

uji

by fairy ogyu

Submission date: 28-Jan-2022 10:21PM (UTC-0500)

Submission ID: 1750408203

File name: uji.docx (287.17K)

Word count: 3136

Character count: 19346

Sentiment Analysis Tweet Pilkada 2020 Saat Pandemi COVID-19 di Media Sosial Twitter Menggunakan Metode ID Convolutional Neural Network

¹Ganjar Tata Pangestu¹, ²Miftahurrahma Rosyda, S.Kom, M.Eng.

¹Fakultas Teknik Industri, Program Studi Teknik Informatika, Universitas Ahamad Dahlan, Kota Yogyakarta, Indonesia
Email: ¹tpangestu.official@gmail.com, ²miftahurrahma.rosyda@tif.uad.ac.id
Email Penulis Korespondensi: tpangestu.official@gmail.com

Abstrak— Pilkada 2020 menjadi perdebatan karena berlangsung di tengah pandemi COVID-19. Munculnya komentar dari berbagai media sosial seperti Twitter. Ada berbagai opini publik yang setuju Pilkada tetap digelar, ada juga opini publik lain yang mendukung penundaan Pilkada hingga pandemi COVID-19 berakhir. Opini yang berbeda tersebut membutuhkan *Sentiment Analysis* yang bertujuan mendapatkan atau mengetahui opini umum Pemilihan Kepala Daerah 2020 di masa pandemi *Coronavirus*. Sebanyak 200 data yang berupa *tweet* yang terbagi menjadi 2 data yaitu 20% data uji dan 80% data latih diperoleh dengan mengambil data dari Twitter menggunakan *twint library*, berdasarkan kata kunci yang telah ditentukan. Kumpulan data yang dihasilkan diklasifikasikan menjadi tiga kelas yaitu positif, netral, dan negatif. Dalam pengujian atau penelitian ini menggunakan *deep learning* yang menggunakan klasifikasi *Convolutional Neural Network*, dikarenakan sudah teruji ampuh dalam kasus pemrosesan Bahasa alami dan dapat mendapatkan hasil yang bagus dalam pengelompokan kalimat.

Kata Kunci: Pemrosesan Bahasa Alami, NLP, Sentimen Analisis, Python, ID Convolutional Neural Network

Abstract— Pilkada 2020 is a debate since it takes place in the midst of the COVID-19 pandemic. The emergence of comments from several social media such as Twitter. There are various public opinions that agree that the Pilkada will still be held, there are also other public opinions that support the postponement of the Pilkada until the COVID-19 pandemic ends. These different opinions require *Sentiment Analysis* which aims to obtain or find out the general opinion of the 2020 Regional Head Election during the *Coronavirus* pandemic. A total of 200 data in the form of tweets which are divided into 2 data, namely 20% test data and 80% training data obtained by retrieving data from Twitter using the *twint library*, based on predetermined keywords. The resulting data set is classified into three classes namely positive, neutral, and negative. In this test or research, *deep learning* uses the *Convolutional Neural Network* classification, because it has been proven effective in the case of natural language processing and can get good results in grouping sentences.

Keywords: Natural Language Processing, NLP, Sentiment Analysis, Python, ID Convolutional Neural Network

1. PENDAHULUAN

Penyelenggaraan pilkada 2020 mendapatkan perdebatan, perdebatan tersebut dikarenakan pelaksanaan pilkada dilakukan pada masa pandemi COVID-19. Kementerian Kesehatan sudah mengeluarkan peraturan pemberlakuan PSBB (pembatasan sosial berskala besar) pada seluruh wilayah dan provinsi Indonesia [1]. Walaupun begitu, KPU tetap merencanakan pelaksanaan pilkada sampai dengan akhir tahun. Dengan adanya peraturan KPU no. 13 tahun 2020 tentang pelaksanaan pemilihan umum serentak lanjutan dalam kondisi bencana non alam COVID-19 dilaksanakan dengan menerapkan protokol kesehatan di setiap pemilihan umumnya.

Dengan memperhatikan dua keadaan yang kondisinya sama-sama *urgent*, banyak komentar bermunculan dari berbagai lini media sosial yang ada contohnya *twitter*. Munculnya kecemasan pada masyarakat di internet dikarenakan jika pemilihan kepala daerah 2020 tetap bersamaan dilakukan, maka dalam kondisi pandemi COVID-19 seperti ini ditakutkan akan mengalami ke lonjakan kasus positif COVID-19. Pelaksanaan pemilihan kepala daerah ini juga dapat berpotensi menyebarkan dan menularkan virus COVID-19. Walaupun begitu tidak mungkin dapat diprediksikan bahwa pandemi COVID-19 akan segera berakhir dalam waktu dekat, sebaliknya pemilihan kepala daerah juga sama pentingnya untuk lekas dilakukan agar untuk masyarakat mendapatkan kedaulatan dan kesejahteraannya [2].

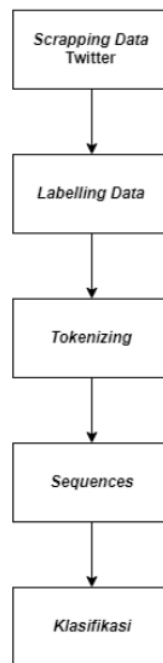
Sentiment Analysis adalah teknik untuk memperoleh pandangan dan informasi khusus dalam suatu kalimat. *Sentiment Analysis* mampu mengetahui perasaan atau sentimen penulis terhadap suatu topik atau objek, apakah netral, negatif, atau positif [3]. *Sentiment Analysis* melibatkan perhitungan dengan menggunakan komputer untuk mengetahui perasaan yang terkait dalam kalimat. NLP atau *Natural Language Processing* terkenal sebagai metode komputer untuk memahami input dari Bahasa manusia [4]. Dengan NLP mesin dapat memahami Bahasa manusia dengan cara menerjemahkan terlebih dahulu Bahasa alami manusia ke bentuk yang dapat dipahami mesin [5]. Pada hasil penelitian milik Young-Seob Jeong dan Hannah Kim memperlihatkan kalau pengimplementasian algoritma *Convolution Neural Network (CNN)* ini benar-benar efektif diterapkan pada NLP karena juga bisa memperoleh hasil yang sangat baik pada penggolongan dalam kalimat [6].

Berdasarkan riset referensi yang sudah dilangsungkan, mendapatkan hasil bahwa masih minim penelitian yang meneliti jumlah epoch yang dipakai pada model CNN. Yang dimaksud epoch adalah berapa kali model *neural*

network telah melihat dataset, karena dalam satu epoch masih terlalu besar untuk dimasukkan (*feeding*) kedalam komputer oleh karena itu maka perlu dilakukan pembagian ke dalam satuan kecil (*batches*) [7]. Pada epoch kecil dapat menghasilkan performa yang buruk dikarenakan penyetaraan indikator yang belum utuh, sedangkan epoch yang besar dapat meningkatkan waktu perhitungan atau komputasi dan risiko dimana data yang digunakan untuk pelatihan adalah yang terbaik. Jadi, jika pengujian dilakukan dengan data yang berbeda maka akan dapat menurunkan akurasi [8].

2. METODOLOGI PENELITIAN

Masalah utama pada klasifikasi sentimen adalah cara mengidentifikasi sentimen dalam sebuah teks, apakah teks memiliki sentimen netral, positif, atau negatif. Gambar 1 adalah gambaran proses pada penelitian ini [9]. Langkah pertama, pengambilan data atau *scraping* data dari twitter, selanjutnya seluruh data diberi label menjadi 3 kelas yaitu positif, negatif, dan netral. Selanjutnya pada proses *pre-processing* memakai teknik NLP. Setelah proses *pre-processing* selesai maka menghasilkan token-token yang akan menjadi *input* pada proses klasifikasi. Pada proses klasifikasi memakai algoritma CNN atau *Convolutional Neural Network* yang merupakan skema dari *deep learning*.



Gambar 1. Alur Penelitian

2.1 Data Scraping

Dataset yang dipakai pada penelitian dikumpulkan dari twitter dengan cara *scraping* yang memakai pemrograman *Python*, menggunakan *library twint* untuk mengumpulkan data melalui proses ekstrasi informasi menggunakan *Hypertext Transfer Protocol (HTTP)*. *Dataset* sejumlah 193 tweet dikumpulkan dengan kata kunci pencarian di antaranya “pilkada 2020”, “covid”, dan “pandemik”. Pada tabel 1 dapat dilihat merupakan contoh dari hasil *scraping* data, data *tweet* yang dikumpulkan dipilih secara *random* yang mengacu pada pemilihan kepala daerah pada saat pandemik *COVID-19*.

Tabel 1. Hasil Contoh Dari *Data Scraping*

Tanggal, Waktu	Tweet
2020-12-15 20:17:49	Nggak heran kalo sampean ngomong tidak terbukti ada klaster baru Covid-19 setelah Pilkada 2020 ... sejak awal pun sampean senang berkelakar tentang Covid-19.

2020-12-17 10:19:04

Pemerintah mengklaim pelaksanaan Pilkada serentak 2020 tak memunculkan klaster baru Covid-19.

Namun, kenyataannya temuan klaster virus korona tersebut mulai bermunculan dan berpotensi terus bertambah. #MediaLawanCovid19

2020-12-15 13:54:19

#WaktunyaLokadata <https://t.co/C7hbnafOrj>
Pilkada 2020 tidak lahirkan klaster baru covid 19
Bravo @KPU_ID

2021-01-20 17:30:05

Semoga tidak ada korban jiwa dan tidak lagi ada yang menganggap remeh, kita berisik soal prokes dan gak ikut "hajatan" itu cuma ihtiar untuk menghindari.. 140 Komisioner KPU Pusat-Daerah
Terpapar Covid di Pilkada 2020
<https://t.co/mTsqRMvB1t>

2.2 Labelling Data

Pelabelan untuk *dataset* dilakukan secara manual. *Dataset* diberikan label netral, positif, dan negatif. Hasil dari pelabelan terdapat 22 tweet dengan sentimen netral, 73 tweet sentimen negatif, dan 98 tweet dengan sentimen positif.

2.3 Tokenizing

Proses operasi memisahkan teks menjadi potongan-potongan berupa *token* [10]. Pemotongan ini bisa berupa potongan kata, huruf, ataupun kalimat. Berikut tabel 2 merupakan contoh proses tokenisasi.

Tabel 2. Contoh dari Proses *Tokenizing*

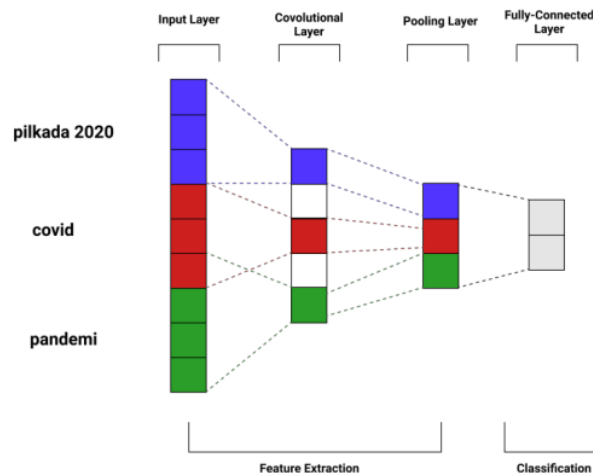
Teks Asli	"cegah penyebaran covid saat pilkada"
Sesudah Tokenisasi	1. 'cegah' 2. 'penyebaran' 3. 'covid' 4. 'saat' 5. 'pilkada'

2.4 Sequences

Suatu proses setelah tokenisasi di mana jika pada proses tokenisasi terdapat sebuah masalah yang di mana sebuah kata memiliki jumlah dan huruf yang sama seperti "amal" dan "lama". Mesin akan membaca kedua kata tersebut adalah sama, maka dari itu kita membutuhkan *sequences*. *Sequences* akan menampung setiap token yang ada ke dalam bentuk *array* [11].

2.5 Klasifikasi

Setelah melewati tahapan *tokenizing* dan *sequences*, kemudian data tersebut diklasifikasikan menggunakan algoritma *Convolutional Neural Network* yang merupakan skema dari *deep learning*. *Convolutional Neural Network* adalah bidang dari Artificial Neural Network (ANN), bidang tersebut dapat memperhitungkan data dan mendapatkan ketepatan yang tinggi [12]. Model *Convolutional Neural Network* banyak menyelesaikan berbagai masalah mengenai pemrosesan gambar, saat sekarang *Convolutional Neural Network* sudah dikembangkan untuk digunakan pada *Natural Language Processing* seperti sentiment analysis, text summary, klasifikasi polaritas emosional, dll [13]. Pada proses CNN ini input berupa tweet yang mengandung unsur "pilkada", "covid", dan "pandemik" yang kemudian akan dijadikan input untuk convolution layer. Setelah pada proses convolutional selesai, hasilnya akan dimasukkan ke dalam aktivasi ReLU. Hasil fungsi dari aktivasi ReLU mengalami proses *down sampling* pada *Pooling Layer*. Lalu hasil dari *Pooling Layer* digunakan sebagai input untuk *Fully Connected Layer*. Berikut Gambar 2 merupakan gambaran arsitektur CNN pada penelitan ini.



Gambar 2. Gambaran Arsitektur CNN

Convolutional Neural Network atau dapat disebut dengan CNN adalah metode dari pembelajaran mesin yang merupakan pengembangan *Multi Layer Perceptron*, sehingga jenis jaringan *Convolutional Neural Network* adalah jenis *multilayer* dapat dikenal atau disebut juga sebagai *fully-connected layer*, karena CNN merupakan jaringan *multilayer* maka hasil dari satu layer akan dijadikan untuk masukkan layer selanjutnya. Biasanya CNN terstruktur dalam 3 layer yaitu *layer input*, *hidden layer*, dan *layer hasil* [12]. Pada penelitian ini menggunakan *Convolutional Neural Network layer* yang sederhana [14], terstruktur dari *convolutional layer*, *max pooling layer*, *dropout layer* serta *fully connected output layer* [15].

Pada *setting* model studi kali ini ditetapkan sejumlah 8 unit *convolutional* dan 300 *vector size*. Pada 8 unit *convolutional layer* diterapkan *filter 16, 32, dan 64*. Adam dipilih sebagai *Optimizer* pada penelitian ini [16] dikarenakan dianggap sebagai optimasi terbaik [17] dan aktifasi *ReLU* atau disebut sebagai *Rectified Linear Unit* yang digunakan untuk aktifasi non-linear dikarenakan mempunyai kelebihan kerumitan yang rendah [18].

3. HASIL DAN PEMBAHASAN

Sebanyak 193 *dataset tweet* dipakai dalam penelitian ini. *Dataset* yang telah dikumpulkan terbagi menjadi tiga kategori yaitu, 0 dengan sentimen netral, 1 dengan sentimen positif, dan 2 dengan sentimen negatif. Tabel 3 merupakan hasil dari pembagian data atau *labelling data*.

Tabel 3. Hasil Proses *Labelling Data*

0 (Netral)	1 (Positif)	2 (Negatif)
30	103	67

Pada tahapan *tokenizing*, *dataset tweet* kalimat dibagi menjadi per kata yang bertujuan mempermudah untuk mengelola teks lebih lanjut, melalui proses *tokenizing* ini dari 200 tweet dipecah menjadi 1.703 kata atau *token*.

```

[[583, 584, 31, 16, 207, 247, 9, 29, 585, 586, 208, 24, 587, 339, 588, 209, 3, 68, 18, 2, 7, 41, 9, 589, 41, 590, 591, 592, 3
40, 119, 248, 341, 593, 594, 249, 174, 59, 1, 8, 342], [595, 60, 42, 15, 104, 37, 32, 2, 9, 1, 3, 4, 5, 6, 596], [597, 343, 3
44, 2, 7, 598, 599, 9, 86, 210, 600, 345, 1, 601, 346, 175, 602, 603, 347, 604, 87, 605, 606, 607, 348, 608, 609, 349, 349,
1, 120, 610, 121, 25, 611, 612, 86, 22, 3, 613], [350, 33, 17, 351, 614, 8, 33, 142, 17, 19, 615, 616, 207, 617, 250, 143, 8,
352, 176, 618, 177, 251, 619, 26, 252, 60, 42, 15, 104, 37, 32, 2, 9, 1, 3, 4, 5, 6, 620], [1, 10, 3, 253, 21, 46, 11, 2, 7,
353, 105, 27, 47, 106, 88, 144, 35, 354, 355, 89, 38, 90, 356, 357, 50, 4, 5, 6, 621], [106, 88, 47, 69, 358, 359, 360, 1, 1
0, 3, 253, 21, 46, 11, 2, 7, 622, 177, 105, 27, 47, 144, 35, 354, 355, 89, 38, 90, 356, 357, 50, 4, 5, 6, 623, 4, 5, 6, 624],
[254, 625, 105, 106, 88, 48, 47, 69, 358, 359, 360, 1, 10, 3, 253, 21, 46, 11, 2, 7, 211, 626, 4, 5, 6, 627], [70, 79, 8, 91,
92, 71, 93, 24, 13, 43, 39, 18, 2, 7, 8, 51, 178, 49, 179, 1, 3, 72, 628, 629, 630, 631, 632, 4, 5, 6, 633], [361, 145, 634,
1, 10, 3, 146, 212, 8, 28, 2, 7, 4, 5, 6, 635, 147, 255], [70, 107, 122, 20, 40, 36, 94, 79, 8, 91, 92, 71, 93, 24, 148, 362,
636, 44, 40, 36, 13, 43, 39, 18, 2, 7, 8, 51, 68, 49, 38, 28, 8, 108, 59, 1, 109, 3, 72, 20, 40, 36, 4, 5, 6, 637], [15, 60,
42, 32, 2, 7, 73, 23, 1, 3, 638, 639, 4, 5, 6, 640, 26, 641, 180, 144, 642, 8, 363, 643, 4, 5, 6, 644, 645, 4, 5, 6, 646, 64
7, 26, 256, 4, 5, 6, 648], [60, 42, 32, 2, 7, 31, 1, 649, 4, 5, 6, 650], [254, 651, 74, 15, 364, 652, 213, 60, 42, 15, 257,
9, 365, 104, 366, 37, 32, 2, 7, 73, 110, 1, 10, 3], [60, 42, 15, 104, 37, 32, 2, 9, 1, 3, 4, 5, 6, 653, 350, 363, 654], [74,
367, 655, 368, 656, 657, 658, 52, 659, 660, 181, 369, 661, 35, 258, 214, 209, 16, 662, 56, 663, 21, 53, 3, 664, 75, 121, 259,
214, 22, 4, 5, 6, 665], [2, 666, 1, 667, 668, 669, 670, 26, 671, 1, 10, 260, 53, 3, 95, 181, 87, 672, 18, 149, 673, 674, 2,
7, 675, 370, 676, 261, 123, 371, 677, 9, 123, 124, 678, 679, 174, 680, 262, 681, 4, 5, 6, 682], [73, 23, 1, 3, 33, 17, 52, 3
0, 25, 2, 7, 4, 5, 6, 683], [15, 60, 42, 32, 2, 7, 31, 1, 10, 3, 4, 5, 6, 684], [372, 685, 686, 687, 688, 689, 690, 691, 1, 1
0, 3, 120, 373, 692, 18, 149, 125, 2, 7, 9, 86, 374, 693, 694, 4, 5, 6, 695], [87, 351, 1, 3, 123, 696, 263, 2, 375, 17, 38,

```

Gambar 3. Contoh Hasil Proses Tokenizing

Setelah proses *tokenizing* selanjutnya dilakukan proses untuk mengubah setiap bentuk kata dari hasil *tokenizing* menjadi bentuk *sequentials* yang ditampung dalam bentuk array. Karena pada setiap kalimat jumlah katanya tidak sama maka dilakukan proses normalisasi dengan cara membuat jumlah kalimatnya fit, yang di mana kalau jumlah katanya lebih maka akan dipotong dan kalau jumlah katanya kurang maka akan diberikan *padding*, maksimal kata pada proses normalisasi ini sebanyak 100 kata.

```

[[[583, 584, 31, 16, 207, 247, 9, 29, 585, 586, 208, 24, 587, 339, 588, 209, 3, 68, 18, 2, 7, 41, 9, 589, 41, 590, 591, 592, 3
40, 119, 248, 341, 593, 594, 249, 174, 59, 1, 8, 342], [595, 60, 42, 15, 104, 37, 32, 2, 9, 1, 3, 4, 5, 6, 596], [597, 343, 3
44, 2, 7, 598, 599, 9, 86, 210, 600, 345, 1, 601, 346, 175, 602, 603, 347, 604, 87, 605, 606, 607, 348, 608, 609, 349, 349,
1, 120, 610, 121, 25, 611, 612, 86, 22, 3, 613], [350, 33, 17, 351, 614, 8, 33, 142, 17, 19, 615, 616, 207, 617, 250, 143, 8,
352, 176, 618, 177, 251, 619, 26, 252, 60, 42, 15, 104, 37, 32, 2, 9, 1, 3, 4, 5, 6, 620], [1, 10, 3, 253, 21, 46, 11, 2, 7,
353, 105, 27, 47, 106, 88, 144, 35, 354, 355, 89, 38, 90, 356, 357, 50, 4, 5, 6, 621], [106, 88, 47, 69, 358, 359, 360, 1, 1
0, 3, 253, 21, 46, 11, 2, 7, 622, 177, 105, 27, 47, 144, 35, 354, 355, 89, 38, 90, 356, 357, 50, 4, 5, 6, 623, 4, 5, 6, 624],
[254, 625, 105, 106, 88, 48, 47, 69, 358, 359, 360, 1, 10, 3, 253, 21, 46, 11, 2, 7, 211, 626, 4, 5, 6, 627], [70, 79, 8, 91,
92, 71, 93, 24, 13, 43, 39, 18, 2, 7, 8, 51, 178, 49, 179, 1, 3, 72, 628, 629, 630, 631, 632, 4, 5, 6, 633], [361, 145, 634,
1, 10, 3, 146, 212, 8, 28, 2, 7, 4, 5, 6, 635, 147, 255], [70, 107, 122, 20, 40, 36, 94, 79, 8, 91, 92, 71, 93, 24, 148, 362,
636, 44, 40, 36, 13, 43, 39, 18, 2, 7, 8, 51, 68, 49, 38, 28, 8, 108, 59, 1, 109, 3, 72, 20, 40, 36, 4, 5, 6, 637], [15, 60,
42, 32, 2, 7, 73, 23, 1, 3, 638, 639, 4, 5, 6, 640, 26, 641, 180, 144, 642, 8, 363, 643, 4, 5, 6, 644, 645, 4, 5, 6, 646, 64
7, 26, 256, 4, 5, 6, 648], [60, 42, 32, 2, 7, 31, 1, 649, 4, 5, 6, 650], [254, 651, 74, 15, 364, 652, 213, 60, 42, 15, 257,
9, 365, 104, 366, 37, 32, 2, 7, 73, 110, 1, 10, 3], [60, 42, 15, 104, 37, 32, 2, 9, 1, 3, 4, 5, 6, 653, 350, 363, 654], [74,
367, 655, 368, 656, 657, 658, 52, 659, 660, 181, 369, 661, 35, 258, 214, 209, 16, 662, 56, 663, 21, 53, 3, 664, 75, 121, 259,
214, 22, 4, 5, 6, 665], [2, 666, 1, 667, 668, 669, 670, 26, 671, 1, 10, 260, 53, 3, 95, 181, 87, 672, 18, 149, 673, 674, 2,
7, 675, 370, 676, 261, 123, 371, 677, 9, 123, 124, 678, 679, 174, 680, 262, 681, 4, 5, 6, 682], [73, 23, 1, 3, 33, 17, 52, 3
0, 25, 2, 7, 4, 5, 6, 683], [15, 60, 42, 32, 2, 7, 31, 1, 10, 3, 4, 5, 6, 684], [372, 685, 686, 687, 688, 689, 690, 691, 1, 1
0, 3, 120, 373, 692, 18, 149, 125, 2, 7, 9, 86, 374, 693, 694, 4, 5, 6, 695], [87, 351, 1, 3, 123, 696, 263, 2, 375, 17, 38,

```

(a)

```

array([[ 583,  584,  31, ...,  0,  0,  0],
       [ 595,  60,  42, ...,  0,  0,  0],
       [ 597, 343, 344, ...,  0,  0,  0],
       ...,
       [ 70, 1682, 1683, ...,  0,  0,  0],
       [  1,  22,  3, ...,  0,  0,  0],
       [1695,  57, 138, ...,  0,  0,  0]])

```

(b)

Gambar 4. Contoh Hasil Proses Sequences, Sebelum Proses Sequences (a) dan Setelah Proses Sequences (b)

Selanjutnya hasil dari proses *sequences* ini dibuat menjadi model latihan menggunakan *word embedding*. Lapisan *convolutional* pertama menggunakan 64 fitur filter, 8 unit *convolutional*, dan menggunakan aktivasi ReLU. Proses *pooling* menggunakan *max-pooling size* yaitu 2. Setelah proses *pooling* tersebut, ada lapisan *drop-out* diikuti lapisan *dense (full connected)* yang berguna untuk *overfitting*. Setelah itu menambahkan lagi *layer perceptron* yang di mana pada lapisan *dense* pertama tadi sebanyak 32 maka di lapisan *dense* kedua ini sebanyak 16. Selanjutnya menambahkan *GlobalMaxPooling1D* yang berguna untuk di mana tiap ada pergeseran filter dapat menentukan nilai maksimumnya [19]. Setelah itu didapatkan lapisan akhir yang terhubung penuh di mana ada *output* dari beberapa kelas yang diklasifikasikan. Fungsi aktivasi *softmax* digunakan untuk mengklasifikasikan kelas yang lebih dari dua, yang di mana pada penelitian ini terdapat tiga kelas.

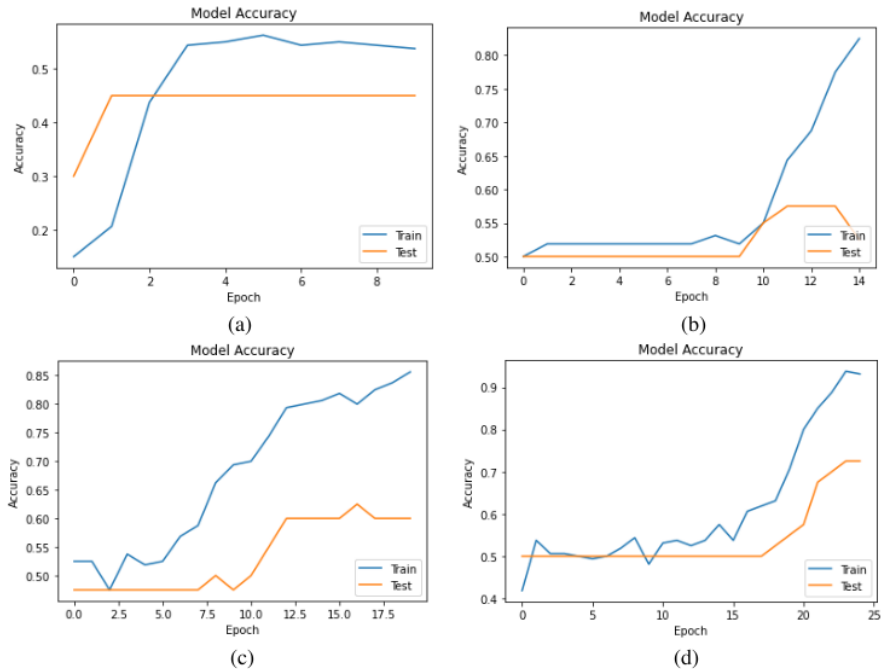
```

Model: "sequential"
-----
Layer (type)                Output Shape                Param #
-----
embedding (Embedding)        (None, 100, 300)           510900
conv1d (Conv1D)              (None, 93, 64)            153664
max_pooling1d (MaxPooling1D) (None, 46, 64)            0
dropout (Dropout)           (None, 46, 64)            0
dense (Dense)                (None, 46, 32)            2880
dropout_1 (Dropout)         (None, 46, 32)            0
dense_1 (Dense)              (None, 46, 16)            528
global_max_pooling1d (Global (None, 16)            0
dense_2 (Dense)              (None, 3)                  51
-----
Total params: 667,223
Trainable params: 667,223
Non-trainable params: 0

```

Gambar 5. Model Latih

Data yang telah dibuat model tersebut terbagi dari 3 kelas (netral, positif, dan negatif) dan akan digunakan sebagai *dataset* untuk pelatihan model *1D CNN* yang akan dibangun. Data tersebut dibagi menjadi *Training Data* dan *Test Data* dengan perbandingan 80:20 yaitu 160 *Training Data* dan 40 *Test Data*. Model *1D CNN* pada penelitian ini menggunakan 1 proses *convolution layer* dan 1 *pooling layer* di mana *convolution* tersebut menggunakan aktivasi ReLU. Dari proses model pelatihan didapatkan akurasi sebesar 53% dengan loss sebesar 0.8832 pada *epoch* 10, akurasi 82.50% dengan loss sebesar 0.5786 pada *epoch* 15, akurasi 85.62% dengan loss sebesar 0.3519 pada *epoch* 20, dan pada *epoch* 25 akurasi sebesar 98.12% dengan loss 0.0832 yang di mana semakin kecil nilai loss maka semakin baik hasilnya.



Gambar 6. Grafik Model Akurasi *Training* dan *Test* Tiap *Epoch*

Dapat dilihat pada tabel 4 merupakan hasil dari pengujian *Test Data*, akurasi terbaik sebanyak 72.50% dengan *loss* 0.8126 dan *epoch* 25. Berdasarkan tabel 4, seiring dengan penambahannya *epoch* maka akurasi mengalami peningkatan. Peningkatan akurasi berkisar 7-12% dari macam-macam *epoch* yang diberikan sebelumnya. Penggunaan *epoch* berbeda dalam proses training mendapatkan hasil kinerja klasifikasi yang berbeda juga. Penerapan pada *epoch* yang kecil mendapati hasil akurasi yang relatif rendah, dikarenakan pada proses adaptasi parameter belum sempurna yang disebabkan oleh waktu iterasi yang terbatas. Ketika jumlah epoch ditingkatkan maka parameter lebih optimal dan hasil yang didapatkan dari hasil klasifikasi bertambah secara bertahap [8].

Tabel 4. Komparasi Akurasi Konvolusi Pada *Epoch* yang Berbeda

	<i>Accuracy</i>
	<i>Loss</i>
<i>Epoch 10</i>	0.4500 (45%) 0.9922
<i>Epoch 15</i>	0.5250 (52.50%) 1.0367
<i>Epoch 20</i>	0.6000 (60%) 0.9181
<i>Epoch 25</i>	0.7250 (72.50%) 0.8126

Pengamatan selanjutnya yaitu uji klasifikasi pada kalimat baru yang tidak terdapat pada *dataset*, untuk melihat apakah label prediksi sesuai dengan label aslinya. Tabel 5 merupakan hasil dari pengamatan tersebut.

Tabel 5. Perbandingan Pada Label Prediksi Dan Label Asli

Tweet	Label Prediksi	Label Asli
Takut banget mau ikut nyoblos karena corona	Negatif	Negatif
Dukung seluruh proses pilkada, tetap jaga jarak dan terapkan 3M	Positif	Positif
Tdk ush dtg nyoblos masih rawan corona	Negatif	Negatif
Masyarakat mendukung proses pilkada 2020 ditengah pandemik covid	Positif	Positif
Masyarakat tdk setuju pilkada dilaksanakan	Negatif	Negatif
mri wjdkan plkada 2020	Negatif	Positif
baiknya kita tunda dulu pilkada tahun ini	Negatif	Negatif
Sungguh lebih baik dibatalkan pilkada tahun ini	Negatif	Negatif
pilkada 2020 akan dilangsungkan pada masa pandemik dan banyak masyarakat yg mendukung	Positif	Positif
walaupun masyarakat takut covid namun partisipasi pemilih pilkada 2020 byk	Positif	Positif

Pada Tabel 5 model diuji dengan kalimat baru, hasil label prediksi pada kalimat baru sesuai dengan label aslinya. Namun, terdapat satu kalimat baru berlabel negatif justru diklasifikasikan ke dalam sentimen positif. Kesalahan klasifikasi tersebut dapat disebabkan oleh kalimat yang mengandung makna ambigu atau makna tersirat, penggunaan bahasa informal, ejaan tidak konsisten, dan terdapat kemiripan kata dalam *Training Data* dan *Test Data*.

4. KESIMPULAN

Dari penelitian “*Sentiment Analysis Tweet Pilkada 2020 Saat Pandemi COVID-19 di Media Sosial Twitter Menggunakan Metode 1D Convolutional Neural Network*” memperoleh hasil akurasi terbanyak sebesar 72.50% dengan *epoch* yang digunakan sebesar 25 *epoch*. Keakuratan dari pengujian meningkat seiring bertambahnya jumlah *epoch* yang diterapkan. Dari *epoch* 10, 15, 20, dan 25 mengalami kenaikan akurasi sebesar 7-12% dari variasi *epoch* sebelumnya. Hasil uji klasifikasi kalimat baru berhasil memberikan label prediksi sesuai dengan label isinya. Namun terdapat satu kesalahan klasifikasi yang disebabkan oleh beberapa faktor, di antaranya terdapat dengan kata dengan makna tersirat atau makna ambigu, penggunaan Bahasa informal, ejaan yang tidak konsisten, serta terdapat kemiripan kata dalam *Training Data* dan *Test Data*. Penelitian ke depan disarankan untuk menambahkan dataset, pada proses

UCAPAN TERIMAKASIH

Alhamdulillah, puji dan syukur saya panjatkan kehadiran Allah SWT atas limpahan rahmat dan karunia-Nya sehingga penelitian ini dapat diselesaikan. Tak lupa shalawat dan salam saya haturkan kepada Nabi besar Muhammad SAW yang telah membawa umat Islam ke jalan yang diridhai Allah SWT.

Penelitian berjudul “*Sentiment Analysis Tweet Pilkada 2020 Saat Pandemi COVID-19 di Media Sosial Twitter Menggunakan Metode 1D Convolutional Neural Network*” ini merupakan salah satu syarat untuk mendapatkan gelar Sarjana Teknik Informatika. Penelitian ini tidak mungkin terselesaikan tanpa adanya dukungan, bimbingan, dan nasehat dari berbagai pihak. Oleh karena itu, izinkan penulis mengucapkan terima kasih kepada:

1. Edy Suryono selaku Ayah yang selalu menyemangati setiap harinya.
2. Suwarni Selaku Ibu yang selalu memberikan motivasi setiap harinya.
3. Miftahurrahma Rosyda sebagai dosen pembimbing penelitian serta skripsi yang sudah memberikan waktunya untuk memberikan bimbingan, saran, serta juga motivasi yang membangun kepada penulis hingga penelitian ini selesai dengan baik.

REFERENCES

- [1] M. Rudiyanto and R. Kasanova, “Dampak Covid-19 Terhadap Implementasi Pendidikan Karakter Impact of Covid-19 on The Implementation of Character Education,” *Jurnal Pendidik. dan Teknol. Indones.*, vol. 1, no. 5, pp. 209–213, 2021.
- [2] S. C. Rizki and Y. A. Hilman, “Menakar Perbedaan Opini Dalam Agenda Pelaksanaan Kontestasi Pilkada Serentak Di Tengah Covid-19,” *J. Ilm. Muqoddimah J. Ilmu Sos. Polit. dan Hummanioramania*, vol. 4, no. 2, p. 143, 2020.
- [3] 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.
- [4] D. W. Otter, J. R. Medina, and J. K. Kalita, “A Survey of the Usages of Deep Learning for Natural Language Processing,” *IEEE Trans. Neural Networks Learn. Syst.*, vol. 32, no. 2, pp. 604–624, 2021.
- [5] A. Rajput, “Natural language processing, sentiment analysis, and clinical analytics,” *Innov. Heal. Informatics A Smart Healthc. Prim.*, no. February, pp. 79–97, 2019.
- [6] H. Kim and Y. S. Jeong, “Sentiment classification using Convolutional Neural Networks,” *Appl. Sci.*, vol. 9, no. 11, pp. 1–14, 2019.
- [7] S. Sharma, “Epoch vs Batch Size vs Iterations,” 2017. [Online]. Available:

<https://towardsdatascience.com/epoch-vs-iterations-vs-batch-size-4dfb9c7ce9c9>. [Accessed: 16-Dec-2021].

- [8] D. Wang *et al.*, “Modulation Format Recognition and OSNR Estimation Using CNN-Based Deep Learning,” *IEEE Photonics Technol. Lett.*, vol. 29, no. 19, pp. 1667–1670, 2017.
- [9] M. Khader, A. Awajan, and G. Al-Naymat, “The impact of natural language preprocessing on big data sentiment analysis,” *Int. Arab J. Inf. Technol.*, vol. 16, no. 3ASpecial Issue, pp. 506–513, 2019.
- [10] A. Bayhaqy, S. Sfenrianto, K. Nainggolan, and E. R. Kaburuan, “Sentiment Analysis about E-Commerce from Tweets Using Decision Tree, K-Nearest Neighbor, and Naïve Bayes,” *2018 Int. Conf. Orange Technol. ICOT 2018*, no. October, 2018.
- [11] M. D. Hisbullah, “Belajar Machine Learning : Natural Language Processing,” 202AD. [Online]. Available: <https://www.anbidev.com/machine-learning-nlp/>. [Accessed: 16-Dec-2021].
- [12] M. Rhanoui, M. Mikram, S. Yousfi, and S. Barzali, “A CNN-BiLSTM Model for Document-Level Sentiment Analysis,” *Mach. Learn. Knowl. Extr.*, vol. 1, no. 3, pp. 832–847, 2019.
- [13] L. xia Luo, “Network text sentiment analysis method combining LDA text representation and GRU-CNN,” *Pers. Ubiquitous Comput.*, vol. 23, no. 3–4, pp. 405–412, 2019.
- [14] A. Pambudi and S. Suprpto, “Effect of Sentence Length in Sentiment Analysis Using Support Vector Machine and Convolutional Neural Network Method,” *IJCCS (Indonesian J. Comput. Cybern. Syst.)*, vol. 15, no. 1, p. 21, 2021.
- [15] S. N. Listyarini and D. A. Anggoro, “Analisis Sentimen Pilkada di Tengah Pandemi Covid-19 Menggunakan Convolution Neural Network (CNN),” *J. Pendidik. dan Teknol. Indones.*, vol. 1, no. 7, pp. 261–268, 2021.
- [16] J. Chen, S. Yan, and K. C. Wong, “Verbal aggression detection on Twitter comments: convolutional neural network for short-text sentiment analysis,” *Neural Comput. Appl.*, vol. 32, no. 15, pp. 10809–10818, 2020.
- [17] P. Nyoman and Putu Kusuma Negara, “Deteksi Masker Pencegahan Covid19 Menggunakan Convolutional Neural Network Berbasis Android,” *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 5, no. 3, pp. 576–583, 2021.
- [18] D. A. N. Taradhita and I. K. G. D. Putra, “Hate speech classification in Indonesian language tweets by using convolutional neural network,” *J. ICT Res. Appl.*, vol. 14, no. 3, pp. 225–239, 2021.
- [19] M. Ahmad, A. Farisi, and W. Astuti, “Klasifikasi Multi-label pada Hadis Sahih Bukhari Terjemahan Bahasa Indonesia Menggunakan Convolutional Neural Networks,” vol. 8, no. 5, pp. 10594–10604, 2021.

ORIGINALITY REPORT

13%

SIMILARITY INDEX

13%

INTERNET SOURCES

2%

PUBLICATIONS

3%

STUDENT PAPERS

PRIMARY SOURCES

1	eprints.ums.ac.id Internet Source	3%
2	adoc.pub Internet Source	1%
3	lokadata.id Internet Source	1%
4	repository.telkomuniversity.ac.id Internet Source	1%
5	Submitted to Trisakti University Student Paper	1%
6	docobook.com Internet Source	1%
7	reza_chan.staff.gunadarma.ac.id Internet Source	1%
8	belajarpython.id Internet Source	<1%
9	digilib.uinsby.ac.id Internet Source	<1%

10	repository.usu.ac.id Internet Source	<1 %
11	forum.detik.com Internet Source	<1 %
12	spcommreports.ohchr.org Internet Source	<1 %
13	www.scribd.com Internet Source	<1 %
14	doku.pub Internet Source	<1 %
15	e-journal.uajy.ac.id Internet Source	<1 %
16	eprints.umm.ac.id Internet Source	<1 %
17	baper.if.uinsgd.ac.id Internet Source	<1 %
18	id.berita.yahoo.com Internet Source	<1 %
19	journal.itelkom-sby.ac.id Internet Source	<1 %
20	repository.ipb.ac.id Internet Source	<1 %

Exclude quotes Off

Exclude matches Off

Exclude bibliography On