk memastikan keakuratan dan keandalan hasil yang diperoleh. Tahap-tahap tersebut meliputi pengumpulan dataset, preprocessing teks, penerapan metode Stochastic Gradient Descent (SGD), penggunaan Fast Text sebagai teknik embedding teks, serta pengujian model. Setiap langkah dirancang

dengan teliti untuk mengatasi tantangan dalam analisis sentimen pada dataset yang terbatas, sekaligus menjamin performa optimal dari model yang dikembangkan.

### 2.1. Tahapan Penelitian

Tahapan penelitian terdiri dari beberapa sub-tahapan yang dimulai dengan pembagian dataset Kaesang , yang kemudian dibagi menjadi dua bagian, yaitu 80% data pelatihan dan 20% data validasi. Selain itu, dilakukan agregasi dengan data eksternal yang digabungkan dengan data pelatihan asli untuk memperkaya dataset. Kedua dataset tersebut kemudian masuk ke dalam tahapan *preprocessing*. Setelah *preprocessing*, data yang telah diproses akan diubah menjadi vektor melalui *Fast Text* untuk mendapatkan representasi vektor dari teks. Data vektor kemudian digunakan untuk melatih model menggunakan algoritma SGD (*Stochastic Gradient Descent*), dengan proses parameter tuning dilakukan untuk mengoptimalkan parameter. Model yang dihasilkan diuji dengan data validasi Kaesang, dan hasil evaluasi diperoleh dalam bentuk classification score berdasarkan metrik *F1-Score*. Model terbaik yang telah di-tuning akan digunakan untuk memprediksi label pada data test Kaesang yang belum memiliki label sentimen. Hasil prediksi selanjutnya akan ditampilkan dengan skor leaderboard untuk membandingkan performa prediksi terhadap data uji. Proses ini secara keseluruhan bertujuan untuk menghasilkan model klasifikasi sentimen yang optimal, dengan memanfaatkan agregasi data ,kombinasi preprocessing serta teknik vektorisasi berbasis Fast Text yang dipadukan dengan algoritma stochastic gradient descent (SGD).




Gambar 1. Alur Sistem Penelitian

### 2.2. Dataset

Table 1. Dataset Penelitian [3]

| No. | *Dataset* | Penggunaan | Jumlah | Distribusi Kelas | | |
|---|---|---|---|---|---|---|
| | | | | Positif | Netral | Negatif |
| 1 | Dataset Kaesang V1 | *Training* | 300 | 100 | 100 | 100 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 2 | Dataset Kaesang V2 | *Training* | 300 | 100 | 100 | 100 |
| 3 | Dataset Vaksinasi Covid-19 | *Training* | 8000 | 463 | 6664 | 873 |
| 4 | Dataset *Open Topic* | *Training* | 7569 | 1505 | 3408 | 2656 |
| 5 | Dataset Kaesang | *Testing* | 924 | - | - | - |

Dataset yang digunakan dalam penelitian ini diambil dari repositori *GitHub* yang disediakan oleh Agustian, yaitu *Small Dataset Sentiment Classification*[1] [3]. Dataset ini terdiri dari data yang terkait dengan klasifikasi sentimen,khususnya data yang berkaitan dengan topik kaesang dan juga data sentiment yang lainnya. Proses pengujian terhadap dataset tersebut dilakukan dengan menggunakan metode *shared task*, yang merupakan pendekatan kolaboratif untuk mengevaluasi performa model atau algoritma dalam suatu tugas yang dibagikan antar peneliti. Dengan pendekatan ini, hasil pengujian menjadi lebih transparan dan dapat dibandingkan dengan model-model lain yang menggunakan dataset serupa.

Pelabelan data merupakan salah satu tantangan utama dalam klasifikasi sentimen, terutama pada dataset berukuran besar. Oleh karena itu, metode *crowd-sourcing* digunakan pada pelabelan dataset penelitian ini yang dikelola langsung oleh penyedia data (*organizer*) yang digunakan untuk anotasi data tweet. Metode ini memungkinkan anotator memberikan label pada sebagian kecil data dengan hasil akhir ditentukan melalui *majority voting*. Pendekatan ini sejalan dengan temuan dalam penelitian terkait, yang menyoroti kompleksitas analisis sentimen serta efektivitas *crowd-sourcing* dengan *majority voting* dalam mengelola data dengan tingkat subjektivitas yang tinggi [8].

### 2.3. Preprocessing

Dataset yang digunakan harus melalui *preprocessing* sebelum diolah dan masuk ketahapan berikutnya[9], tahap *preprocessing* merupakan langkah penting dalam klasifikasi teks sentimen untuk memastikan bahwa data yang digunakan dapat diproses secara optimal oleh model. Dalam penelitian ini, beberapa teknik preprocessing diterapkan, meliputi cleaning, capitalize word, normalisasi, dan stemming. Proses cleaning dilakukan dengan menghapus karakter atau simbol yang tidak diperlukan, seperti tanda baca, angka, serta tautan yang tidak relevan dengan konteks sentimen. Capitalize word digunakan untuk mengonversi semua kata menjadi huruf kecil agar tidak terjadi perbedaan interpretasi berdasarkan huruf kapital. Selanjutnya, normalisasi diterapkan untuk menyamakan variasi kata yang memiliki arti serupa, misalnya, penggunaan kata "gak" menjadi "tidak". Teknik stemming juga diterapkan untuk mengubah kata menjadi bentuk dasarnya, sehingga mengurangi kompleksitas teks. Dalam upaya menemukan kombinasi preprocessing terbaik, penelitian ini melakukan berbagai eksperimen dengan mengaktifkan dan menonaktifkan sebagian langkah preprocessing, seperti hanya menggunakan cleaning dan normalisasi, atau menggabungkan semua tahap tersebut. Setiap kombinasi diuji untuk melihat pengaruhnya terhadap skor validasi, dengan tujuan menemukan tahapan preprocessing yang memberikan hasil terbaik dalam meningkatkan akurasi klasifikasi sentimen. Evaluasi dilakukan dengan memantau hasil pada data valid Komposisi preprocessing yang digunakan, ini mengacu pada pengalaman hasil optimal yang dicapai pada penelitian yang dilakukan sebelumya [10]. Berikut langkah – langkah dalam text processing

Table 2.Tahapan Preprocessing

| Pre Processing | Sebelum | Sesudah |
|---|---|---|
| Cleaning dan Case Folding | **@Dennysiregar7** saya setuju Kaesang masuk **@psi_id** karena **@PDI_Perjuangan TIDAK SEGERA MENDESAK** mensahkan **#RUU**perampasanAset **@Dennysiregar7** kalian pada diam mengenai #SuratPresiden ini | saya setuju kaesang masuk karena **tidak segera mendesak** mensahkan ruuperampasanaset kalian pada diam mengenai suratpresiden ini |
| Stopword Removal | PSI Semarang optimistis Kaesang **bisa** tarik anak muda berpolitik https://t.co/YAnXJki7bY | psi semarang optimistis kaesang tarik anak muda berpolitik |
| Normalisasi | @KompasTV Sbener e gk usah susah susah **kakean** polah mas kaesang...asal PSI dukung pak ganjar jdi presiden otomatis PSI bnyak yg pilih | @kompastv sbener e tidak usah susah susah **kebanyakan** polah mas kaesang...asal psi dukung pak ganjar jadi presiden otomatis psi bnyak yang pilih |
| Stemming | Siapa**pun** tdk asa yg bs tebak sekalipun **Peng**amat Politik spt apa Ketum baru PSI. Dlm **se**kejap akan jd Trend Setter di dunia **Per**politik**an**. Dan para Elit PSI akan kwb spt ini : Tidak akan **men**dukung anies itu kan pd | siapa tdk asa yg bs tebak sekalipun amat politik spt apa tum baru psi dlm kejap akan jd trend setter di dunia politik dan para elit psi akan kwb spt ini tidak akan dukung anies itu kan pd |

[1] https://github.com/s4gustian/Small_DataSet_Sentiment_Classification

| | | |
|---|---|---|
| | saat Grace &amp Giring yg kd Ketum, Tp Ketum kami saat ini ya Kaesang"" | saat grace amp giring yg kd tum tp tum kami saat ini ya kaesang |
| Lexicon Based (Emoji dan Emoticon | Harapannya min 50 RT aja kok **:)** Jangan lupa juga buat yg mau ikutan, link form akan ada jam 17.00 yaaa 💜 💜 Ayuk rayain ultah JIMIN di new normal event ini 😳 | harapan min 50 rt aja kok **senyum** jangan lupa juga buat yg mau ikut, link form akan ada jam 17.00 yaaa **hati ungu hati ungu** ayuk rayain ultah jimin di new normal event ini **wajah agak tersenyum** |
| Tokenisasi | Hasilnya memang cetar membahana. Tak menunggu lama, Erick Thohir langsung menggebrak dengan membenahi internal Kementerian BUMN, merombak direksi dan komisaris di sejumlah BUMN dengan sasaran pertamanya adalah PT Pertamina (Persero), | ["hasil", "cetar", "membahana", "tak", "menunggu", "lama", "erick", "thohir","menggebrak","membenahi", "internal", "kementerian", "bumn", "merombak", "direksi", "komisaris", "sejumlah","bumn","sasaran", "pertama","pt","pertamina","persero"] |

Emoji dan emotikon, seperti simbol senyum berperan penting dalam pengembangan leksikon sentimen dan peningkatan akurasi model analisis sentimen. Penelitian sebelumnya telah menunjukkan bahwa emotikon dan emoji berperan dalam membangun leksikon sentimen dan melatih pengklasifikasi untuk analisis sentimen [11]. Misalnya, emotikon senyum dapat memberikan konteks emosional yang tidak selalu tersampaikan melalui teks biasa. Integrasi simbol-simbol ini ke dalam analisis sentimen memungkinkan model untuk lebih sensitif terhadap nuansa emosional sehingga menghasilkan klasifikasi yang lebih akurat. Pada penelitian lain yang dilakukan oleh Anatoly Surikov menggunakan emoji dan emotikon mengalami kenaikan akurasi sebanyak 6% dari accuracy 85% menggunakan emotional indicator dan Word2vec menjadi 91% hal menunjukkan bahwa indikator emotional berarti sangat penting dan dapat mempengaruhi analisis sentimen [12].

### 2.4. Fast Text (word embedding)

Dalam penelitian ini, *fast text* digunakan sebagai model bahasa representasi teks menjadi vektor. *Fast text* merupakan kamus *open source* yang dikembangkan oleh Tim *Facebook Research Lab* sebagai metode yang efektif dan cepat dalam melakukan vektorisasi kata maupun klasifikasi teks [13][14]. Fitur ini memperluas model *word embedding* konvensional dengan mempertimbangkan *subword*, memungkinkan model untuk mengenali kata-kata yang jarang ditemui atau baru melalui komposisi *subword* yang menyusunnya. Fitur *Fast Text* menunjukkan keunggulan dalam menangani kosa kata yang kaya morfologi, di mana kata-kata dipecah menjadi n-gram karakter untuk menghasilkan embedding yang lebih kaya. Hal ini memungkinkan *Fast Text* untuk mengatasi masalah yang dihadapi, seperti keterbatasan dalam menangani kata-kata baru atau jarang [15]. Oleh karena itu, *Fast Text* dipilih sebagai metode yang tepat untuk mendukung penelitian ini.

### 2.5. Klasifikasi Stochastic Gradient Descent

Metode penelitian ini menggunakan library Scikit-learn[16] SGD Classifier (Stochastic Gradient Descent) sebagai algoritma klasifikasi untuk membangun model prediksi. Dalam penelitian ini, tuning parameter dilakukan pada dua parameter penting, yaitu alpha dan penalty. Alpha merupakan parameter regularisasi yang menentukan besarnya penalti yang dikenakan terhadap bobot model untuk mencegah overfitting. Nilai alpha yang lebih tinggi memberikan regularisasi yang lebih kuat, yang dapat membantu mengurangi varians model dengan risiko meningkatkan bias. Sedangkan penalty menentukan jenis regularisasi yang diterapkan. Penalty yang umum digunakan adalah L1 dan L2. Regularisasi L1 membuat beberapa koefisien menjadi nol, yang efektif dalam seleksi fitur, sedangkan regularisasi L2 mengecilkan semua koefisien secara seragam, menjaga stabilitas model tanpa menghilangkan fitur apapun. Rumus Stochastic Gradient Descent (SGD) [17], Algoritma ini memperbarui bobot model menggunakan rumus dasar sebagai berikut:

$$\boldsymbol{\theta = \theta - \eta \cdot \nabla_{\theta} J\left(\theta; x^{(i)}, y^{(i)}\right)} \quad (1)$$

di mana:

| | |
|---|---|
| $\boldsymbol{\theta}$ | : Menyatakan parameter model yang ingin kita optimalkan. |
| $\boldsymbol{\eta}$ | : Menyatakan learning rate, yang menentukan seberapa besar langkah yang diambil dalam setiap iterasi. |
| $\boldsymbol{\nabla_{\theta} J(\theta; x^{(i)}, y^{(i)})}$ | : Simbol ini menunjukkan gradien dari fungsi kerugian J terhadap parameter θ. Gradien ini mengukur seberapa besar perubahan pada parameter θ harus dilakukan untuk meminimalkan fungsi kerugian |
| $\boldsymbol{x^{(i)}\ dan\ y^{(i)}}$ | : $x^{(i)}$ Merupakan fitur input dari contoh contoh ke-i dalam dataset<br>$y^{(i)}$ Merupakan label output dari contoh ke-i dalam dataset. |

Rumus ini memungkinkan model untuk meminimalkan fungsi kerugian secara bertahap, dengan memilih sampel acak dari dataset. SGD sering kali menggunakan fungsi kerugian seperti hinge loss untuk SVM dan log loss untuk regresi logistik.

### 2.6. Parameter Tuning Menggunakan *Grid Search*

*Grid search* merupakan pendekatan sistematis dalam menemukan kombinasi parameter optimal untuk meningkatkan kinerja model pembelajaran mesin, teknik grid search akan mengeksekusi setiap kombinasi yang mungkin dari semua nilai hyperparameter ke dalam konfigurasi grid [18]. Dalam hal ini, tiga parameter utama yang disesuaikan melalui Grid Search adalah loss, alpha, dan penalty. Parameter loss menentukan fungsi kerugian yang digunakan oleh model untuk meminimalkan kesalahan prediksi, di mana opsi umum meliputi 'hinge'. Sementara itu, alpha berfungsi sebagai regulasi yang membantu model menghindari overfitting dengan menghukum bobot besar pada parameter. Penalty mengacu pada jenis regularisasi yang diterapkan, seperti L2 (Ridge), L1 (Lasso), atau elasticnet, yang menggabungkan keduanya.

Dengan menggunakan Grid Search, berbagai kombinasi dari nilai-nilai ini dieksplorasi secara menyeluruh untuk menemukan pengaturan parameter terbaik yang menghasilkan kinerja model tertinggi, performa setiap kombinasi dinilai dengan menggunakan set validasi yang telah ditetapkan pada set pelatihan [19]. Grid search kemudian menghasilkan konfigurasi yang memberikan kinerja terbaik selama proses validasi berdasarkan metrik evaluasi yang dipilih, seperti akurasi atau F1 Score.

Table 3. Parameter *Stochastic Gradient Descent*

| Parameter | Nilai yang diuji |
|---|---|
| Loss | Hinge (default) |
| Alpha | {‘0.0001’, ‘0.001’, ‘0.01’, ‘0.1’, ‘1’} |
| Pinalty | {'l2', 'l1', 'elasticnet'} |

### 2.7. Evaluasi

Evaluasi dalam pengukuran Confusion Matrix adalah pengujian yang memungkinkan Anda mencatat seberapa benar atau salah hasil prediksi suatu algoritma saat melakukan klasifikasi [20]. Pengukuran ini melibatkan beberapa metrik utama seperti accuracy, precision, recall, dan F1 Score. F1 Score digunakan sebagai acuan nilai utama karena merupakan gabungan dari precision dan recall, memberikan keseimbangan antara keduanya. Ini sangat berguna saat menghadapi dataset yang tidak seimbang, di mana satu kelas jauh lebih dominan daripada kelas lainnya. F1 Score memberikan penilaian yang lebih adil dengan mempertimbangkan kesalahan false positive dan false negative, sehingga lebih cocok untuk mengukur kinerja model dalam situasi yang kompleks.

$$\textit{f1-Score} = 2\ x\ \frac{\textit{Precission}\ x\ \textit{Recall}}{\textit{Precission}+\textit{Recall}} \qquad (2)$$

## 3. HASIL DAN PEMBAHASAN

### 3.1. Eksperimen Setup

Pada eksprimen ini data kaesang v1 dan kaesang v2 yang merupakan data train utama yang di split dengan perbandingan (80:20) menjadi data train dan validasi. Data validasi di pakai untuk melakukan pengecekan nilai f1 score untuk mencari nilai optimasi terbaik berdasarkan model optimasi yang dilakukan sebelum nantinya model diuji kedalam data testing menggunakan pengujian leaderboard.

Table 4. Hasil Penelitian Model Optimal

| No. | Agregasi Data | Tahapan Preprocessing | | | | Parameter Tuning | | | F1 Score (Validation) |
|---|---|---|---|---|---|---|---|---|---|
| | | Stopword | Stemming | Normalisasi | Lexicon | Loss | Alpha | Penalty | |
| 1 | Kaesang v1 : 300<br>Kaesang v2 : 300<br>Covid 19 : 0<br>Open Topik : 900 | Tidak | Tidak | Tidak | Tidak | Hinge | 0.01 | l1 | 57.50 % |
| 2 | Kaesang v1 : 300<br>Kaesang v2 : 300<br>Covid 19 : 900<br>Open Topik : 0 | Tidak | Tidak | Tidak | Tidak | Hinge | 0.01 | l1 | 56.53 % |
| 3 | Kaesang v1 : 300<br>Kaesang v2 : 300<br>Covid 19 : 0<br>Open Topik : 1200 | Ya | Tidak | Tidak | Tidak | Hinge | 0.1 | l2 | **58.94%** |
| 4 | Kaesang v1 : 300<br>Kaesang v2 : 300 | Tidak | Tidak | Ya | Ya | Hinge | 0.1 | l2 | 57.00 % |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | Covid 19 : 600<br>Open Topik : 1200 | | | | | | | | |
| 5 | Kaesang v1 : 300<br>Kaesang v2 : 300<br>Covid 19 : 600<br>Open Topik : 1200 | Tidak | Ya | Ya | Tidak | Hinge | 0.1 | l2 | 55.80 % |
| 6 | Kaesang v1 : 300<br>Kaesang v2 : 300<br>Covid 19 : 600<br>Open Topik : 1200 | Ya | Tidak | Tidak | Ya | Hinge | 0.1 | l2 | 54.76 % |
| 7 | Kaesang v1 : 300<br>Kaesang v2 : 300<br>Covid 19 : 600<br>Open Topik : 1200 | Ya | Ya | Tidak | Tidak | Hinge | 0.1 | l1 | **64.97%** |
| 8 | Kaesang v1 : 300<br>Kaesang v2 : 300<br>Covid 19 : 1200<br>Open Topik : 2700 | Ya | Ya | Ya | Tidak | Hinge | 0.01 | l2 | 54.33% |
| 9 | Kaesang v1 : 300<br>Kaesang v2 : 300<br>Covid 19 : 1200<br>Open Topik : 2700 | Ya | Ya | Ya | Ya | Hinge | 0.01 | l2 | **58.81%** |

Tahapan Optimasi pertama yaitu agregasi data dimana kita mencari komposisi dataset yang bisa kita gunakan sebagai data training yang kemudian model training akan digunakan untuk klasifikasi dari data validasi yang telah di split sebelumnya. Pengaturan kombinasi teknik preprocessing dalam model dengan mempertimbangkan beberapa tahapan utama yaitu penghapusan stopword, stemming, normalisasi, dan analisis emosional. Pada beberapa pengaturan, digunakan sebagian tahapan seperti hanya penghapusan stopword dan stemming, tanpa normalisasi atau analisis emosional. Sebaliknya, ada juga pengaturan di mana seluruh tahapan preprocessing diterapkan secara lengkap untuk memaksimalkan kualitas teks yang diolah. Tujuan dari variasi ini adalah untuk mengevaluasi dampak dari setiap kombinasi preprocessing terhadap analisis dan klasifikasi data. Kemudian langkah optimasi selanjutnya dilakukan parameter tuning dengan menggunakan grid search untuk mencari semua kemungkinan parameter yang diuji cobakan untuk mencari nilai parameter dengan hasil paling baik.

### 3.2. Hasil Pengujian Model Optimasi

Table 5. Pengujian Model Klasifikasi pada Leaderboard

| **Model** | **Pengujian Model Optimal** | | | |
|---|---|---|---|---|
| | **Jumlah Data Validasi** | **F1 Score (Validation)** | **Jumlah Data Testing** | **F1 Score (Testing)** |
| Run 1 (3) | 120 | 58.94% | 924 | 49.74% |
| Run 2 (7) | 120 | 64.97% | 924 | 49.41% |
| Run 3 (9) | 120 | 58.81% | 924 | 52.13% |

Berdasarkan tabel hasil pengujian, pengaruh penerapan teknik preprocessing dan penambahan dataset terhadap F1 Score validasi dan F1 Score testing. Pada Run 1, preprocessing yang diterapkan hanya berupa penghapusan stopword (Stopword), tanpa ada penerapan stemming, normalisasi, ataupun analisis emosional. Dataset yang digunakan pada Run 1 terdiri dari Kaesang v1 dengan jumlah 300 data dan Open Topik dengan jumlah 1200 data. Teknik preprocessing yang sederhana ini menghasilkan F1 Score validasi sebesar 58.94% dan F1 Score testing sebesar 49.74%, dengan menunjukkan kestabilan model. Pada Run 2, teknik preprocessing ditingkatkan dengan menambahkan stemming, tetapi tanpa normalisasi atau analisis emosional. Dataset yang digunakan pada Run 2 mencakup tambahan dataset Kaesang Final sebanyak 300 data , Covid 19 sebanyak 600 data dan open topic sebanyak 1200 data . Penambahan teknik preprocessing serta dataset ini meningkatkan F1 Score validasi menjadi 64.97%, namun F1 Score testing tetap berada di 49.41%. Perbedaan ini menunjukkan adanya *overfitting* karena model tidak dapat mempertahankan performa yang sama pada data uji. Pada Run 3, semua teknik preprocessing diterapkan, termasuk penghapusan stopword, stemming, normalisasi, dan analisis emosional. Dataset yang digunakan ditambah lagi dengan Open Topik menjadi 2700 data, serta mempertahankan dataset sebelumnya, yaitu Covid 19. Hasil pengujian menunjukkan F1 Score validasi sedikit menurun menjadi 58.81%, namun terdapat peningkatan pada F1 Score testing sebesar 52.13%.

### 3.3. Pengujian Shared Task

Data uji yang disediakan dalam shared task ini tidak dilengkapi dengan label. Proses penilaian dilakukan dengan mengunggah file prediksi ke sistem leaderboard, di mana skor evaluasi secara otomatis akan ditampilkan. Skor resmi yang digunakan dalam

shared task ini adalah F1-score, karena mempertimbangkan keseimbangan kelas dalam data uji. *F1-score* merupakan metrik evaluasi yang umum digunakan dalam berbagai penelitian *shared task*.

Table 6. Perbandingan Pengujian Leaderboard

| Peneliti | Metode | F1-Score | Accuracy | Precision | Recall |
|---|---|---|---|---|---|
| Safrizal [21] | SVM + Fasttext | 53.59% | 62.73% | 53.01% | 59.62% |
| **Penelitian ini** | SGD + Fast Text | **52.13%** | **59.96%** | **52.99%** | **63.76%** |
| Yoga El S. [5] | SVM + TF - IDF | 51.96% | 61.97% | 52.31% | 58.37% |
| Organizer [3] | SVM + TF - IDF | 51.28% | 61.21% | 52.89% | 57.22% |
| Admin [3] | Baseline | 40.38% | 45.45% | 49.53% | 48.80% |

Baseline dalam penelitian klasifikasi teks merujuk pada model sederhana yang berfungsi sebagai titik acuan awal untuk membandingkan performa model yang lebih kompleks. Model baseline ini memberikan gambaran tentang performa minimum yang diharapkan dan membantu menilai apakah model baru atau metode yang diusulkan dapat memberikan peningkatan signifikan. Dalam baseline yang dijalankan oleh Admin, yang hanya menggunakan data kaesang tanpa optimasi lebih lanjut, performa terlihat cukup rendah, dengan F1 Score sebesar 40.38%.

Selain melakukan pembahan data eksternal dalam proses klasifikasi, penelitian ini juga melakukan tahapan proses pada *preprocessing* , dan hal yang dilakukan untuk meninggkat hasil dari pengujian ini juga menggunakan parameter tuning dari metode yang digunakan untuk mendapatkan parameter dengan versi terbaik dengan menggunakan grid search. Pada perbandingan pengujian diatas menunjukkan berbagai pendekatan dalam sebuah leaderboard klasifikasi. Metode yang diuji melibatkan model yang diterapkan oleh berbagai peneliliti, termasuk pendekatan baseline.

Dalam penelitian ini, metode yang digunakan adalah kombinasi dari SGD (*Stochastic Gradient Descent*) dan *Fast Text*, yang dibandingkan dengan metode SVM (*Support Vector Machine*) menggunakan fitur TF-IDF sebagai pembanding utama dari *Organizer*, serta pendekatan *baseline* oleh Admin yang hanya menggunakan data dari Kaesang tanpa melakukan optimasi. Hasil pengujian menunjukkan bahwa model SGD + FastText dalam penelitian ini mencapai F1 Score (macro) sebesar 52.13% dan akurasi 59.96%, dengan nilai *precision* 52.99% dan *recall* 63.76%. Meskipun nilai akurasi sedikit lebih rendah dibandingkan model *Organizer* yang menggunakan SVM + TF-IDF (61.21%), model ini menunjukkan peningkatan pada aspek recall yang lebih tinggi sebesar 63.76%.

Hal ini mengindikasikan bahwa pendekatan FastText dalam penelitian ini berhasil mengidentifikasi lebih banyak sampel positif dengan benar, meskipun sedikit mengorbankan akurasi keseluruhan. Sebagai pembanding, metode yang digunakan oleh tim Organizer dengan SVM + TF-IDF mencapai F1 Score sebesar 51.28%, dengan precision yang sebanding namun recall yang lebih rendah dibandingkan penelitian ini. Hal ini menegaskan pentingnya penggunaan teknik optimasi dan dataset tambahan, seperti data terkait Covid dan Open Topic, untuk meningkatkan kinerja model klasifikasi. Secara keseluruhan, penelitian ini menunjukkan pendekatan yang lebih baik pada beberapa metrik dibandingkan dengan baseline.

## 4. KESIMPULAN

Penelitian ini, menunjukkan bahwa teknik agregasi data dapat secara efektif menghasilkan data pelatihan baru dari dataset asli yang terbatas. Melalui teknik ini, penelitian berhasil meningkatkan hasil *f1-score* dengan menambahkan variasi dataset. Selain itu, penelitian ini juga menerapkan berbagai teknik optimasi lainnya, seperti kombinasi *preprocessing* dan pengaturan komposisi data eksternal di luar topik data uji, yang kemudian diuji menggunakan metode *grid search* untuk menemukan parameter terbaik. Hasil eksperimen menunjukkan peningkatan yang signifikan dalam performa klasifikasi. Pada baseline, model hanya mencapai akurasi sebesar 40%, namun dengan optimasi teknik-teknik tersebut, akurasi meningkat hingga mencapai 52.13%, melewati *f1-score* yang dicapai oleh pihak organizer 51.28% menggunakan metode SGD *Classifier*. Peningkatan ini menunjukkan pentingnya strategi optimasi yang diterapkan untuk mencapai hasil yang lebih baik dalam tugas klasifikasi sentimen pada dataset terbatas.

## DAFTAR PUSTAKA

[1] M. Yusrizal and T. B. Sasongko, "Analisis Sentimen Masyarakat Terhadap Presiden dan Calon Presiden Terpilih 2024 Menggunakan Naïve Bayes," *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 8, no. 3, p. 1673, Jul. 2024, doi: 10.30865/mib.v8i3.7882.

[2] I. H. Hasibuan, E. Budianita, S. Agustian, and P. Pizaini, "Klasifikasi Sentimen Komentar Youtube Tentang Pembatalan Indonesia Sebagai Tuan Rumah Piala Dunia U-20 Menggunakan Algoritma Naïve Bayes Classifer," *Jurnal Sistem Komputer dan Informatika (JSON)*, vol. 5, no. 2, p. 249, Dec. 2023, doi: 10.30865/json.v5i2.7096.

[3] S. Agustian *et al.*, "New Directions in Text Classification Research: Maximizing The Performance of Sentiment Classification from Limited Data Arah Baru Penelitian Klasifikasi Teks: Memaksimalkan Kinerja Klasifikasi Sentimen dari Data Terbatas," 2024. [Online]. Available: https://github.com/s4gustian/Small_DataSet_Sentiment_Classification

[4] N. Narayan, M. Biswal, P. Goyal, and A. Panigrahi, "Hate Speech and Offensive Content Detection in Indo-Aryan Languages: A Battle of LSTM and Transformers," Dec. 2023.

[5] Y. El Saputra, S. Agustian, and S. Ramadhani, "KLIK: Kajian Ilmiah Informatika dan Komputer Klasifikasi Sentimen SVM Dengan Dataset yang Kecil Pada Kasus Kaesang Sebagai Ketua Umum PSI," *Media Online)*, vol. 4, no. 6, pp. 2902–2908, 2024, doi: 10.30865/klik.v4i6.1944.

[6] H.-T. Duong and T.-A. Nguyen-Thi, "A review: preprocessing techniques and data augmentation for sentiment analysis," *Comput Soc Netw*, vol. 8, no. 1, p. 1, Dec. 2021, doi: 10.1186/s40649-020-00080-x.

[7] K. Kenyon-Dean *et al.*, "Sentiment Analysis: It's Complicated!," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2018, pp. 1886–1895. doi: 10.18653/v1/N18-1171.

[8] M. Ihsan, Benny Sukma Negara, and Surya Agustian, "LSTM (Long Short Term Memory) for Sentiment COVID-19 Vaccine Classification on Twitter," *Digital Zone: Jurnal Teknologi Informasi dan Komunikasi*, vol. 13, no. 1, pp. 79–89, May 2022, doi: 10.31849/digitalzone.v13i1.9950.

[9] A. Permana, S. S.-J. C. (Computer Science, and undefined 2023, "Perbandingan algoritma k-nearst neighbor dan naïve bayes pada aplikasi shopee," *ejurnal.umri.ac.idAO Permana, S SaepudinJurnal CoSciTech (Computer Science and Information Technology), 2023•ejurnal.umri.ac.id*, Accessed: Nov. 19, 2024. [Online]. Available: https://ejurnal.umri.ac.id/index.php/coscitech/article/view/4474

[10] P. Yohana, S. Agustian, and S. Kurnia Gusti, "Klasifikasi Sentimen Masyarakat terhadap Kebijakan Vaksin Covid-19 pada Twitter dengan Imbalance Classes Menggunakan Naive Bayes," *SNTIKI : Seminar Nasional Teknologi Informasi Komunikasi dan Industri*, vol. 26, pp. 69–80, Oct. 2022, [Online]. Available: https://lp2m.unmul.ac.id/webadmin/public/upload/files/9584b64517cfe308eb6b115847cbe8e7.pdf

[11] M. Fernández-Gavilanes, J. Juncal-Martínez, S. García-Méndez, E. Costa-Montenegro, and F. J. González-Castaño, "Creating emoji lexica from unsupervised sentiment analysis of their descriptions," *Expert Syst Appl*, vol. 103, pp. 74–91, 2018, doi: https://doi.org/10.1016/j.eswa.2018.02.043.

[12] A. Surikov and E. Egorova, "Alternative method sentiment analysis using emojis and emoticons," *Procedia Comput Sci*, vol. 178, pp. 182–193, 2020, doi: https://doi.org/10.1016/j.procs.2020.11.020.

[13] A. Joulin, E. Grave, P. Bojanowski, and T. Mikolov, "Bag of Tricks for Efficient Text Classification," Jul. 2016.

[14] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching Word Vectors with Subword Information," *Trans Assoc Comput Linguist*, vol. 5, pp. 135–146, Dec. 2017, doi: 10.1162/tacl_a_00051.

[15] S. F. Sabbeh and H. A. Fasihuddin, "A Comparative Analysis of Word Embedding and Deep Learning for Arabic Sentiment Classification," *Electronics (Basel)*, vol. 12, no. 6, p. 1425, Mar. 2023, doi: 10.3390/electronics12061425.

[16] F. Pedregosa FABIANPEDREGOSA *et al.*, "Scikit-learn: Machine Learning in Python Gaël Varoquaux Bertrand Thirion Vincent Dubourg Alexandre Passos PEDREGOSA, VAROQUAUX, GRAMFORT ET AL. Matthieu Perrot," 2011. [Online]. Available: http://scikit-learn.sourceforge.net.

[17] S. Ruder, "An overview of gradient descent optimization algorithms," Sep. 2016.

[18] D. M. Belete and M. D. Huchaiah, "Grid search in hyperparameter optimization of machine learning models for prediction of HIV/AIDS test results," *International Journal of Computers and Applications*, vol. 44, no. 9, pp. 875–886, Sep. 2022, doi: 10.1080/1206212X.2021.1974663.

[19] Y. N. Fuadah, M. A. Pramudito, and K. M. Lim, "An Optimal Approach for Heart Sound Classification Using Grid Search in Hyperparameter Optimization of Machine Learning," *Bioengineering*, vol. 10, no. 1, p. 45, Dec. 2022, doi: 10.3390/bioengineering10010045.

[20] R. Firdaus, J. Satria, B. B.-J. C. (Computer, and undefined 2022, "Klasifikasi Jenis Kelamin Berdasarkan Gambar Mata Menggunakan Algoritma Convolutional Neural Network (CNN)," *ejurnal.umri.ac.id*, Accessed: Nov. 19, 2024. [Online]. Available: https://ejurnal.umri.ac.id/index.php/coscitech/article/view/4360

[21] S. Agustian and A. Nazir, "Klasifikasi Sentimen Terhadap Pengangkatan Kaesang Sebagai Ketua Umum Partai PSI Menggunakan Metode Support Vector Machine," *Technology and Science (BITS)*, vol. 6, no. 1, 2024, doi: 10.47065/bits.v6i1.5340.