



## Analisis Perbandingan Algoritma LSTM dan Naive Bayes untuk Analisis Sentimen

Auliya Rahman Isnain<sup>#1</sup>, Heni Sulistiani<sup>#2</sup>, Bagus Miftaq Hurohman<sup>#3</sup>, Andi Nurkholis<sup>#4</sup>, Styawati<sup>#5</sup>

<sup>#</sup>Faculty of Engineering and Computer Science, Universitas Teknokrat Indonesia  
Jl. ZA. Pagar Alam No. 9-11, Labuhan Ratu, Bandar Lampung

<sup>1</sup>auliyarahman@teknokrat.ac.id

<sup>2</sup>henisulistiani@teknokrat.ac.id

<sup>3</sup>bagus\_miftaq\_hurohman@teknokrat.ac.id

<sup>4</sup>andinh@teknokrat.ac.id

<sup>5</sup>styawati@teknokrat.ac.id

**Abstrak**— *New Normal* merupakan sebuah sebutan bagi kebijakan pemerintah untuk mengizinkan masyarakatnya melakukan aktifitas seperti biasa di tengah *pandemi Covid-19* yang sedang melanda dengan tetap memperhatikan protokol kesehatan. Kebijakan ini menimbulkan berbagai tanggapan dari masyarakat terutama di media sosial *twitter*. Untuk itu, diperlukan proses analisis sentimen untuk melakukan pemrosesan terhadap teks yang didapat dari *twitter*. Analisis sentimen adalah bentuk representasi dari *text mining* dan *text processing*. Pada penelitian ini melakukan perbandingan kinerja metode *Long Short Term Memory* dengan *Naive Bayes* terhadap analisis sentimen Kebijakan *New Normal*. Hasil yang diperoleh dari penelitian ini yaitu metode LSTM memiliki kinerja yang lebih baik bila dibandingkan dengan *Naive Bayes*. Metode LSTM menghasilkan nilai akurasi, presisi dan *recall* sebesar 83.33%. Sedangkan metode *Naive Bayes* memiliki nilai akurasi, presisi dan *recall* sebesar 82%.

**Kata kunci**— Analisis sentimen, LSTM, *Naive Bayes*, *New Normal*

### I. PENDAHULUAN

Kampanye pemerintah terhadap *new normal* menimbulkan berbagai tanggapan dari masyarakat. Sebagian ada yang menanggapinya secara positif dan sebagian lagi menanggapinya negatif. Hal ini dapat dilihat dari berbagai postingan disosial media yang ada, karena masyarakat dapat secara bebas memberikan pendapat maupun aspirasinya melalui media sosial yang mereka miliki [1]. Salah satu sosial media yang populer untuk menyalurkan pendapat bebas maupun aspirasi adalah *Twitter* [2]. Pada media sosial *twitter*, Pendapat masyarakat dapat digunakan sebagai sarana untuk menyimpulkan keadaan yang sedang terjadi di suatu wilayah [3]. Dalam hal ini, wilayah yang ditinjau adalah wilayah yang sudah mulai melakukan kegiatan *new normal*. Banyaknya unggahan di *twitter* seputar *new normal* menjadi kumpulan data yang sangat membantu untuk dijadikan sebagai sarana pendukung dalam pembuatan

kebijakan. Untuk itu, diperlukan suatu analisis sentimen terhadap kumpulan komentar dan unggahan tersebut guna mengetahui perasaan yang diberikan terhadap topik atau objek di media sosial [4]. Metode yang digunakan dalam analisis sentimen sangat beragam diantaranya, yaitu *Support Vector Machine* [5][6][7], *Naive Bayes* [4][8][9], *Long Short Term Memory* [10][11][12]. Diantara metode-metode tersebut, penelitian ini berfokus pada metode *Long Short Term Memory* (LSTM). LSTM memiliki kelebihan diantara metode yang lainnya, dimana metode LSTM dirancang untuk mampu menyimpan informasi jangka panjang, membaca, serta memperbarui informasi sebelumnya [12]. Dari hasil *literature review* yang dilakukan, belum ada penelitian yang membahas mengenai perbandingan metode LSTM dan *Naive Bayes* untuk analisis sentimen terhadap kebijakan *new normal*.

Hasil dari pencarian referensi terdapat penelitian yang membahas mengenai perbandingan metode *Naive Bayes* dan SVM untuk klasifikasi penyakit diabetes melitus [13]. Hasilnya metode *Naive Bayes* memiliki kinerja yang lebih rendah dibandingkan metode SVM. Namun, jika metode *Naive Bayes* dibandingkan dengan metode K-Nearest Neighbor untuk klasifikasi artikel berbahasa Indonesia [14] menghasilkan bahwa metode *Naive Bayes* memiliki kinerja yang lebih baik. Sedangkan dalam penelitian yang membahas tentang perbandingan algoritma LSTM dan *Naive Bayes* dalam sentimen analisis terhadap brand *indomie* [15] menunjukkan bahwa metode LSTM memiliki kinerja yang lebih baik dibandingkan *Naive Bayes*.

Setelah melakukan kajian pustaka dan literatur review, diperoleh informasi bahwa belum ada penelitian yang membahas mengenai perbandingan metode LSTM dengan *Naive Bayes* untuk analisis sentimen terhadap kebijakan *new normal*. Maka dari itu, dalam penelitian ini akan membahas tentang perbandingan kedua metode tersebut. Tujuan dari penelitian ini yaitu agar dapat diketahui metode mana yang memiliki kinerja paling baik dengan nilai akurasi tertinggi dalam menganalisis kebijakan *new normal*.

II. TINJAUAN PUSTAKA

Analisis sentimen dapat diartikan sebagai sebuah proses otomatisasi penggalian sikap, opini, dan emosi yang terkandung didalam suatu data teks [16]. Analisis sentimen merupakan proses untuk mengolah suatu data teks yang akan dikategorikan ke dalam sentimen positif dan sentimen negatif.

Berdasarkan proses pendekatannya analisis sentimen terbagi menjadi dua, yaitu : 1) Pendekatan Machine Learning yang merupakan pendekatan dimana analisis sentimen dilakukan dengan cara memproses data yang sebelumnya sudah dibagi menjadi data positif dan data negatif. 2) Pendekatan *Lexicon-based* adalah merupakan sebuah pendekatan dimana dalam melakukan analisis sentimen dilakukan dengan cara memberikan nilai sentimen dari suatu kata berdasarkan kamus *lexicon*. Sedangkan berdasarkan level sumber datanya, maka analisis sentimen dapat dibagi menjadi tiga [17], yaitu : 1) Level Dokumen yaitu Secara garis besar fokus utamanya adalah menganggap seluruh isi dokumen sebagai sentimen positif atau negative dimana pada level ini mengasumsikan bahwa seluruh opini dalam dokumen berisi tentang satu entitas saja. 2) Level Kalimat adalah jenis ini menentukan sentimen pada suatu kalimat bernilai sentimen positif, negatif, atau netral. Sentimen netral berarti kalimat tersebut bukanlah opini. Yang ke 3) Level Entitas dan Aspek. Level ini tidak mampu melakukan analisis pada konstruksi bahasa melainkan pada opini itu sendiri.

A. Long-Short Term Memory (LSTM)

Dibuat oleh Sepp Hochreiter dan Jurgen Schmidhuber pada tahun 1997. Metode LSTM merupakan variasi dari *Recurrent Neural Network* (RNN) dengan kelebihan, diantaranya mampu menyimpan informasi jangka panjang, membaca, serta memperbarui informasi sebelumnya dan juga mampu menangani masalah *vanishing gradient* dalam pelatihan yang biasanya terjadi pada metode variasi RNN yang lainnya [18]. LSTM secara umum terdiri dari *cell*, *input gate*, *output gate* dan *forget gate* [19].

Pada penelitian sebelumnya [12], LSTM digunakan untuk melakukan analisis sentiment terhadap ulasan novel dan memiliki hasil yang lebih baik dibandingkan *naïve bayes*. Keunggulan penting LSTM adalah kemampuan mengingat dari sekuens jangka panjang (ukuran data) yang sulit dicapai dengan metode dengan fitur yang masih tradisional [20]. Kemampuannya mengingat didukung oleh *gating system* yang dimiliki LSTM. Yang mana didalam LSTM terdapat *cell state* yang berfungsi sebagai memori penyimpanan yang dinalainya dapat dimanipulasi menggunakan *gating system* tersebut [19].

B. Word2vec

Pada proses sentimen analisis, diperlukan fitur untuk merepresentasikan kata kedalam vektor angka. Pertama kali dipublikasikan oleh Google pada tahun 2013, word2vec merupakan dua lapisan jaringan syaraf untuk memproses data teks [21]. Untuk masukannya sendiri (input) disebut corpus dan output yang dihasilkan adalah

satu set vektor. Word2vec memiliki metode, yaitu : *Continuous Bag of Word* (CBOW) bertujuan untuk memprediksi kata yang diberikan oleh kata-kata disekitarnya [22]. Metode ini menggunakan current word sebagai target dan konteks sebagai input dan *Skip-gram*.

Bertujuan memprediksi jendela kata yang diberikan oleh kata tunggal. Berbanding terbalik dengan CBOW, Skip-gram menggunakan current word sebagai input untuk memprediksi konteks sebagai target.

III. METODE PENELITIAN

A. Data

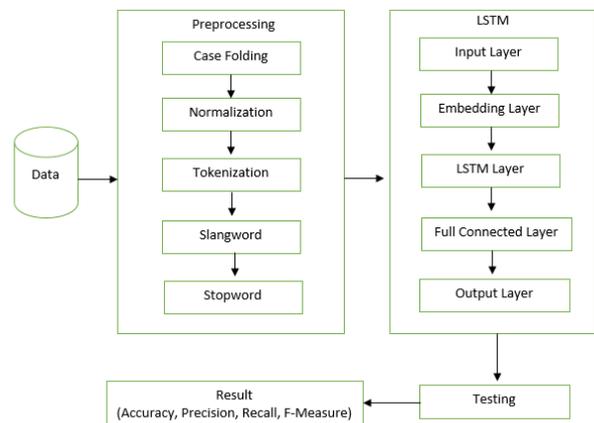
Dalam penelitian ini, data yang didapatkan bersumber dari *crawling* data dari *Twitter*. Dataset diambil dengan melakukan *crawling* data pada twitter sejak tanggal 6 Juli 2020 sampai 25 Juli 2020. Data tersebut diambil dengan menggunakan kata kunci “#NewNormal”. Data yang telah berhasil diambil akan disimpan dengan format *.csv*. Hasil pengambilan data yang didapat dapat dilihat pada Tabel 1: Tabel 1 Contoh hasil *crawling* data di twitter.

TABEL I  
HASIL CRAWLING DATA DI TWITTER

Tanggapan
1291524338768805891,1291524338768805891,2020-08-07 06:59:32 SE Asia Standard Time,2020-08-07,06:59:32,+0700,492958388,radiogloriafm,Gloria Paramitha FM,,"Hidup dalam kenormalan baru bukan berarti sudah terbebas dari Covid-19, tapi jaga protokol kesehatan. Pakai masker dari luar dan dari dalam tubuh cegah segala virus denganâ€¦

Data berupa teks yang didapat masih berupa teks tidak terstruktur. Oleh karena itu, dilakukan tahap *preprocessing* yang akan membuat data teks menjadi lebih bersih dan terstruktur. Hasil yang diharapkan setelah melewati tahap *processing* adalah teks yang terstruktur sehingga polaritas dari teks dapat ditentukan. Seperti yang ditunjukkan pada Tabel 1.

Tahapan penelitian yang dilakukan dapat dilihat pada Gambar 1 berikut:



Gambar. 1 Tahapan penelitian

B. Preprocessing

Tahap ini merupakan proses penyederhanaan tanggapan agar lebih mudah saat melakukan proses selanjutnya. Tahap *preprocessing* ini dibagi menjadi 5 tahap, yaitu : 1) *Case folding* dalam dokumen teks yang didapat, tidak semuanya menggunakan penggunaan huruf yang konsisten. Oleh karena itu *case folding* diperlukan untuk mengubah karakter huruf pada tanggapan menjadi karakter huruf kecil seluruhnya. (*lowercase*), 2) Normalisasi fitur tahap ini merupakan tahapan dimana karakter khusus seperti tanda baca titik (.), koma (,), tanda tanya (?), tanda seru (!), tanda petik (‘), dan sebagainya), angka numerik (0-9), dan simbol lainnya (\*,#,%, dan sebagainya) dihilangkan, 3) Tokenisasi, tahapan ini bertujuan untuk memecah tanggapan menjadi satuan kata. Proses ini dilakukan dengan melihat spasi antar kalimat, berdasarkan spasi tersebut setiap kata akan dipecah [4], 4) Konversi *slangword* Pada tahap ini, kata yang termasuk *slangword* atau kata tidak baku (bahasa gaul) diubah ke dalam bentuk baku sesuai KBBI. Yang termasuk dalam kategori *slangword* adalah kata tidak baku, singkatan-singkatan, atau salah penulisannya [13] dan yang terakhir, 5) Penghapusan *stopword* menghapus kata-kata yang tidak deskriptif seperti yang, ke, dari, atau, dan sebagainya.

C. Pengujian

Pada penelitian ini, pengujian dilakukan dengan metode *confusion matrix*. Parameter yang diukur adalah akurasi, presisi, *recall*, dan *f-measure*. Akurasi adalah nilai kemiripan antara nilai prediksi dan nilai sebenarnya. Presisi adalah tingkat kebenaran antara informasi yang diminta pengguna dan yang diberikan oleh sistem. *Recall* adalah tingkat keberhasilan dari sistem dalam menemukan kembali suatu informasi. *F-measure* merupakan bentuk parameter gabungan antara *recall* dan presisi untuk keberhasilan *retrival*.

IV. HASIL DAN PEMBAHASAN

Data yang digunakan dalam tahapan pemrosesan sebanyak 1.823 data. Kemudian, data tersebut diberi label secara manual dengan nilai 1 sebagai sentimen positif dan nilai 0 sebagai sentimen negative. Dari proses pelabelan tersebut didapatlah data kelas positif sebanyak 1.028 dan data kelas negatif sebanyak 795 data. Keseimbangan antara kelas positif dan negatif sangat berpengaruh terhadap akurasi yang didapatkan [23]. Oleh karena itu jumlah dari kelas positif dikurangi menjadi 795 data agar seimbang terhadap kelas negatif. Total dataset yang digunakan pada akhirnya berjumlah 1.590 data. Dataset tersebut akan dibagi menjadi data training dan data testing dengan pembagian data training sebesar 80% dan data testing sebesar 20%. Pembagian yang telah dilakukan dapat dilihat pada Tabel 2.

TABEL III  
PEMBAGIAN DATA ACAK

Data	jumlah
Training	12772
Testing	318

A. Training WordVec Indonesia

Model yang dihasilkan dari proses Word2Vec akan digunakan untuk mengubah kamus kata dari dataset menjadi bentuk vektor. bagian `sentences = word2vec.LineSentence` adalah bagian dimana file `word2vec` yang telah diekstrak ke dalam format txt dipanggil sebagai input. Bagian `id_w2v = word2vec.Word2Vec` berfungsi sebagai fungsi untuk melakukan proses pembuatan model dimana parameter `sg=1` menunjukkan arsitektur pembuatannya menggunakan skipgram.

B. Konveksi Vektor

Di dalam LSTM, terdapat sebuah layer yang bernama *embedding layer*. Layer ini berfungsi untuk mengubah inputan menjadi vektor. Didalam layer inilah `word2vec` digunakan. Dibagian `load_word2vec_format`, model `word2vec` yang telah dibuat sebelumnya dipanggil. Bagian `embedding_matrix=word2vec_model.wv.syn0` menunjukkan kode untuk menetapkan *embedding matrix*. Pada baris kode

```
top_words=embedding_matrix.shape[0],
X_train dan X_test akan diubah ke bentuk vektor dua dimensi. Setiap bagian dari proses ini akan diterapkan di embedding layer LSTM.
```

C. Pengujian LSTM Berdasarkan Batch Size

Pengujian *batch size* dilakukan dengan mengubah besar nilai *batch size* dalam rentang tertentu untuk mendapatkan nilai optimal. *Batch size* sendiri adalah pembagian jumlah sampel data yang disebarkan pada sebuah *neural network* [24]. Pada pengujian ini, ditetapkan *neuron* sebanyak 100, *epoch* sebanyak 5 dan nilai *batch size* yang akan diujikan, yaitu 32, 64, dan 128. Dari pengujian tersebut didapatlah hasil yang ditunjukkan pada Tabel 3.

TABEL IIIII  
HASIL PENGUJIAN BATCH SIZE

BS	AK (%)	RE (%)	PR (%)	FM (%)	T (detik)
32	82.39	82.38	82.38	82.40	2112
64	82.39	83.38	82.43	82.39	1045
128	83.33	83.33	83.33	83.31	560

Keterangan :  
 BS : *Batch size*  
 AK : Akurasi  
 RE : *Recall*  
 PR : Presisi  
 FM : *F-measure*  
 T : *time/waktu*

Dapat dilihat dari Tabel 3 bahwa hasil akurasi tertinggi didapat oleh *batch size* 128 dengan akurasi sebesar 83.33%, *recall* 83.33%, *presisi* 83.33%, dan *f-measure* 83.33%. Berdasarkan waktu yang diperlukan, semakin besar *batch size* semakin sedikit waktu yang diperlukan untuk melakukan pemrosesan.

D. Pengujian LSTM Berdasarkan Nilai Neuron

Pengujian *neuron* dilakukan dengan mengubah besar nilai *neuron* dalam rentang tertentu untuk mendapatkan nilai optimal. Pada pengujian ini, ditetapkan *batch size* sebesar 128 dan *epoch* sebanyak 5 karena berdasarkan pengujian sebelumnya mendapatkan nilai akurasi tertinggi pada *neuron* 100. Pada pengujian ini, nilai *neuron* akan ditetapkan sebanyak 75, 100, dan 125 dengan nilai 100 sebagai nilai tengah dan acuan dari pengujian sebelumnya. Dari pengujian tersebut didapatlah hasil yang ditunjukkan pada Tabel 4. Dari Tabel ini ditunjukkan bahwa N merupakan parameter Neuron yang diuji, sedangkan AK adalah Akurasi, RE merupakan Recall, PR dan FM adalah Presisi dan F-Measure dan T adalah waktu dalam detik untuk mengasilkan nilai akurasi.

TABEL IV  
HASIL PENGUJIAN NEURON

N	AK (%)	RE (%)	PR (%)	FM (%)	T (detik)
75	81.76	81.76	81.84	81.77	457
100	83.33	83.33	83.33	83.31	560
125	81.45	81.44	81.77	81.45	563

Dapat dilihat dari Tabel 4 diatas bahwa hasil akurasi tertinggi didapat oleh *neuron* 100 dengan akurasi sebesar 83.33%, *recall* 83.33%, *presisi* 83.33%, dan *f-measure* 83.33%. Berdasarkan waktu yang diperlukan, semakin besar *neuron* semakin lama waktu yang diperlukan untuk melakukan pemrosesan.

E. Pengujian LSTM Berdasarkan Nilai Epoch

Pengujian *epoch* dilakukan dengan mengubah besar nilai *epoch* dalam rentang tertentu untuk mendapatkan nilai optimal. Pada pengujian ini, ditetapkan *neuron* sebanyak 100 dan nilai *epoch* akan dimulai dari 5 lalu ditingkatkan sampai 25 dengan kelipatan 5. Dari pengujian tersebut didapatlah hasil yang ditunjukkan pada Tabel 5.

TABEL V  
HASIL PENGUJIAN EPOCH

E	AK (%)	RE (%)	PR (%)	FM (%)	T (detik)
5	83.33	83.33	83.33	83.31	560
10	81.76	81.76	81.75	81.75	1018
15	80.19	80.18	80.18	80.17	1386
20	79.87	79.87	79.86	79.86	2107
25	80.50	80.50	80.50	80.50	2560

Keterangan :  
E : Epoch  
AK: Akurasi  
RE : Recall

PR : Presisi  
FM: F-measure  
T : time/waktu

Dapat dilihat dari Tabel 5, bahwa hasil akurasi tertinggi didapat oleh *epoch* 5 dengan akurasi sebesar 83.33%, *recall* 83.33%, *presisi* 83.33%, dan *f-measure* 83.33%. Berdasarkan waktu yang diperlukan, semakin besar *epoch* semakin lama waktu yang diperlukan untuk melakukan pemrosesan.

F. Pengujian Algoritma Naïve Bayes

Pengujian Naïve Bayes dilakukan untuk memanfaatkan *library* sklearn pada python. Data yang digunakan pada klasifikasi ini dibagi menjadi data *training* sebanyak 80% dan data *testing* sebanyak 20%. Skenario pembagian data tersebut mampu menghasilkan nilai akurasi yang tinggi pada metode Naive Bayes [25]. Selanjutnya, metode yang digunakan dalam ekstraksi fitur adalah TF-IDF. Pada pengujian ini didapatlah hasil yang ditunjukkan pada Tabel 6.

TABEL VI  
HASIL PENGUJIAN NAIVE BAYES

AK (%)	RE (%)	PR (%)	FM (%)	T (detik)
82	82	82	82	11

Keterangan :  
AK : Akurasi  
RE : Recall  
PR : Presisi  
FM : F-measure  
T : time/waktu

Dapat dilihat pada Tabel 6, bahwa akurasi yang didapat sebesar 82%, *recall* 82%, *presisi* 82%, dan *f-measure* 82%.

G. Perbandingan Hasil Sentimen

Pada Tabel 7, ditunjukkan hasil dari pengujian tiap metode. Berdasarkan nilai akurasinya, LSTM memiliki nilai akurasi yang lebih baik, yaitu 83.33%. Sedangkan Naïve Bayes memiliki nilai akurasi sebesar 82%. Hal ini menunjukkan bahwa LSTM menghasilkan nilai akurasi yang lebih baik dibandingkan Naïve Bayes.

TABEL VII  
PERBANDINGAN HASIL SENTIMEN

M	AK (%)	RE (%)	PR (%)	FM (%)	T (detik)
LSTM	83.33	83.33	83.33	83.31	560
Naïve	82	82	82	82	11

Keterangan :  
M : Metode  
AK : Akurasi  
RE : Recall  
PR : Presisi  
FM : F-measure  
T : time/waktu

## V. KESIMPULAN

Berdasarkan penelitian yang telah dilakukan, diperoleh beberapa kesimpulan. Metode *Long-Short Term Memory* menghasilkan nilai akurasi yang lebih baik dibanding metode naïve bayes dengan nilai akurasi, presisi dan *recall* sebesar 83.33%. Sedangkan metode Naïve Bayes hanya memiliki nilai akurasi, presisi dan *recall* sebesar 82%. Metode LSTM memiliki kinerja yang lebih baik sebesar 1.33% bila dibandingkan dengan metode LSTM. Untuk penelitian selanjutnya, dapat menerapkan algoritma peningkatan parameter agar memiliki nilai akurasi lebih dari 83%.

## REFERENSI

- [1] L. Septiani and Y. Sibaroni, "Sentiment Analysis Terhadap Tweet Bernada Sarkasme Berbahasa Indonesia," *J. Linguist. Komputasional*, vol. 2, no. 2, pp. 62–67, 2019, doi: 10.26418/jlk.v2i2.23.
- [2] Y. Rafita, "Social Network Analysis Dalam Melihat Kecenderungan Pemberitaan Pada Akun Twitter '@detikcom' dan '@Metro\_TV,'" *Khazanah*, vol. 6, no. 2, pp. 67–81, 2014, doi: 10.20885/khazanah.vol6.iss2.art7.
- [3] M. Syarifuddin, "Analisis Sentimen Opini Publik Terhadap Efek Psbb Pada Twitter Dengan Algoritma Decision Tree-Knn-Naïve Bayes," *INTI Nusa Mandiri*, vol. 15, no. 1, pp. 87–94, 2020, doi: 10.33480/inti.v15i1.1433.
- [4] B. Gunawan, H.S. Pratiwi and E.S. Pratama, "Sistem Analisis Sentimen pada Ulasan Produk Menggunakan Metode Naïve Bayes", *Jurnal Edukasi dan Penelitian Informatika*, Vol. 4 No. 2, Desember 2018.
- [5] U. Rofiqoh, R.S. Perdana, and M.A. Fauzi, "Analisis Sentimen Tingkat Kepuasan Pengguna Penyedia Layanan Telekomunikasi Seluler Indonesia Pada Twitter Dengan Metode Support Vector Machine dan Lexicon Based Features", *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, Vol. 1, No. 12, Desember 2017, hlm. 1725-1732.
- [6] N. Fitriyah, B. Warsito, and D.A.I. Maruddani, "Analisis Sentimen Gojek Pada Media Sosial Twitter Dengan Klasifikasi Support Vector Machine (SVM)", *JURNAL GAUSSIAN*, Volume 9, Nomor 3, Tahun 2020, Halaman 376 -390.
- [7] I. Taufik, and S.A. Pamungkas, "Analisis Sentimen Terhadap Tokoh Publik Menggunakan Algoritma Support Vector Machine (SVM)", *Jurnal "LOG!K@"*, Jilid 8, No. 1, 2018, Hal. 69 - 79.
- [8] D.G. Nugroho, Y.H. Chrisnanto, and A. Wahana, "Analisis Sentimen Pada Jasa Ojek Online Menggunakan Metode Naïve Bayes", *Prosiding SNST ke-7 Tahun 2016*, Fakultas Teknik Universitas Wahid Hasyim Semarang.
- [9] F. Ratnawati, "Implementasi Algoritma Naïve Bayes Terhadap Analisis Sentimen Opini Film Pada Twitter", *Jurnal Inovtek Polbeng - Seri Informatika*, Vol. 3, No. 1, Juni 2018.
- [10] J. Nurvania, Jondri, and K.M. Lhaksamana, "Analisis Sentimen Pada Ulasan di TripAdvisor Menggunakan Metode Long Short-Term Memory (LSTM)", *e-Proceeding of Engineering : Vol.8, No.4 Agustus 2021*.
- [11] C.K. Poetra, S.F. Pane, and Rd. N.S. Fatonah, "Meningkatkan Akurasi Long-Short Term Memory (LSTM) Pada Analisis Sentimen Vaksin Covid-19 di Twitter dengan Glove", *Jurnal Telamika*, Vol. 16 No. 2, Institut Teknologi Harapan Bangsa Bandung, 2021.
- [12] M. A. Nurrohmat and A. SN, "Sentiment Analysis of Novel Review Using Long Short-Term Memory Method," *IJCCS (Indonesian J. Comput. Cybern. Syst.*, vol. 13, no. 3, p. 209, 2019, doi: 10.22146/ijccs.41236.
- [13] H. Apriyani and Kurniati, "Perbandingan Metode Naïve Bayes dan Support Vector Machine dalam Klasifikasi Penyakit Diabetes Melitu", *Jour. of Inf. Tech. Ampera*, Vol. 1, No. , December 2020.
- [14] R.N. Devita, H.W. Herwanto and A.P. Wibawa, "Perbandingan Kinerja Metode Naïve Bayes dan K-Nearest Neighbor untuk Klasifikasi Artikel Berbahasa Indonesia", *J. Pengemb. Teknol. Inf. dan Ilmu Komput. Univ. Brawijaya*, Vol. 5, No. 4, September 2018.
- [15] Elisabeth, " Perbandingan Sentimen Analisis Terhadap Brand Indomie Menggunakan Naive Bayes dan Long Short Term Memory (LSTM)", *Program Studi Sistem Informasi, Fakultas Teknik dan Informatika, Universitas Multimedia Nusantara*, 2018.
- [16] S.R.M. Pakpahan, I. Indriati and M. Marji, "Analisis Sentimen Tentang Opini Performa Klub Sepak Bola pada Dokumen Twitter Menggunakan Support Vector Machine dengan Perbaikan Kata Tidak Baku", *J. Pengemb. Teknol. Inf. dan Ilmu Komput. Univ. Brawijaya*, Vol. No. 7, 2019.
- [17] B. Liu, "Sentiment Analysis : A Multi-Faceted Problem," no. 1, 2010.
- [18] R. Feldman and J. Sanger, "The Text Mining Handbook". Cambridge, 2007.
- [19] A. Khumaidi, R. Raafi'udin and I.P. Solihin, "Pengujian Algoritma Long Short Term Memory untuk Prediksi Kualitas Udara dan Suhu Kota Bandung", *Jurn. Telematika*, vol. 15, No. 1, Institut Teknologi Harapan Bangsa, 2020.
- [20] L. Zaman, S. Sumpeno, and M. Hariadi, "Analisis Kinerja LSTM dan GRU sebagai Model Generatif untuk Tari Remo," *J. Nas. Tek. Elektro dan Teknol. Inf.*, vol. 8, no. 2, p. 142, 2019, doi: 10.22146/jnteti.v8i2.503.
- [21] M. Rizki, S. Basuki, and Y. Azhar, "Implementasi Deep Learning menggunakan Arsitektur Long Short Term Memory (LSTM) untuk Prediksi Curah Hujan Kota Malang", Vol. 2, No. 3 Maret 2020," *Jurnal Repositor*, vol. 2, no. 3, 2020.
- [22] D. Qiu, H. Jiang and S. Chen, "Fuzzy Information Retrieval Based on Continuous Bag-of-Words Model", *Symmetry*, 12, 225, 2020.
- [23] A. R. Isnain, A. Sihabuddin, and Y. Suyanto, "Bidirectional Long Short Term Memory Method and Word2vec Extraction Approach for Hate Speech Detection," *IJCCS (Indonesian J. Comput. Cybern. Syst.*, vol. 14, no. 2, p. 169, 2020, doi: 10.22146/ijccs.51743.
- [24] M. H. Azhar, P. P. Adikara, and Y. A. Sari, "Analisis Sentimen pada Ulasan Hotel dengan Fitur Score Representation dan Identifikasi Aspek pada Ulasan Menggunakan K-Modes," *J. Pengemb. Teknol. Inf. dan Ilmu Komput. Univ. Brawijaya*, vol. 2, no. 9, pp. 2777–2782, 2018.
- [25] M. Akbar, "Perbandingan Algoritme Naïve Bayes Classifier Dan Knearest Neighbors Pada Prediksi Pergerakan Mata Uang Dollar Amerika (USD) Terhadap Harga Emas", *Jurusan Teknik Elektro, Fakultas Teknologi Industri, Universitas Islam Indonesia*, 2020.