Analisis Sentimen pada X Terhadap Pilkada 2024 Menggunakan Ekspansi Fitur FastText dan CNN dengan Optimasi Bat Algorithm

Dzaki Afin Firdaus
Fakultas Informatika
Universitas Telkom, Bandung
afinfirdaus@students.telkomuniversity.ac.id

Erwin Budi Setiawan
Fakultas Informatika
Universitas Telkom, Bandung
erwinbudisetiawan@telkomuniversity.ac.id

Abstrak

Pilkada 2024 merupakan momentum penting dalam demokrasi Indonesia yang akan menentukan arah pembangunan daerah. Dalam konteks ini, analisis sentimen dapat menjadi alat yang efektif untuk memahami opini publik terhadap calon pemimpin dan isu-isu yang berkaitan dengan Pilkada. Penelitian ini bertujuan untuk mengimplementasikan sistem analisis sentimen menggunakan Convolutional Neural Network (CNN) yang dioptimalkan dengan Bat Algorithm dan ekspansi fitur menggunakan FastText. Metode ini diterapkan pada data tweet berbahasa Indonesia yang dikumpulkan selama periode Pilkada 2024. Hasil evaluasi menunjukkan bahwa akurasi tertinggi diperoleh dengan menggunakan max feature sebesar 15.000 (73,01%), konfigurasi Uni-Bigram (73,30%), dan ekspansi fitur menggunakan FastText dengan korpus Tweet + IndoNews pada Top 1 (73,82%). Optimasi menggunakan Bat Algorithm memberikan peningkatan sebesar 0,05% (73,82% menjadi 73,87%), yang menunjukkan bahwa FastText secara signifikan meningkatkan akurasi model. Bat Algorithm terbukti efektif dalam mengoptimalkan parameter model dan memberikan kontribusi positif dalam peningkatan kinerja. Penelitian ini menunjukkan bahwa penggunaan FastText dapat memperbaiki akurasi model analisis sentimen, sementara Bat Algorithm juga memberikan kontribusi yang berharga dalam optimasi model.

Kata kunci: analisis sentimen, CNN, bat algorithm, fasttext, pilkada 2024, optimasi

Abstract

The 2024 Regional Elections (Pilkada) is a significant moment in Indonesia's democracy that will determine the direction of regional development. In this context, sentiment analysis can serve as an effective tool to understand public opinion towards the candidates and issues related to the Pilkada. This study aims to implement a sentiment analysis system using Convolutional Neural Network (CNN) optimized with Bat Algorithm and feature expansion using FastText. This method was applied to Indonesian-language tweet data collected during the 2024 Pilkada period. The evaluation results show that the highest accuracy was achieved using a max feature of 15,000 (73.01%), a Uni-Bigram configuration (73.30%), and feature expansion using FastText with the Tweet + IndoNews corpus at Top 1 (73.82%). Optimization with Bat Algorithm resulted in a 0.05% improvement (from 73.82% to 73.87%), indicating that FastText significantly improved the model's accuracy. Bat Algorithm proved effective in optimizing the model's parameters and made a positive contribution to performance improvement. This study demonstrates that the use of FastText can significantly improve the accuracy of sentiment analysis models, while Bat Algorithm also provides valuable contributions to model optimization.

Keywords: sentiment analysis, CNN, bat algorithm, fasttext, pilkada 2024, optimization

1. PENDAHULUAN

Latar Belakang

Pemilihan Kepala Daerah (Pilkada) adalah proses di mana masyarakat memilih secara langsung gubernur, bupati, dan/atau walikota, yang telah diatur dalam Undang-Undang Nomor 32 tahun 2004 tentang pemerintahan daerah. Proses pemilihan ini menunjukkan pentingnya demokrasi langsung, di mana rakyat memiliki hak untuk menentukan pemimpinnya. Namun, birokrasi yang buruk dapat menghambat jalannya proses demokrasi, bahkan mengancam hak tersebut, karena birokrasi memegang peranan strategis dalam pengambilan keputusan dan pelayanan publik yang mendukung proses pemilihan tersebut. Dengan demikian, keberhasilan Pilkada sangat bergantung pada kualitas birokrasi yang ada dalam pemerintahan [1].

Pilkada merupakan salah satu momen penting dalam demokrasi Indonesia, hasil Pilkada akan menentukan arah pembangunan dan kesejahteraan masyarakat di daerah selama lima tahun kedepan[19]. Setiap calon berhak memiliki tim kampanye yang bertugas mempromosikan visi misi, program, dan janji kepada masyarakat[19]. Oleh karena itu penting bagi pemangku kepentingan untuk memahami opini publik terhadap Pilkada 2024 agar dapat mengambil keputusan yang tepat. Namun, untuk dapat memilih pemimpin yang tepat masyarakat memerlukan informasi yang akurat dan komprehensif mengenai para kandidat[20].

Opini publik yang beredar di sosial media khususnya X, dapat menjadi informasi yang berharga [2]. Dengan mengetahui sentimen publik yang beredar di media sosial kita dapat mengetahui apa saja yang menjadi perhatian masyarakat, rekam jejak para kandidat, dan harapan masyarakat terhadap calon pemimpinnya. Analisis

sentimen dapat membantu untuk mengidentifikasi isu-isu penting yang menjadi perhatian masyarakat, informasi ini dapat digunakan untuk meningkatkan partisipasi pemilih dan meningkatkan kualitas demokrasi[21].

Dalam penelitian sebelumnya yang dilakukan oleh Haque dkk yang membandingkan beberapa metode Neural Network, CNN menunjukan kinerja yang signifikan dibandingkan dengan LSTM dan LSTM-CNN untuk analisis sentimen[3]. Untuk memprediksi sentimen positif, negatif, dan netral, langkah pertama adalah mengidentifikasi sentimen tersebut, yang dilakukan menggunakan CNN[3]. Menganalisis sentimen dalam data X dibatasi oleh adanya teks tidak terstruktur, namun pendekatan ektraksi fitur *Term Frequency Inverse Document Frequency* (TF-IDF) digunakan untuk memberi bobot pada setiap kata dan representasi setiap kata[22].

Dalam pemrosesan bahasa alami, vektor kata digunakan untuk merepresentasikan kata dalam format padat. Representasi ini biasanya disebut *word embeddings*, meskipun tidak tergolong baru *word embeddings* populer sejak google merilis Word2vec pada 2013 kemudian muncul *word embeddings* lain seperti glove dan fasttext[16]. CNN tidak dapat menerima masukan bahasa alami manusia secara langsung oleh karena itu fasttext memproses bahasa alami menjadi vektor yang akan diteruskan kedalam program[4]. Penelitian lain yang dilakukan oleh Chalkidis dkk [5] untuk meneliti adaptasi awal analisis hukum menggunakan *deep learning* dan *word embeddings*, menyimpulkan bahwa fasttext dapat menangani out-of-vocabulary (OOV) dan kata-kata langka yang tidak dapat diatasi Word2vec dan glove.

Penelitian yang telah dilakukan oleh RA Rudiyanto dkk [6] menggunakan CNN dan *Particle Swarm Optimization* untuk menganalisis sentimen menyatakan bahwa kinerja akurasi sering mengalami penurunan ketika berhadapan dengan data yang ambigu, kata-kata yang sulit dikenali, dan kondisi serupa lainnya. Oleh karena itu, diperlukan perpaduan algoritma yang optimal[6]. Penggunaan PSO tidak hanya meningkatkan akurasi dan nilai skor f1 tetapi juga menunjukan perkembangan yang signifikan dibandingkan dengan metode lain yaitu CNN, TF-IDF, dan Word2vec. Metode *bat algorithm* yang mempunyai kemampuan ekolokasi kelelawar dapat membantu CNN untuk mengoptimalkan hyperparameter CNN , memilih fitur yang relevan, dan mengoptimalkan arsitektur CNN untuk meningkatkan kinerja klasifikasi sentimen[7].

Penelitian sebelumnya yang dilakukan oleh Jahan dkk mengevaluasi lima algoritma machine learning pada dataset yang terdiri dari 100.000 tweet untuk analisis sentimen, menunjukkan bahwa Logistic Regression dan Support Vector Machines (SVM) meraih akurasi tertinggi (86.22%), diikuti oleh Random Forest (82.59%)[23]. Sementara itu, Naive Bayes dan Gradient Boosting memiliki akurasi lebih rendah (masing-masing 70.45% dan 69.96%). Semua model menghadapi kesulitan dalam mengklasifikasikan sentimen negatif, yang menunjukkan tantangan dalam menangkap nuansa emosional pada data media sosial, dan mendorong pengembangan lebih lanjut untuk meningkatkan akurasi analisis sentimen[23].

Dalam penelitian yang dilakukan oleh Reddy dkk *Bat Algorithm* (BAT) digunakan untuk mengoptimalkan *Economic Load Dispatch* (ELD) dalam sistem tenaga listrik[24]. ELD adalah proses yang krusial untuk memastikan operasi pembangkitan tenaga yang efisien dan kost-efektif, dengan tujuan untuk mengurangi biaya operasional dan mendistribusikan beban secara optimal di antara beberapa generator dalam sistem tenaga[24]. BAT, yang terinspirasi oleh perilaku kelelawar dalam mencari mangsa, diterapkan untuk menemukan solusi optimal dalam masalah distribusi beban ini. Dengan demikian, BAT digunakan untuk mengoptimasi sistem tenaga listrik agar lebih efisien, mengurangi biaya operasional, dan meningkatkan kinerja pembangkitan tenaga secara keseluruhan.

Penelitian ini bertujuan untuk memperbaiki dan meningkatkan hasil penelitian sebelumnya dengan mengatasi masalah data ambigu, kata-kata sulit dikenali, dan optimasi parameter melalui model CNN yang digabungkan dengan word embedding FastText dan Bat Algorithm Optimization, guna mencapai akurasi lebih tinggi. Meskipun analisis sentimen Pilkada telah banyak dikembangkan, beberapa aspek seperti optimasi hyperparameter dan penanganan kata jarang muncul masih belum optimal. Oleh karena itu, penelitian ini berfokus pada implementasi sistem analisis sentimen Pilkada 2024 yang lebih akurat dengan memanfaatkan deep learning dan algoritma Bat yang diharapkan dapat meningkatkan kinerja model.

Topik dan Batasannya

Penelitian ini mengimplementasikan sistem analisis sentimen untuk Pilkada 2024 di Indonesia menggunakan Convolutional Neural Network (CNN), ekspansi fitur FastText, dan *Bat Algorithm* sebagai optimasi. Analisis dilakukan pada tweet berbahasa Indonesia dari media sosial X selama periode Pilkada, sementara tweet di luar periode atau berbahasa lain tidak dipertimbangkan.

Tujuan

Tujuan utama dari penelitian ini adalah untuk mengimplementasikan analisis sentimen pada data media sosial X menggunakan metode Convolutional Neural Networks (CNN) yang dioptimalkan dengan *Bat Algorithm* dan Ekspansi Fitur melalui FastText. Penelitian ini juga bertujuan untuk mengukur nilai akurasi implementasi FastText, serta sebelum dan sesudah optimasi dengan *Bat Algorithm*.

2. KAJIAN TEORI

Penelitian ini mengacu pada studi sebelumnya yang mempertimbangkan objek, metode, hasil, dan kesimpulan. Salah satunya adalah penelitian oleh Ibraham Kaibi dkk [8], yang membandingkan tiga model word embeddings (Glove, Word2vec, dan FastText) yang digabungkan dengan enam algoritma Machine Learning (Gaussian-Naïve Bayes, Linear SVC, NuSVC, Logistic Regression, SGD, dan Random Forest) pada dataset X berbahasa Arab. Hasil percobaan menunjukkan bahwa FastText

menghasilkan performa lebih baik dibandingkan Glove atau Word2vec.

Penelitian oleh Haque M dkk [3] membandingkan arsitektur CNN, LSTM, dan LSTM-CNN untuk klasifikasi sentimen pada ulasan film di platform IMDb. Hasilnya menunjukkan CNN mencapai akurasi tertinggi 91%, unggul 2% dibandingkan LSTM dan 1% dibandingkan LSTM-CNN. Studi ini merekomendasikan penggunaan CNN untuk pemrosesan bahasa alami dalam penelitian selanjutnya.

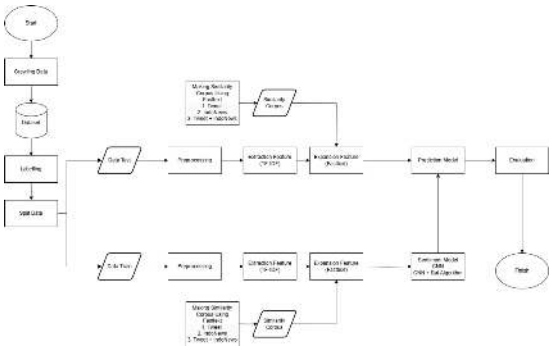
Penelitian oleh T Darwassh Hanawy Hussein dkk [7] mengaplikasikan Bat Algorithm dan Convolutional Neural Network untuk merencanakan perjalanan ambulans di sistem kota cerdas, dengan tujuan mentransfer pasien secara rahasia, akurat, dan cepat. Bat Algorithm digunakan untuk mencari jalur efektif antara lokasi kecelakaan dan rumah sakit, namun hasil akurasinya tidak dicantumkan.

Penelitian oleh E Utami dkk [9] meningkatkan kinerja sistem Machine Learning dalam klasifikasi sentimen menggunakan Bat Algorithm untuk memberikan bobot pada kata-kata dalam korpus. Hasil eksperimen menunjukkan peningkatan akurasi dari 32,82% menjadi 76,63% setelah pengaplikasian BA. Namun, jurnal tersebut tidak menjelaskan rinci tentang pengumpulan data, jumlah data, atau kriteria pemilihannya.

Penelitian ini menggunakan metode CNN karena akurasinya yang lebih tinggi dibandingkan metode lain. Bat Algorithm dipilih untuk mengoptimalkan hyperparameter, memilih fitur relevan, dan meningkatkan arsitektur CNN. Kombinasi BA-CNN belum banyak digunakan untuk analisis sentimen X dalam bahasa Indonesia.

3. METODE

Sistem analisis sentimen dimulai dengan crawling data dari media sosial X untuk mengumpulkan informasi yang disimpan dalam dataset. Data diberi label manual untuk menentukan sentimen (positif, negatif, atau netral) dan dibagi menjadi data latih dan uji. Data diproses dengan pembersihan dan normalisasi teks, diikuti ekstraksi fitur menggunakan TF-IDF dan ekspansi fitur dengan FastText. Model CNN digunakan untuk klasifikasi, sementara Bat Algorithm diterapkan untuk optimasi parameter guna meningkatkan akurasi. Akhirnya, sistem mengevaluasi kinerja model berdasarkan akurasi prediksi. Gambar 1 menunjukkan rancangan sistem.



GAMBAR 1.
Alur Kerja Model

3.1 Crawling data

Crawling adalah teknik untuk mengumpulkan data dari media sosial X berdasarkan kata kunci. Proses ini dilakukan selama kampanye Pilkada 2024, menghasilkan total 93.408 data mentah terkait topik tersebut. Rincian distribusi data berdasarkan kata kunci terdapat pada Tabel 1.

TABEL 1.
Distribusi Data Hasil Crawling

Kata Kunci	Total
pilkada	69.583
Pilkada Jakarta 2024	4.021
#pilkadaserentak2024	2.368
#pilkada2024	10.522
pemilihan kepala daerah	3.458
pilgub	3.456
Total	93.408

Penulis juga melakukan scraping pada berbagai situs berita Indonesia seperti Detik, Tribun, Kompas, dan lainnya antara Januari hingga Oktober 2024. Proses ini menghasilkan 126.247 berita terkait "Pilkada 2024" yang mencakup headline, tanggal, isi berita, dan tautan. Data ini akan digunakan untuk membangun corpus.

3.2 Labelling Data

Data yang telah dicrawling akan diberi label secara manual oleh tiga orang dengan prinsip majority vote untuk memastikan akurasi. Label yang diberikan mencakup sentimen Positif, Netral, dan Negatif, dengan definisi masing-masing sebagai berikut:

1. Label Negatif : Data mengandung kata kasar, hinaan, fitnah, cemoohan, atau kecenderungan menentang Pilkada 2024.
2. Label Netral : Data bersifat netral, hanya berisi informasi faktual atau deskripsi tanpa sikap terhadap Pilkada 2024.
3. Label Positif : Data berisi kata dukungan, promosi, pujian, atau kecenderungan mendukung Pilkada 2024.

3.3 Split Data

Proses split data membagi dataset yang telah dilabeli menjadi dua bagian, yaitu 90% untuk data latih dan 10% untuk data uji. Pembagian ini memungkinkan model belajar dari sebagian besar data dan menguji akurasi serta kemampuan generalisasi pada data yang belum pernah dilihat.

3.4 Pre-Processing

Pre-processing adalah tahapan untuk membersihkan dan mengubah data mentah agar siap digunakan dalam analisis lebih lanjut.

3.4.1 Cleansing Data

Cleansing data adalah tahap pembersihan dengan menghapus elemen tidak relevan seperti tanda baca, URL, username, angka, mention, hashtag, dan simbol, untuk memastikan data hanya berisi kata-kata relevan.

3.4.2 Case Folding

Case folding adalah proses mengubah semua huruf kapital menjadi huruf kecil untuk memperlakukan kata yang sama meski berbeda kapitalisasi, seperti "Pilkada" dan "pilkada".

3.4.3 Stopword Removal

Stopword removal adalah proses menghapus kata-kata tidak penting seperti "dan", "atau", "dengan", untuk fokus pada kata-kata yang lebih relevan dalam analisis.

3.4.4 Tokenization

Tokenization adalah proses memecah teks menjadi unit terkecil (token) seperti kata atau frasa untuk memudahkan pemrosesan oleh model.

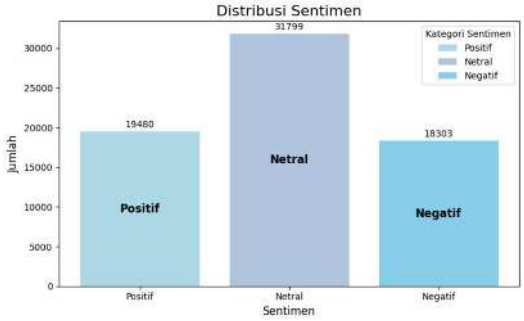
3.4.5 Stemming

Stemming adalah proses mengubah kata ke bentuk dasarnya dengan menghapus afiks, untuk mengeliminasi variasi kata yang memiliki makna sama.

3.4.6 Normalisasi

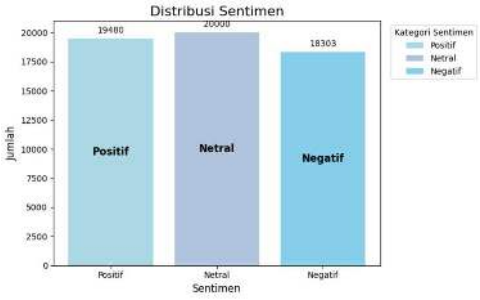
Normalisasi adalah proses mengubah teks ke bentuk standar dengan mengonversi huruf besar ke kecil, menghapus angka, tanda baca, dan karakter khusus, serta menormalkan kata-kata tidak baku untuk meningkatkan efektivitas analisis dan akurasi prediksi.

Setelah melalui tahap preprocessing dan penghapusan duplikat, jumlah data yang awalnya 93.408 berkurang menjadi 69.669. Hasil distribusi sentimen setelah preprocessing menunjukkan ketidakseimbangan, seperti yang dapat dilihat pada Gambar 2, yang menggambarkan distribusi sentimen pasca-preprocessing.



GAMBAR 2
Distribusi Sentimen setelah Preprocessing

Dengan berbagai pertimbangan, maka diputuskan untuk mengurangi jumlah data dengan label netral (0.0) menjadi 20.000 supaya label netral tidak mendominasi.



GAMBAR 3
Distribusi Sentimen setelah preprocessing

Gambar 3 menunjukkan persebaran label pada dataset akhir yang akan digunakan dalam penelitian ini. Setelah dilakukan preprocessing dan penghapusan data duplikat, jumlah data mentah yang awalnya sebanyak 93.408 berkurang menjadi 57.783. Label data telah cukup seimbang di antara ketiganya.

3.5 Ekstraksi Fitur

Dalam penelitian ini TF-IDF digunakan untuk ekstraksi fitur dengan tujuan mengidentifikasi kata-kata penting dalam dataset, library sklearn digunakan untuk menghitung nilai TF-IDF. Dengan menggunakan TF-IDF, fitur yang diekstraksi lebih representatif terhadap konteks dokumen, sehingga model yang dibangun dapat lebih akurat dalam memahami dan memprediksi kategori teks. Selain itu, TF-IDF juga membantu dalam mengurangi dampak kata-kata umum yang tidak informatif sehingga analisis sentimen pada X menjadi lebih efektif dan efisien [13]. Nilai TF-IDF dapat diperoleh dengan perhitungan rumus berikut.

$$TF - IDF(t,d) = tf_{t,d} \times \log\left(\frac{N}{df_t}\right) \quad (1)$$

Rumus TF-IDF terdiri dari dua elemen utama, yaitu Term Frequency (TF) dan Inverse Document Frequency (IDF). TF menggambarkan seberapa sering kata t muncul dalam dokumen d, yang menggambarkan frekuensi kemunculan kata tersebut dalam dokumen tersebut. Di sisi lain, IDF mengukur

kelangkaan kata t dalam seluruh koleksi dokumen dengan cara menghitung berapa banyak dokumen yang mengandung kata tersebut.

3.6 Ekspansi Fitur menggunakan FastText

Dalam penelitian ini, teknik ekspansi fitur dilakukan dengan memanfaatkan model FastText untuk meningkatkan representasi kata. FastText, yang merupakan bagian dari modul gensim, menggunakan informasi subkata untuk merepresentasikan kata-kata secara lebih efektif, terutama ketika word embeddings tradisional kurang optimal[14]. FastText terbukti lebih unggul dalam menangani kata-kata yang tidak terdapat dalam kamus, karena ia memperhitungkan bentuk subkata untuk menciptakan representasi kata yang lebih informatif[15]. Model ini melibatkan penggunaan korpus yang terdiri dari berbagai dataset, termasuk kumpulan kata dari tweet dan berita. Korpus yang digunakan dalam penelitian ini terdiri dari korpus tweet, korpus IndoNews, dan gabungan keduanya. Tabel 2 menunjukkan jumlah kosakata yang terdapat dalam masing-masing korpus yang telah disiapkan.

TABEL 2.
Jumlah Kosakata pada Corpus

Corpus Similarity	Total
Tweet	57.744
Indonews	126.247
Tweet + Indonews	183.991

Korpus-korpus tersebut digunakan untuk menilai kesamaan kata-kata tertentu dalam konteks yang lebih luas[16]. Tabel 3 menunjukkan contoh kesamaan kata pada korpus yang telah dibangun, di mana kata-kata yang serupa akan dikelompokkan bersama, memperkuat hubungan antar kata dalam analisis.

TABEL 3.
Corpus Similarity Top 10 Tweet

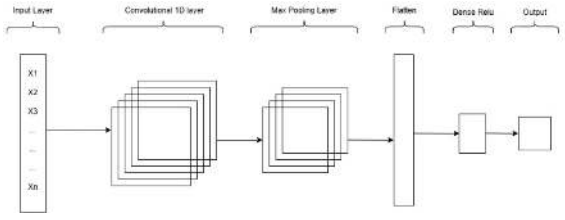
Kata	Top 1	Top 2	Top 3	Top 4	Top 5
rivalitas	brutalitas	loyalitas	totalitas	elektabilitas	faktualitas
	Top 6	**Top 7**	**Top 8**	**Top 9**	**Top 10**
	fatalitas	spiritualitas	realitas	orisinalitas	kualitas

Setelah korpus similarity terbentuk, langkah berikutnya adalah melakukan ekspansi fitur. Dalam penelitian ini, ekspansi fitur dilakukan menggunakan FastText untuk mengidentifikasi kata-kata yang belum terdeteksi sebelumnya. Tujuan dari ekspansi fitur ini adalah untuk mengatasi masalah ketidaksesuaian kosakata, di mana kata-kata yang memiliki nilai 0 pada vektor akan digantikan dengan kata-kata yang mirip berdasarkan peringkat kesamaan dalam korpus. Dengan demikian, kata yang sebelumnya bernilai 0 akan memiliki nilai yang lebih tinggi, yaitu 1, dan menjadi lebih representatif[25]. Sebagai contoh terdapat sebuah *tweet* berisi kalimat "brutalitas polisi memang tidak bisa dibiarkan, …". Dalam analisis ini, kata "rivalitas" awalnya memiliki nilai fitur 0 dalam representasi *tweet,* yang berarti kata tersebut tidak muncul dalam *tweet.* Namun, dengan menggunakan model FastText, ditemukan bahwa "rivalitas" memiliki beberapa kata yang dianggap mirip, seperti "brutalitas", "loyalitas", "totalitas", "elektabilitas", "faktualitas", "fatalitas", "spiritualitas", "realitas", "orisinalitas", "kualitas". Karena dalam *tweet* tersebut terdapat kata "brutalitas", yang merupakan salah satu kata mirip dengan "rivalitas", maka "rivalitas" dianggap relevan dengan isi *tweet* tersebut. Akibatnya, nilai fitur "rivalitas" dalam representasi *tweet* diubah dari 0 menjadi 1.

3.7 *Convolutional Neural Network* (CNN)

Convolutional Neural Network (CNN) adalah salah satu teknik dalam Deep Learning yang banyak digunakan dalam berbagai bidang, seperti pengolahan gambar, pengenalan pola, dan klasifikasi teks[18]. CNN mampu mempelajari fitur yang berkaitan satu sama lain dan menggabungkannya untuk melakukan klasifikasi. Dalam Natural Language Processing, CNN mengubah teks menjadi representasi matriks, di mana setiap baris menunjukan kata dalam kalimat dan setiap kolom mewakili dimensi kata tersebut [10].

Metode konvolusi CNN terinspirasi dari sistem saraf biologis [11] dan melibatkan beberapa lapisan pemrosesan seperti convolutional dan maxpooling layers secara bersamaan untuk mengekstraksi representasi hirarkis dari input secara bertahap. Dengan demikian, CNN mengintegrasikan tugas klasifikasi dan ekstraksi fitur menjadi satu kesatuan, menciptakan model yang lebih efektif dalam analisis sentimen serta tugas-tugas lain dalam NLP. Gambar 4 adalah model klasifikasi teks menggunakan CNN [12].



GAMBAR 4
Model Klasifikasi Teks CNN

3.8 Bat Algorithm Optimization

Bat Algorithm (BA) adalah algoritma optimasi berbasis metaheuristik yang meniru perilaku kelelawar dalam mencari mangsa dengan memanfaatkan

ekolokasi. Dalam proses optimasi, BA bekerja dengan menginisialisasi populasi kelelawar pada posisi awal tertentu, yang mewakili solusi awal dalam ruang pencarian. Setiap kelelawar memiliki kecepatan dan posisi yang diperbarui secara iteratif berdasarkan tiga komponen utama: frekuensi, kecepatan, dan posisi. BA dibangun oleh Yang xin she [17]berdasarkan peraturan berikut:

1. Kelelawar menggunakan ekolokasi untuk mendeteksi jarak dan hambatan di sekitar mereka.
2. Kelelawar terbang acak dengan kecepatan tetap, menghasilkan suara dengan frekuensi tetap namun panjang gelombangnya bisa berubah, serta menyesuaikan intensitasnya sesuai jarak dengan mangsa.
3. Kerasnya bunyi dapat berubah, dari nilai besar positif ke nilai konstan minimum.

Dalam penelitian ini, BA digunakan untuk mencari kombinasi parameter optimal pada CNN, seperti learning rate, dropout rate, jumlah filter, ukuran kernel, dan unit di fully connected layer, dengan mengarahkan kelelawar ke konfigurasi parameter yang memberikan performa terbaik berdasarkan metrik evaluasi.

```
Bat Algorithm
Objective function f(x), x =(x1,...,xd )^T
Initialize the bat population xi ( i = 1, 2, ...,n) and vi
Define pulse frequency fi at xi
Initialize pulse rates ri and the loudness Ai
while ( t < Max number of iterations)
Generate new solutions by adjusting frequency,
and updating velocities and locations/solutions [equations (2) to (4)]
    if (rand> ri)
    Select a solution among the best solutions
    Generate a local solution around the selected best solution
    end if
    Generate a new solution by flying randomly
    if (rand< Ai & f(xi) < f(x*))
    Accept the new solutions
    Increase ri and reduce Ai
    end if
Rank the bats and find the current best x*
end while
Postprocess results and visualization
```

GAMBAR 5
Pseudocode untuk program Bat Algorithm

Gambar 5 menunjukkan pseudocode Bat Algorithm (BA), yang terinspirasi dari cara kelelawar berburu. Algoritma ini menginisialisasi populasi kelelawar dengan parameter seperti posisi, kecepatan, frekuensi pulsa, laju pulsa, dan intensitas suara. Selama iterasi, kelelawar menyesuaikan parameter untuk mencari solusi optimal. Jika nilai acak lebih besar dari laju pulsa, kelelawar memilih solusi terbaik dan memperbaiki solusi lokal. Jika lebih kecil, solusi baru diterima jika memenuhi kriteria. Setelah iterasi, solusi terbaik ditemukan dan divisualisasikan. Algoritma ini mengeksplorasi dan mengeksploitasi solusi untuk menemukan hasil optimal.

3.9 Evaluasi

Evaluasi dilakukan menggunakan confusion matrix, yang membandingkan nilai aktual dan prediksi dalam klasifikasi sentimen untuk tiga kelas: Positive, Neutral, dan Negative. Setiap baris mewakili nilai aktual, sementara kolom menunjukkan nilai prediksi model. Tabel 4 memperlihatkan perbandingan ini.

TABEL 4.
Confusion Matrix

Actual Values	**Predicted Values**		
	Positive	*Neutral*	*Negative*
Positive	TP(*Positive*)	FN(*Neutral*)	FN(*Negative*)
Neutral	FP(*Positive*)	TN(*Neutral*)	FN(*Negative*)
Negative	FP(*Positive*)	FN(*Neutral*)	TN(*Negative*)

Confusion matrix digunakan untuk menghitung skor model seperti akurasi, presisi, recall, dan f1-score. Akurasi mengukur ketepatan prediksi model secara keseluruhan. Presisi mengukur akurasi prediksi kelas positif, recall membandingkan kategori positif yang terklasifikasi true positif, dan f1-score memberikan ukuran seimbang antara presisi dan recall.

$$Akurasi = \frac{TP + TN}{TP + FN + TN + FP} \quad (2)$$

$$Presisi = \frac{TP}{TP + FP} \quad (3)$$

$$Recall = \frac{TP}{TP + FN} \quad (4)$$

$$F1 - Score = \frac{2\ x\ precision\ x\ recall}{precision\ +\ recall} \quad (5)$$

4. HASIL DAN PEMBAHASAN

Penelitian ini melibatkan lima skenario pengujian:

1. Skenario I, Pengujian rasio split data menggunakan TF-IDF dan CNN untuk mendapatkan nilai baseline.
2. Skenario II, Pengujian konfigurasi max feature dengan TF-IDF dan CNN untuk akurasi terbaik.
3. Skenario III, Pengujian n-gram dalam ekstraksi fitur TF-IDF dan CNN untuk akurasi terbaik.
4. Skenario IV, Pengujian ekspansi fitur menggunakan FastText untuk meningkatkan akurasi.
5. Skenario V, Penerapan Bat Algorithm untuk meningkatkan akurasi model.

4.1 Skenario 1

Pada skenario ini, diuji rasio pembagian data 90:10, 80:20, dan 70:30 menggunakan TF-IDF dan CNN untuk mencapai akurasi tertinggi.

TABEL 5
Skenario Split Data

Split Ratio	Akurasi (%)
90:10	**70,64**
80:20	70,48
70:30	70,21

Berdasarkan Tabel 5, rasio 90:10 memberikan akurasi tertinggi 70,64%, yang akan digunakan untuk skenario selanjutnya.

4.2 Skenario 2

Berdasarkan hasil skenario 1, dilakukan pengujian konfigurasi max feature dengan nilai 5.000, 10.000, 15.000, 20.000, 25.000, dan 38.557 untuk mencapai akurasi tertinggi.

TABEL 6.
Skenario Max Feature

Max Feature	Akurasi(%)
5.000	70,65
10.000	72,66
15.000	**73,01**
20.000	72,42
25.000	71,92
38.557 (*baseline*)	70,64

Berdasarkan Tabel 6, nilai max feature 15.000 menghasilkan akurasi tertinggi 73,01%, selisih +2,37% dari baseline, dan akan diterapkan pada skenario selanjutnya.

4.3 Skenario 3

Pada skenario ini, hasil dari skenario 2 diuji dengan kombinasi n-gram (unigram, bigram, trigram, uni-bigram, dan uni-bi-trigram) untuk menemukan konfigurasi dengan akurasi tertinggi. Tabel 7 menunjukkan hasil pengujian pada skenario 3.

TABEL 7.
Skenario Konfigurasi n-gram

n-gram	Akurasi (%)
Unigram	73,14
Bigram	68,76
Trigram	63,31
Uni-Bigram	**73,30**
Uni-Bi-Trigram	72,73

Berdasarkan Tabel 7, konfigurasi Uni-Bigram menghasilkan akurasi tertinggi 73,30%, dan akan digunakan untuk skenario selanjutnya.

4.4 Skenario 4

Pada skenario keempat, dilakukan ekspansi fitur menggunakan FastText dengan tiga jenis korpus: Tweet, IndoNews, dan kombinasi keduanya (Tweet + IndoNews), berdasarkan tingkat similarity untuk top 1, 5, 10, dan 15. Tabel 8 menunjukkan hasil pengujian ini.

TABEL 8.
Skenario Ekspansi Fitur dengan FastText

Rank	Akurasi (%)		
	Tweet	IndoNews	Tweet + IndoNews
Top 1	**73,27 (+2,63)**	73,60 (+2,96)	**73,82 (+3,18)**
Top 5	73,08 (+2,44)	73,70 (+3,06)	73,71 (+3,07)
Top 10	73,16 (+2,52)	73,47 (+2,83)	73,68 (+3,04)
Top 15	72,42 (+1,78)	**73,85 (+3,21)**	73,55 (+2,91)

Hasil pengujian pada Tabel 8 menunjukkan akurasi tertinggi pada Top 1 Tweet (73,27%), Top 15 IndoNews (73,85%), dan Top 1 Tweet + IndoNews (73,82%). Berdasarkan hasil ini, penulis memilih Top 1 dari korpus Tweet + IndoNews sebagai pilihan optimal untuk skenario selanjutnya.

4.5 Skenario 5

Skenario 5 bertujuan mengoptimalkan kinerja model CNN dari skenario 4 menggunakan Bat Algorithm untuk mencari parameter hiperoptimal, seperti learning rate, dropout rate, jumlah filter, ukuran kernel, dan unit dense. Setelah parameter optimal ditemukan, model dilatih ulang untuk memvalidasi hasilnya. Tabel 9 menunjukkan parameter optimal yang ditemukan:

TABEL 9.
Nilai Optimal Parameter

Parameter	**Nilai Optimal**
Learning Rate	0.001
Dropout Rate	0.2178
Jumlah Filter	107
Kernel Size	3
Dense Units	39

Parameter yang ditemukan oleh Bat Algorithm (BA) memberikan konfigurasi optimal untuk model CNN. Learning rate 0.001 menjaga keseimbangan konvergensi, dropout rate 0.2178 mengurangi overfitting, jumlah filter 107 menangkap fitur spasial kompleks, ukuran kernel 3 menangkap detail tanpa meningkatkan kompleksitas, dan 39 unit dense di fully connected layer memastikan model kuat tanpa overfitting. Kombinasi ini

menciptakan keseimbangan antara pembelajaran dan generalisasi, menghasilkan model yang efisien dan efektif.

TABEL 10.
Hasil Akurasi Skenario Bat Algorithm

Rank	Akurasi (%)	
	Sebelum BA	Setelah BA
Top 1	73,82	73,87 (+0,05)

Tabel 10 menunjukkan peningkatan akurasi sebesar 0,05%, dari 73,82% menjadi 73,87%, setelah menggunakan Bat Algorithm (BA) pada Top 1 dari rank similarity corpus yang menggabungkan tweet dan IndoNews.

4.6 Analisis Hasil Pengujian

Gambar 5 menampilkan hasil pengujian yang memperlihatkan peningkatan akurasi model pada setiap skenario pengujian yang dilakukan.

GAMBAR 6.
Fluktuasi Setiap Skenario

Berdasarkan Gambar 6, penelitian ini mengoptimalkan model CNN melalui lima skenario. Skenario I menghasilkan baseline akurasi 70,64% dengan rasio 90:10. Skenario II meningkatkan akurasi menjadi 73,01% dengan 15.000 fitur. Skenario III mencapai akurasi 73,30% dengan konfigurasi Uni-Bigram. Skenario IV menggunakan ekspansi fitur FastText dan korpus Tweet + IndoNews, menghasilkan akurasi 73,82%. Skenario V mengoptimalkan model dengan Bat Algorithm, mencapai akurasi 73,87%, naik 0,05% dari sebelumnya.

5. KESIMPULAN

Penelitian ini mengidentifikasi strategi peningkatan akurasi model CNN dalam klasifikasi teks melalui lima skenario pengujian, dengan tujuan utama menerapkan analisis sentimen pada data media sosial X menggunakan CNN yang dioptimalkan Bat Algorithm dan ekspansi fitur FastText. Data sebanyak 93.526 tweet berbahasa Indonesia terkait Pilkada 2024 dikumpulkan melalui crawling menggunakan tweet-harvest, dan fitur diekstraksi menggunakan TF-IDF. Kontribusi utama penelitian ini adalah pengujian Bat Algorithm untuk mengoptimalkan CNN dan evaluasi ekspansi fitur dengan FastText, yang berhasil meningkatkan akurasi hingga 73,82%. Penerapan Bat Algorithm memberikan peningkatan akurasi sebesar 0,05%. Penelitian ini menunjukkan potensi besar kombinasi CNN, Bat Algorithm, dan FastText dalam analisis sentimen media sosial, dengan saran untuk memperbanyak dataset dan eksplorasi metode optimasi lainnya.

REFERENSI

[1] M. Yusuf, D. Subekti, M. Wahid, and M. Saadah, “Bureaucrat’s Political Activities in the 2020 Simultaneous Regional Elections in Indonesia: What They Express on Social Media?,” no. Icsp, pp. 164–171, 2023, doi: 10.2991/978-2-38476-194-4_18.

[2] B. Bansal and S. Srivastava, “On predicting elections with hybrid topic based sentiment analysis of tweets,” Procedia Comput. Sci., vol. 135, pp. 346–353, 2018, doi: 10.1016/j.procs.2018.08.183

[3] M. R. Haque, S. Akter Lima, and S. Z. Mishu, “Performance Analysis of Different Neural Networks for Sentiment Analysis on IMDb Movie Reviews,” 3rd Int. Conf. Electr. Comput. Telecommun. Eng. ICECTE 2019, pp. 161–164, 2019, doi: 10.1109/ICECTE48615.2019.9303573.

[4] I. Santos, N. Nedjah, and L. De Macedo Mourelle, “Sentiment analysis using convolutional neural network with fasttext embeddings,” 2017 IEEE Lat. Am. Conf. Comput. Intell. LA-CCI 2017 - Proc., vol. 2018-Janua, pp. 2–6, 2017, doi: 10.1109/LA-CCI.2017.8285683.

[5] I. Chalkidis and D. Kampas, “Deep learning in law: early adaptation and legal word embeddings trained on large corpora,” Artif. Intell. Law, vol. 27, no. 2, pp.171–198, 2019, doi: 10.1007/s10506-018-9238-9

[6] R. A. Rudiyanto and E. B. Setiawan, “Sentiment Analysis Using Convolutional Neural Network (CNN) and Particle Swarm Optimization on Twitter,” JITK (Jurnal Ilmu Pengetah. dan Teknol. Komputer), vol. 9, no. 2, pp. 188–195, 2024, doi: 10.33480/jitk.v9i2.5201.

[7] T. Darwassh Hanawy Hussein, M. Frikha, S. Ahmed, and J. Rahebi, “BA-CNN: Bat Algorithm-Based Convolutional Neural Network Algorithm for Ambulance Vehicle Routing in Smart Cities,” Mob. Inf. Syst., vol. 2022, 2022, doi: 10.1155/2022/7339647.

[8] I. Kaibi, E. H. Nfaoui, and H. Satori, “A comparative evaluation of word embeddings techniques for twitter sentiment analysis,” 2019 Int. Conf. Wirel. Technol. Embed. Intell. Syst. WITS 2019, pp. 1–4, 2019, doi: 10.1109/WITS.2019.8723864.

[9] E. Utami, S. Raharjo, O. M. A. Alsyaibani, and C. Adipradana, “Machine Learning Optimization using Bat Algorithm to Classify Sentiment of Twitter Users,” 2022

Int. Conf. Bus. Anal. Technol. Secur. ICBATS 2022, pp. 1–5, 2022, doi: 10.1109/ICBATS54253.2022.9759029.

[10] F. Alfariqi, W. Maharani, and J. H. Husen, "Klasifikasi Sentimen pada Twitter dalam Membantu Pemilihan Kandidat Karyawan dengan Menggunakan Convolutional Neural Network dan Fasttext Embeddings," e-Proceeding Eng., vol. 7, no. 2, pp. 8052–8062, 2020.

[11] H. K. Farid, E. B. Setiawan, and I. Kurniawan, "Implementation Information Gain Feature Selection for Hoax News Detection on Twitter using Convolutional Neural Network (CNN)," Indones. J. Comput., vol. 5, no. 3, pp. 23–36, 2020, doi: 10.34818/INDOJC.2020.5.3.506.

[12] J. Liu, H. Ma, X. Xie, and J. Cheng, "Short Text Classification for Faults Information of Secondary Equipment Based on Convolutional Neural Networks," Energies, vol. 15, no. 7, 2022, doi: 10.3390/en15072400.

[13] A. I. Kadhim, "Term Weighting for Feature Extraction on Twitter: A Comparison between BM25 and TF-IDF," 2019 Int. Conf. Adv. Sci. Eng. ICOASE 2019, pp. 124–128, 2019, doi: 10.1109/ICOASE.2019.8723825.

[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, 2017, doi: 10.1162/tacl_a_00051.

[15] Grave, E., Bojanowski, P., Mikolov, T., & Joulin, A. (2018). Learning Word Vectors for 157 Languages. Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics (EACL).

[16] Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. Proceedings of the International Conference on Learning Representations (ICLR).

[17] X. S. Yang, "A new metaheuristic Bat-inspired Algorithm," Stud. Comput. Intell., vol. 284, pp. 65–74, 2010, doi: 10.1007/978-3-642-12538-6_6.

[18] Rani, S., Bashir, A. K., Alhudhaif, A., Koundal, D., & Gunduz, E. S. (2022). An efficient CNN-LSTM model for sentiment detection in# BlackLivesMatter. Expert Systems with Applications, 193, 116256

[19] Komisi Pemilihan Umum (KPU). (2020). Peran Opini Publik dalam Pemilu. Retrieved from https://www.kpu.go.id/berita/baca/11867/kampanye-edukatif-dan-programatif-dorong-demokrasi-indonesia-semakin-baik

[20] Fahum. (2020). Pentingnya Kampanye Edukatif dan Programatif dalam Pemilu. Retrieved from https://fahum.umsu.ac.id/info/pengertian-kampanye-pemilu

[21] Setiawan, H. D., & Djafar, T. M. (2023). Partisipasi politik pemilih muda dalam pelaksanaan demokrasi di Pemilu 2024. Populis: Jurnal Sosial dan Humaniora, 8(2), 201-213.

[22] Gifari, O. I., Adha, M., Hendrawan, I. R., & Durrand, F. F. S. (2022). Analisis Sentimen Review Film Menggunakan TF-IDF dan Support Vector Machine. Journal of Information Technology, 2(1), 36-40.

[23] Jahan, I., Islam, M. N., Hasan, M. M., & Siddiky, M. R. (2024). Comparative analysis of machine learning algorithms for sentiment classification in social media text. World J. Adv. Res. Rev, 23(3), 2842-2852.

[24] Reddy, M. R. P., Kumar, D. S., Sanjay, S. R., Kumar, P. M., & Bharath, Y. S. (2024, April). Achieving Cost Efficiency and Load Sharing in Power Systems: The Superiority of Adaptive BAT Algorithm. In 2024 International Conference on Communication, Computing and Internet of Things (IC3IoT) (pp. 1-5). IEEE.

[25] Setiawan, E. B., Widyantoro, D. H., & Surendro, K. (2016, October). Feature expansion using word embedding for tweet topic classification. In 2016 10th International Conference on Telecommunication Systems Services and Applications (TSSA) (pp. 1-5). IEEE.