



Artikel Penelitian

## Penerapan 1D-CNN untuk Analisis Sentimen Ulasan Produk Kosmetik Berdasar Female Daily Review

Erwin Yudi Hidayat <sup>a,\*</sup>, Devioletta Handayani <sup>b</sup>

<sup>a,b</sup> Fakultas Ilmu Komputer, Universitas Dian Nuswantoro, Semarang, 50131, Indonesia

### INFORMASI ARTIKEL

#### Sejarah Artikel:

Diterima Redaksi: 10 Maret 2022

Revisi Akhir: 11 Januari 2023

Diterbitkan Online: 15 Januari 2023

### KATA KUNCI

1D-CNN,  
Analisis sentiment,  
Text mining

### KORESPONDENSI

E-mail: [erwin@dsn.dinus.ac.id](mailto:erwin@dsn.dinus.ac.id)\*

### A B S T R A C T

Pada tahun 2020 tercatat sekitar 797 industri kosmetik berskala besar maupun kecil yang terdapat di Indonesia. Berdasarkan tahun sebelumnya, angka ini naik 4.87%. Kondisi ini menyebabkan munculnya persaingan perusahaan kosmetik, salah satunya adalah Emina. Berbagai media digunakan sebagai sarana untuk menyampaikan sentimen atau opini masyarakat. Pihak perusahaan dapat memanfaatkan sentimen untuk mengetahui umpan balik masyarakat terhadap *brand* mereka. *Website Female Daily Review* menjadi salah satu *platform* yang digunakan untuk menampung segala bentuk opini mengenai produk kecantikan. Proses pengambilan data dari *website* pada penelitian ini menggunakan *web scraping*. Dari 11119 data ulasan yang didapatkan diperlukan analisis opini, emosi, dan sentimennya dengan memanfaatkan *text mining* untuk identifikasi serta mengekstrak suatu topik. Analisis sentimen dapat membantu mengetahui tingkat kepuasan pengguna terhadap suatu *brand* kosmetik. Algoritma yang digunakan adalah *1D-Convolutional Neural Network* (1D-CNN). Sebelum dilakukan klasifikasi data, perlu diterapkan *text preprocessing* agar *dataset* mentah menjadi lebih terstruktur. Hasil dari klasifikasi sentimen dibagi ke dalam 3 kategori yaitu positif, negatif, dan netral. Berdasarkan eksperimen dalam membangun model analisis sentimen menggunakan 1D-CNN sebanyak 30 percobaan, didapatkan model terbaik dalam menganalisis sentimen dengan akurasi sebesar 80.22%.

## 1. PENDAHULUAN

Kehadiran kosmetik memberi kemudahan khususnya bagi kaum wanita dalam mempercantik diri. Industri kosmetik tanah air saat ini terus mengalami peningkatan jumlah peminat. Hal ini dipicu oleh gaya hidup generasi milenial yang mempunyai tuntutan untuk selalu tampil menarik. *Chief Executive Officer* (CEO) Social Bella, John Marco Rasjid, merasa optimis jika industri produk kecantikan mempunyai potensi untuk terus tumbuh. Menurut John, dikutip dari Euromonitor, pada tahun 2019 pasar kecantikan dan perawatan diri yang ada di Indonesia dapat mencapai US\$ 6.03 miliar. Pada tahun 2022, angka tersebut diperkirakan dapat tumbuh menjadi US\$8.46 miliar. Pemerintah telah mencatat, terdapat sekitar 797 industri kosmetik baik berskala besar maupun berskala kecil dan menengah yang terdapat di Indonesia. Berdasarkan tahun sebelumnya, angka ini

naik 4.87%. Dari 797 kosmetik nasional, 294 di antaranya sudah terdaftar di Badan Koordinasi Penanaman Modal (BKPM) [1]. Hal inilah yang menyebabkan munculnya persaingan perusahaan kosmetik baik skala besar maupun kecil.

Salah satu produk kosmetik di Indonesia yang ikut bersaing adalah Emina. Emina merupakan nama *brand* produk kosmetik dari PT. Paragon Technology and Innovation. Produk kecantikan ini tergolong baru dirilis pada tahun 2015. Saat ini berbagai media digunakan sebagai sarana untuk menyampaikan sentimen atau opini masyarakat. Pihak perusahaan dapat memanfaatkan sentimen untuk mengetahui umpan balik dari masyarakat terhadap *brand* mereka. *Brand* dianggap menjadi salah satu aset tidak terwujud yang paling berharga serta manajemen *brand* adalah prioritas penting untuk pihak manajemen perusahaan maupun organisasi [2].

*Female Daily Review* merupakan salah satu portal yang digunakan untuk menampung segala bentuk opini mengenai produk kecantikan. Berdasar analisis *website* yang dilakukan oleh platform Similar Web pada tahun 2018, *website Female Daily Review* mendapatkan peringkat 20.973 dunia dan 523 di Indonesia dengan total 3.700.000 pengunjung pada bulan Februari 2018 [3]. Pada *website* ini pengguna dapat mencurahkan pengalamannya selama menggunakan produk kosmetik secara bebas. Meskipun sudah disediakan ulasan berbentuk *score* bintang pada *website* tersebut, terkadang pengguna memberi ulasan bintang yang tidak sesuai dengan ulasan komentar yang ditulis. Dari banyaknya ulasan komentar pun tidak langsung dapat menghasilkan sebuah pengetahuan.

Untuk mendapatkan pengetahuan baru yang tersembunyi dari beberapa kumpulan teks maka perlu dilakukan teknik pendekatan algoritmik menggunakan komputer atau biasa disebut dengan *text mining*. Teknik ini menjadi bagian dari keilmuan temu kembali informasi yang digunakan untuk data yang memiliki tipe teks cenderung tidak terstruktur [4]. Perlu dilakukan analisis opini, emosi, dan sentimennya dengan memanfaatkan analisis sentimen yang menerapkan *text mining* dalam mengidentifikasi serta mengekstrak tentang suatu topik [5]. Dengan kata lain, analisis sentimen dapat membantu untuk mengetahui tingkat kepuasan pengguna terhadap suatu *brand* kosmetik.

Berbagai metode telah dikembangkan dan diterapkan untuk menganalisis sentimen diantaranya *Support Vector Machine*, *Naïve Bayes*, *K-Nearest Neighbour*, serta *Deep Learning*. cara kerja dari analisis sentimen yaitu dengan mengklasifikasikan opini positif, negatif, serta netral. Sehingga nantinya dapat dijadikan acuan untuk meningkatkan suatu layanan maupun dalam meningkatkan kualitas produk [6]. Penelitian terkait analisis sentimen pernah dilakukan oleh [7], [8], [9], dan [10].

Alasan penulis menggunakan algoritma *1D-Convolutional Neural Network* (1D-CNN), karena menurut penelitian [11] hasil akurasi yang dihasilkan oleh algoritma 1D-CNN lebih tinggi daripada akurasi algoritma lainnya seperti *Naïve Bayes*, RNN, DCNN, RCNN yaitu sebesar 90%. Jika ditinjau dari segi kecepatan algoritma dalam memprediksi sentimen, metode *Naïve Bayes* lebih unggul daripada 1D-CNN. Namun, pada penelitian ini yang dibutuhkan adalah ketepatan algoritma dalam memprediksi sentimennya. Sehingga algoritma 1D-CNN lebih tepat diterapkan untuk analisis sentimen ulasan produk kosmetik.

*Convolutional Neural Network* (CNN) merupakan jaringan saraf tiruan *deep feed-forward* di mana tiap simpulnya tidak membentuk siklus. Umumnya, CNN digunakan untuk *computer vision* [12], namun algoritma ini sekarang banyak digunakan untuk mengolah *Natural Language Processing* (NLP) [13]. Pola yang dihasilkan dapat berupa ekspresi dan CNN dapat mengidentifikasinya dalam kalimat tertentu terlepas dari posisi kata-kata tersebut. *Layer* yang digunakan pada CNN terdapat 3 jenis yaitu *convolutional layer*, *pooling layer*, dan *fully-connected layer* [14].

Pada penelitian ini dilakukan klasifikasi sentimen dalam 3 kategori yaitu positif, negatif, dan netral menggunakan algoritma 1D-CNN. Adapun topik dalam penelitian ini adalah menentukan

analisis sentimen terhadap produk kosmetik dari *brand* Emina berdasar situs *Female Daily Review*.

## 2. METODE

### 2.1. Sumber Data

Penelitian ini menggunakan sumber data berupa ulasan produk *brand* Emina pada *website Female Daily Review* (<https://reviews.femaledaily.com/>). Ribuan ulasan produk kosmetik dengan *brand* Emina tersedia pada *website Female Daily Review*. Ulasan produk yang diambil pada penelitian ini adalah ulasan dari beberapa produk kosmetik dengan *brand* Emina yang dipilih secara acak. Data tersebut kemudian akan dilakukan *preprocessing*, *labeling*, dan analisis sentimen.

### 2.2. Metode Pengumpulan Data

Dalam penelitian ini prosedur pengumpulan data yang digunakan ada dua. Pertama adalah studi pustaka. Studi pustaka dilakukan dengan cara mengumpulkan teori-teori yang berkaitan dalam penulisan skripsi yang kemudian akan dijadikan bahan untuk melengkapi penelitian ini. Sumber teori berasal dari hasil penelitian (jurnal dan prosiding), buku referensi, serta artikel-artikel yang berkaitan. Selain itu peneliti juga mengunjungi situs yang berkaitan dengan pengaplikasian *text mining* dan algoritma CNN.

Kedua, observasi. Dalam melakukan observasi, peneliti melakukan pengamatan serta pengambilan data dari *Female Daily Review* mengenai produk kosmetik Emina. Peneliti mengambil data dari laman produk Emina secara acak pada situs *Female Daily Review*. Proses pengambilan datanya dibantu *library* Python yaitu Selenium dan BeautifulSoup kemudian data-data tersebut disimpan pada *file comma separated values* (.csv)

### 2.3. Metode yang Diusulkan

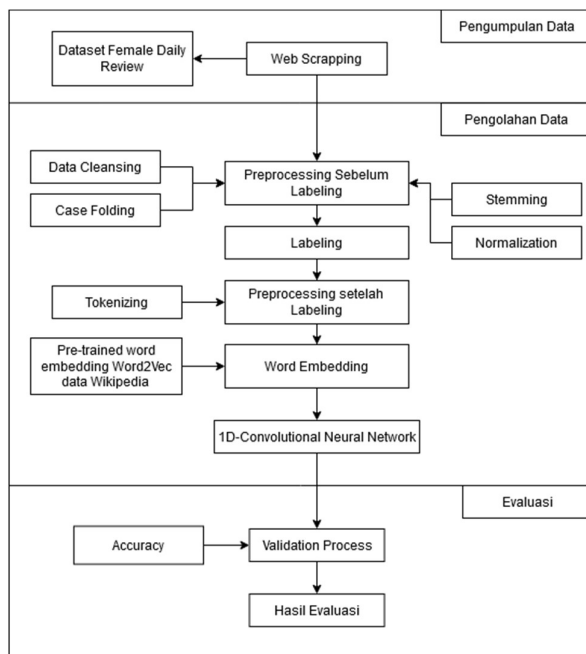
Pada penelitian ini metode yang diusulkan yaitu mengimplementasikan algoritma 1D-CNN dalam menganalisis sentimen produk kosmetik pada *brand* Emina. Sebelum data ulasan dilakukan analisis sentimen, data tersebut harus melewati *preprocessing* dan *labeling* terlebih dahulu.

Pada tahapan *preprocessing* sebelum *labeling* terdapat beberapa tahapan seperti *data cleansing*, *case folding*, *normalization*, serta *stemming*. Kemudian pada tahap *labeling*, penelitian ini akan menggunakan aplikasi hasil penerapan penelitian dari [15] yang dapat mengukur kekuatan sentimen dengan mengkombinasikan metode *SentiStrength*, *hybrid TF-IDF*, dan *cosine similarity*. Tahap selanjutnya adalah *tokenizing* yang menghasilkan daftar *vocabulary* penyusun kalimat pada dataset. Kemudian dataset akan diubah ke dalam bentuk vektor tetapi sebelumnya perlu menambahkan *Out of Vocabulary* (OOV) token pada indeks ke-0 untuk menyamakan panjang vektornya. Setelah itu perlu dilakukan *word embedding* untuk perhitungan jumlah serta frekuensi kata yang muncul pada sebuah dokumen [16]. Penelitian ini memanfaatkan *pre-trained word embedding Word2Vec* berdasar data Wikipedia yang tersedia pada Github. Setelah itu data tersebut digunakan untuk proses pelatihan dan pengujian model.

## 2.4. Tahapan Penelitian

Penelitian diawali dengan *web scraping* data ulasan produk yang diambil ulasannya dipilih secara acak. Dilanjutkan dengan tahap *preprocessing* yang dibagi menjadi 2 yaitu *preprocessing* sebelum dilakukan *labeling* dan *preprocessing* sesudah dilakukan *labeling*.

Tahap *preprocessing* sebelum dilakukan *labeling* di antaranya adalah *case folding* sebagai tahap proses dalam menyamakan *case* dalam sebuah *dataset* agar dapat mempermudah dalam melakukan pencarian, serta *stemming* sebagai tahap untuk mencari bentuk dasar dari tiap kata. Tahap selanjutnya yaitu melakukan *labeling* kekuatan sentimennya pada data ulasan produk menggunakan aplikasi dari penelitian [15]. Kemudian tahap *preprocessing* setelah dilakukan *labeling* data adalah *tokenizing*. *Tokenizing* merupakan tahap mengkonversi teks menjadi urutan integer indeks kata dan menghasilkan daftar *vocabulary* kata penyusun pada dataset. *Out of Vocabulary* (OOV) token pada indeks ke-0 kemudian ditambahkan untuk menyamakan panjang vektor. Penelitian ini memanfaatkan *pre-trained word embedding* Word2Vec berdasar data Wikipedia dalam melakukan *word embedding*. Setelah itu data berbentuk vektor dibagi lagi menjadi data uji dan data latih. Tahap terakhir baru dilakukan pelatihan serta pengujian data menggunakan algoritma 1D-CNN. Setelah selesai proses pelatihan serta pengujian data, maka perlu diukur akurasinya.



Gambar 1. Tahapan Penelitian

### 2.4.1. Web Scraping

*Web scraping* merupakan teknik untuk mengekstrak data dari *World Wide Web* (WWW) dan kemudian disimpan ke dalam sistem *file* atau basis data yang nantinya data tersebut dapat diambil kembali untuk dilakukan analisis. *Web Scraping* dapat dilakukan secara manual oleh pengguna atau secara otomatis oleh bot atau *web crawler* [17]. Data yang dibutuhkan pada penelitian ini adalah ulasan produk dengan *brand* Emina pada *Female Daily Review*. Pengambilan data dibantu dengan *library* Python

Selenium dan BeautifulSoup. Di mana *library* BeautifulSoup ini mencari jumlah data yang kalimat ulasannya terpotong ditandai dengan *link Read more*. Kemudian *library* Selenium dapat melakukan klik pada *tag* HTML yang merujuk pada *Read more* untuk dikirim ke *driver*. Jika sudah terbuka semua reviewnya maka *library* Selenium akan membantu untuk mengumpulkan data ulasan produk yang merujuk ke *class* HTML bersangkutan, lalu dimasukkan ke dalam sebuah *array*. Untuk perpindahan laman dapat memanfaatkan *id page* dan perulangan. Data yang sudah terkumpul pada satu *array* kemudian dikonversikan ke bentuk *csv*. Tabel 1 adalah 3 sample data dari *dataset* mentah ulasan produk *brand* Emina.

Tabel 1. Dataset Mentah Ulasan Produk

No.	Dataset Mentah Ulasan Produk
1	aku ga pernah pake blush yang ga cream gini sih alias compact blush gtuu, jadi aku mau share pengalaman aku pake ini, yaa aplikasiin nya agak susah dan ngotorin tangan, warna nya lumayan pigmented di kulit tapi ga smudgeproof dan cuman up to 5 hour doang hehe, sometime kulit aku juga jadi lebih oily karena produk ini:(
2	Ini cukup bagus sih untuk cream blush dengan harga yang murah. Sekarang juga pilihan warnanya makin beragam kan! Akan cocok bagi yang suka pake natural makeup. Pertama kali beli ini itu yang shade-shade awal kalo gasalah pink sama apa gitu. Nah awal2 tuh merasa agak tricky pake produknya, entah kurang banyak atau kebanyakan. Tapi kayanya shade-shade baru lebih aplicable dibanding yg shade dulu ya. Trus juga harganya muraaaaah bgt sih jd good deal bagi pemakai make up pemula dan untuk sehari2.
3	kirain mirip ky product nya burnburn yg skali swipe lgsg keluar. ini pakai berkali kali baru keluar warna nya. itupun brasa nya anget gitu. ky gk nyaman lah. trus ngerasa nya jd ngegeser si foundi krn kan hrs berkali tap tap si product utk ngeratain. mgkin klo utk remaja yg kulit nya mulus tanpa pakai foundi nah ini cocok.. krn kliatan natural lah

### 2.4.2. Preprocessing Sebelum Labeling

*Preprocessing* memiliki tujuan untuk mengubah data teks yang tidak terstruktur menjadi data yang terstruktur dan sesuai kebutuhan. Pada proses *preprocessing* terdapat beberapa tahap di antaranya *data cleansing*, *case folding*, *normalization*, dan *stemming*. Adapun diagram alir dari *preprocessing* sebelum dilakukan *labeling* seperti pada Gambar 2.

*Data cleansing* merupakan proses untuk menghilangkan kata-kata yang tidak dibutuhkan. Pada penelitian ini *data cleansing* menghapus ulasan produk yang sama atau duplikat, simbol-simbol, tanda baca, *tag* (@), spasi ganda serta angka. Komponen yang dihapus berdasarkan *dataset* mentah di atas antara lain '5', '-', '!', 'ø', 'ÿ', ':(', dan '2', serta tanda titik (.) dan koma (,).

*Case folding* merupakan proses untuk menyamakan *case* dalam sebuah *dataset* agar dapat mempermudah dalam melakukan pencarian. Tidak semua *dataset* teks konsisten dalam menggunakan huruf kapital. Sehingga peran *case folding* ini penting untuk mengkonversikan keseluruhan teks ke dalam bentuk standar (huruf kecil atau *lowercase*).

*Normalization* atau normalisasi adalah proses pengolahan data dengan memanfaatkan kamus dalam mengubah kata yang tidak sesuai dengan arti serta makna sebenarnya. Kamus manual

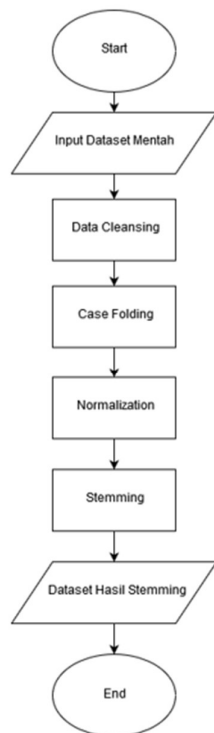
digunakan pada saat normalisasi untuk menunjukkan penerjemahan arti serta makna dari kata-kata *slank* atau gaul. Singkatan diubah menjadi kata yang memiliki makna sesuai dengan Kamus Besar Bahasa Indonesia (KBBI) [18]. Sampel dari kamus yang digunakan sebagai acuan normalisasi disajikan pada Tabel 2.

Tabel 2. Contoh Kata Baku dan Kata Tidak Baku

Kata Tidak Baku	Kata Baku
krn	karena
ga, ngga, gak, gk	tidak
pake	pakai
gt, gitu	begitu
udah	sudah
bgt	sangat
hrs	harus

*Stemming* merupakan proses untuk mencari bentuk dasar dari tiap kata. Pada penelitian ini proses *stemming* dibantu menggunakan *library* Python Sastrawi yang merupakan pengembangan dari *library* PHP Sastrawi. *Library* yang diterapkan adalah menggunakan algoritma Nazief dan Adriani. *Dataset* yang sudah dilakukan *stemming* disimpan pada *file* dengan format *.csv* untuk digunakan pada tahap selanjutnya.

Gambar 2. Diagram Alir *Preprocessing* Sebelum *Labeling*



### 2.4.3. Labeling

Tahap *labeling* merupakan proses untuk menentukan kekuatan sentimennya pada data ulasan produk menggunakan aplikasi dari penelitian [15] yang mengkombinasikan metode *SentiStrength*, *Hybrid TF-IDF* dan *Cosine Similarity* untuk mengekstraksi ringkasan sentimen positif, negatif, dan netral. Metode *SentiStrength* digunakan untuk mendapatkan skor kekuatan

sentimen dan mengklasifikasi data dari *dataset* ke dalam kelas positif, negatif dan netral. Data dari *dataset* bersentimen positif dan negatif diringkas dengan cara pemeringkatan data menggunakan *Hybrid TF-IDF* yang dikombinasi dengan skor kekuatan sentimen, kemudian menghilangkan data yang mirip menggunakan *Cosine Similarity*. Hasil pengujian memperlihatkan bahwa kombinasi *SentiStrength*, *Hybrid TF-IDF*, dan *Cosine Similarity* mampu menghasilkan ringkasan sentimen dengan akurasi yang lebih baik dibandingkan menggunakan *Hybrid TF-IDF* saja.

Tabel 3. *Dataset* Hasil *Stemming*

No.	<i>Dataset</i> Ulasan Produk Hasil <i>Stemming</i>
1	saya tidak pernah pakai blush yang tidak cream gin sih alias compact blush gtuu jadi saya mau share alam saya pakai ini yaa aplikasiin nya agak susah dan ngotorin tangan warna nya lumayan pigmented di kulit tetapi tidak smudgeproof dan cuman up to hour doang hehe sometime kulit saya juga jadi lebih oily karena produk ini
2	ini cukup bagus sih untuk cream blush dengan harga yang murah sekarang juga pilih warna makin agam kan akan cocok bagi yang suka pakai natural makeup pertama kali beli ini itu yang shadeshade awal kalau gasalah pink sama apa begitu nah awal tuh rasa agak tricky pakai produk entah kurang banyak atau banyak tetapi kaya shadeshade baru lebih aplicable banding yang shade dulu ya terus juga harga muraaaaah sangat sih jadi good deal bagi pakai make up mula dan untuk hari
3	kirain mirip ky product nya burnburn yang skali swipe lgsg kluar ini pakai kali kali baru kluar warna nya itu brasa nya anget gituh ky tidak nyaman lah terus ngerasa nya jadi ngegeser si foundi karena kan harus kali tap tap si product untuk ngeratain mgkin kalau untuk remaja yang kulit nya mulus tanpa pakai foundi nah ini cocok karena kliatan natural lah

Tabel 4. *Dataset* Setelah *Labeling*

No.	<i>Dataset</i> Ulasan Produk	Label Sentimen
1	saya tidak pernah pakai blush yang tidak cream gin sih alias compact blush gtuu jadi saya mau share alam saya pakai ini yaa aplikasiin nya agak susah dan ngotorin tangan warna nya lumayan pigmented di kulit tetapi tidak smudgeproof dan cuman up to hour doang hehe sometime kulit saya juga jadi lebih oily karena produk ini	Netral
2	ini cukup bagus sih untuk cream blush dengan harga yang murah sekarang juga pilih warna makin agam kan akan cocok bagi yang suka pakai natural makeup pertama kali beli ini itu yang shadeshade awal kalau gasalah pink sama apa begitu nah awal tuh rasa agak tricky pakai produk entah kurang banyak atau banyak tetapi kaya shadeshade baru lebih aplicable banding yang shade dulu ya terus juga harga muraaaaah sangat sih jadi good deal bagi pakai make up mula dan untuk hari	Positif
3	kirain mirip ky product nya burnburn yang skali swipe lgsg kluar ini pakai kali kali baru kluar warna nya itu brasa nya anget gituh ky tidak nyaman lah terus ngerasa nya jadi ngegeser si foundi karena kan harus kali tap tap si product untuk ngeratain mgkin kalau untuk remaja yang kulit nya mulus tanpa pakai foundi nah ini cocok karena kliatan natural lah	Negatif

Penggunaan penelitian *labeling* oleh [15] masih relevan digunakan sebagai alat *labeling* otomatis pada penelitian ini, karena penelitian tersebut berbasis kamus kata di mana nilai sentimen dari suatu kata tidak akan berubah-ubah. Contohnya kata “benci” dan “jelek”, sentimennya tetap negatif. Begitu juga dengan kata “bagus” dan “cantik” yang nilai sentimennya tetap positif.

#### 2.4.4. Preprocessing Setelah Labeling

Pada tahap ini, *tokenizing* dilakukan. *Tokenizing* yaitu tahap pemotongan kalimat berdasarkan tiap kata yang menyusunnya sehingga didapatkan urutan indeks *integer* per kata dan menghasilkan daftar *vocabulary* kata penyusun pada *dataset*. Setelah itu perlu dilakukan konversi *dataset* ke bentuk vektor. Jumlah kata yang digunakan pada tiap ulasan berbeda-beda. Pada proses mengubah *dataset* ke dalam bentuk vektor, panjang vektor datanya harus tetap. Sehingga perlu ditambah OOV pada index ke-0 untuk mengisi kekosongan serta menyamakan panjang vektor datanya. Berikut ini adalah contoh *tokenizing* berdasar *dataset* ulasan produk yang telah diberi label.

Tabel 5. Daftar *Vocabulary*

Indeks	Kata
0	(OOV)
1	saya
2	tidak
3	pernah
4	pakai
5	blush
6	yang
7	cream

#### 2.4.5. Word Embedding

*Word embedding* merupakan langkah untuk mengubah *index vocabulary* menjadi bentuk *array* yang berisikan nilai-nilai fitur dengan memanfaatkan *pre-trained word embedding* Word2Vec data Wikipedia dengan jumlah fitur 300 untuk tiap indeksnya.

Tabel 6. Nilai *Filter Word Embedding*

	Wanita	Lipstik	Merah
Fitur Gender	1	0.01	0.02
Fitur Kosmetik	0.02	0.98	0.1
Fitur Wama	0.01	0.02	0.97

Misalkan *dataset* yang terdiri dari kata penyusunnya: “wanita”, “lipstik”, dan “merah”. Kemudian jumlah fiturnya ditentukan, misalkan sebanyak 3. Word2Vec memiliki peran untuk mempelajari pola yang terbentuk dari frekuensi kata yang muncul pada *dataset*. Misalkan ternyata Word2Vec dapat menyimpulkan bahwa 3 fitur tersebut adalah gender, kosmetik dan warna dengan nilai fitur yang bervariasi, seperti pada Tabel 6.

Berdasarkan Tabel 6, maka “wanita” memiliki fitur [1, 0.02, 0.01], “lipstik” = [0.01, 0.98, 0.02], dan “merah” = [0.02, 0.1, 0.97]. Nilai fitur yang telah didapatkan kemudian digunakan sebagai *node* pada saat *training* dan *testing* model.

#### 2.4.6. Pengujian dan Pengukuran Model

*Dataset* terlebih dahulu dibagi menjadi *dataset* uji dan *dataset* latih, dengan prosentase *dataset* uji dan *dataset* latih 75% data latih dan 25% data uji. Uji coba dilakukan guna mendapatkan model terbaik untuk menganalisis sentimen. Pengujian model

<https://doi.org/10.25077/TEKNOSI.v8i3.2022.153-163>

dilakukan dengan cara melakukan uji coba jumlah *convolutional layer*, *pooling layer*, *hidden layer*, jumlah simpul pada *hidden layer*, *learning rate* (0.1 - 0.001) serta *dropout* (0.1 - 0.5). Model yang dipilih adalah model yang memiliki akurasi yang tinggi.

Tabel 7. Contoh Data Kategori Aktual dan Kategori Prediksi

No	Kategori Aktual	Kategori Prediksi
1	Netral	Netral
2	Positif	Netral
3	Positif	Positif
4	Netral	Positif
5	Negatif	Negatif
6	Negatif	Negatif

Evaluasi kinerja klasifikasi dihitung untuk membandingkan hasil klasifikasi sentimen. Tabel 7 menyajikan contoh dari data dengan kategori aktual dan kategori prediksi.

$$\text{Akurasi} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \times 100 \% \quad (1)$$

Akurasi diperoleh dengan Persamaan 1. Maka diperoleh:

$$\text{Akurasi} = \frac{4}{6} \times 100 \% = 66,67 \%$$

## 2.5. Algoritma 1D-CNN

*Convolutional Neural Network* merupakan jaringan saraf tiruan *deep feed-forward* dengan tiap simpulnya tidak membentuk siklus. Pola yang dihasilkan dapat berupa ekspresi dan CNN dapat mengidentifikasinya dalam kalimat tertentu terlepas dari posisi kata-kata tersebut. *Layer* yang digunakan pada CNN terdapat 3 jenis yaitu *convolutional layer*, *pooling layer*, dan *fully-connected layer*.

*Convolutional layer* berisi serangkaian filter yang ukurannya tetap yang digunakan untuk mengkonvolusikan data. *Output* dari *convolutional layer* adalah *feature maps*. Berikut ini adalah persamaan operasi *convolutional*:

$$\text{FM}_{a,b} = \text{bias} + \sum_c^C \sum_d^D Z_{c,d} + X_{a+c-1,b+d-1} \quad (2)$$

*Pooling layer* memastikan bahwa jaringan hanya fokus pada pola yang paling penting serta data dirangkum dengan menggeser jendela melintasi *feature maps*, kemudian menerapkan beberapa operasi *linear* atau *non linear* pada data yang ada pada jendela. *Pooling layer* memiliki fungsi untuk mengurangi dimensi dari *feature maps* yang akan digunakan pada *layer* selanjutnya.

$$f_h(0, \text{FM}_{a,b}) = \max(0, \text{FM}_{a,b}) = \begin{cases} \text{FM}_{a,b}, & \text{jika } \text{FM}_{a,b} \geq 0, \\ 0 & \text{jika } \text{FM}_{a,b} < 0, \end{cases} \quad (3)$$

*Layer* terakhir yang digunakan adalah *fully-connected layer*. *layer* ini digunakan untuk memahami pola yang dihasilkan dari *layer* sebelumnya. *Neuron* pada *layer* ini memiliki koneksi penuh ke semua aktivasi pada *layer* sebelumnya. Metode CNN juga menggunakan fungsi aktivasi yang dilakukan ketika berada di antara *convolutional layer* dan *pooling layer*. Aktivasi di antara kedua *layer* tersebut menggunakan fungsi aktivasi ReLU. Sedangkan untuk fungsi aktivasi *output* menggunakan softmax. Persamaan fungsi aktivasi ReLU terdapat pada Persamaan 3.

Fungsi aktivasi softmax mempunyai tujuan untuk mendapatkan hasil klasifikasi serta menghasilkan nilai yang diinterpretasi



sebagai probabilitas yang belum dinormalisasi untuk tiap kelas [19]. Nilai kelas yang dihitung dengan menggunakan fungsi softmax ditunjukkan oleh Persamaan 4.

$$y_{ijk} = \frac{e^{x_{ijk}}}{\sum_{t=1}^D e^{x_{ijt}}} \quad (4)$$

Fungsi terakhir adalah *loss function* untuk menghitung *loss* (nilai *error*) dengan menggunakan *categorical cross-entropy*. Persamaan 5 adalah *loss function* yang dimaksud.

$$L_{\log}(Y, Y_{\text{pred}}) = -\frac{1}{N} \sum_{i=1}^N \sum_{c=1}^C I_{y_i \in C_c} \log P_{\text{model}}[y_i \in C_c] \quad (5)$$

### 3. HASIL

#### 3.1. Web Scraping

Penelitian ini diawali dengan *web scraping*, yang biasa digunakan yaitu parsing HTML. Sekarang ini sudah banyak *library* pendukung untuk mempermudah *scraping* data. *Library* BeautifulSoup yang digunakan membantu dalam proses mengekstraksi data. Secara *syntax* pun *library* BeautifulSoup lebih mudah untuk dipelajari. Namun, *library* tersebut tidak mampu menyelesaikan permasalahan ulasan produk pada *website Female Daily Review* yang masih teringkas (ditandai dengan kata kunci 'Read more'). Oleh karena itu, peneliti menggunakan *library* Python yang lain yaitu Selenium. Selenium memiliki peran sebagai otomasi peramban sekaligus membantu dalam ekstraksi data terutama membantu membuka ulasan produk yang masih teringkas.

Ulasan produk yang diambil pada penelitian ini adalah ulasan produk kosmetik dengan *brand* Emina. Pengumpulan data dilakukan sejak bulan Oktober 2020 dan terkumpul sekitar 11119 data dari 10 produk Emina yang dipilih secara acak. Adapun sumber *website* yang digunakan adalah *Female Daily Review*.

Langkah pertama dalam mengimplementasikan *web scraping* adalah *import* semua *library* yang dibutuhkan dan menginisialisasi *array* kosong untuk menampung data ulasan produk. Otomatisasi perpindahan halaman saat *scraping* data memanfaatkan perulangan dan *link* yang disertai *id page*.

Dari variabel *link* yang ada maka *driver* dapat mengakses *website* produk yang dituju. *Library* BeautifulSoup membantu dalam mencari *wrapper* yang berisikan ulasan produk dan menghitung berapa jumlah ulasan produk yang masih teringkas yang ditandai dengan kata kunci 'Read more'.

*Driver* mencari elemen HTML yang mempunyai *tag span* dengan *class read-more* sebanyak ulasan produk yang masih teringkas. Kemudian *driver* akan mengeksekusi script dengan cara melakukan klik pada *elemen* yang dituju sehingga ulasan dapat terlihat secara lengkap. Setelah itu *driver* mencari elemen HTML ulasan produk dengan *class text-content* untuk kemudian dimasukkan ke dalam sebuah *array* yang sudah dilakukan inisialisasi.

```
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC
from webdriver_manager.chrome import ChromeDriverManager

from urllib.request import urlopen

from bs4 import BeautifulSoup

import pandas as pd

driver = webdriver.Chrome(executable_path='C:\Program Files\Python3\Scripts\chromedriver.exe')

review_produk=[]
```

Gambar 3. Import Library dan Inisialisasi Array Penampung

```
sc_review_final = {'review': review_produk}
df = pd.DataFrame(sc_review_final, columns=['review'])
print(df)
df.to_csv('test.csv', sep=',', encoding='utf-8', index=False)
```

Gambar 4. Konversi Array Ulasan Produk menjadi File .csv

Ulasan produk yang telah dikumpulkan pada suatu *array* lalu dikonversikan ke dalam bentuk *.csv*, sehingga *file* atau *dataset* tersebut dapat diakses kembali untuk langkah penelitian selanjutnya.

#### 3.2. Preprocessing Sebelum Labeling

Langkah *preprocessing* sebelum *labeling* pada penelitian ini terdiri dari *data cleansing*, *case folding*, *normalization*, dan *stemming*.

Tabel 8. Contoh Hasil *Preprocessing* Sebelum Labeling

Tahap	Data Input	Data Hasil
<b>Preprocessing</b>		
<b>Data Cleansing</b>	Definisi kecil kecil cabe rawit! Walaupun ukurannya kecil tapi bagus bgt ya ampunnn, pigmented dan tahan lama! Varian warnanya bagus bagus dan diblendnya gampang banget. Produk lokal yang udah jadi andalan banyak orang nih, bener bener harus punyaa!	Definisi kecil kecil cabe rawit Walaupun ukurannya kecil tapi bagus bgt ya ampunnn pigmented dan tahan lama Varian warnanya bagus bagus dan diblendnya gampang banget Produk lokal yang udah jadi andalan banyak orang nih bener bener harus punyaa
<b>Case Folding</b>	Definisi kecil kecil cabe rawit Walaupun ukurannya kecil tapi bagus bgt ya ampunnn pigmented dan tahan lama Varian warnanya bagus bagus dan diblendnya gampang banget Produk lokal yang udah jadi andalan banyak orang nih bener bener harus punyaa	definisi kecil kecil cabe rawit walaupun ukurannya kecil tapi bagus bgt ya ampunnn pigmented dan tahan lama varian warnanya bagus bagus dan diblendnya gampang banget produk lokal yang udah jadi andalan banyak orang nih bener bener harus punyaa
<b>Normalization</b>	definisi kecil kecil cabe rawit walaupun ukurannya kecil tapi bagus bgt ya ampunnn pigmented dan tahan lama varian warnanya bagus bagus dan diblendnya gampang	definisi kecil kecil cabai rawit walaupun ukurannya kecil tetapi bagus sangat ya ampunnn pigmented dan tahan lama varian warnanya bagus bagus dan diblendnya

	banggett produk lokal yang udah jadi andalan banyak orang nih bener bener harus punyaa	gampang banggett produk lokal yang sudah jadi andalan banyak orang nih benar benar harus punyaa
<b>Stemming</b>	definisi kecil kecil cabai rawit walaupun ukurannya kecil tetapi bagus sangat ya ampunnn pigmented dan tahan lama varian warnanya bagus bagus dan diblendnya gampang banggett produk lokal yang sudah jadi andalan banyak orang nih benar benar harus punyaa	definisi kecil kecil cabai rawit walaupun ukur kecil tetapi bagus sangat ya ampunnn pigmented dan tahan lama varian warna bagus bagus dan diblendnya gampang banggett produk lokal yang sudah jadi andal banyak orang nih benar benar harus punyaa

Dataset mentah yang terdiri dari 11119 data ulasan produk setelah melewati tahap *preprocessing* menyisakan data sejumlah 11084 data yang sudah terstruktur dan akan digunakan pada tahap selanjutnya

### 3.3. Labeling

Tabel 9. Jumlah Data per Kelas Klasifikasi

Kode Sentimen	Kelas Klasifikasi	Jumlah Data
0	Netral	1666
1	Positif	8089
2	Negatif	1329

Pada tahap labeling peneliti menggunakan implementasi dari penelitian [15] yang mengkombinasikan metode *SentiStrength*, *hybrid TF-IDF* dan *cosine similarity* untuk mengekstraksi ringkasan sentimen positif, negatif, dan netral yang tersedia secara *open source* pada Github. Dari 11084 data ulasan produk yang tersisa didapatkan jumlah data per kelas klasifikasi seperti pada Tabel 9.

### 3.4. Preprocessing Setelah Labeling

*Tokenizing* merupakan tahap untuk memisah tiap kata penyusun kalimat data ulasan produk. Jika daftar *vocabulary* sudah terbentuk maka setiap kalimat pada dataset perlu diubah ke bentuk vektor. Setiap ulasan produk memiliki jumlah kata penyusun yang berbeda dan perlu ditambahkan OOV token. Penelitian ini menambahkan *vocabulary* OOV pada indeks ke-0 untuk menyamakan panjang vektor. Adapun jumlah daftar *vocabulary* yang didapatkan dari dataset ditambah dengan kata 'OOV' adalah 13446 kata.

### 3.5. Word Embedding

Setelah *dataset* diubah ke dalam bentuk vektor, *index vocabulary* dilakukan *embedding* berdasar nilai fitur yang memanfaatkan *pre-trained word embedding* Word2Vec menggunakan data dari Wikipedia, sehingga nilai dari setiap fitur sudah tersedia. Adapun jumlah fitur atau dimensi yang dipilih adalah sejumlah 300 fitur untuk tiap *index vocabulary*.

### 3.6. Pengujian dan Pengukuran Model

Pengujian dan pengukuran model pada penelitian ini dilakukan dengan melakukan beberapa eksperimen. Peneliti melakukan eksperimen sebanyak 30 kali untuk mendapatkan akurasi model

<https://doi.org/10.25077/TEKNOSI.v8i3.2022.153-163>

terbaik dengan cara mengubah susunan *layer*, *epoch*, serta mengubah nilai dari *learning rate*. Adapun rasio yang digunakan sebagai data *testing* dan data *training* adalah sebesar 25:75.

Tabel 10. Contoh Model Eksperimen

Eks ke-	LR	Susunan Layer	E-poch	Hasil
1	0.01	Conv1D(32,3,ReLU) MaxPooling1D Dense(16,ReLU) Dense (32,ReLU) Dense (8,ReLU) GlobalMaxPooling1D	20	Loss: 0.2332 Accuracy: 0.9155 Val_loss: 0.9151 Val_Acc: 0.6994
2	0.001	Conv1D(32,3,ReLU) MaxPooling1D Dense(16,ReLU) Dense (32,ReLU) Dense (8,ReLU) GlobalMaxPooling1D	20	Loss: 0.6693 Accuracy: 0.7438 Val_loss: 0.7692 Val_acc: 0.7200
3	0.008	Conv1D(32,3,ReLU) MaxPooling1D Dropout(0.3) Dense(64,ReLU) Dense (32,ReLU) Dense(16,ReLU) GlobalMaxPooling1D	60	Loss: 0.3030 Accuracy:0.8771 Val_loss:0.5957 Val_acc:0.7759
4	0.008	Conv1D(32,3,ReLU) MaxPooling1D Dropout(0.5) Dense(64,ReLU) Dense (32,ReLU) Dense(16,ReLU) GlobalMaxPooling1D	60	Loss: 0.4223 Accuracy: 0.8344 Val_loss: 0.6180 Val_acc: 0.7643
5	0.01	Conv1D(32,3,ReLU) MaxPooling1D Dropout(0.5) Dense(32,ReLU)-L2 Dense (64,ReLU)-L2 Dropout(0.2) Dense(16,ReLU)-L2 GlobalMaxPooling1D	50	Loss: 0.5642 Accuracy: 0.7869 Val_loss: 0.6050 Val_acc: 0.7690

## 4. PEMBAHASAN

Berdasarkan Tabel 10, penentuan besar *epoch*, *learning rate*, *dropout* pada *hidden layer* mempengaruhi nilai akurasi. Pada eksperimen ke-12 peneliti mendapatkan hasil akurasi mendapatkan *test accuracy* sebesar 80.22% sedangkan *training accuracy* sebesar 84.64%. Ketika jumlah *epoch* diperbesar pada eksperimen ke-13 ternyata *test accuracy* turun menjadi 79.36% dan *training accuracy* naik menjadi 86.44%. Model tersebut mempunyai indikasi *overfit* karena *training accuracy* mempunyai jarak lumayan jauh dengan *test accuracy*.

Mulai dari eksperimen ke-14 sampai 20, peneliti melakukan percobaan dengan memperkecil nilai *learning rate* pada angka 0.001 dengan nilai *epoch* yang berubah-ubah. Dimulai dari nilai *epoch* 60 (eksperimen ke-14) sampai dengan 300 (eksperimen ke-19) mendapatkan hasil *test accuracy* yang terus meningkat. Namun pada eksperimen ke-20 dengan besar *epoch* 400, model yang dibangun mempunyai indikasi *overfit* karena *training accuracy* terlampaui jauh lebih tinggi daripada *testing accuracy*. Adapun akurasi yang dicapai pada model dengan *learning rate* 0.001 tidak lebih tinggi dari model dengan *learning rate* 0.008

dan epoch 60. Kecilnya *learning rate* dan besarnya *epoch* ternyata tidak menjamin hasil akurasi yang didapatkan menjadi lebih baik.

Pada eksperimen ke-1 sampai dengan ke-20 fungsi aktivasi yang digunakan adalah ReLU. Kemudian pada eksperimen ke-21 sampai dengan 25 peneliti menggunakan susunan *layer* dan *learning rate* (0.008) sama dengan eksperimen ke-12 tetapi fungsi aktivasi yang digunakan diubah menjadi Tanh, Sigmoid, dan LeakyRelu. Hasilnya, akurasi yang didapatkan masih tidak lebih tinggi dari model eksperimen ke-12 yang menggunakan fungsi aktivasi ReLU. Saat eksperimen ke-26 sampai dengan 30, fungsi aktivasi yang digunakan adalah LeakyRelu karena berdasarkan eksperimen sebelumnya hasil *testing accuracy* yang didapatkan lebih baik daripada model yang menggunakan fungsi aktivasi Tanh (eksperimen ke-21 sampai dengan 22) dan Sigmoid (eksperimen ke-23 sampai dengan 24) serta nilai *learning rate* diperkecil pada angka 0.001.

Model yang dibangun pada eksperimen ke-26 ternyata masih *underfitting* sehingga peneliti mencoba memperbesar nilai *epoch*. Kemudian eksperimen ke-27 sampai dengan 29 nilai dari *epoch* masih terus diperbesar karena *testing accuracy* mengalami peningkatan dan selisih *training accuracy* serta *testing accuracy* tidak terlalu jauh. Selanjutnya, pada eksperimen ke-30 hasil *testing accuracy* yang didapatkan justru turun jika dibandingkan dengan eksperimen ke-29. Dari eksperimen 21 sampai dengan 30 *testing accuracy* yang didapatkan pun ternyata tidak lebih tinggi dari eksperimen ke-12 yang menggunakan fungsi aktivasi ReLU. Oleh karena itu penggunaan fungsi aktivasi ReLU lebih tepat untuk membangun model penelitian ini.

## 5. KESIMPULAN

Penelitian ini berhasil mengimplementasikan algoritma 1D-CNN dalam menganalisis sentimen ulasan produk kosmetik pada *website Female Daily Review*. Penelitian ini dimulai dengan proses *scraping* data pada *website Female Daily Review* mengenai produk kosmetik emina dan mendapatkan data ulasan produk sejumlah 11119 data. *Dataset* tersebut masih mentah sehingga perlu dilakukan *preprocessing* seperti *data cleansing*, *case folding*, *normalization*, serta *stemming*. Tahap ini bertujuan untuk menghilangkan *noise* yang berpengaruh pada saat proses *mining* dan menyisakan *dataset* sejumlah 11084 data. Setelah itu dilakukan labeling dengan memanfaatkan penelitian yang sudah ada [15], menghasilkan 1666 kelas netral, 8089 kelas positif, dan 1329 kelas negatif. Data yang sudah dikelompokkan ke dalam 3 kelas dilakukan *tokenizing* yang menghasilkan kumpulan kata penyusun atau *vocabulary* dari *dataset* ulasan produk. Jumlah daftar *vocabulary* yang didapatkan beserta dengan kata 'OOV' untuk menyamakan panjang vektor adalah sejumlah 13446 kata. *Dataset* diubah ke dalam bentuk vektor yang tersusun dari *index vocabulary* dan dilakukan *embedding* berdasar nilai fitur yang memanfaatkan *pre-trained word embedding* Word2Vec menggunakan data dari Wikipedia, sehingga nilai dari setiap fitur sudah tersedia. Adapun jumlah fitur atau dimensi yang dipilih adalah sejumlah 300 fitur untuk tiap indeks. Pada pengujian dan pengukuran model menggunakan algoritma 1D-CNN, peneliti melakukan eksperimen sebanyak 30 kali untuk mendapatkan akurasi model terbaik dengan mengubah susunan *layer*, *epoch*,

serta mengubah nilai dari *learning rate*. Rasio yang digunakan sebagai *data testing* dan *data training* adalah sebesar 25:75.

Dari 30 eksperimen yang dilakukan, didapatkan model terbaik untuk menganalisis sentimen terdapat pada eksperimen ke-12 yaitu ketika *learning rate* 0.008, susunan *layer* Conv1D (128,3), MaxPooling 1D, Dropout (0.5), Dense (64), Dense (32), Dropout (0.2), Dense (16), GlobalMaxPooling 1D, *epoch* sebesar 60, dan menghasilkan akurasi sebesar 80.22%. Akurasi yang didapatkan pada penelitian ini tidak sebaik penelitian yang dilakukan oleh [11], hal ini disebabkan oleh adanya *imbalanced* data dari *dataset* yang digunakan di mana selisih jumlah data tiap kelas berbeda jauh. Akurasi model yang diperoleh juga masih lebih rendah dibandingkan [20], di mana pada penelitian tersebut CNN dikombinasikan dengan LSTM. Begitu pula dibandingkan dengan [21] yang menggunakan *Multi-Channel* CNN dengan *Multi-Head Attention Mechanism*, akurasi dari model yang digunakan dalam penelitian ini masih belum lebih baik. Hal ini disebabkan karena selain CNN, dua penelitian terakhir menggunakan metode lain untuk mendapatkan hasil akurasi yang lebih terpercaya.

## DAFTAR PUSTAKA

- [1] K. P. R. Indonesia, "Kementerian Perindustrian Republik Indonesia," 2020.
- [2] B. Gunawan, H. S. Pratiwi, and E. E. Pratama, "Sistem Analisis Sentimen pada Ulasan Produk Menggunakan Metode Naive Bayes," *J. Edukasi dan Penelit. Inform.*, vol. 4, no. 2, pp. 113–118, 2018.
- [3] Z. Salsabil and M. Arfa, "Efektifitas website Femaledaily.com dalam memenuhi kebutuhan informasi pengguna," *J. Ilmu Perpust. - Univ. Diponegoro*, vol. 8, no. 2, pp. 199–210, 2018.
- [4] A. Priyanto and M. R. Ma'arif, "Implementasi Web Scrapping dan Text Mining untuk Akuisisi dan Kategorisasi Informasi dari Internet (Studi Kasus: Tutorial Hidroponik)," *Indones. J. Inf. Syst.*, vol. 1, no. 1, pp. 25–33, 2018, doi: [10.24002/ijis.v1i1.1664](https://doi.org/10.24002/ijis.v1i1.1664).
- [5] R. Siringoringo, "Text Mining dan Klasterisasi Sentimen Pada Ulasan Produk Toko Online," *Tek. Inform. Univ. Prima Indones. Medan*, vol. 2, pp. 1–6, 2019.
- [6] Y. Cahyono, "Analisis Sentiment pada Sosial Media Twitter Menggunakan Naive Bayes Classifier dengan Feature Selection Particle Swarm Optimization dan Term Frequency," *Jurnal Informatika Universitas Pamulang*, vol. 2, no. 1, p. 14, 2017. doi: [10.32493/informatika.v2i1.1500](https://doi.org/10.32493/informatika.v2i1.1500).
- [7] N. D. Mentari, M. A. Fauzi, and L. Muflikhah, "Analisis Sentimen Kurikulum 2013 Pada Sosial Media Twitter Menggunakan Metode K-Nearest Neighbor dan Feature Selection Query Expansion Ranking," *J. Pengemb. Teknol. Inf. dan Ilmu Komput. Univ. Brawijaya*, vol. 2, no. 8, pp. 2739–2743, 2018.
- [8] A. Deviyanto and M. D. R. Wahyudi, "Penerapan Analisis Sentimen Pada Pengguna Twitter Menggunakan Metode K-Nearest Neighbor," *JISKA (Jurnal Inform. Sunan Kalijaga)*, vol. 3, no. 1, p. 1, 2018, doi: [10.14421/jiska.2018.31-01](https://doi.org/10.14421/jiska.2018.31-01).
- [9] H. Ardian and S. Kosasi, "Analisis Sentimen Pada Review Produk Kosmetik Bahasa Indonesia Dengan Metode Naive Bayes," *J. ENTER*, vol. 2, no. 1, pp. 306–320, 2019.
- [10] K. D. Adhine Salsabila, A. Ghiffarie, R. P. Baistama, M. I. Variadi, and M. D. Rhajendra, "Analisis Sentimen



- Terhadap Produk The Body Shop Tea Tree Oil,” *J. Teknol. dan Manaj. Inform.*, vol. 5, no. 2, 2019, doi: [10.26905/jtmi.v5i2.3684](https://doi.org/10.26905/jtmi.v5i2.3684).
- [11] M. Rajwadi, C. Glackin, J. Wall, and N. Cannings, “Explaining Sentiment Classification,” no. December, 2019, doi: [10.21437/Interspeech.2019-2743](https://doi.org/10.21437/Interspeech.2019-2743).
- [12] D. Bhatt *et al.*, “Cnn variants for computer vision: History, architecture, application, challenges and future scope,” *Electron.*, vol. 10, no. 20, pp. 1–28, 2021, doi: [10.3390/electronics10202470](https://doi.org/10.3390/electronics10202470).
- [13] W. Yin, K. Kann, M. Yu, and H. Schütze, “Comparative Study of CNN and RNN for Natural Language Processing,” 2017, [Online]. Available: <http://arxiv.org/abs/1702.01923>
- [14] N. A. Shafirra and I. Irhamah, “Klasifikasi Sentimen Ulasan Film Indonesia dengan Konversi Speech-to-Text (STT) Menggunakan Metode Convolutional Neural Network (CNN),” *J. Sains dan Seni ITS*, vol. 9, no. 1, 2020, doi: [10.12962/j23373520.v9i1.51825](https://doi.org/10.12962/j23373520.v9i1.51825).
- [15] D. H. Wahid and A. SN, “Peringkasan Sentimen Esktraktif di Twitter Menggunakan Hybrid TF-IDF dan Cosine Similarity,” *IJCCS (Indonesian J. Comput. Cybern. Syst.*, vol. 10, no. 2, p. 207, 2016, doi: [10.22146/ijccs.16625](https://doi.org/10.22146/ijccs.16625).
- [16] Y. D. Prabowo, T. L. Marselino, and M. Suryawiguna, “Pembentukan Vector Space Model Bahasa Indonesia Menggunakan Metode Word to Vector,” *J. Buana Inform.*, vol. 10, no. 1, p. 29, 2019, doi: [10.24002/jbi.v10i1.2053](https://doi.org/10.24002/jbi.v10i1.2053).
- [17] B. Zhao, “Encyclopedia of Big Data,” *Encycl. Big Data*, pp. 3–5, 2020, doi: [10.1007/978-3-319-32001-4](https://doi.org/10.1007/978-3-319-32001-4).
- [18] E. Sihite, R. D. Ramadhani, M. Zidny, and R. Adhitama, “Text Processing Clustering dalam Menentuan Profesi Berdasarkan Data Twitter,” pp. 103–108, 2018.
- [19] E. N. Arrofiqoh and H. Harintaka, “Implementasi Metode Convolutional Neural Network Untuk Klasifikasi Tanaman Pada Citra Resolusi Tinggi,” *Geomatika*, vol. 24, no. 2, p. 61, 2018, doi: [10.24895/jig.2018.24-2.810](https://doi.org/10.24895/jig.2018.24-2.810).
- [20] A. U. Rehman, A. K. Malik, B. Raza, and W. Ali, “A Hybrid CNN-LSTM Model for Improving Accuracy of Movie Reviews Sentiment Analysis,” *Multimed. Tools Appl.*, vol. 78, no. 18, pp. 26597–26613, 2019, doi: [10.1007/s11042-019-07788-7](https://doi.org/10.1007/s11042-019-07788-7).
- [21] Y. Feng and Y. Cheng, “Short Text Sentiment Analysis Based on Multi-Channel CNN with Multi-Head Attention Mechanism,” *IEEE Access*, vol. 9, pp. 19854–19863, 2021, doi: [10.1109/ACCESS.2021.3054521](https://doi.org/10.1109/ACCESS.2021.3054521).

## NOMENKLATUR

TP	<i>True Positive</i>
TN	<i>True Negative</i>
FP	<i>False Positive</i>
FN	<i>False Negative</i>
FM <sub>a,b</sub>	<i>feature map ke-a,b</i>
bias	bias pada <i>feature map</i>
Z <sub>c,d</sub>	bobot pada <i>convolution kernel ke-c,d</i>
X	input
a	1,2, ..., A. A adalah panjang <i>pixel feature map</i>
b	1,2, ..., B. B adalah lebar <i>pixel feature map</i>
c	1,2, ..., C. C ukuran panjang <i>convolution kernel</i>
d	1,2, ..., D. D ukuran lebar <i>convolution kernel</i> .
y <sub>ijk</sub>	vektor yang berisi nilai antara 0 dan 1
x	vektor yang berisi nilai yang didapatkan dari lapisan <i>fully-connected</i> terakhir

<https://doi.org/10.25077/TEKNOSI.v8i3.2022.153-163>

$y_i \in C_c$  fungsi indikator dari pengamatan ke ‘i’ yang termasuk dalam kategori ke-‘c’

$P_{\text{model}}[y_i \in C_c]$  fungsi indikator dari pengamatan ke ‘i’ yang termasuk dalam kategori ke-‘c’

## BIODATA PENULIS

### Erwin Yudi Hidayat

Menamatkan pendidikan sarjana di Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Dian Nuswantoro, sebagai Sarjana Komputer (S.Kom). Studi jenjang S2 diselesaikan di Universiti Teknikal Malaysia Melaka (UTeM), dengan konsentrasi pada bidang kecerdasan buatan. Penulis tertarik dalam penelitian *machine learning*, *deep learning*, dan pengolahan citra digital. Saat ini menjadi staf pengajar pada almamater di tempat kuliah S1 dulu ditempuh.

### Devioletta Handayani

Penulis adalah alumni Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Dian Nuswantoro, Semarang. Ketertarikan akan *text mining* dan kecerdasan buatan, mengantarkan penulis terhadap penelitian analisis sentimen. Gelar S.Kom yang diperoleh diselesaikan dalam waktu 3,5 tahun dan lulus dengan predikat *cumlaude*.

## LAMPIRAN

Tabel 11. Model Eksperimen

Eks ke-	LR	Susunan Layer	E-poch	Hasil
1	0.01	Conv1D(32,3,ReLU)	20	Loss: 0.2332
		MaxPooling1D		Accuracy:
		Dense(16,ReLU)		0.9155
		Dense (32,ReLU)		Val_loss: 0.9151
		Dense (8,ReLU)		Val_Acc: 0.6994
		GlobalMaxPooling1D		
2	0.001	Conv1D(32,3,ReLU)	20	Loss: 0.6693
		MaxPooling1D		Accuracy:
		Dense(16,ReLU)		0.7438
		Dense (32,ReLU)		Val_loss: 0.7692
		Dense (8,ReLU)		Val_acc: 0.7200
		GlobalMaxPooling1D		
3	0.008	Conv1D(32,3,ReLU)	60	Loss: 0.3030
		MaxPooling1D		Accuracy:0.8771
		Dropout(0.3)		Val_loss:0.5957
		Dense(64,ReLU)		Val_acc:0.7759
		Dense (32,ReLU)		
		Dense(16,ReLU)		
		GlobalMaxPooling1D		
4	0.008	Conv1D(32,3,ReLU)	60	Loss: 0.4223
		MaxPooling1D		Accuracy:
		Dropout(0.5)		0.8344
		Dense(64,ReLU)		Val_loss: 0.6180
		Dense (32,ReLU)		Val_acc: 0.7643
		Dense(16,ReLU)		
		GlobalMaxPooling1D		
5	0.01	Conv1D(32,3,ReLU)	50	Loss: 0.5642
		MaxPooling1D		Accuracy:
		Dropout(0.5)		0.7869
		Dense(32,ReLU)-L2		Val_loss: 0.6050
		Dense (64,ReLU)-L2		Val_acc: 0.7690
		Dropout(0.2)		
		Dense(16,ReLU)-L2		
		GlobalMaxPooling1D		

6	0.008	Conv1D(32,3,ReLU) MaxPooling1D Dropout(0.5) Dense(32,ReLU)-L2 Dense (64,ReLU)-L2 Dropout(0.2) Dense(16,ReLU)-L2 GlobalMaxPooling1D	50	Loss: 0.5932 Accuracy: 0.7741 Val_loss: 0.6087 Val_acc: 0.7737	Dropout(0.2) Dense(16) GlobalMaxPooling1D
7	0.01	Conv1D(32,3,ReLU) MaxPooling1D Dropout(0.5) Dense(32,ReLU)-L2 Dense (64,ReLU)-L2 Dropout(0.2) Dense(16,ReLU)-L2 GlobalMaxPooling1D	80	Loss: 0.5282 Accuracy: 0.7977 Val_loss: 0.5866 Val_acc: 0.7770	Conv1D(128,3) MaxPooling1D Dropout(0.5) Dense(64) Dense (32) Dropout(0.2) Dense(16) GlobalMaxPooling1D
8	0.01	Conv1D(32,3,ReLU) MaxPooling1D Dropout(0.5) Dense(32,ReLU)-L2 Dense(64,ReLU)-L2 Dropout(0.2) Dense(16,ReLU)-L2 GlobalMaxPooling1D	80	Loss: 0.5235 Accuracy: 0.8082 Val_loss: 0.6047 Val_acc: 0.7784	Conv1D(128,3,ReLU) MaxPooling1D Dropout(0.5) Dense(64,ReLU) Dense (32,ReLU) Dropout(0.2) Dense(16,ReLU) GlobalMaxPooling1D
9	0.01	Conv1D(32,3,ReLU) MaxPooling1D Dropout(0.5) Dense(32,ReLU)-L2 Dense (64,ReLU)-L2 Dropout(0.2) Dense(16,ReLU)-L2 GlobalMaxPooling1D	60	Loss: 0.5453 Accuracy: 0.7935 Val_loss: 0.5611 Val_acc: 0.7813	Conv1D(128,3,ReLU) MaxPooling1D Dropout(0.5) Dense(64,ReLU) Dense (32,ReLU) Dropout(0.2) Dense(16,ReLU) GlobalMaxPooling1D
10	0.008	Conv1D(32,3,ReLU) MaxPooling1D Dropout(0.5) Dense(32,ReLU)-L2 Dense (64,ReLU) Dropout(0.3) Dense(16,ReLU) GlobalMaxPooling1D	80	Loss: 0.5522 Accuracy: 0.7980 Val_loss: 0.5917 Val_acc: 0.7846	Conv1D(128,3,ReLU) MaxPooling1D Dropout(0.5) Dense(64,ReLU) Dense (32,ReLU) Dropout(0.2) Dense(16,ReLU) GlobalMaxPooling1D
11	0.008	Conv1D(128,3,ReLU) MaxPooling1D Dropout(0.5) Dense(64,ReLU) Dense (32,ReLU) Dense(16,ReLU) GlobalMaxPooling1D	60	Loss: 0.3377 Accuracy: 0.8704 Val_loss: 0.5595 Val_acc: 0.7907	Conv1D(128,3,ReLU) MaxPooling1D Dropout(0.5) Dense(64,ReLU) Dense (32,ReLU) Dropout(0.2) Dense(16,ReLU) GlobalMaxPooling1D
12	0.008	Conv1D(128,3,ReLU) MaxPooling1D Dropout(0.5) Dense(64,ReLU) Dense (32,ReLU) Dropout(0.2) Dense(16,ReLU) GlobalMaxPooling1D	60	Loss: 0.3893 Accuracy: 0.8464 Val_loss: 0.5180 Val_acc: 0.8022	Conv1D(128,3,tanh) MaxPooling1D Dropout(0.5) Dense(64,tanh) Dense (32,tanh) Dropout(0.2) Dense(16,tanh) GlobalMaxPooling1D
13	0.008	Conv1D(128,3,ReLU) MaxPooling1D Dropout(0.5) Dense(64,ReLU) Dense (32,ReLU) Dropout(0.2) Dense(16,ReLU) GlobalMaxPooling1D	80	Loss: 0.3460 Accuracy: 0.8644 Val_loss: 0.5176 Val_acc: 0.7936	Conv1D(128,3,tanh) MaxPooling1D Dropout(0.5) Dense(64,tanh) Dense (32,tanh) Dropout(0.2) Dense(16,tanh) GlobalMaxPooling1D
14	0.001	Conv1D(128,3) MaxPooling1D Dropout(0.5) Dense(64) Dense (32)	60	Loss: 0.6773 Accuracy:0.7345 Val_loss:0.6266 Val_acc:0.7560	Conv1D(128,3,ReLU) MaxPooling1D Dropout(0.5) Dense(64,ReLU) Dense (32,ReLU) Dropout(0.2) Dense(16,ReLU) GlobalMaxPooling1D
15	0.001	Conv1D(128,3) MaxPooling1D Dropout(0.5) Dense(64) Dense (32) Dropout(0.2) Dense(16) GlobalMaxPooling1D	80	Loss: 0.6315 Accuracy:0.7539 Val_loss:0.6049 Val_acc:0.7636	Conv1D(128,3,ReLU) MaxPooling1D Dropout(0.5) Dense(64,ReLU) Dense (32,ReLU) Dropout(0.2) Dense(16,ReLU) GlobalMaxPooling1D
16	0.001	Conv1D(128,3,ReLU) MaxPooling1D Dropout(0.5) Dense(64,ReLU) Dense (32,ReLU) Dropout(0.2) Dense(16,ReLU) GlobalMaxPooling1D	100	Loss: 0.6199 Accuracy:0.7519 Val_loss:0.5930 Val_acc:0.7647	Conv1D(128,3,ReLU) MaxPooling1D Dropout(0.5) Dense(64,ReLU) Dense (32,ReLU) Dropout(0.2) Dense(16,ReLU) GlobalMaxPooling1D
17	0.001	Conv1D(128,3,ReLU) MaxPooling1D Dropout(0.5) Dense(64,ReLU) Dense (32,ReLU) Dropout(0.2) Dense(16,ReLU) GlobalMaxPooling1D	150	Loss: 0.5582 Accuracy:0.7730 Val_loss:0.5757 Val_acc:0.7755	Conv1D(128,3,ReLU) MaxPooling1D Dropout(0.5) Dense(64,ReLU) Dense (32,ReLU) Dropout(0.2) Dense(16,ReLU) GlobalMaxPooling1D
18	0.001	Conv1D(128,3,ReLU) MaxPooling1D Dropout(0.5) Dense(64,ReLU) Dense (32,ReLU) Dropout(0.2) Dense(16,ReLU) GlobalMaxPooling1D	200	Loss: 0.5037 Accuracy:0.7973 Val_loss:0.5565 Val_acc:0.7791	Conv1D(128,3,ReLU) MaxPooling1D Dropout(0.5) Dense(64,ReLU) Dense (32,ReLU) Dropout(0.2) Dense(16,ReLU) GlobalMaxPooling1D
19	0.001	Conv1D(128,3,ReLU) MaxPooling1D Dropout(0.5) Dense(64,ReLU) Dense (32,ReLU) Dropout(0.2) Dense(16,ReLU) GlobalMaxPooling1D	300	Loss: 0.4197 Accuracy:0.8375 Val_loss:0.5407 Val_acc:0.7827	Conv1D(128,3,ReLU) MaxPooling1D Dropout(0.5) Dense(64,ReLU) Dense (32,ReLU) Dropout(0.2) Dense(16,ReLU) GlobalMaxPooling1D
20	0.001	Conv1D(128,3,ReLU) MaxPooling1D Dropout(0.5) Dense(64,ReLU) Dense (32,ReLU) Dropout(0.2) Dense(16,ReLU) GlobalMaxPooling1D	400	Loss: 0.3145 Accuracy:0.8749 Val_loss:0.5375 Val_acc:0.7871	Conv1D(128,3,ReLU) MaxPooling1D Dropout(0.5) Dense(64,ReLU) Dense (32,ReLU) Dropout(0.2) Dense(16,ReLU) GlobalMaxPooling1D
21	0.008	Conv1D(128,3,tanh) MaxPooling1D Dropout(0.5) Dense(64,tanh) Dense (32,tanh) Dropout(0.2) Dense(16,tanh) GlobalMaxPooling1D	60	Loss: 0.3869 Accuracy:0.8414 Val_loss: 0.6090 Val_acc: 0.7874	Conv1D(128,3,tanh) MaxPooling1D Dropout(0.5) Dense(64,tanh) Dense (32,tanh) Dropout(0.2) Dense(16,tanh) GlobalMaxPooling1D
22	0.008	Conv1D(128,3,tanh) MaxPooling1D Dropout(0.5) Dense(64,tanh) Dense (32,tanh) Dropout(0.2) Dense(16,tanh) GlobalMaxPooling1D	80	Loss: 0.3473 Accuracy: 0.8572 Val_loss: 0.6892 Val_acc: 0.7802	Conv1D(128,3,ReLU) MaxPooling1D Dropout(0.5) Dense(64,ReLU) Dense (32,ReLU) Dropout(0.2) Dense(16,ReLU) GlobalMaxPooling1D
23	0.008	Conv1D(128,3,sigmoid)	60	Loss: 0.7738	Conv1D(128,3,sigmoid)

