

# PENDEKATAN SENTIMEN BERBASIS ASPEK PADA ULASAN SIRKUIT MANDALIKA MENGGUNAKAN CNN DAN REPRESENTASI FASTTEXT

(*Aspect-Based Sentiment Approach to Mandalika Circuit Reviews Using CNN and FastText Representation*)

Ida Bagus Ryand Wirayana Manuaba\*<sup>[1]</sup>, Ramaditia Dwiyanaputra<sup>[1]</sup>, Mohammad Zaenuddin Hamidi<sup>[1]</sup>

<sup>[1]</sup>Dept Informatics Engineering, Mataram University

Jl. Majapahit 62, Mataram, Lombok NTB, INDONESIA

Email: bagusryand77@gmail.com, [rama, mzhamidi]@unram.ac.id

## Abstract

Reviews are texts that contain an assessment or comment on something and can be used to provide more in-depth information. This research aims to analyze community reviews of the Mandalika Circuit using the aspect-based sentiment analysis technique CNN method. The CNN model is trained using two types of word embedding, namely Keras and FastText, and supported by the Multilabel Stratified K-Fold Cross Validation method to ensure an even distribution of data on each label and produce a stable accuracy evaluation. The results show that most Indonesians show neutral sentiments towards Mandalika Circuit and CNN with FastText word embedding has a higher average accuracy than CNN with Keras word embedding for aspect and sentiment classification tasks with 84% for aspect and 72% for sentiment.

**Keywords:** Aspect-Based Sentiment Analysis, Mandalika Circuit, Social Media X, CNN, FastText

\*Correspondence Author

## 1. PENDAHULUAN

Mandalika menjadi salah satu destinasi prioritas yang ditandai dengan dibangunnya banyak tempat pariwisata seperti yang sudah terkenal yaitu Sirkuit Mandalika. Pada akhir tahun 2019 Sirkuit Mandalika ini dibangun dengan biaya 1,1 triliun rupiah. Panjang lintasannya mencapai 4,3 kilometer dan dapat menampung sekitar 200.000 orang [1].

Indonesia telah menjadi tuan rumah dalam dua event olahraga balap internasional seperti MotoGP dan WSBK (*World Superbike*), diantaranya pada tanggal 12-14 November 2021 untuk World Superbike dan 18-20 Maret 2022 untuk MotoGP [2], [3], [4]. Hal ini tentu secara tidak langsung akan meningkatkan perekonomian provinsi Nusa Tenggara Barat khususnya di kawasan Mandalika.

Dalam berlangsungnya event di Sirkuit Mandalika ini, tentu banyak masyarakat yang memberikan pendapatnya ataupun ulasan terhadap sirkuit ini. Ulasan ataupun pendapat masyarakat ini dapat bernilai positif maupun negatif. Hal ini bisa jadi bermanfaat bagi Mandalika Grand Prix Association sebagai bahan evaluasi untuk memperbaiki ataupun meningkatkan kualitas dari Sirkuit Mandalika. Mandalika Grand Prix Association (MGPA) adalah

perusahaan ITDC, yang bertugas untuk mengoperasikan dan mengelola Sirkuit Mandalika [5]. Untuk mendapatkan ulasan yang cukup mengenai Sirkuit Mandalika ini, penulis akan mencoba melakukan *crawling* dari platform X.

Platform X yang dulunya bernama Twitter adalah platform yang memungkinkan seseorang untuk berinteraksi dan tetap terhubung melalui pesan cepat. Pengguna dapat membagikan konten, termasuk foto, video, tautan, dan teks [6]. Laporan We Are Social menyatakan bahwa pada bulan Oktober 2023, sekitar 27,5 juta orang di negara Indonesia menggunakan platform Twitter, juga dikenal sebagai X. Dengan jumlah ini, Indonesia menempati peringkat keempat di seluruh dunia untuk penggunaan platform tersebut [7]. Melakukan analisis sentimen berbasis aspek pada pengguna platform X memiliki potensi yang menjanjikan berdasarkan data di atas. Analisis sentimen berbasis aspek akan mengidentifikasi berbagai aspek (atribut atau komponen) dalam setiap teks, kemudian menentukan tingkat sentimen yang sesuai untuk masing-masing aspek [8]. Sehingga nantinya di dalam satu ulasan terdapat aspek apa yang dibahas dan tingkat sentimennya.

Secara umum, perusahaan banyak menggunakan analisis sentimen berbasis aspek karena memberikan

keunggulan bagi perusahaan untuk mengetahui lebih lanjut tentang apa yang benar-benar dibutuhkan pelanggan. Terdapat beberapa algoritma *machine learning* yang dapat digunakan seperti *Random Forest*, *SVM*, *Logistic Regression*, *Naive Bayes*, hingga menggunakan algoritma *Deep Learning*.

Salah satu contoh dalam analisis sentimen berbasis aspek yang menggunakan algoritma *deep learning* adalah CNN (*Convolutional Neural Networks*). CNN adalah *neural network* yang dapat digunakan untuk mengenali dan mendeteksi objek pada sebuah citra atau gambar. CNN adalah salah satu teknik yang terinspirasi dari cara manusia melihat atau menghasilkan persepsi visual. Selain digunakan untuk mengklasifikasikan citra atau gambar yang berupa dua dimensi, CNN juga dapat digunakan untuk mengklasifikasikan teks yang representasi datanya berupa satu dimensi. Beberapa penelitian sebelumnya telah menggunakan CNN untuk klasifikasi teks seperti klasifikasi sentimen pada ulasan film Indonesia [9], analisis sentimen pada data Twitter topik politik [10], dan klasifikasi teks pada data Twitter terhadap pandemi *COVID-19* [11]. Arsitektur CNN pada klasifikasi teks juga tidak jauh berbeda dengan arsitektur CNN pada klasifikasi data gambar. Pada CNN untuk klasifikasi teks memiliki tahap *word embedding*, *convolutional*, *pooling*, *dense*, *dropout* dan *output*.

FastText merupakan salah satu *library* yang dapat digunakan pada tahapan *word embedding*. FastText dikembangkan oleh Facebook yang memiliki keunggulan dapat menangani kata yang tidak pernah dilihat sebelumnya. FastText dirancang agar dapat menghasilkan representasi kata dan melakukan klasifikasi teks dengan lebih baik.

## 2. TINJAUAN PUSTAKA

Beberapa penelitian sebelumnya telah melakukan analisis data teks menggunakan metode CNN dan menghasilkan akurasi yang cukup baik. Contohnya penelitian oleh Yuliska, dkk. yang mengidentifikasi pengaruh kompleksitas arsitektur CNN terhadap hasil klasifikasi sentimen data saran. Menggunakan *Word2Vec model* untuk mendapatkan *word embedding* atau representasi kata. Terdapat dua jaringan CNN yang digunakan, jaringan pertama, yang menggunakan dua lapisan *convolutional* dan penggunaan *Maxpooling* setelah lapisan *convolutional* kedua. Jaringan kedua juga menggunakan dua *Convolutional* serta *Maxpooling* setelah masing-masing *Convolutional*. Hasil penelitian menunjukkan bahwa jaringan kedua dapat mengklasifikasi sentimen dengan sangat baik [12].

Desya, dkk. melakukan penelitian mengenai sentimen masyarakat terhadap pemindahan Ibukota Negara Indonesia. Tujuan utama dari penelitian yang dibahas dalam artikel ini adalah untuk mendapatkan perbandingan hasil antara dua metode ekstraksi fitur, yaitu GloVe dan FastText, dalam analisis sentimen terhadap komentar YouTube mengenai rencana pemindahan ibu kota negara Indonesia. Hasil penelitian menunjukkan bahwa metode ekstraksi fitur GloVe lebih akurat daripada FastText dengan akurasi 76.1% [13].

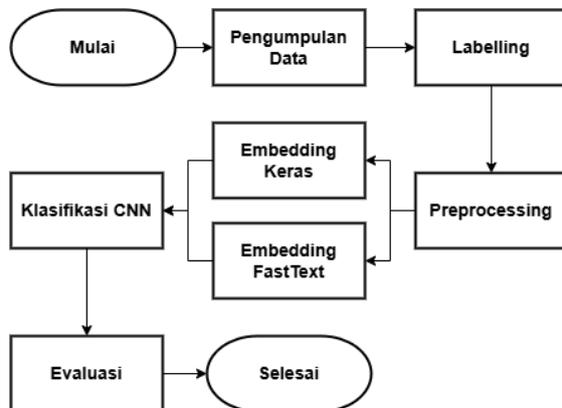
Penelitian selanjutnya oleh Sukma, dkk. telah melakukan analisis sentimen pilkada di tengah pandemi *Covid-19* dengan menggunakan CNN. Tujuan penelitian ini adalah untuk mengetahui persepsi masyarakat tentang penyelenggaraan Pilkada, khususnya apakah mereka setuju (sentimen positif) atau tidak setuju (sentimen negatif). Hasil yang diperoleh dari penerapan algoritma *Convolution Neural Network (CNN)* dalam analisis sentimen menunjukkan bahwa metode ini dapat mencapai akurasi tertinggi sebesar 90% dengan menggunakan empat *layer* konvolusi. Penelitian ini juga menunjukkan bahwa semakin banyak *epoch* yang digunakan, akurasi model cenderung menjadi lebih baik [14].

Arolina, dkk. melakukan penelitian yang mengidentifikasi ulasan tentang fitur musik aplikasi Instagram yang mengandung sentimen negatif dan positif, serta untuk mengukur performa model CNN yang digunakan. Pengumpulan data ulasan dilakukan dengan menggunakan teknik *web scraping* pada Google Play Store [15].

Pada kesempatan ini, penulis akan menggunakan CNN untuk mengklasifikasikan aspek dan sentimen sebab CNN memiliki kemampuan menangkap fitur lokal dalam data teks yang berguna untuk memahami pola sehingga dapat menentukan makna yang terkandung pada teks tersebut. FastText digunakan penulis untuk mendukung CNN dalam menentukan makna yang terkandung pada teks. Sehingga penelitian ini bertujuan untuk melihat bagaimana sentimen di masyarakat Indonesia mengenai Sirkuit Mandalika serta menguji kinerja dari model CNN yang menggunakan *embedding* Keras dan *embedding* FastText. Analisis sentimen memberikan manfaat pada pihak pengelola Sirkuit Mandalika sebab memberikan pemahaman bagaimana respons masyarakat terhadap berbagai aspek Sirkuit seperti layanan, fasilitas, dan pengalaman saat berada di Sirkuit. Kontribusi penulis dalam penelitian ini adalah penggunaan *embedding* FastText serta *multilabel stratified k-fold cross validation* dalam *multitask learning* CNN.

### 3. METODE PENELITIAN

Rancangan penelitian akan disajikan pada gambar berikut.



Gambar 1. Rancangan Penelitian

#### 3.1. Pengumpulan Data

Tahap awal penelitian adalah mengumpulkan data teks ulasan yang akan digunakan untuk analisis sentimen berbasis aspek adalah data yang berasal dari platform X pada tahun 2023. Cara pengambilan data pada platform X ini adalah dengan teknik *crawling*. Teknik *Crawling* adalah sebuah teknik pengumpulan data dari web untuk melakukan pengumpulan informasi [16].

#### 3.2. Labelling Data

Data ulasan yang telah didapatkan melalui proses *crawling* pada platform X tentunya belum memiliki label aspek dan sentimen. Label ini diperlukan sebelum melakukan pelatihan dan klasifikasi menggunakan metode CNN. Proses pelabelan data terbagi menjadi dua yaitu pelabelan aspek kemudian pelabelan sentimen. Proses anotasi dan pelabelan ini dilakukan oleh tiga orang, satu seorang *expert* dengan latar belakang akademik bidang linguistik dan dua mahasiswa teknik informatika dengan mengikuti panduan anotasi. Belum dilakukan *inter-annotator agreement*, ini menjadi salah satu keterbatasan pada proses anotasi. Jadi, untuk meminimalisir potensi bias, hasil anotasi mahasiswa di *review* oleh ahli bahasa.

#### 3.3. Preprocessing

Setelah dilakukan *labelling* pada setiap data teks ulasan, tahap selanjutnya adalah *preprocessing* yaitu mengubah data teks menjadi data yang siap untuk pengolahan *machine learning*. Tahapan *preprocessing* terdiri dari:

##### 3.3.1. Cleaning Data

Membersihkan data dari *mention*, *hashtag*, URL, dan karakter spesial. Kemudian mengubah semua huruf pada kalimat menjadi huruf kecil atau *lowercase*. Tahap ini menggunakan *library Regular Expression* yang menyediakan fungsionalitas untuk mencari pola tertentu di dalam teks yang dirancang untuk melakukan pencarian pola secara cepat dan efisien.

##### 3.3.2. Normalisasi

Proses menormalisasikan teks dengan cara mengubah kata yang tidak baku menjadi kata baku. Proses ini juga mengoreksi kata singkatan dan kata yang kurang jelas atau *typo*. Teknik normalisasi yang digunakan adalah *Substitution-based Normalization* dengan menggunakan kamus yang dapat mengganti kata atau frasa yang berbentuk singkatan atau tidak baku menjadi bentuk yang lebih baku.

##### 3.3.3. Stopwords

Proses ini merupakan proses untuk menghilangkan kata-kata yang tidak bermakna secara signifikan dalam sebuah kalimat. *Library* yang akan digunakan untuk menjalankan proses ini adalah *library Sastrawi*. *Library Sastrawi* adalah modul sederhana dalam bahasa pemrograman Python yang berfungsi untuk mengubah kata-kata dalam bahasa Indonesia ke bentuk baku atau sesuai dengan standar kamus. *Sastrawi* merupakan *library NLP* yang disiapkan khusus untuk bahasa Indonesia [17].

##### 3.3.4. Tokenisasi

Tokenisasi adalah pembagian teks menjadi bagian-bagian kecil yang disebut *token* [13]. Dengan memecah teks menjadi *token-token*, komputer dapat lebih mudah memahami dan memproses data teks.

##### 3.3.5. Stemming

Proses menyederhanakan kata menjadi bentuk dasarnya atau penghapusan imbuhan pada kata. Proses ini melibatkan penghapusan awalan (*prefix*) dan akhiran (*suffix*) dari kata-kata untuk menormalkannya dan meningkatkan efektivitas tugas-tugas pemrosesan bahasa alami. *Stemming* juga menggunakan *library Sastrawi*.

#### 3.4. Embedding

Salah satu yang membedakan antara CNN yang memproses data gambar dengan CNN yang memproses data teks adalah adanya tahap *word embedding* pada arsitektur CNN yang memproses data

teks. *Word embedding* adalah lapisan yang mengubah data berdimensi tinggi menjadi data berdimensi rendah dalam kasus *machine learning* dan *deep learning*. Ini adalah salah satu cara untuk membuat data teks lebih bermakna dengan mengubahnya menjadi sebuah *array* atau vektor yang terdiri dari kumpulan angka sehingga *machine learning* dapat memprosesnya. Dengan metode *word embedding*, data teks dapat menjadi sebuah vektor yang terdiri dari angka-angka kecil yang mengandung lebih banyak informasi untuk memahami makna setiap kata.

### 3.4.1. Keras *Embedding Layer*

Percobaan pertama, penulis akan menggunakan *word embedding* dari Keras yang merupakan *word embedding* secara *default* bila tidak menggunakan *word embedding* khusus seperti Word2Vec, FastText, atau Glove. *Word Embedding* ini bekerja dengan cara mengonversi data teks menjadi representasi vektor. Nilai representasi vektor awal dinisialisasi secara acak dan akan diperbaharui selama proses pelatihan guna memperbaiki representasi vektor [18].

### 3.4.2. FastText

Kemudian percobaan kedua, penulis akan menggunakan model *word embedding*. FastText adalah *library open-source* dan gratis yang dikeluarkan oleh Facebook dan memungkinkan pengguna untuk melakukan representasi teks ataupun klasifikasi teks secara efisien dan cepat. FastText bekerja dengan memanfaatkan *sub-word embedding*. Saat FastText menerima kata yang tidak pernah ada di kamus, FastText akan mengonversi kata tersebut menjadi *sub-word* kemudian vektor dari *sub-word* ini akan digabungkan untuk membentuk vektor yang baru [13].

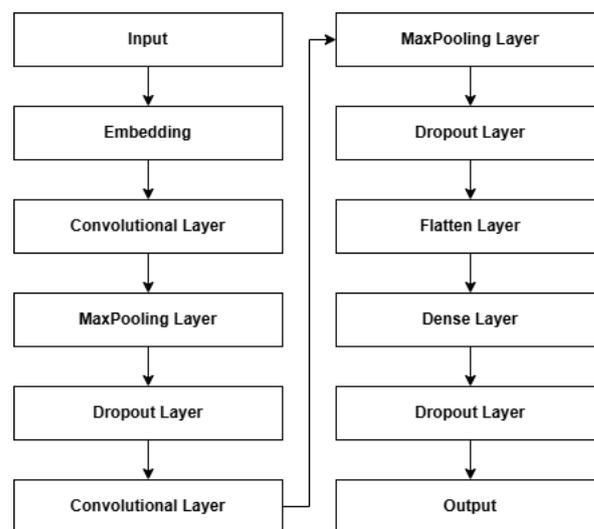
Model *word embedding* ini membutuhkan dataset awal sebagai bahan untuk melatih model. Dataset yang akan digunakan dalam membuat model *word embedding* FastText berbahasa Indonesia adalah data artikel bahasa Indonesia yang paling terbaru yang ada di Wikipedia.

### 3.5. Klasifikasi CNN

Setelah data melewati proses *preprocessing* dan menyiapkan model *word embedding* FastText, langkah selanjutnya yaitu proses pelatihan menggunakan model CNN. Model CNN yang akan digunakan oleh penulis adalah model *multitask learning* yang akan mengeluarkan dua *output* label (aspek dan sentimen) secara paralel. *Multitask learning* atau pembelajaran multitugas adalah jenis teknik pembelajaran mesin dimana model dilatih untuk melakukan beberapa tugas

secara bersamaan. Dalam *deep learning*, pembelajaran multitugas mengacu pada pelatihan jaringan saraf untuk melakukan beberapa tugas dengan membagi beberapa lapisan dan parameter jaringan ke seluruh tugas.

Arsitektur CNN yang digunakan adalah *DoubleMax CNN* dengan penambahan *dropout* pada setiap *pooling layer*. *DoubleMax CNN* dipilih karena memiliki lapisan *pooling* ganda sehingga lebih baik dalam mengurangi dimensi data namun tetap mempertahankan informasi penting. Berikut adalah gambar arsitektur CNN.



Gambar 2. Arsitektur CNN

Arsitektur CNN dimulai dengan *embedding* yang berfungsi untuk merepresentasikan kata ke dalam bentuk vektor. Kemudian *convolutional layer* 1D pertama dengan jumlah filter 128, *kernel size* 5, dan *activation function* ReLU. Setelah konvolusi, dilakukan *MaxPooling* dan *Dropout* dengan ukuran *pooling* 2 dan ukuran *dropout* 0.7. Kemudian melakukan *convolutional layer* 1D kedua dengan jumlah filter 256, *kernel size* 5, dan *activation function* ReLU. Sama seperti sebelumnya, setelah konvolusi, dilakukan *MaxPooling* dan *Dropout* dengan ukuran yang sama. Setelah itu dilakukan *flatten* guna mengubah *output* dari layer sebelumnya menjadi *array* satu dimensi. Selanjutnya masuk ke dalam *fully connected layer* atau *dense* dengan jumlah neuron 64. Langkah terakhir adalah menyisipkan lagi *layer dropout* dengan ukuran 0.7.

*Output* yang akan dikeluarkan ada dua yaitu aspek dan sentimen. Untuk aspek menggunakan fungsi aktivasi *sigmoid* sedangkan untuk sentimen menggunakan fungsi aktivasi *softmax*.

Arsitektur ini bersifat *multitask learning* yang dimana akan mengerjakan dua tugas secara bersamaan. Tugas pertama untuk klasifikasi aspek dan

tugas kedua untuk klasifikasi sentimen. Dua tugas tersebut menggunakan arsitektur yang sama. *Stratified K-Fold Cross Validation* akan digunakan dalam proses klasifikasi ini guna memastikan data dilatih dan diuji dengan distribusi kelas yang sama. Variasi *Stratified K-Fold Cross Validation* yang akan digunakan adalah *Multilabel Stratified K-Fold Cross Validation* karena data yang dimiliki mempunyai lebih dari satu label yaitu label aspek dan label sentimen. *Multilabel Stratified K-Fold Cross Validation* adalah *library* python yang berguna untuk melakukan pelatihan dan pengujian dalam beberapa *fold*. Setiap *fold* akan melakukan pelatihan dan pengujian dengan persebaran data yang berbeda sehingga diharapkan model akan menunjukkan hasil akurasi yang paling akurat. *Fold* yang akan digunakan berjumlah lima *fold*. Dataset akan terbagi menjadi tiga diantaranya *training set*, *validation set*, dan *testing set*. Data *training* berguna untuk melakukan pelatihan sedangkan data *validation* dan data *testing* berguna untuk mengukur akurasi model.

### 3.6. Evaluasi

Evaluasi kinerja model dilakukan dengan cara *Confusion Matrix* dan *Classification Report*. *Confusion Matrix* adalah sebuah cara untuk mengevaluasi kinerja model. *Confusion Matrix* dapat menunjukkan seberapa baik model dapat melakukan klasifikasi data. Untuk masalah klasifikasi biner, *Confusion Matrix* biasanya digunakan. Namun, metode ini juga dapat diterapkan pada masalah klasifikasi yang melibatkan lebih dari dua kelas [19].

*Classification Report* adalah ringkasan metrik-metrik kunci yang digunakan untuk menilai kualitas prediksi dari model klasifikasi. Metrik-metrik ini meliputi *precision*, *recall*, *F1 score*, dan *support*. *Precision* mengukur rasio prediksi positif yang benar terhadap total prediksi positif. *Recall* mengukur rasio prediksi positif yang benar terhadap total kasus sebenarnya dalam kelas tersebut. *F1 score* adalah rata-rata dari *recall* dan *precision*, yang memberikan keseimbangan antara keduanya. *Support* menunjukkan jumlah kejadian aktual dari setiap kelas dalam dataset.

## 4. HASIL DAN PEMBAHASAN

### 4.1. Dataset

Data teks ulasan yang berhasil dikumpulkan dari platform X sebanyak 3.422 data dan disimpan ke dalam file berformat csv. Selanjutnya data tersebut akan dilabelkan untuk label aspek dan sentimennya. Pelabelan hanya mengambil dua kelas aspek yaitu

aspek ekonomi dan aspek politik. Ini dikarenakan banyaknya ulasan yang membahas tentang keadaan ekonomi masyarakat di sekitar Sirkuit Mandalika dan suasana politik saat Sirkuit Mandalika dibangun dan mulai beroperasi. Kemudian untuk sentimen terdapat tiga kelas yaitu kelas negatif, netral, dan positif.

Jika teks ulasan berisi kata-kata yang berkaitan dengan ekonomi seperti "biaya", "bayar", "pengeluaran", "harga" maka teks ulasan tersebut akan dilabelkan aspek ekonomi. Sedangkan teks ulasan yang berisi topik mengenai politik seperti peraturan, kebijakan pemerintah, presiden, nama anggota partai politik akan diberi label aspek politik.

Kemudian untuk pelabelan sentimen negatif, netral, ataupun positif ditentukan melalui emosi atau sikap yang terkandung di dalam teks ulasan. Teks yang menunjukkan pujian atau kepuasan akan diberi label sentimen positif namun jika teks berisi keluhan, kritik, atau ketidakpuasan akan diberi label sentimen negatif. Jika teks ulasan tidak menunjukkan emosi yang positif maupun negatif, maka akan dilabeli sentimen netral.

TABEL I. Jumlah Kelas Pada Setiap Label

Aspek	Sentimen	Jumlah
Ekonomi	Negatif	648
Ekonomi	Netral	1058
Ekonomi	Positif	342
Politik	Negatif	598
Politik	Netral	460
Politik	Positif	316

Jumlah data yang memiliki kelas aspek ekonomi adalah 2.048, sedangkan data yang memiliki kelas aspek politik adalah 1.374. Untuk label sentimen, data paling banyak berada di kelas ekonomi netral dengan 1.058 data dan paling sedikit berada di kelas politik positif dengan 316 data.

### 4.2. Preprocessing

Data mengalami *preprocessing* dalam beberapa tahap yaitu: *Cleaning*, Normalisasi, *Stopwords*, Tokenisasi, dan *Stemming*. Berikut merupakan contoh data yang telah melewati tahap *preprocessing*.

TABEL II. Contoh Data *Preprocessing*

Sebelum <i>Preprocessing</i>	Sesudah <i>Preprocessing</i>
@detikcom pak sirkuit mandalika kapan di sindir pak rakyat NTB menanam pisang pak katanya Tanah mrk di rampas dan tsk di bayar..sekian Pak laporan selesai...	pak sirkuit mandalika kapan sindir pak rakyat ntb tanam pisang pak kata tanah rampas tidak di bayarsekian pak lapor selesai
@ieumal @starfess Udah tau niih. Katanya mau k Sirkuit Mandalika buat shooting MV. wkwkwwkwk	udah tau niih kata mau k sirkuit mandalika buat shooting mv wkwkwwkwk
Electrizen Pagelaran Internasional pertama World SuperBike (WSBK) tahun 2023 pada bulan Maret mendatang yang digelar di Sirkuit Mandalika dimana sebagai gardu induk yang mensuplai Sirkuit Mandalika Gardu Induk Kuta sudah semestinya memiliki sistem yang andal untuk menunjang <a href="https://t.co/u0FIQzBwTV">https://t.co/u0FIQzBwTV</a>	electrizen pagelaran internasional pertama world superbike wsbk tahun 2023 bulan maret datang gelar sirkuit mandalika sebagai gardu induk suplai sirkuit mandalika gardu induk kuta mesti milik sistem andal tunjang
Konon Katanya Sirkuit Mandalika ngabisin Dana Hampir 2 Trilyun tp Lahan Warga kok nda Semuanya terbayar? Pada Kemana tu Duit larinya? Yg Lebih Aneh lagi kok \$epertinya KPK sama sekali tak Tertarik nyelidikin Malahan lbh Sibuk Nguyek2 Dana Formula E yg cuma habis 150 Miliar?	konon kata sirkuit mandalika ngabisin dana hampir 2 trilyun lahan warga kok nda semua bayar mana tu duit lari lebih aneh kok epertinya kpk sama sekali tarik nyelidikin malah lbh sibuk nguyek2 dana formula e cuma habis 150 miliar
Acara parade rider MotoGP di Jakarta termasuk saat dijamu Presiden Jokowi bikin bos Dorna Carmelo Ezpeleta merasa sangat bangga! #Mandalika #MotoGP <a href="https://t.co/jzMUysP9Kp">https://t.co/jzMUysP9Kp</a>	acara parade rider motogp jakarta masuk jamu presiden jokowi bikin bos dorna carmelo ezpeleta rasa sangat bangga

### 4.3. Word Embedding FastText

Untuk menggunakan model *word embedding* FastText sebenarnya ada beberapa cara diantaranya dengan menggunakan *Pretrained* model FastText atau melatih model FastText sendiri. *Pretrained* model adalah model yang telah dilatih pada dataset besar dan umum sebelumnya, sehingga dapat digunakan sebagai dasar untuk menyelesaikan tugas-tugas baru dengan lebih efisien. Namun menggunakan *pretrained* model ini memakan RAM yang sangat besar karena memiliki dataset yang banyak. Alternatifnya adalah penulis akan menggunakan package Fasttext python guna membuat model *word embedding* Fasttext tersendiri. Untuk itu penulis mengambil data artikel berbahasa Indonesia dari Wikipedia sebanyak 150.000 artikel.

Setelah itu, pelatihan model *word embedding* FastText berbahasa Indonesia dimulai.

```
# proses pelatihan model
model = fasttext.train_unsupervised('wiki-id-formatted.txt', model='skipgram')
```

Gambar 3. Pelatihan *Word Embedding*

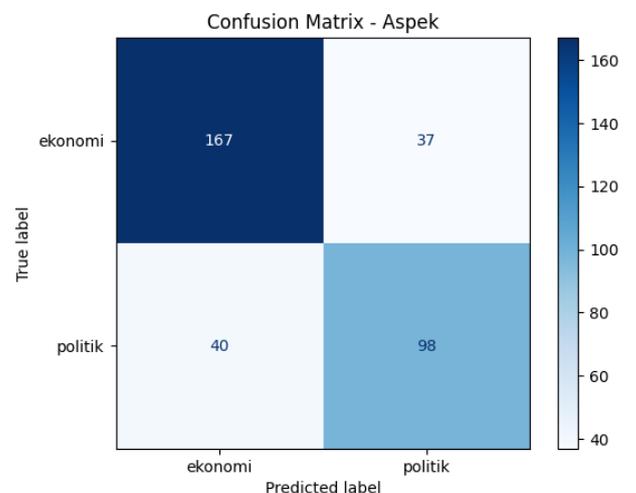
Algoritma yang dipilih untuk membuat model adalah *skipgram* dan ukuran dimensi secara *default* adalah 100.

### 4.4. Evaluasi CNN

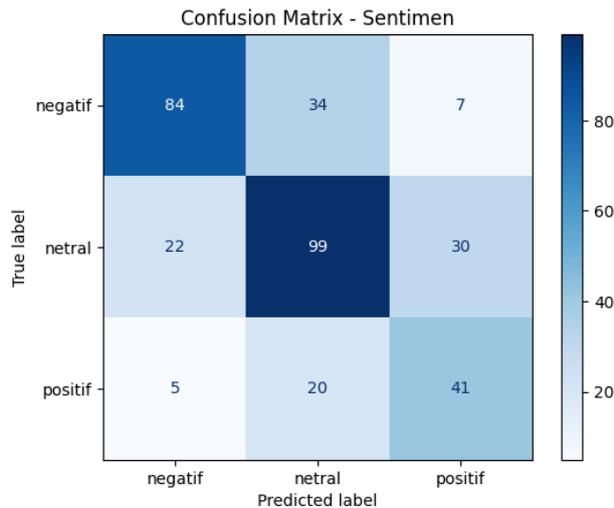
#### 4.4.1. Dengan *Embedding* Keras

Hasil rata-rata akurasi yang didapatkan pada CNN yang menggunakan *embedding* Keras adalah 79% untuk tugas klasifikasi label aspek dan 70% untuk tugas klasifikasi label sentimen. Dalam lima kali *fold*, akurasi aspek tertinggi didapatkan pada *fold* kedua yaitu 82%. Kemudian akurasi sentimen tertinggi didapatkan pada *fold* kedua yaitu 74%.

Berikut merupakan hasil *confusion matrix*.



Gambar 4. *Confusion Matrix* Aspek *Embedding* Keras



Gambar 5. *Confusion Matrix Sentimen Embedding Keras*

Berdasarkan *confusion matrix* di atas terlihat bahwa pada label aspek model dapat memprediksi 167 data kelas ekonomi dan 98 data kelas politik dengan benar. Terlihat model cukup baik dalam memprediksi data kelas ekonomi dibandingkan politik. Untuk label sentimen, model berhasil memprediksi 84 data kelas negatif, 99 data kelas netral, dan 41 data kelas positif dengan benar. Terlihat model cukup baik dalam memprediksi kelas negatif dan netral.

Classification Report - Aspek				
	precision	recall	f1-score	support
ekonomi	0.81	0.82	0.81	204
politik	0.73	0.71	0.72	138
accuracy			0.77	342
macro avg	0.77	0.76	0.77	342
weighted avg	0.77	0.77	0.77	342

Gambar 6. *Classification Report Aspek Embedding Keras*

Classification Report - Sentimen				
	precision	recall	f1-score	support
negatif	0.76	0.67	0.71	125
netral	0.65	0.66	0.65	151
positif	0.53	0.62	0.57	66
accuracy			0.65	342
macro avg	0.64	0.65	0.64	342
weighted avg	0.66	0.65	0.66	342

Gambar 7. *Classification Report Sentimen Embedding Keras*

*Classification report* menunjukkan bahwa pada label aspek performa model untuk memprediksi kelas ekonomi lebih baik dibandingkan kelas politik. Ini

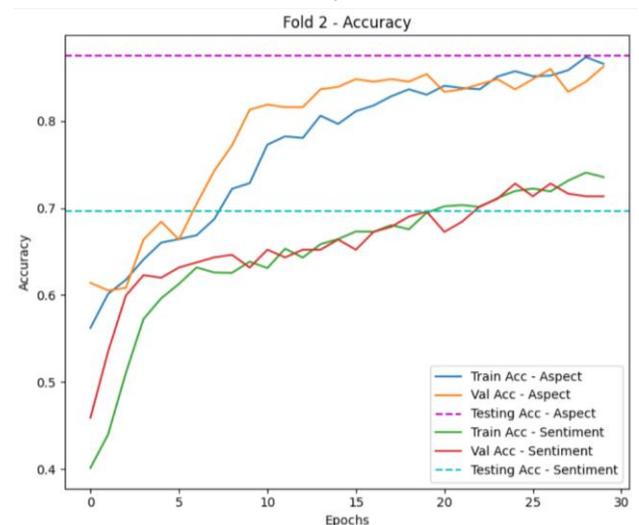
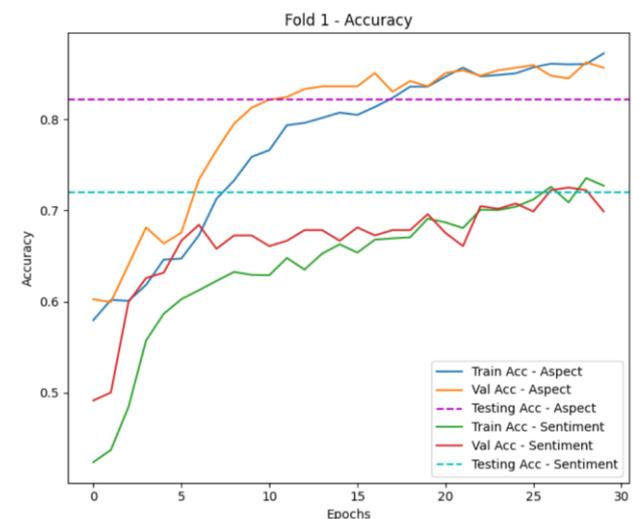
terlihat pada nilai *f1-score* kelas ekonomi 0,81 sedangkan kelas politik mendapatkan nilai 0,72.

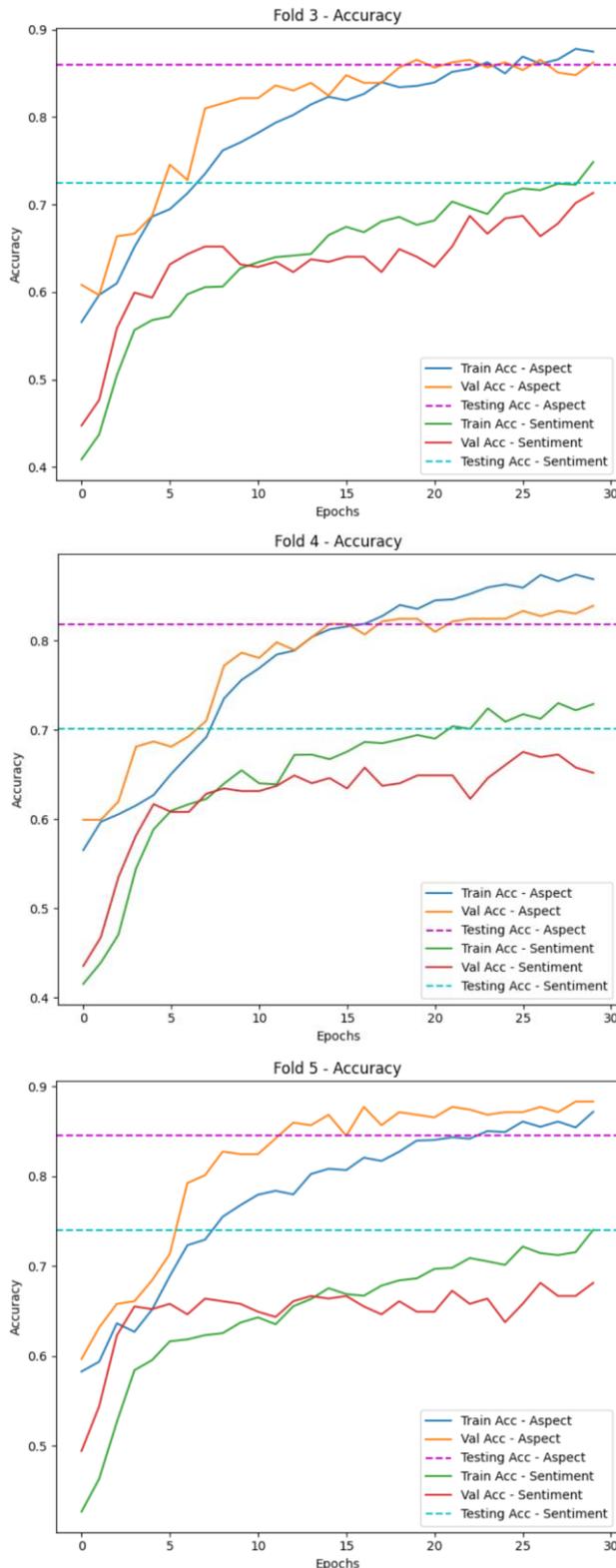
Lalu untuk label sentimen, terlihat model mendapatkan nilai *f1-score* 0,57 pada kelas positif; 0,65 pada kelas netral; dan 0,71 pada kelas negatif. Model menunjukkan performa yang lebih tinggi dalam memprediksi kelas negatif.

#### 4.4.2. Dengan *Embedding* FastText

Rata-rata Hasil akurasi pada CNN yang menggunakan model *embedding* FastText adalah 84% untuk tugas klasifikasi label aspek dan 72% untuk tugas klasifikasi label sentimen. Dengan lima kali *fold* yang digunakan akurasi aspek tertinggi didapatkan pada *fold* kedua yaitu 87%. Kemudian untuk akurasi sentimen tertinggi didapatkan pada *fold* kelima yaitu 74%.

Berikut penulis akan menampilkan grafik akurasi label aspek dan sentimen untuk CNN yang menggunakan *embedding* FastText pada setiap *fold*.

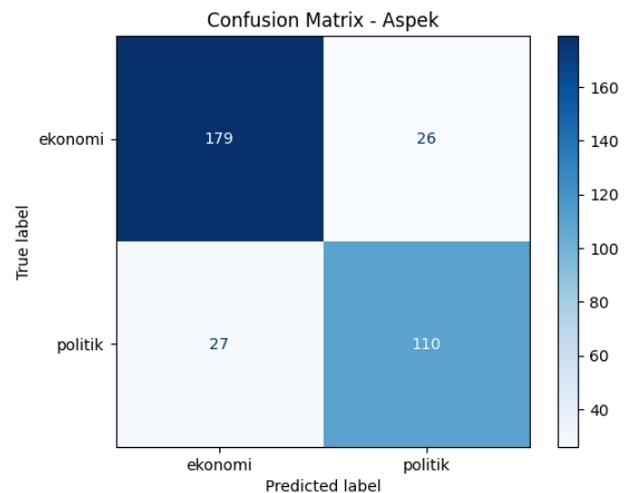




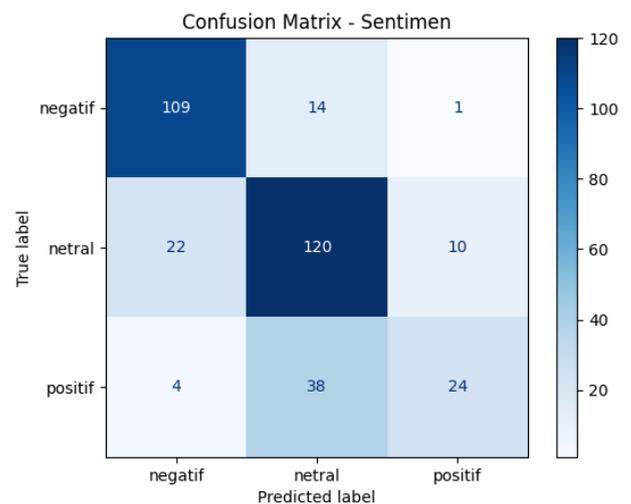
Gambar 8. Grafik Akurasi CNN Embedding FastText

Lima grafik di atas, sumbu X menunjukkan jumlah *epoch* yang telah dilakukan selama pelatihan sedangkan sumbu Y menunjukkan nilai akurasi model pada data *training*, *validation*, dan *testing*. Penulis

telah melakukan beberapa percobaan untuk mengurangi dan menambah jumlah *epoch* namun jumlah *epoch* yang menghasilkan akurasi paling optimal adalah sebanyak 30. Terlihat bahwa akurasi pada data *training* meningkat seiring bertambahnya *epoch* yang berarti model semakin baik dalam belajar dari data *training*. Kemudian akurasi pada data *validation*, untuk label aspek kesenjangan pada akurasi *training* dan *validation* cukup kecil sehingga dapat dikatakan model tidak *overfitting*. Namun untuk label sentimen terlihat adanya kesenjangan pada *fold* keempat dan kelima tetapi tidak terlalu besar.



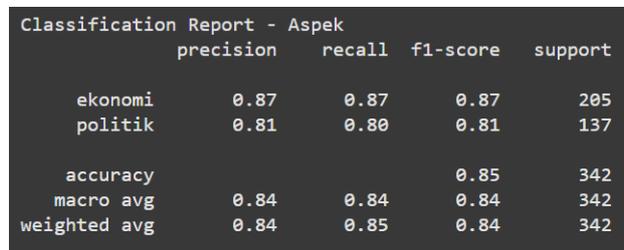
Gambar 9. Confusion Matrix Aspek Embedding FastText



Gambar 10. Confusion Matrix Sentimen Embedding FastText

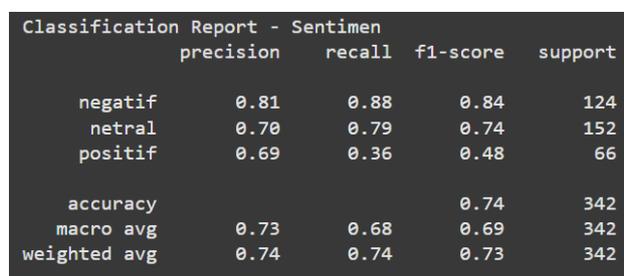
Pada gambar di atas terlihat pada label aspek model dapat memprediksi 179 data kelas ekonomi dan 110 data kelas politik dengan benar. Model lebih baik dalam memprediksi data kelas ekonomi dibandingkan

kelas politik. Kemudian untuk label sentimen model dapat memprediksi 109 data kelas negatif, 120 data kelas netral, dan 24 kelas positif dengan benar. Terlihat model cukup baik dalam memprediksi kelas negatif dan netral namun cukup kesulitan untuk memprediksi kelas positif.



	precision	recall	f1-score	support
ekonomi	0.87	0.87	0.87	205
politik	0.81	0.80	0.81	137
accuracy			0.85	342
macro avg	0.84	0.84	0.84	342
weighted avg	0.84	0.85	0.84	342

Gambar 11. Classification Report Aspek Embedding FastText



	precision	recall	f1-score	support
negatif	0.81	0.88	0.84	124
netral	0.70	0.79	0.74	152
positif	0.69	0.36	0.48	66
accuracy			0.74	342
macro avg	0.73	0.68	0.69	342
weighted avg	0.74	0.74	0.73	342

Gambar 12. Classification Report Sentimen Embedding FastText

Berdasarkan *classification report* di atas terlihat bahwa performa model dalam label aspek untuk memprediksi kelas politik sedikit lebih rendah dibandingkan kelas ekonomi. Ini terlihat pada nilai *f1-score* kelas politik 0,81 sedangkan kelas ekonomi mendapatkan 0,87.

Lalu untuk label sentimen, terlihat model mengalami kesulitan dalam memprediksi kelas positif namun cukup mumpuni untuk memprediksi kelas negatif. Nilai *f1-score* yang dihasilkan adalah 0,48 untuk kelas positif, 0,74 untuk kelas netral, dan 0,84 untuk kelas negatif.

Hasil akurasi yang rendah pada kelas positif disebabkan oleh ketidakseimbangan data. Data teks ulasan positif memiliki jumlah yang lebih sedikit dibandingkan data teks ulasan netral dan negatif. Ini menyebabkan model lebih sering belajar pada kelas yang dominan daripada kelas minoritas.

## 5. KESIMPULAN DAN SARAN

### 5.1. Kesimpulan

Berdasarkan hasil penelitian, sebagian besar masyarakat Indonesia menunjukkan sentimen yang bersifat netral terhadap Sirkuit Mandalika.

Akurasi yang didapatkan dengan menggunakan *word embedding* FastText lebih baik daripada *word embedding default* dari Keras. Ini terlihat dari akurasi model CNN yang didapatkan. Pada CNN yang menggunakan *embedding* Keras rata-rata akurasi label aspek yang didapatkan adalah 79% dan 70% untuk rata-rata akurasi label sentimen. Tetapi pada CNN yang menggunakan *embedding* FastText rata-rata akurasi yang didapatkan pada kedua label meningkat yaitu 84% untuk label aspek dan 72% untuk label sentimen. Model menunjukkan efektivitas yang lebih baik pada tugas klasifikasi aspek daripada tugas klasifikasi sentimen.

### 5.2. Saran

Hasil penelitian di atas terlihat bahwa kedua model CNN (model yang menggunakan *embedding* Keras dan *embedding* FastText) mendapatkan nilai *f1-score* yang cukup rendah untuk memprediksi kelas positif pada label sentimen. Hal ini dikarenakan data ulasan yang memiliki sentimen positif lebih sedikit atau bisa dikatakan tidak seimbang dengan data ulasan yang memiliki sentimen netral dan negatif.

Melihat hasil yang didapatkan, augmentasi data penting dilakukan karena dapat menambah variasi data minoritas guna menyeimbangkan data pada setiap kelas khususnya dalam hal ini data kelas positif. Namun karena keselarasan pada tujuan awal penelitian yaitu untuk mengevaluasi model dan representasi teks serta waktu penelitian yang terbatas, implementasi augmentasi data belum dapat dilakukan pada tahap ini. Untuk penelitian selanjutnya disarankan agar menggunakan teknik augmentasi data agar model semakin baik dalam mempelajari pola yang adil dan representatif dari setiap kelas sehingga diharapkan nilai akurasi pada setiap label akan meningkat.

### DAFTAR PUSTAKA

- [1] Adi Wikanto, "Fakta tentang Sirkuit Mandalika, akan diresmikan Jokowi hari ini (12/11)."
- [2] V. Rahmadana dan H. A. Prakoso, "PENYELENGGARAAN PAGELARAN OLAH RAGA BALAP DI SIRKUIT MANDALIKA SEBAGAI UPAYA PENINGKATAN NATION BRANDING INDONESIA," Jurnal MODERAT, vol. 8, no. 2.
- [3] Rosi Oktari, "Sirkuit Mandalika Siap Jadi Tuan Rumah WSBK 2021."
- [4] Yedi Supriadi, "Tanggal berapa MotoGP Mandalika 2022? Ini Jadwal Resmi Balapan MotoGP 2022 Khusus Sabtu dan Minggu."
- [5] N. P. Sri, W. Susila, dan S. Isnaini, "Medkom: Jurnal Media dan Komunikasi Strategi Public Relations

- Mandalika Grand Prix Association (MGPA) dalam Manajemen Event World Superbike 2022”, [Daring]. Tersedia pada: <https://e-journal.unair.ac.id/Medkom>
- [6] R. Merdiansah dan A. Ali Ridha, “Analisis Sentimen Pengguna X Indonesia Terkait Kendaraan Listrik Menggunakan IndoBERT,” *Jurnal Ilmu Komputer dan Sistem Informasi (JIKOMSI)*, vol. 7, no. 1, hlm. 221–228, 2024.
- [7] Cindy Mutia Annur, “Ada 27 Juta Pengguna Twitter di Indonesia, Terbanyak ke-4 Global.”
- [8] C. H. Yutika, A. Adiwijaya, dan S. Al Faraby, “Analisis Sentimen Berbasis Aspek pada Review Female Daily Menggunakan TF-IDF dan Naïve Bayes,” *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 5, no. 2, hlm. 422, Apr 2021, doi: 10.30865/mib.v5i2.2845.
- [9] N. Ayu Shafirra dan Irhamah, “Klasifikasi Sentimen Ulasan Film Indonesia dengan Konversi Speech-to-Text (STT) Menggunakan Metode Convolutional Neural Network (CNN),” *JURNAL SAINS DAN SENI ITS*, Vol. 9, No. 1, (2020)
- [10] A. Permana Putra dan A. Farrah Syafira, “Analisis Sentimen Data Twitter Topik Politik Dengan Metode Naive Bayes Dan Convolutional Neural Networks (Cnn),” *Jurnal Ilmiah Wahana Pendidikan*, Oktober 2023, 9 (20), 36-41, doi: <https://doi.org/10.5281/zenodo.8396579>.
- [11] N. Alvi Hasanah, N. Suciati, dan D. Purwitasari, “Pemantauan Perhatian Publik terhadap Pandemi COVID-19 melalui Klasifikasi Teks dengan Deep Learning”, *JURNAL RESTI (Rekayasa Sistem dan Teknologi Informasi)*, Vol. 5 No. 1 (2021) 193 - 202
- [12] D. Hidayatul Qudsi, J. Hakim Lubis, K. Umam Syaliman, dan N. Fadilah Najwa, “ANALISIS SENTIMEN PADA DATA SARAN MAHASISWA TERHADAP KINERJA DEPARTEMEN DI PERGURUAN TINGGI MENGGUNAKAN CONVOLUTIONAL NEURAL NETWORK”, doi: 10.25126/jtiik.202184842.
- [13] D. R. Putri, E. Y. Puspaningrum, dan H. Maulana, “KLASIFIKASI SENTIMEN TENTANG PEMINDAHAN IBU KOTA NEGARA INDONESIA DENGAN CONVOLUTIONAL NEURAL NETWORK MENGGUNAKAN GLOVE DAN FASTTEXT,” *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 12, no. 3, Agu 2024, doi: 10.23960/jitet.v12i3.4882.
- [14] S. N. Listyarini dan D. A. Anggoro, “Analisis Sentimen Pilkada di Tengah Pandemi Covid-19 Menggunakan Convolution Neural Network (CNN),” *Jurnal Pendidikan dan Teknologi Indonesia*, vol. 1, no. 7, hlm. 261–268, Jul 2021, doi: 10.52436/1.jpti.60.
- [15] A. Rinardi, C. Sri Kusuma Aditya, dan D. Rizki Chandranegara, “Analisis Sentimen Ulasan Fitur Music Aplikasi Instagram Pada Google Play Store Menggunakan Metode Convolutional Neural Network (CNN),” *REPOSITOR*, vol. 6, no. 4, hlm. 415–424, 2024.
- [16] T. Informasi dan P. Negeri Malang, “IMPLEMENTASI TEKNIK CRAWLING UNTUK PENGUMPULAN DATA DARI MEDIA SOSIAL TWITTER PRAMANA YOGA SAPUTRA,” 2017. [Daring]. Tersedia pada: [www.quicksprout.com](http://www.quicksprout.com)
- [17] M. U. Albab, Y. Karuniawati P, dan M. N. Fawaiq, “Optimization of the Stemming Technique on Text preprocessing President 3 Periods Topic,” vol. 20, no. 2, hlm. 1–10, 2023, doi: 10.26623/transformatika.v20i2.5374.
- [18] Jason Brownlee, "How to Use Word Embedding Layers for Deep Learning with Keras", *Deep Learning for Natural Language Processing*.
- [19] M. Irfani dan S. Khomsah, “Analisis Sentimen Berbasis Aspek pada EDOM Pembelajaran Menggunakan Metode CNN dan Word2vec,” *Jurnal Sistem dan Teknologi Informasi (JustIN)*, vol. 12, no. 3, hlm. 413, Jul 2024, doi: 10.26418/justin.v12i3.75610.