JUSTIN
Jurnal Sistem dan Teknologi Informasi
p-ISSN : 2460-3562 / e-ISSN : 2620-8989
DOI: 10.26418/justin.v12i3.75610
Vol. 12, No. 3, Juli 2024

# Analisis Sentimen Berbasis Aspek pada EDOM Pembelajaran Menggunakan Metode CNN dan Word2vec

Muhammad Irfani[#1], Siti Khomsah[#2]
[a]*Fakultas Informatika, Institut Teknologi Telkom Purwokerto*
*Jl. D.I Panjaitan No. 128 Purwokerto 53147, Jawa Tengah - Indonesia*
[1]20104037@ittelkom-pwt.ac.id
[2]siti@ittelkom-pwt.ac.id

**Abstrak**

Penelitian ini secara khusus mengeksplorasi analisis sentimen berbasis aspek pada Evaluasi Dosen Oleh Mahasiswa (EDOM) di Institut Teknologi Telkom Purwokerto (ITTP) dengan jumlah dataset sebanyak 5116. Dengan menerapkan metode *Convolutional Neural Network* (CNN) dan Word2Vec, tujuan utama penelitian adalah mengidentifikasi aspek-aspek yang muncul dalam sentimen opini mahasiswa terkait EDOM ITTP. Selain itu, penelitian ini berupaya mengevaluasi akurasi model klasifikasi sentimen berbasis aspek menggunakan kombinasi CNN dan Word2Vec, *confussion matrix* digunakan untuk mengukur tingkat akurasi model. Proses penelitian melibatkan penerapan teknik oversampling untuk mengatasi ketidakseimbangan data pada kelas sentimen dengan jumlah data. Dalam menanggulangi permasalahan tersebut, variasi metode oversampling, seperti SMOTE, Random Oversampling, ADASYN, SMOTE-NC, dan Borderline SMOTE, diimplementasikan. Hasil penelitian menunjukkan peningkatan signifikan dalam akurasi model CNN setelah menerapkan algoritma oversampling, mengukuhkan keberhasilan implementasi sentimen berbasis aspek pada EDOM ITTP. Penelitian ini memberikan kontribusi berharga dalam pemahaman dan pengembangan analisis sentimen, terutama dalam konteks pembelajaran, dengan mempertimbangkan aspek-aspek spesifik dalam opini mahasiswa. Temuan ini dapat menjadi dasar bagi perkembangan lebih lanjut dalam meningkatkan pengalaman evaluasi dosen oleh mahasiswa di lingkungan pendidikan tinggi.

**Kata kunci**: *Deep Learning*, CNN, word2vec, analisis sentimen, ABSA.

# Aspect-Based Sentiment Analysis in EDOM Learning Using CNN and Word2vec Methods

**Abstract**

This research specifically explores aspect-based sentiment analysis on Lecturer Evaluation by Students (EDOM) at the Telkom Purwokerto Institute of Technology (ITTP) with a total dataset of 5116. By applying the Convolutional Neural Network (CNN) and Word2Vec methods, the main aim of the research is to identify aspects- aspects that emerge in student opinion sentiment regarding EDOM ITTP. In addition, this research seeks to evaluate the accuracy of aspect-based sentiment classification models using a combination of CNN and Word2Vec, a confusion matrix is used to measure the level of model accuracy. The research process involves applying oversampling techniques to overcome data imbalances in sentiment classes with the amount of data. In overcoming this problem, various oversampling methods, such as SMOTE, Random Oversampling, ADASYN, SMOTE-NC, and Borderline SMOTE, were implemented. The research results show a significant increase in the accuracy of the CNN model after applying the oversampling algorithm, confirming the success of implementing aspect-based sentiment in EDOM ITTP. This research provides a valuable contribution to the understanding and development of sentiment analysis, especially in the learning context, by considering specific aspects of student opinion. These findings can serve as a basis for further developments in improving the experience of lecturer evaluation by students in higher education settings.

**Keywords**: *Deep Learning*, CNN, word2vec, analisis sentimen, ABSA.

## I. PENDAHULUAN

Peran penting dosen dalam mengajar mahasiswa tak terbantahkan. Mereka bertanggung jawab memberikan pengajaran berkualitas, membimbing mahasiswa mencapai tujuan akademik, dan mengembangkan keterampilan nyata. Survei Evaluasi Dosen Oleh Mahasiswa (EDOM) penting untuk meningkatkan performa dosen. Dengan EDOM, mahasiswa memberikan umpan balik tentang pengalaman

Submitted 18-01-2024; Revised 02-05-2024; Accepted 02-05-2024

belajar, mengidentifikasi kekuatan dan kelemahan dosen, serta memberikan saran perbaikan. Survei EDOM merupakan hal yang umum dilakukan berbagai perkuliahan tinggi, termasuk IT Telkom Purwokerto (ITTP). Survei EDOM berbentuk pertanyaan kuantitatif dan kualitatif. Pertanyaan kuantitatif lebih mudah untuk dianalisis karena dapat diukur secara langsung dengan data numerik, sementara pertanyaan kualitatif melibatkan jawaban teks yang lebih panjang dan memerlukan analisis konten yang lebih mendalam untuk memahami sentimen atau opini yang terkandung dalam teks tersebut, dengan adanya penelitian ini, komentar mahasiswa dalam EDOM itu biasanya dipengaruhi aspek-aspek yang ada dalam proses pembelajaran. Berdasarkan data yang diperoleh dari EDOM ITTP dan hasil survei yang penulis lakukan, maka penulis membuat sebuah model aspect based sentiment analysis untuk mengetahui sentimen dan aspek-aspek yang mempengaruhi kepuasan mahasiswa terhadap proses pembelajaran di ITTP.

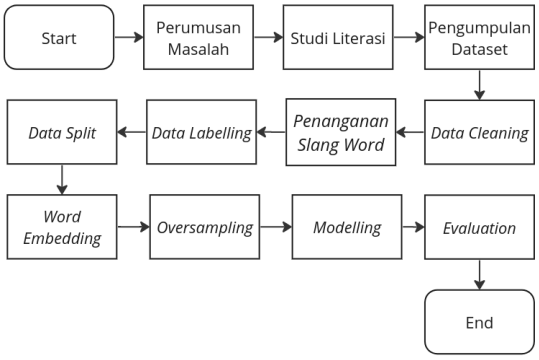
.

Data yang diperoleh dari EDOM belum bisa diolah secara kuantitatif, terutama untuk data survei pada bagian pengisian ulasan. Untuk mengetahui informasi yang lebih mendalam terkait kepuasan mahasiswa, dalam melakukan analisa tersebut bisa menggunakan metode *Sentiment Analysis* pada data EDOM. *Sentiment Analysis* adalah studi komputasi mengenai pendapat, penilaian, sikap, dan emosi terhadap entitas termasuk individu, isu, subjek maupun peristiwa dan atributnya.[1]. Sentimen dapat dibagi menjadi 2 kelas yaitu *positive* dan *negative*, untuk lebih dalam lagi dalam kelas sentimen dapat dikenali aspek-aspek yang menyebabkan opini itu mengarah pada kelas positif dan negatif. *Aspect-based sentiment analysis* (ABSA) dapat membantu menganalisis apa saja yang mempengaruhi komentar individu[2]. Komentar mahasiswa dalam edom itu biasanya dipengaruhi aspek-aspek yang ada dalam proses pembelajaran. Berdasarkan data yang diperoleh dari EDOM ITTP dan hasil survei yang penulis lakukan, , maka penulis mengajukan sebuah model *aspect based sentiment analysis* untuk mengetahui sentimen dan aspek-aspek yang mempengaruhi kepuasan mahasiswa terhadap proses pembelajaran di ITTP.

Data yang penulis peroleh adalah data yang diambil dari survei EDOM di ITTP, untuk melakukan perancangan *sentiment analysis* ini penulis akan menggunakan bahasa pemrograman *Python*. Metode yang penulis pakai nantinya akan menggunakan CNN untuk melakukan analisa apakah data dari survei tersebut apakah termasuk sentimen negatif atau positive, CNN merupakan jenis arsitektur jaringan syaraf tiruan yang khusus dirancang untuk memproses data *grid*, seperti gambar atau data berdimensi tinggi lainnya. CNN terinspirasi oleh cara pengolahan visual pada sistem saraf manusia dan secara khusus dirancang untuk mendeteksi pola visual kompleks. [3][13], penulis juga akan menggunakan word2vec, word2vec adalah metode *word embedding* yang dikembangkan oleh Mikolov dan sering digunakan dalam klasifikasi sentimen yang telah di-pretrained. Metode ini memiliki kemampuan untuk menangkap makna semantik teks dengan cara mewakili setiap kata dalam bentuk vektor yang memiliki kedekatan makna yang mirip. [5], Word2Vec mempunyai 2 arsitektur, yang pertama adalah arsitektur Skip-gram dan *Continuous Bag of Words* (CBOW), pada penelitian ini penulis akan menggunakan arsitektur dengan menggunakan CBOW, Prosesnya melibatkan pembelajaran dari teks untuk memahami hubungan antar kata. Model ini menggunakan jendela konteks yang mencakup beberapa kata sekitar kata target, dan dari situ, mencoba memprediksi kata target tersebut. Metode ini pertama kali dikemukakan oleh T.Mikolov dkk. [11]

Teknik pengujian yang akan penulis pakai adalah *confusion matrix*, *confusion matrix* adalah sebuah metode untuk melakukan evaluasi kinerja model klasifikasi dalam machine learning. *Confusion matrix* memberikan gambaran tentang seberapa baik model dapat melakukan klasifikasi data ke dalam kategori yang benar. *Confusion matrix* biasanya digunakan pada masalah klasifikasi biner, di mana ada dua kelas yang mungkin: kelas positif dan kelas negatif. Namun, konsep ini juga dapat diterapkan pada masalah klasifikasi dengan lebih dari dua kelas.[4].

## II. METODOLOGI PENELITIAN




Gambar 1 Metodologi Penelitian

Dalam Gambar 1 merupakan tahapan dari penelitian yang meliputi Perumusan Masalah, Studi Literasi, Pengumpulan Dataset, *Data Cleaning*, Penanganan *Slang Word*, *Data Labelling*, *Data Split*, *Word Embedding*, *Oversampling*, *Modelling* dan *Evaluation*

### *A. Perumusan Masalah*

Komentar-komentar mahasiswa pada EDOM di ITTP dipengaruhi oleh berbagai aspek terkait proses pembelajaran, sehingga perlu komputasi untuk mengetahui aspek apa saja yang mempengaruhi proses pembelajaran mahasiswa, menggunakan algoritma *deep learning* CNN dan *word embedding* menggunakan Word2Vec.

### *B. Studi Literature*

Penelitian pertama mengenai penerapan Analisis Sentimen dengan CNN dalam menganalisis ulasan bahasa Indonesia dilakukan oleh [6] pada tahun 2021. Tujuan penelitian tersebut adalah untuk menganalisis pendapat masyarakat

terkait produk di restoran tertentu. Penelitian ini menggunakan kombinasi model CNN dan *Contextualized Word Embedding*, yang kemudian dibandingkan dengan kombinasi model CNN dan *Traditional Word Embedding*. Pengujian klasifikasi dilakukan menggunakan tiga aspek model, yaitu BERT-CNN, ELMo-CNN, dan Word2vec-CNN. Hasil penelitian menyatakan bahwa klasifikasi sentimen memberikan hasil terbaik pada model BERT-CNN, dengan nilai precision sebesar 0.89, recall sebesar 0.89, dan f1-score sebesar 0.91.

Penelitian yang kedua tentang penerapan analisa sentimen pada gunung semeru menggunakan menggunakan data dari Google *Maps User Reviews*, model klasifikasi yang digunakan adalah menggunakan SVM, *Complement Naïve Bayes*, *Logistic Regression*, dan *transfer learning* dari *pre-trained* BERT, IndoBERT dan mBERT, kemudian untuk aspek yang digunakan pada penelitian tersebut adalah atraksi, fasilitas, akses, dan harga. Berdasarkan hasil percobaan menunjukkan bahwa *transfer learning* dari model IndoBERT menghasilkan hasil yang baik dengan akurasi dan F1-Score berturut-turut mencapai 91,48% dan 71,56%. Selain itu, di antara berbagai model lainnya yang digunakan, model SVM memberikan hasil terbaik dengan akurasi sebesar 89,16% dan F1-Score sebesar 62,23%[7].

Penelitian yang ketiga tentang implementasi analisa sentimen menggunakan metode STM, LSTM-CNN, CNN-LSTM dan Word2Vec pada media online dengan dataset yang diambil dari artikel, yaitu dari Detik *Finance*, penelitian tersebut bertujuan untuk melakukan klasifikasi berdasarkan sentimen positif dan negative, hasil dari penelitian tersebut menunjukkan bahwa pengujian dengan metode LSTM (*Long Short Term Memory*) mempunyai akurasi 62%, LSTM-CNN 65% dan CNN-LSTM mempunyai 74%[8]

Penelitian yang keempat tentang perbandingan pengaruh panjang kalimat dalam analisis sentimen menggunakan SVM dan CNN, dalam penelitian[9] menyatakan bahwa penggunaan panjang kalimat pada dataset akan mempengaruhi performa pada algoritma SVM dan CNN jika menggunakan word2vec, sedangkan jika menggunakan model SVM dengan TFIDF performa tidak begitu terpengaruh oleh panjang kalimat, meskipun seperti itu, kombinasi metode SVM+TFIDF memiliki waktu proses yang cepat dibandingkan metode lainnya. Adapun metode CNN+Word2vec menghasilkan performansi yang baik dengan nilai akurasi sebesar 0,94%, presisi sebesar 0,95%, recall sebesar 0,96%, dan f1-*score* sebesar 0,95%.

Penelitian kelima tentang optimisasi *sentiment analysis* dengan CNN dengan kombinasi *text preprocessing* dan word2vec yang dilakukan[10]. Penelitian tersebut bertujuan untuk melakukan identifikasi model yang paling akurat dan yang paling efektif dalam melakukan sentiment analysis, data yang dipakai menggunakan 20.986 ulasan dari 720 produk di marketplace shopee, pada penelitian tersebut menemukan bahwa kombinasi teknik *preprocessing* ketiga (*case folding, punctuation removal, word normalizer*, dan *stemming*), menyatakan bahwa kombinasi parameter word2vec kedua (size 50, window 2, hs 0, dan negative 10), dan kombinasi parameter CNN keempat (kernel size 2, *dropout* 0.2, dan learning rate 0.01) memiliki akurasi terbaik sebesar 99.00%, presisi 98.96%, dan *recall* 98.96%.

Penelitian keenam membahas tentang penggunaan NLP untuk keperluan analisa sentimen, penelitian ini akan menggunakan beberapa model deep learning, yaitu CNN, RNN (*Recurrent Neural Network*) dan LSTM. Tujuan dari penelitian ini adalah mengetahui perbedaan performa dari model *deep learning*. Hasil dari penelitian ini menunjukkan bahwa penggunaan CNN menempati posisi terbaik dengan akurasi training sebesar 97%, kemudian untuk posisi kedua ada pada *bidirectional* LSTM dengan nilai akurasi *training* mencapai 94%[12]. Berbagai penelitian telah menunjukkan bahwa penggunaan CNN meningkatkan akurasi model, seperti pada analisis sentimen berbasis aspek pada ulasan restoran di Indonesia [6], analisis sentimen produk di pasar online [10] dan Perbandingan Model *Deep Learning* untuk Klasifikasi *Sentiment Analysis*, dengan Teknik *Natural Language Processing*[12]. Selain itu, penggunaan Word2Vec sebagai metode word embedding juga terbukti efektif dalam meningkatkan akurasi model klasifikasi, seperti pada analisis sentimen dengan LSTM-CNN pada media online [8]. Temuan ini mengindikasikan potensi penggunaan CNN dan Word2Vec dalam meningkatkan akurasi model dalam berbagai konteks analisis sentimen.

### *C. Pengumpulan Dataset*

Dataset yang digunakan pada project ini adalah menggunakan data Evaluasi Dosen Oleh Mahasiswa (EDOM) dari tahun ajaran 21/22 dan 22/23, sebelum memasuki tahap data cleaning, maka diperlukan datasets berupa data komentar pada survei data EDOM, didapatkan data sebagai berikut:

Tabel 1

Sampel dataset mentah

| No | Dataset Mentah |
|---|---|
| 1 | Ibu mengajar matkul dengan baik semoga kedepannya bisa lebih baik lagi |
| 2 | tetap semangat pak, terima kasih sudah mengajar saya selama satu semester |
| 3 | Tidak ada masukan dari Saya Bu, karena saya dapat memahami materi yang Ibu jabarkan selama Semester 1 ini Bu |
| 4 | sudah sangat baik |
| 5 | Materi yang diberikan sudah sangat jelas, penjelasan mudah dipahami |
| 6 | Sudah Sangat Baik Dalam Pembelajaran |
| 7 | ? |
| 8 | - |
| 9 | - |

### *D. Data Cleaning*

Setelah mendapatkan dataset, maka diperlukan *data cleaning* sebelum data digunakan untuk tahap selanjutnya, perlunya data cleaning karena digunakan untuk melakukan

penghapusan data yang tidak diperlukan yang bisa mempengaruhi performa dari model. Pada proses *data cleaning* ini akan dilakukan penyaringan kata untuk menghilangkan simbol-simbol, kolom yang kosong, karakter khusus dan emotikon. Proses ini bertujuan agar saat pengolahan data tidak terjadi error yang tinggi dikarenakan data yang tidak penting. Contoh data yang akan dihapus adalah hastag (#), bintang (*) dan masih banyak lagi karakter khusus yang akan dihapus.

### *E. Penanganan Slang Word*

Setelah melewati *data cleaning*, maka tahap selanjutnya adalah melakukan penanganan *slang word*, *slang word* merupakan kata-kata yang tidak formal, penanganan *slang word* sendiri merupakan cara agar meningkatkan kualitas dari data sebelum memasuki tahap selanjutnya. Ilustrasi Penanganan *Slang Word* ada pada tabel 2.

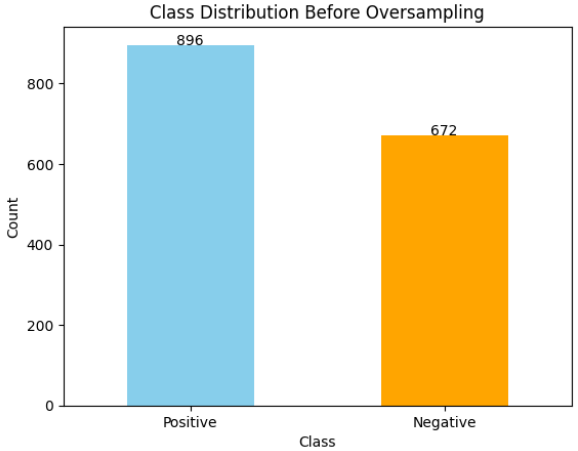
Tabel 2

Ilustrasi Penanganan *Slang Word*

| Teks mengandung slang | Teks tidak mengandung slang |
|---|---|
| Kurangi marah ya ibu, biar awet muda dan anak mahasiswa nya senang sama budos ny | Kurangi marah ya ibu, biar awet muda dan anak mahasiswa nya senang sama ibu dosen nya |

### *F. Data Labelling*

Data yang sudah melewati penanganan *slang word* akan melewati pelabelan data terlebih dahulu. Hal ini dilakukan agar sistem dapat memahami makna dari setiap kata yang akan diujikan. *Labelling* untuk mencari apakah kalimat tersebut merupakan kalimat *positive* dan *negative* dilakukan secara otomatis, kemudian untuk *labelling* aspek apa yang terkandung dalam data yang sudah ada label *negative* atau *positive* akan menggunakan *labelling* data secara manual oleh manusia. Tahap pertama akan dilakukan pelabelan dengan cara menggunakan model sentimen yang sudah ada, yaitu pre-trained model dari website huggingface dengan nama bert-base-indonesian-1.5G-sentiment-analysis-smsa. Kemudian dilanjut dengan melakukan pelabelan manual untuk mencari aspek apa saja yang ada dalam teks tersebut, melakukan pengecekan pada setiap label dan mengganti label Netral menjadi Positif atau Negatif secara manual, setelah itu dilanjut dengan penghapusan kata-kata yang tidak bermakna secara aspek.



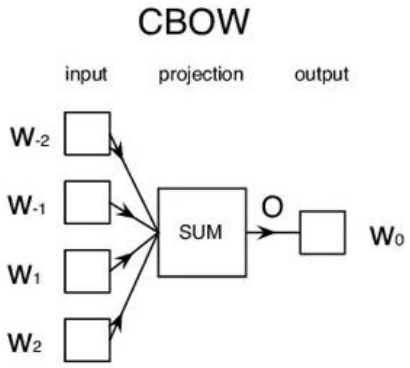

Gambar 2. Dataset Sebelum Oversampling

### *G. Data Split*

Data *splitting* pada klasifikasi berbasis teks untuk model analisis sentimen dan pembuatan model *aspect* melibatkan pembagian dataset teks menjadi dua subset yang berbeda, yaitu untuk tujuan pelatihan (*training*) sebanyak 80%, untuk tujuan pengujian (*testing*) sebanyak 20%. Tujuan utama dari pembagian data ini adalah untuk menguji kinerja model pada data yang belum pernah dilihat sebelumnya dan memastikan bahwa model mampu menggeneralisasi informasi dengan baik pada data teks baru. Pada tahap ini juga akan dilakukan label mapping, dimana label mapping akan mengubah label yang ada di dalam dataset menjadi label angka.

### *H. Word Embedding*

Word2Vec adalah salah satu metode *word embedding* yang berguna untuk menjadikan kata menjadi sebuah vektor. Arsitektur Word2vec hanya terdiri dari 3 layer yaitu *Input*, *Projection* (*Hidden Layer*) dan *Output*. Pada penelitian ini akan menggunakan arsitektur CBOW, CBOW adalah salah satu metode dalam model Word2Vec, yang merupakan suatu teknik dalam pengolahan bahasa alami (*Natural Language Processing*/NLP) untuk menghasilkan representasi vektor kata (*word embeddings*) dari teks. Ilustrasi dari CBOW dapat dilihat pada gambar 3




Gambar 3 Contoh Implementasi CBOW

### *I. Oversampling*

Pada tahap ini akan dilakukan *oversampling* setelah *word embedding*, alasan utama dilakukannya *oversampling* setelah *word embedding* adalah untuk meningkatkan representasi dan pembelajaran dari kelas-kelas minoritas setelah proses representasi kata-kata (*word embedding*) dilakukan.

Hal ini dapat mencegah model CNN cenderung memihak satu kelas dan akan membantu meningkatkan akurasi pada model CNN [14].

Pada tahap ini akan dilakukan percobaan untuk mencari metode atau algoritma oversampling yang terbaik untuk model CNN nanti, berikut merupakan tabel experiment

yang akan digunakan untuk mencari algoritma oversampling terbaik:

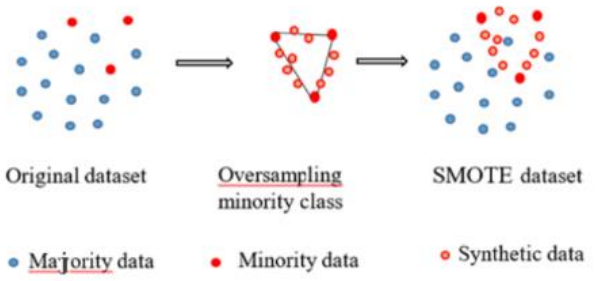
Tabel 3

Algoritma Oversampling yang akan digunakan

| Algoritma *Oversampling* |
|---|
| SMOTE |
| Random |
| ADASYN |
| SMOTE-NC |
| Borderline SMOTE |

Berikut merupakan penjelasan algoritma dari tabel 3:

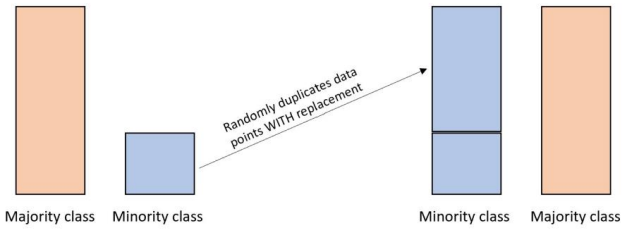
1. SMOTE (*Synthetic Minority Over-sampling Technique*) adalah teknik dalam *machine learning* untuk menangani ketidakseimbangan kelas. Cara kerjanya melibatkan pemilihan instansi minoritas, identifikasi tetangga terdekat, dan pembuatan instansi sintetis dengan mempertimbangkan perbedaan nilai fitur. Langkah terakhirnya adalah menghubungkan instansi asli dengan instansi sintetis, menciptakan dataset yang seimbang. SMOTE membantu mencegah bias model terhadap kelas mayoritas dan meningkatkan kemampuan model untuk menggeneralisasi pada contoh kelas minoritas. Meskipun berguna, penggunaan SMOTE memerlukan pertimbangan dan evaluasi hati-hati pada dampaknya terhadap kinerja model. [16] Berikut merupakan ilustrasi dari SMOTE:




Gambar 4. Ilustrasi SMOTE

2. Random Oversampling

*Random Oversampling* adalah teknik *oversampling* di mana instansi dari kelas minoritas secara acak dipilih dan disalin untuk meningkatkan jumlahnya. Tujuannya adalah menciptakan keseimbangan antara kelas mayoritas dan minoritas dengan meningkatkan proporsi instansi kelas minoritas. Meskipun sederhana, metode ini dapat membantu model *machine learning* untuk memahami dan memprediksi kelas minoritas dengan lebih baik. [17] Berikut merupakan ilustrasi dari *random oversampling*:




Gambar 5. Ilustrasi Random Oversampling

3. ADASYN (*Adaptive Synthetic Sampling*) adalah varian dari teknik SMOTE yang dirancang untuk menangani ketidakseimbangan kelas dalam *machine learning*. ADASYN memperkenalkan elemen adaptif dengan memberikan bobot pada setiap instansi minoritas berdasarkan tingkat kesulitannya dalam klasifikasi. Ini berarti instansi yang lebih sulit untuk diklasifikasikan menerima lebih banyak perhatian dalam pembuatan instansi sintetis. [18] Berikut merupakan algoritma dari ADASYN:

(1) Hitung tingkat ketidakseimbangan kelas:

Gunakan rumus d = ms/ml, di mana ms adalah jumlah instansi kelas minoritas dan ml adalah jumlah instansi kelas mayoritas. Nilai d berada di rentang (0, 1].

(2) Jika d < dth (suatu ambang batas tertentu):

(a) Hitung jumlah instansi sintetis yang perlu dihasilkan untuk kelas minoritas:

$G = (ml - ms) \times \beta$, di mana $\beta \in [0, 1]$ adalah parameter untuk tingkat keseimbangan yang diinginkan.

(b) Untuk setiap instansi xi di kelas minoritas, temukan K tetangga terdekat berdasarkan jarak Euclidean dan hitung rasio ri:

ri = Δi/K, di mana Δi adalah jumlah tetangga terdekat xi yang termasuk dalam kelas mayoritas.

(c) Normalisasi ri sehingga rˆi = ri/ms i=1, sehingga rˆi merupakan distribusi kepadatan (rˆi = 1).

(d) Hitung jumlah instansi sintetis yang perlu dihasilkan untuk setiap instansi minoritas xi:

gi = ˆri × G.

(e) Untuk setiap instansi minoritas xi, hasilkan gi instansi sintetis dengan langkah-langkah berikut:

Lakukan perulangan dari 1 hingga gi:

(i) Pilih secara acak satu instansi minoritas, xzi, dari K tetangga terdekat xi.

(ii) Hasilkan instansi sintetis:

si = xi + (xzi – xi) × λ, di mana (xzi – xi) adalah vektor perbedaan di ruang n dimensi, dan λ adalah angka acak di [0, 1].

Akhiri perulangan.

Algoritma ini bertujuan untuk menghasilkan instansi sintetis di sekitar instansi minoritas yang memiliki kepadatan rendah, sehingga menciptakan keseimbangan kelas yang lebih baik dalam dataset.[19]

4. SMOTE-NC (SMOTE *for Nominal and Continuous features*) adalah variasi dari SMOTE yang dirancang khusus untuk mengatasi ketidakseimbangan kelas dalam dataset yang memiliki fitur kategorikal (nominal) dan fitur kontinu. SMOTE tradisional dirancang untuk mengatasi ketidakseimbangan pada fitur kontinu, namun SMOTE-NC memperluas konsep tersebut untuk dapat menangani fitur kategorikal juga. Berbeda dengan SMOTE yang hanya bekerja pada fitur kontinu, SMOTE-NC memperlakukan fitur kontinu dan kategorikal secara terpisah. Proses pembuatan instansi sintetis di SMOTE-NC menggabungkan teknik SMOTE untuk fitur kontinu dan teknik lain seperti pertukaran nilai untuk fitur kategorikal.[20] Berikut merupakan algoritma dari SMOTE-NC

Tabel 4

Algoritma SMOTE-NC

```
Input:
  t = jumlah sampel kelas minoritas dalam set pelatihan
  n% = jumlah oversampling
  k = jumlah tetangga terdekat yang akan dipertimbangkan
  s = total jumlah instansi dalam set pelatihan
  c = jumlah variabel kontinu dalam dataset
  m = median deviasi standar fitur kontinu ketika c > 0

ir = t/s

untuk setiap fitur kategorikal, lakukan
  untuk setiap "l" pada label yang berbeda, lakukan
    e = jumlah total instansi dengan label "l" dalam set pelatihan
    e0 = e * ir
    o = jumlah instansi minoritas dengan label "l" dalam set pelatihan
    χ = (o - e0) / e0

    jika c > 0,
      l = χ * m
    else,
      l = χ
  end
 end
end

Terapkan SMOTE (t, n, k)

Untuk poin data sintetis, nilai atribut kategorikal ditentukan sebagai nilai mayoritas di antara tetangga terdekatnya.
Inverse-encode nilai kategorikal ke label aslinya.
```

5. Borderline-SMOTE adalah variasi dari teknik SMOTE yang difokuskan pada pembuatan instansi sintetis hanya di sekitar batas keputusan (*borderline*) antara kelas minoritas dan mayoritas. Berbeda dengan SMOTE yang secara acak memilih instansi minoritas, Borderline-SMOTE lebih selektif dalam menghasilkan instansi sintetik. Secara singkat, Borderline-SMOTE bertujuan untuk meningkatkan kualitas pembuatan instansi sintetis dengan memfokuskan pada titik-titik batas antara kelas. Hal ini dapat membantu mengatasi ketidakseimbangan kelas dengan lebih efektif karena hanya memperkenalkan instansi sintetis di area yang penting untuk memperbaiki masalah klasifikasi pada kelas minoritas. [21] Berikut merupakan algoritma dari Borderline-SMOTE [22]:

(i). Set pelatihan adalah T, kelas minoritas adalah P, dan mayoritas adalah M.

(ii). Untuk setiap pi (i=1,2,3,...,pnum) dalam kelas minoritas P:

mencari m-tetangga terdekat dari seluruh set pelatihan T (minoritas dan mayoritas).

(iii). Untuk setiap Pi, s tetangga terdekatnya dari k tetangga terdekat dalam P dipilih secara acak.

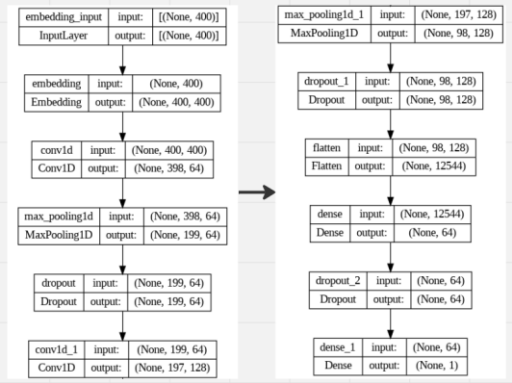
(iv). Sistem menghitung perbedaan difj antara P'i dan s tetangga terdekat dari P kemudian mengalikan perbedaan ini dengan angka acak rj antara 0 dan 1, dan hasil perkalian ditambahkan ke P'i. Data sintetis baru ditandai sebagai:

$$\text{Synthetic}_j = P_i' + r_j \times \text{dif}_j \text{dimana}$$
$$j = 1, 2, ..., s$$

*J. Modeling*

Proses analisis data pada penelitian ini dilakukan dengan menggunakan python dengan *tools Google Colab*. Data yang sudah melewati proses *word embedding* dan *oversampling* selanjutnya akan dilakukan modelling dengan menggunakan algoritma CNN. Tingkat akurasi yang dihasilkan dari proses ini sangat dipengaruhi oleh proses *text preprocessing* (*data cleaning*, penanganan

*slang word, sop word removing*) dan *data labelling*. apabila proses tersebut tidak dilakukan secara benar maka tingkat akurasi akan terpengaruh. Berikut merupakan contoh diagram arsitektur CNN yang akan digunakan ada pada Gambar 5.




Gambar 6 Struktur Model CNN

Pada gambar di atas merupakan struktur model secara berurutan dengan menggunakan kelas *Sequential*. Prosesnya dimulai dengan penambahan *layer embedding*, yang berfungsi untuk menangani representasi kata dalam bentuk vektor. Setelahnya, kita menambahkan layer Conv1D dengan 64 *filter*, ukuran *kernel* 3, dan menggunakan fungsi aktivasi ReLU. Langkah ini diikuti oleh layer MaxPooling1D dengan ukuran pooling 2, yang membantu mereduksi dimensi data. Untuk menerapkan regularisasi, kita menyisipkan *dropout* setelah layer ini. Kemudian menambahkan layer Conv1D lainnya dengan 128 *filter*, ukuran *kernel* 3, dan fungsi aktivasi ReLU. Seperti sebelumnya, kita menggunakan layer MaxPooling1D untuk mereduksi dimensi, diikuti oleh *dropout* untuk regularisasi.

Agar data dapat diolah lebih lanjut, kita memasukkan layer *Flatten*, yang mengubah matriks data menjadi vektor satu dimensi. Setelah tahap ini, kita memasukkan layer *Dense* dengan 64 unit dan fungsi aktivasi ReLU untuk melakukan operasi *non-linear*. *Dropout* kembali digunakan untuk mengurangi kemungkinan *overfitting* pada tahap ini. Akhirnya, kita menambahkan *layer Dense* terakhir dengan 1 unit dan aktivasi sigmoid. Ini adalah langkah krusial untuk masalah klasifikasi biner, dimana aktivasi sigmoid digunakan untuk menghasilkan *output* dalam rentang 0 hingga 1, sesuai dengan probabilitas kelas yang diinginkan. Dengan demikian, blok ini membentuk fondasi model CNN yang terstruktur dan dioptimalkan untuk penyelesaian masalah klasifikasi.

### *K. Evaluation*

Setelah proses pemodelan selesai dilakukan, proses akan dilanjut dengan proses *testing* dengan data *testing* yang sudah dipisah dengan dataset utama dan evaluasi untuk mendapatkan nilai performa model terbaik. Validasi model digunakan untuk menampilkan nilai akurasi dari model yang sudah dilakukan. Proses validasi dan evaluasi pada model ini menggunakan *Confusion Matrix*. *Library* yang digunakan untuk mendukung penggunaan *Confusion Matrix* adalah sklearn. *Library* yang dirujuk akan dipanggil dengan menggunakan *syntax* sklearn.metrics.confusion matrix, *Confusion Matrix* akan menghasilkan nilai *accuracy*, *precision*, *recall* dan *f1 score* [15].

## III. HASIL DAN PEMBAHASAN

### *A. Hasil Pelabelan Sentimen dan Aspek*

Tahap ini bertujuan untuk melakukan analisis sentimen pada dataset teks berbahasa Indonesia setelah proses pembersihan kata-kata slang. Pertama, dataset dibaca dari file CSV yang telah dibersihkan. Selanjutnya, dilakukan inisialisasi untuk menggunakan model analisis sentimen BERT berbahasa Indonesia yang telah di-pretrain dengan dukungan GPU. Model tersebut diperoleh dari Hugging Face menggunakan https://huggingface.co/ayameRushia/bert-base-indonesian-1.5G-sentiment-analysis-smsa. Proses analisis sentimen dilakukan pada seluruh data teks dalam dataset, dan hasilnya berupa label sentimen dan skor sentimen diekstrak dari output model. Hasil analisis sentimen kemudian ditambahkan ke dalam *DataFrame*. Setelah proses ini selesai, *DataFrame* yang telah diperbarui disimpan ke dalam file Excel baru. Tahap ini merupakan bagian dari proses pengembangan model dan penilaian sentimen dalam dataset. Setelah dataset dibersihkan dari kata-kata slang dan diberi label sentimen, data tersebut siap untuk digunakan dalam tahap analisis atau tugas pemrosesan teks lainnya. Berikut merupakan sampel dari hasil data labelling sentimen.

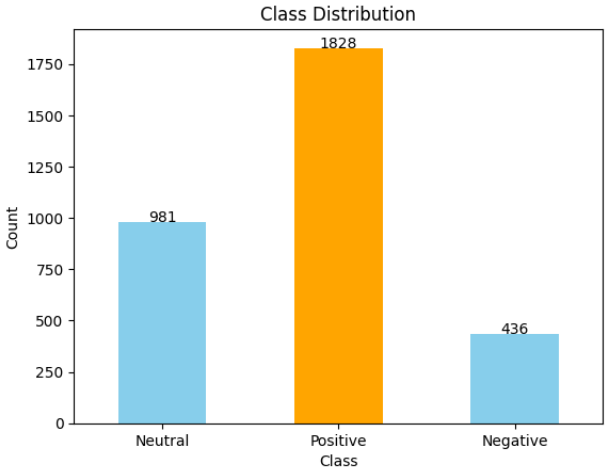
Tabel 5

Sampel hasil data labelling sentimen

| Teks | Sentimen |
|---|---|
| Ibu mengajar matkul dengan baik semoga kedepannya bisa lebih baik lagi | Positive |
| tetap semangat pak, terima kasih sudah mengajar saya selama satu semester | Positive |
| Tidak ada masukan dari Saya Bu, karena saya dapat memahami materi yang Ibu jabarkan selama Semester 1 ini Bu | Positive |
| sudah sangat baik | Positive |
| Materi yang diberikan sudah sangat jelas, penjelasan mudah dipahami | Positive |
| Sudah Sangat Baik Dalam Pembelajaran | Positive |
| terimakasih atas materi yang telah disampaikan | Neutral |
| Tidak ada masukan, karena semua sangat baik. | Positive |
| lebih ditingkatkan kembali dalam mengajar nya | Negative |
| beri jeda waktu di tengah perkuliahan untuk mahasiswa mendalami materi yang disampaikan | Neutral |

Dari hasil tabel 5 merupakan hasil setelah melakukan pelabelan dengan menggunakan *pre-trained* model di *huggingface*. Teks yang terindikasi sebagai label positif memiliki contoh seperti ini “Ibu mengajar matkul dengan baik semoga kedepannya bisa lebih baik lagi”, pada teks tersebut merupakan contoh dari teks yang mengandung label positif karena pada teks tersebut mahasiswa memberikan sentimen yang baik terhadap cara mengajar

dosen kepada mahasiswa, kemudian untuk contoh teks yang mengandung sentimen negatif akan seperti ini “lebih ditingkatkan kembali dalam mengajar nya”, pada kalimat ini mengandung permintaan agar dosen lebih meningkatkan lagi cara mengajar dosen terhadap mahasiswa. Berikut merupakan grafik dari hasil sentimen




Gambar 7. Hasil Pelabelan Sentimen

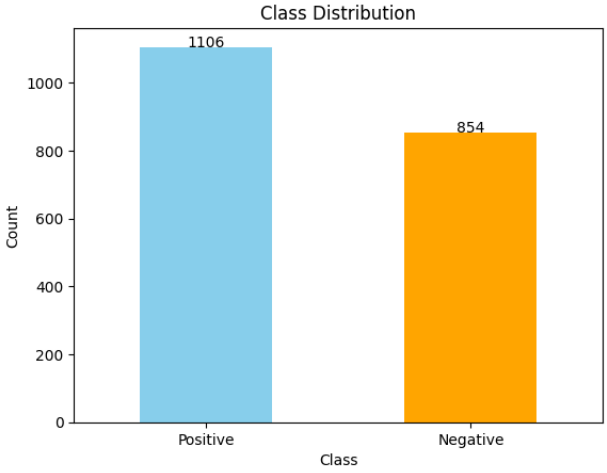
Dari hasil pelabelan yang didapatkan, mendapatkan sejumlah data sebanyak 981 untuk kelas netral, 1828 untuk kelas positif dan 436 untuk kelas negatif, setelah itu akan dilakukan penggantian label Netral menjadi Positif atau Negative, pada tahap ini juga akan dilakukan pengecekan pada setiap label yang ada. Tabel 6 merupakan hasil setelah mengganti label Netral menjadi label yang lebih cocok dan melakukan pengecekan pada setiap label.

Tabel 6

Sampel hasil menganti label Neutral ke label yang lebih cocok

| Teks | Sentimen |
|---|---|
| Ibu mengajar matkul dengan baik semoga kedepannya bisa lebih baik lagi | Positive |
| tetap semangat pak, terima kasih sudah mengajar saya selama satu semester | Positive |
| Tidak ada masukan dari Saya Bu, karena saya dapat memahami materi yang Ibu jabarkan selama Semester 1 ini Bu | Positive |
| sudah sangat baik | Positive |
| Materi yang diberikan sudah sangat jelas, penjelasan mudah dipahami | Positive |
| Sudah Sangat Baik Dalam Pembelajaran | Positive |
| terimakasih atas materi yang telah disampaikan | Neutral |
| Tidak ada masukan, karena semua sangat baik. | Positive |
| lebih ditingkatkan kembali dalam mengajar nya | Negative |
| beri jeda waktu di tengah perkuliahan untuk mahasiswa mendalami materi yang disampaikan | Negative |

Pada tabel 5 mengandung teks dengan label *neutral*, pada penelitian ini label *neutral* tidak dibutuhkan, maka karena itu label neutral harus diubah ke dalam label yang lebih cocok, pada contoh tabel diatas, teks “beri jeda waktu di tengah perkuliahan untuk mahasiswa mendalami materi yang disampaikan” dapat dikaitkan dengan sentimen negatif, karena mahasiswa meminta pengajar untuk memberikan jeda waktu kepada mahasiswa agar mahasiswa dapat mendalami materi. Berikut merupakan keseluruhan hasil dari tabel 6.




Gambar 8. Hasil pengecekan dan penggantian label netral

Pada gambar 8 didapatkan bahwa untuk kelas netral sudah tidak ada dan diganti dengan kelas lainnya, jumlah total data yang didapatkan adalah 1,106 untuk kelas positif dan 854 untuk kelas negatif. Setelah proses pelabelan dengan *pre-trained* model dan penggantian label Neutral ke label yang lebih sesuai selesai, maka akan dilanjut ke tahap pelabelan untuk mencari aspek apa yang terkandung di dalam teks, pada proses ini dilakukan secara manual oleh penulis. Tabel 7 merupakan sampel hasil dari pelabelan aspek.

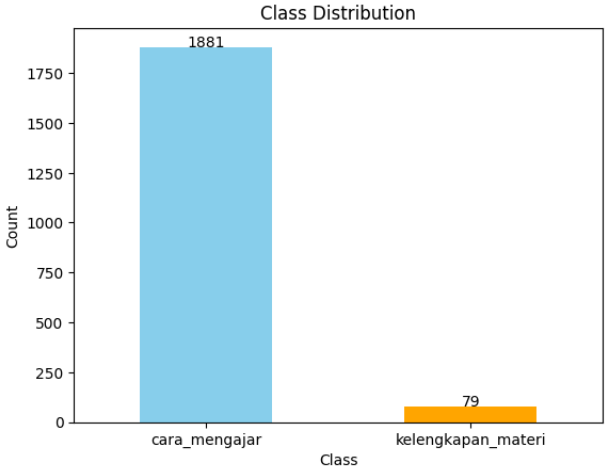
Tabel 7

Sampel hasil pelabelan aspek

| Teks | Label Sentimen | Label Aspek |
|---|---|---|
| Ibu mengajar matkul dengan baik semoga kedepannya bisa lebih baik lagi | Positive | cara_mengajar |
| tetap semangat pak, terima kasih sudah mengajar saya selama satu semester | Positive | cara_mengajar |
| Tidak ada masukan dari Saya Bu, karena saya dapat memahami materi yang Ibu jabarkan selama Semester 1 ini Bu | Positive | kelengkapan_materi |
| sudah sangat baik | Positive | cara_mengajar |
| Materi yang diberikan sudah sangat jelas, penjelasan mudah dipahami | Positive | cara_mengajar |
| Sudah Sangat Baik Dalam Pembelajaran | Positive | cara_mengajar |

| | | |
|---|---|---|
| terimakasih atas materi yang telah disampaikan | Neutral | cara_mengajar |
| Tidak ada masukan, karena semua sangat baik. | Positive | cara_mengajar |
| lebih ditingkatkan kembali dalam mengajar nya | Negative | cara_mengajar |
| beri jeda waktu di tengah perkuliahan untuk mahasiswa mendalami materi yang disampaikan | Negative | cara_mengajar |

Aspek yang ada dalam penelitian ini adalah aspek cara mengajar dan kelengkapan materi, teks dengan indikasi label cara mengajar adalah sebagai berikut, “lebih ditingkatkan kembali dalam mengajar nya”, teks tersebut masuk kedalam label cara mengajar karena pada teks tersebut mahasiswa memberikan saran ke pengajar untuk lebih meningkatkan cara mengajar pengajar kepada mahasiswa, kemudian untuk contoh label kelengkapan materi adalah “Tidak ada masukan dari Saya Bu, karena saya dapat memahami materi yang Ibu jabarkan selama Semester 1 ini Bu”, teks tersebut masuk ke dalam label kelengkapan materi karena dalam teks tersebut mahasiswa sudah memahami materi yang diberikan oleh pengajar, dalam arti materi yang diberikan oleh pengajar sudah lengkap dan dimengerti oleh mahasiswa. Berikut merupakan grafik distribusi dari label aspek:




Gambar 9. Grafik distribusi kelas aspek

Dalam dua *dataset* yang diberikan, terlihat adanya ketidakseimbangan data antara kelas positif dan negatif pada dataset pertama serta antara kelas "Cara Mengajar" dan "Kelengkapan Materi" pada dataset kedua. Pada dataset pertama, kelas positif memiliki 1,106 sampel, sedangkan kelas negatif hanya memiliki 854 sampel. Ketidakseimbangan ini bisa menjadi perhatian serius dalam pelatihan model karena dapat menyebabkan model cenderung lebih memihak pada kelas mayoritas, yaitu kelas positif.

Sementara itu, pada dataset kedua, kelas "Cara Mengajar" memiliki 1,881 sampel, sementara kelas "Kelengkapan Materi" hanya memiliki 79 sampel. Kembali, ketidakseimbangan ini dapat menyulitkan model untuk mengenali dan mempelajari pola dari kelas minoritas, dalam hal ini adalah kelas "Kelengkapan Materi". Untuk mengatasi masalah ketidakseimbangan dalam kedua dataset tersebut, pendekatan *oversampling* dapat diterapkan. Dengan *oversampling*, jumlah sampel pada kelas minoritas akan ditingkatkan sehingga menciptakan proporsi yang lebih seimbang antara kelas positif dan negatif pada dataset pertama, serta antara kelas "Cara Mengajar" dan "Kelengkapan Materi" pada dataset kedua. Hal ini akan membantu model kecerdasan buatan untuk belajar dengan lebih baik dari kedua kelas, meningkatkan kemampuannya dalam membuat prediksi yang akurat untuk kelas minoritas.

## *B. Evaluasi Model*

Tabel 8

Hasil akurasi setiap Oversampling pada Dataset Sentimen

| *Oversampling Algorithm* | Akurasi |
|---|---|
| *SMOTE* | 0.84 |
| *Random* | 0.82 |
| *ADASYN* | 0.83 |
| *SMOTE-NC* | 0.83 |
| *Borderline SMOTE* | 0.84 |

Kesimpulan dari tabel 8 adalah berdasarkan hasil *confusion matrix* pada *dataset* sentimen, dapat disimpulkan bahwa semua metode *oversampling* yang digunakan, yaitu SMOTE, *Random Oversampling*, ADASYN, SMOTE-NC, dan *Borderline* SMOTE, memberikan kinerja yang cukup baik dalam meningkatkan akurasi model. Secara keseluruhan, akurasi yang dicapai oleh masing-masing algoritma oversampling berada di kisaran 0.82 hingga 0.84. Meskipun terdapat perbedaan kecil dalam akurasi, perbedaan tersebut mungkin tidak signifikan secara praktis.

Dalam konteks penanganan ketidakseimbangan kelas, penggunaan *oversampling* mampu menghasilkan model yang lebih seimbang dan mampu mengatasi tantangan kelas minoritas. Oleh karena itu, pemilihan metode *oversampling* dapat disesuaikan dengan karakteristik data dan kebutuhan spesifik dari masalah klasifikasi yang dihadapi. Dengan akurasi yang relatif tinggi pada semua metode, penting untuk mempertimbangkan faktor-faktor lain seperti interpretabilitas model, dan kecepatan konvergensi dalam menentukan metode *oversampling* yang paling sesuai untuk kasus tertentu.

Tabel 9

Hasil Akurasi setiap Oversampling pada Dataset Aspek

| *Oversampling Algorithm* | Akurasi |
|---|---|

| | |
|---|---|
| *SMOTE* | 0.88 |
| *Random* | 0.94 |
| *ADASYN* | 0.87 |
| *SMOTE-NC* | 0.89 |
| *Borderline SMOTE* | 0.93 |

Berdasarkan hasil *Confusion Matrix* pada model aspek dengan penerapan berbagai algoritma *oversampling*, dapat disimpulkan bahwa semua metode *oversampling*, yaitu SMOTE, *Random Oversampling*, ADASYN, SMOTE-NC, dan *Borderline* SMOTE, memberikan kinerja yang sangat baik dalam meningkatkan akurasi model. Akurasi yang dicapai oleh masing-masing algoritma *oversampling* berada pada kisaran tinggi, yaitu antara 0.87 hingga 0.94. Keberhasilan ini menunjukkan bahwa *oversampling* mampu secara signifikan meningkatkan kemampuan model dalam mengklasifikasikan data pada aspek yang diidentifikasi.

Pada percobaan diatas, penulis menggunakan fungsi *Confusion Matrix* untuk melakukan evaluasi model dengan mencetak *confusion matrix* setiap kali proses modelling selesai. *Confusion matrix* adalah sebuah tabel yang digunakan untuk mengukur kinerja model klasifikasi. Fungsi *Confusion Matrix* menggunakan *library* seaborn untuk membuat heatmap dari *confusion matrix* yang mencakup empat nilai: *True Positive (TP), True Negative (TN), False Positive (FP),* dan *False Negative (FN).* Setiap nilai pada *confusion matrix* merepresentasikan jumlah data yang diklasifikasikan dengan benar atau salah oleh model. Selanjutnya, fungsi mengembalikan nilai TP, TN, FP, dan FN, yang dapat digunakan untuk analisis lebih lanjut.

Dalam implementasi konkretnya, setelah model sentimen dan aspek melakukan prediksi pada data uji (y_pred), fungsi *Confusion Matrix* akan mencetak dan menampilkan *confusion matrix* dalam bentuk *heatmap*. Pada akhirnya, nilai TP, TN, FP, dan FN diambil dan digunakan untuk evaluasi performa model.

## IV. KESIMPULAN

Berdasarkan dari tahap analisis yang dilakukan penulis, didapatkan kesimpulan bahwa. Implementasi CNN untuk klasifikasi sentimen dan aspek dalam teks telah dirancang dengan baik. Model menggunakan lapisan *Embedding*, Conv1D, MaxPooling1D, dan Dense untuk merangkum struktur arsitektur. Untuk mencegah *overfitting*, penulis menggunakan lapisan *Dropout* setelah lapisan Conv1D dan Dense. Proses pelatihan dilakukan pada data yang telah di *resampling*, dengan penggunaan fungsi *loss binary crossentropy*, *optimizer* Adam, dan metrik akurasi. Pengoptimalan model dilakukan dengan menyesuaikan *learning rate* menggunakan *callback* ReduceLROnPlateau. Keseluruhan, implementasi ini mencerminkan upaya yang sistematis dan berhati-hati dalam mengatasi masalah klasifikasi sentimen dan aspek pada data teks, dengan fokus pada generalisasi model melalui teknik regularisasi dan hasil akurasi yang diperoleh dari dataset yang sedikit dengan menggunakan arsitektur *oversampling* memberikan tingkat akurasi dan loss yang cukup baik.

## DAFTAR PUSTAKA


[1] B. Liu and L. Zhang, ‘A Survey of Opinion Mining and Sentiment Analysis’, in Mining Text Data, C. C. Aggarwal and C. Zhai, Eds. Boston, MA: Springer US, 2012, pp. 415–463.

[2] Muhammad Arief Rahman, Herman Budianto, and E. I. Setiawan, “Aspect Based Sentimen Analysis Opini Publik Pada Instagram dengan Convolutional Neural Network”, INSYST, vol. 1, no. 2, pp. 50–57, Dec. 2019.

[3] JS. R. S. W. Wijaya Arya, ‘Klasifikasi Citra Menggunakan Convolutional Neural Network (Cnn) pada Caltech 101’, JURNAL TEKNIK ITS, vol. 5. 2016.

[4] L. A. Andika and P. A. N. Azizah, ‘Analisis Sentimen Masyarakat terhadap Hasil Quick Count Pemilihan Presiden Indonesia 2019 pada Media Sosial Twitter Menggunakan Metode Naive Bayes Classifier’. 2019.

[5] D. I. Af et al., ‘Pengaruh Parameter Word2Vec terhadap Performa Deep Learning pada Klasifikasi Sentimen’, vol. 6. 2021.

[6] P. R. Amalia and E. Winarko, ‘Aspect-Based Sentiment Analysis on Indonesian Restaurant Review Using a Combination of Convolutional Neural Network and Contextualized Word Embedding’, IJCCS (Indonesian Journal of Computing and Cybernetics Systems), vol. 15, p. 285, 7 2021.

[7] C. A. Bahri and L. H. Suadaa, ‘Aspect-Based Sentiment Analysis in Bromo Tengger Semeru National Park Indonesia Based on Google Maps User Reviews’, IJCCS (Indonesian Journal of Computing and Cybernetics Systems), vol. 17, p. 79, 2 2023.

[8] D. T. Hermanto, A. Setyanto, and E. T. Luthfi, ‘Algoritma LSTM-CNN untuk Sentimen Klasifikasi dengan Word2vec pada Media Online LSTM-CNN Algorithm for Sentiment Clasification with Word2vec On Online Media’, Citec Journal, vol. 8. pp. 64–77, 2021.

[9] A. Pambudi and S. Suprapto, ‘Effect of Sentence Length in Sentiment Analysis Using Support Vector Machine and Convolutional Neural Network Method’, IJCCS (Indonesian Journal of Computing and Cybernetics Systems), vol. 15, p. 21, 1 2021.

[10] E. Utami, ‘OPTIMIZING SENTIMENT ANALYSIS OF PRODUCT REVIEWS ON MARKETPLACE USING A COMBINATION OF PREPROCESSING TECHNIQUES, WORD2VEC, AND CONVOLUTIONAL NEURAL NETWORK OPTIMISASI ANALISIS SENTIMEN ULASAN PRODUK PADA MARKETPLACE DENGAN KOMBINASI TEKNIK PREPROCESSING, WORD2VEC, DAN CONVOLUTIONAL NEURAL NETWORK’, Jurnal Teknik Informatika (JUTIF), vol. 4, pp. 101–107, 2023.

[11] T. Mikolov, K. Chen, G. Corrado, and J. Dean, ‘Efficient Estimation of Word Representations in Vector Space’, 1 2013.

[12] F. P. Rachman, H. Santoso, and A. History, ‘Jurnal Teknologi dan Manajemen Informatika Perbandingan Model Deep Learning untuk Klasifikasi Sentiment Analysis dengan Teknik Natural Languange Processing’, Jurnal Teknologi dan Manajemen Informatika, vol. 7. pp. 103–112, 2021.

[13] I. Bakti and M. Firdaus, “Arsitektur CNN InceptionResNet-V2 Untuk Pengelompokan Pneumonia Chest X-Ray,” jukomtek, pp. 35–42, Jan. 2023, doi: 10.58290/jukomtek.v1i2.66.

[14] R. Nursyahfitri, C. Rozikin, and R. I. Adam, “Penerapan Metode SMOTE dalam Klasifikasi Daerah Rawan Banjir di Karawang Menggunakan Algoritma Naive Bayes,” justin, vol. 10, no. 4, p. 339, Dec. 2022, doi: 10.26418/justin.v10i4.46935.

[15] F. A. Ramadhan, S. H. Sitorus, and T. Rismawan, “Penerapan Metode Multinomial Naïve Bayes untuk Klasifikasi Judul Berita Clickbait dengan Term Frequency - Inverse Document Frequency,” justin, vol. 11, no. 1, p. 70, Jan. 2023, doi: 10.26418/justin.v11i1.57452.

[16] N. Cahyana, S. Khomsah, and A. S. Aribowo, “Improving Imbalanced Dataset Classification Using Oversampling and Gradient Boosting,” in 2019 5th International Conference on Science in Information Technology (ICSITech), Yogyakarta, Indonesia: IEEE, Oct. 2019, pp. 217–222. doi: 10.1109/ICSITech46713.2019.8987499.

[17] A. Viloria, O. B. P. Lezama, and N. Mercado-Caruzo, ‘Unbalanced data processing using oversampling: Machine Learning’, Procedia Computer Science, vol. 175, pp. 108–113, 2020.

[18] N. Cahyana, S. Khomsah, and A. S. Aribowo, “Improving Imbalanced Dataset Classification Using Oversampling and Gradient Boosting,” in 2019 5th International Conference on Science in Information Technology (ICSITech), Yogyakarta, Indonesia: IEEE, Oct. 2019, pp. 217–222. doi: 10.1109/ICSITech46713.2019.8987499.
[19] H. He, Y. Bai, E. A. Garcia, and S. Li, ‘ADASYN: Adaptive synthetic sampling approach for imbalanced learning’, 2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence). IEEE, pp. 1322–1328, 6 2008.
[20] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, “SMOTE: Synthetic Minority Over-sampling Technique,” jair, vol. 16, pp. 321–357, Jun. 2002, doi: 10.1613/jair.953.
[21] N. Cahyana, S. Khomsah, and A. S. Aribowo, “Improving Imbalanced Dataset Classification Using Oversampling and Gradient Boosting,” in 2019 5th International Conference on Science in Information Technology (ICSITech), Yogyakarta, Indonesia: IEEE, Oct. 2019, pp. 217–222. doi: 10.1109/ICSITech46713.2019.8987499.
[22] M. Mukherjee and M. Khushi, “SMOTE-ENC: A Novel SMOTE-Based Method to Generate Synthetic Data for Nominal and Continuous Features,” ASI, vol. 4, no. 1, p. 18, Mar. 2021, doi: 10.3390/asi4010018.