

# Journal of Dinda

Kelompok Keahlian Rekayasa Data  
Institut Teknologi Telkom Purwokerto

Vol. 2 No. 1 (2022) 46 - 51

ISSN Media Elektronik: 2809-8064

---

## Deteksi Sarkasme Pada Judul Berita Berbahasa Inggris Menggunakan Algoritme Bidirectional LSTM

**Muhammad David Hilmawan**

Teknik Informatika, Fakultas Informatika, Institut Teknologi Telkom Purwokerto  
18102023@ittelkom-pwt.ac.id

### Abstract

Sarcasm is the use of harsh words to hurt another person, in the form of ridicule or harsh ridicule. The word sarcasm is derived from the Greek word *sarkasmos* which means “to tear flesh like a dog”, “biting lips in anger”, or “speaking bitterly”. Sarcasm may or may not be ironic, but what is certain is that this style of language is always hurtful and unpleasant to hear. In this study, a classification model will be made to predict sarcasm in English news headlines because news titles use standard words and there are no spelling mistakes, making news headlines an appropriate dataset for sarcasm detection. Bidirectional Long Short-Term Memory (BiLSTM) algorithm, which is one of the deep learning algorithms used in research to create a classification model. This model is then compared with the Long Short-Term Memory (LSTM) algorithm model to validate the advantages of the BiLSTM algorithm over the basic LSTM algorithm. The model achieved 82.55% on validation accuracy, 82.36% on validation recall, 79.53% on validation precision, and 80.92% validation f1 score.

Keywords: sarcasm, BiLSTM, LSTM, deep learning, f1-score

### Abstrak

Sarkasme adalah penggunaan kata-kata pedas untuk menyakiti hati orang lain, berupa cemoohan atau ejekan kasar. Kata sarkasme diturunkan dari kata Yunani *sarkasmos* yang berarti “merobek-robek daging seperti anjing”, “menggigit bibir karena marah”, atau “berbicara dengan kepahitan”. Sarkasme dapat bersifat ironis, atau tidak, tetapi yang pasti adalah bahwa gaya bahasa ini selalu akan menyakiti hati dan kurang enak didengar. Pada penelitian ini akan dibuat model klasifikasi untuk memprediksi sarkasme pada judul berita berbahasa inggris dikarenakan judul berita menggunakan kata baku dan tidak ada salah pengejaan kata, menjadikan judul berita sebuah dataset yang tepat untuk dilakukan deteksi sarkasme. Algoritme *Bidirectional Long Short-Term Memory* (BiLSTM) yang merupakan salah satu algoritme *deep learning* digunakan pada penelitian untuk membuat model klasifikasi. Model ini lalu dibandingkan dengan model algoritme *Long Short-Term Memory* (LSTM) untuk memvalidasi keunggulan dari algoritme BiLSTM daripada algoritme LSTM dasar. Didapatkan akurasi validasi dari model sebesar 82,55%, *precision* validasi sebesar 82,36%, *recall* validasi sebesar 79,53%, dan *f1 score* validasi sebesar 80,92%.

Kata kunci: sarkasme, BiLSTM, LSTM, deep learning, f1-score

© 2022 Jurnal DINDA

## 1. Pendahuluan

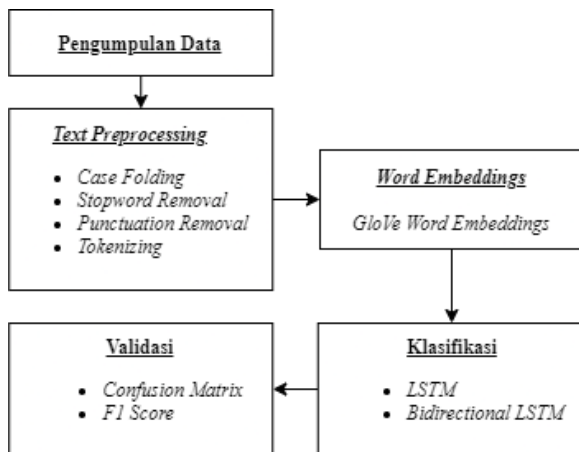
Dalam Kamus Besar Bahasa Indonesia, sarkasme adalah penggunaan kata-kata pedas untuk menyakiti hati orang lain, berupa cemoohan atau ejekan kasar [1]. Kata sarkasme diturunkan dari kata Yunani *sarkasmos* yang berarti “merobek-robek daging seperti anjing”, “menggigit bibir karena marah”, atau “berbicara dengan kepahitan” [2]. Sarkasme merupakan suatu acuan yang lebih kasar dari ironi dan sinisme dan mengandung kepahitan serta celaan yang getir. Sarkasme dapat bersifat ironis, atau tidak, tetapi yang pasti adalah bahwa gaya bahasa ini selalu akan menyakiti hati dan kurang enak didengar [3]. Gaya bahasa sarkasme dapat dicirikan sebagai berikut [2]:

- 1) Maknanya mengandung olok-olokan, ejekan, atau sindiran
- 2) Mengatakan makna yang bertentangan
- 3) Mengandung kepahitan dan kurang enak didengar
- 4) Lebih kasar dibandingkan dengan gaya bahasa ironis dan gaya bahasa sinisme.

Penelitian ini bertujuan untuk membuat model klasifikasi dengan algoritme *Bidirectional Long Short-Term Memory* (BiLSTM) untuk deteksi gaya bahasa sarkasme pada judul berita berbahasa inggris. Judul berita diambil sebagai dataset dikarenakan penulisan judul berita yang menggunakan kata baku dan tidak informal, judul berita juga tidak memiliki salah pengejaan kata yang memudahkan proses pengolahan dataset untuk pembuatan model.

## 2. Metode Penelitian

Berikut ini merupakan alir metode yang digunakan pada penelitian.



Gambar 1. Diagram Alir Penelitian

### 2.1. Dataset

Dataset yang digunakan dalam penelitian ini adalah dataset judul berita berbahasa inggris yang diambil dari dua situs. Judul berita yang berisi sarkasme didapatkan dari *TheOnion* dan judul berita yang tidak berisi sarkasme diambil dari *HuffPost*. Judul berita disusun menggunakan kata baku dan tidak ada salah pengetikan sehingga memudahkan proses penelitian [4]. Dataset terdapat 3 kolom yaitu “*is\_sarcastic*” yang berisi nilai 1 jika judul berita berupa sarkasme dan 0 jika judul berita tidak berupa sarkasme, “*headline*” yang berisi judul berita tersebut, dan “*article\_link*” yang berisi tautan dari judul berita tersebut.

Tabel 1. Tabel Dataset Penelitian

is_sarcastic	headline	Article
1	thirtysomething scientists unveil doomsday clock of hair loss	<a href="https://www.theonion.com/thirtysomething-scientists-unveil-doomsday-clock-of-hai-1819586205">https://www.theonion.com/thirtysomething-scientists-unveil-doomsday-clock-of-hai-1819586205</a>
0	dem rep. totally nails why congress is falling short on gender, racial equality	<a href="https://www.huffingtonpost.com/entry/donna-edwards-inequality_us_57455f7e4b055bb1170b207">https://www.huffingtonpost.com/entry/donna-edwards-inequality_us_57455f7e4b055bb1170b207</a>
0	eat your veggies: 9 deliciously different recipes	<a href="https://www.huffingtonpost.com/entry/eat-your-veggies-9-delici_b_8899742.html">https://www.huffingtonpost.com/entry/eat-your-veggies-9-delici_b_8899742.html</a>
1	inclement weather prevents liar from getting to work	<a href="https://local.theonion.com/inclement-weather-prevents-liar-from-getting-to-work-1819576031">https://local.theonion.com/inclement-weather-prevents-liar-from-getting-to-work-1819576031</a>
1	mother comes pretty close to using word 'streaming' correctly	<a href="https://www.theonion.com/mother-comes-pretty-close-to-using-word-streaming-cor-1819575546">https://www.theonion.com/mother-comes-pretty-close-to-using-word-streaming-cor-1819575546</a>

### 2.2. Case Folding

*Case folding* adalah proses mengubah semua huruf kapital pada dokumen menjadi tidak kapital [5]. *Case folding* dilakukan untuk menyamakan semua kata karena kata yang sama tetapi penggunaan huruf kapital yang berbeda dapat mempengaruhi hasil dari penelitian.

### 2.3. Stopword Removal

*Stopword* diperkenalkan pertama kali oleh H.P. Luhn pada tahun 1958. *Stopword* adalah kumpulan kata yang sering muncul dalam sebuah dokumen dan terdapat sedikit informasi yang biasanya tidak dibutuhkan [6]. Menghapus *stopword* tidak hanya mengurangi kompleksitas komputasi, menghapus *stopword* juga dapat meningkatkan kualitas hasil yang didapatkan [7]. Pada penelitian ini dataset merupakan bahasa inggris,

contoh *stopword* bahasa inggris antara lain seperti *an, and, any, are, the, for,* dan yang lainnya.

#### 2.4. Punctuation Removal

Tanda baca pada dokumen tidak memiliki arti yang bermakna pada suatu kalimat. Maka lebih baik tanda baca untuk dihilangkan [8].

#### 2.4. Tokenizing

*Tokenizing* adalah proses pemisahan setiap kata pada suatu dokumen dimana potongan-potongan tersebut dinamakan dengan *token* [9].

#### 2.5. Word Embeddings

*Word embeddings* merupakan salah satu metode *deep learning* yang sangat berguna untuk menyusun representasi kata pada suatu dokumen menjadi sebuah vektor. Metode ini dapat menangkap hubungan sintaksis dan semantik antar kata yang ada pada suatu dokumen [10]. Pada penelitian ini digunakan *GloVe Word Embeddings*. *GloVe Word Embeddings* dibuat berdasarkan log bilinear model dan menggabungkan keuntungan dari metode *local window* dan faktorisasi matriks global [11].

#### 2.6. Long Short-Term Memory (LSTM) Neural Network

*Long Short-Term Memory (LSTM)* adalah sebuah evolusi dari *Recurrent Neural Network (RNN)* yang diperkenalkan pertama kali oleh Hochreiter dan Schmidhuber untuk mengatasi kekurangan dari RNN dengan cara menambahkan interaksi tambahan pada tiap modul. LSTM dapat mempelajari dependensi *long-term* dan mengingat informasi dalam jangka waktu yang lama [12]. LSTM dapat menghubungkan informasi sebelumnya dan informasi selanjutnya yang dikendalikan melalui 3 *gate* yaitu *input gate*, *output gate* dan *forget gate*. *Input gate* dan *output gate* berfungsi untuk mengatur alur data yang masuk dan keluar dalam *network* sementara *forget gate* akan menghapus atau melupakan informasi yang berbobot rendah [13]. Perhitungan LSTM dapat dirumuskan sebagai berikut [14], [15].

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (1)$$

$$\tilde{C}_t = \tanh(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (2)$$

$$C_t = f_t C_{t-1} + i_t \tilde{C}_t \quad (3)$$

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (4)$$

$$O_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (5)$$

$$h_t = O_t \tanh C_t \quad (6)$$

dimana

$i_t$  = *input gate*

$f_t$  = *forget gate*

$O_t$  = *output gate*

$\sigma$  = fungsi sigmoid

$x_t$  = *input saat ini*

$h_t$  = *output baru*

$h_{t-1}$  = *output sebelumnya*

$\tilde{C}_t$  = *kondisi sel saat ini*

$C_t$  = *kondisi sel baru*

$C_{t-1}$  = *kondisi sel sebelumnya*

$W_i$  = *bobot matrix input gate*

$b_i$  = *bias input gate LSTM*

$W_f$  = *bobot matrix forget gate*

$b_f$  = *bias forget gate LSTM*

$W_o$  = *bobot matrix output gate*

$b_o$  = *bias output gate LSTM*.

#### 2.7. Bidirectional LSTM

*Bidirectional LSTM (BiLSTM)* merupakan sebuah improvisasi dari LSTM dimana BiLSTM akan memeriksa semua informasi dari arah depan maupun arah belakang, tidak seperti LSTM yang hanya memeriksa informasi dari arah depan saja [16].

#### 2.8. Confusion Matrix

*Confusion Matrix* menampilkan dimana model klasifikasi “kebingungan (*confused*)” saat melakukan prediksi. *Confusion matrix* dapat menampilkan kesalahan prediksi model klasifikasi dan juga tipe dari kesalahannya. Berikut merupakan tampilan dari *confusion matrix* [17].

Tabel 2. Tabel *Confusion Matrix*

		Predicted Class	
		Positive	Negative
Actual Class	Positive	True Positive (TP)	False Negative (FN)
	Negative	False Positive (FP)	True Negative (TN)

dimana

TP = jumlah data positif yang diprediksi benar

FP = jumlah data positif yang diprediksi salah

FN = jumlah data negatif yang diprediksi benar

TN = jumlah data negatif yang diprediksi benar.

Dari *confusion matrix* dapat ditemukan proporsi kasus dengan hasil positif yang diprediksi benar (*precision*), proporsi kasus positif yang diprediksi benar (*recall*), dan akurasi dengan perumusan sebagai berikut [18]:

$$Precision = \frac{TP}{TP+FP} \quad (7)$$

$$Recall = \frac{TP}{TP+FN}$$

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$$

### 2.9. F1 Score

F1 Score menggabungkan nilai dari *precision* dan *recall*. F1 Score merupakan nilai yang menandakan apabila model yang dibangun memiliki *precision* dan *recall* yang baik [19]. F1 Score dapat dirumuskan sebagai berikut [20]:

$$F_1 = 2 \cdot \frac{Precision \cdot Recall}{Precision+Recall}$$

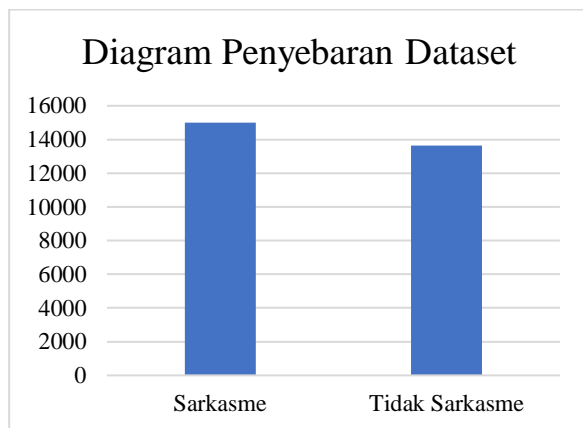
## 3. Hasil dan Pembahasan

### 3.1. Eksplorasi Data

Dataset yang digunakan berjumlah 28619 data yang terdiri dari 14985 judul yang bermakna sarkasme dan 13634 judul berita yang tidak bermakna sarkasme. Berikut merupakan tabel dan diagram penyebaran data pada dataset.

Tabel 3. Tabel Penyebaran Dataset

Kelas	Jumlah Data
Sarkasme	14985
Tidak Sarkasme	13634
Total	28619



Gambar 2. Diagram Penyebaran Dataset

### 3.2. Preprocessing

Pada proses *preprocessing* dataset dilakukan *case folding*, *stopword removal*, *punctuation removal*, dan *tokenizing*. Pada *case folding*, *stopword removal*, dan *punctuation removal* semua kata pada dataset akan diubah menjadi huruf kecil, dihilangkan kata yang tidak bermakna seperti *and*, *then*, *also*, *a*, *an*, dan lain sebagainya, dan juga dihilangkan semua tanda baca yang ada.

(8) Tabel 4. Proses *Case Folding*, *Stopword Removal*, dan *Punctuation Removal*

Sebelum	Sesudah
thirtysomething scientists unveil doomsday clock of hair loss	thirtysomething scientists unveil doomsday clock hair loss
dem rep. totally nails why congress is falling short on gender, racial equality	dem rep totally nails congress falling short gender racial equality
eat your veggies: 9 deliciously different recipes	eat veggies 9 deliciously different recipes
inclement weather prevents liar from getting to work	inclement weather prevents liar getting work
mother comes pretty close to using word 'streaming' correctly	mother comes pretty close using word streaming correctly

(9)

Setelah dilakukan proses *case folding*, *stopword removal*, dan *punctuation removal*, dataset lalu dilakukan proses *tokenizing*.

Tabel 5. Proses *Tokenizing*

Sebelum	Sesudah
thirtysomething scientists unveil doomsday clock hair loss	'thirtysomething', 'scientists', 'unveil', 'doomsday', 'clock', 'hair', 'loss'
dem rep totally nails congress falling short gender racial equality	'dem', 'rep', 'totally', 'nails', 'congress', 'falling', 'short', 'gender', 'racial', 'equality'
eat veggies 9 deliciously different recipes	'eat', 'veggies', '9', 'deliciously', 'different', 'recipes'
inclement weather prevents liar getting work	'inclement', 'weather', 'prevents', 'liar', 'getting', 'work'
mother comes pretty close using word streaming correctly	'mother', 'comes', 'pretty', 'close', 'using', 'word', 'streaming', 'correctly'

### 3.3. Pemodelan

Pada penelitian ini digunakan algoritme LSTM dan BiLSTM untuk pembuatan model klasifikasi. Model LSTM dan BiLSTM dibuat secara identikal agar bisa dibandingkan secara adil. Berikut merupakan *model summary* dari model LSTM dan BiLSTM.

Tabel 5. Tabel *Model Summary* LSTM

Layer	Output Shape	Parameter
Embedding layer	(None, 32, 100)	2850300
LSTM Layer	(None, 32, 64)	42240
LSTM Layer	(None, 32)	12416
Dense Layer	(None, 1)	33

Tabel 6. Tabel *Model Summary* BiLSTM

Layer	Output Shape	Parameter
Embedding layer	(None, 32, 100)	2850300
Bidirectional LSTM Layer	(None, 32, 128)	84480
Bidirectional LSTM Layer	(None, 64)	41216
Dense Layer	(None, 1)	65

Dataset di bagi menjadi 80% untuk pelatihan model dan 20% untuk validasi model. Proses pelatihan model dilakukan sebanyak 10 *epoch* dengan *batch size* 128.

### 3.4 Hasil

Dari penelitian ini didapatkan hasil akurasi, *precision*, *recall*, dan *f1 score* sebagai berikut:

Tabel 7. Hasil Model LSTM

	Data	
	Pelatihan	Validasi
Akurasi	83,76%	81,90%
<i>Precision</i>	83,68%	80,78%
<i>Recall</i>	82,39%	80,17%
<i>F1 Score</i>	83,03%	80,47%

Tabel 8. Hasil Model BiLSTM

	Data	
	Pelatihan	Validasi
Akurasi	87,62%	82,55%
<i>Precision</i>	87,22%	82,36%
<i>Recall</i>	86,78%	79,53%
<i>F1 Score</i>	87,00%	80,92%

## 4. Kesimpulan

Dari penelitian yang sudah dilakukan dapat diambil kesimpulan bahwa algoritme *Bidirectional LSTM* memiliki keunggulan daripada algoritma *LSTM* walaupun hanya sedikit. Kedua algorithme mendapatkan hasil akurasi yang baik dimana *Bidirectional LSTM* mendapatkan akurasi validasi sebesar 82,55% dan *f1 score* sebesar 80,92% dan algoritma *LSTM* mendapatkan akurasi validasi sebesar 81,90% dan *f1 score* sebesar 80,47%.

## Daftar Rujukan

- [1] "Sarkasme," *KBBI Daring*, 2021. <https://kbbi.kemdikbud.go.id/entri/sarkasme>.
- [2] A. N. Cahyo, T. A. A. Manullang, and M. Isnain, "Analisis Penggunaan Gaya Bahasa Sarkasme pada Lagu Bahaya Komunis Karangan Jason Ranti," *Asas J. Sastra*, vol. 9, no. 1, pp. 6–22, 2020, doi: 10.24114/ajs.v9i1.18329.
- [3] S. Lutfiyani, B. E. Purwanto, and S. Anwar, "Sarkasme Pada Media Sosial Twitter Dan Implikasinya Terhadap Pembelajaran Bahasa Indonesia di SMA," *J. Bahasa, Sastra Indones. dan Pengajarannya*, vol. 1, no. 2, pp. 269–284, 2020.
- [4] R. Misra and P. Arora, "Sarcasm Detection using Hybrid Neural Network," 2019, doi: 10.13140/RG.2.2.32427.39204.
- [5] M. A. Rosid, A. S. Fitriani, I. R. I. Astutik, N. I. Mulloh, and H. A. Gozali, "Improving Text Preprocessing for Student Complaint Document Classification Using Sastrawi," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 874, no. 1, 2020, doi: 10.1088/1757-899X/874/1/012017.
- [6] J. Kaur and P. Kaur Buttar, "A Systematic Review on Stopword Removal Algorithms," *Int. J. Futur. Revolut. Comput. Sci. Commun. Eng.*, no. April, pp. 207–210, 2018, [Online]. Available: <http://www.ijfrcsce.org>.
- [7] H. Fani, M. Bashari, F. Zarrinkalam, E. Bagheri, and F. Al-Obeidat, "Stopword detection for streaming content," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 10772 LNCS, pp. 737–743, 2018, doi: 10.1007/978-3-319-76941-7\_70.
- [8] G. Singh, B. Kumar, L. Gaur, and A. Tyagi, "Comparison between Multinomial and Bernoulli Naïve Bayes for Text Classification," *2019 Int. Conf. Autom. Comput. Technol. Manag. ICACTM 2019*, pp. 593–596, 2019, doi: 10.1109/ICACTM.2019.8776800.
- [9] A. L. Fairuz, R. D. Ramadhani, and N. A. Tanjung, "Analisis Sentimen Masyarakat Terhadap COVID-19 Pada Media Sosial," *J. DINDA*, vol. 1, no. 1, pp. 10–12, 2021, [Online]. Available: <http://journal.itelkom-pwt.ac.id/index.php/dinda/article/view/180>.
- [10] S. M. Rezaeinia, R. Rahmani, A. Ghodsi, and H. Veisi, "Sentiment analysis based on improved pre-trained word embeddings," *Expert Syst. Appl.*, vol. 117, pp. 139–147, 2019, doi: 10.1016/j.eswa.2018.08.044.
- [11] M. Ibrahim, S. Gauch, O. Salman, and M. Alqahatani, "Enriching consumer health vocabulary using enhanced glove word embedding," *CEUR Workshop Proc.*, vol. 2619, 2020.
- [12] X. H. Le, H. V. Ho, G. Lee, and S. Jung, "Application of Long Short-Term Memory (LSTM) neural network for flood forecasting," *Water (Switzerland)*, vol. 11, no. 7, 2019, doi: 10.3390/w11071387.
- [13] F. Shahid, A. Zameer, and M. Muneeb, "Predictions for COVID-19 with deep learning models of LSTM, GRU and Bi-LSTM," *Chaos, Solitons and Fractals*, vol. 140, p. 110212, 2020, doi: 10.1016/j.chaos.2020.110212.
- [14] M. Z. Islam, M. M. Islam, and A. Asraf, "A combined deep CNN-LSTM network for the detection of novel coronavirus (COVID-19) using X-ray images," *Informatics Med. Unlocked*, vol. 20, p. 100412, 2020, doi: 10.1016/j.imu.2020.100412.
- [15] Ramadhan, Nur Ghaniaviyanto, Nia Annisa Ferani Tanjung, and Faisal Dharma Adhinata. "Implementation of LSTM-RNN for Bitcoin Prediction." *Indonesia Journal on Computing (Indo-JC)* 6.3 (2021): 17-24. doi.org/10.34818/INDOJC.2021.6.3.592

- [16] Y. Bin, Y. Yang, F. Shen, N. Xie, H. T. Shen, and X. Li, “Describing video with attention-based bidirectional LSTM,” *IEEE Trans. Cybern.*, vol. 49, no. 7, pp. 2631–2641, 2019, doi: 10.1109/TCYB.2018.2831447.
- [17] I. Markoulidakis, G. Kopsiaftis, I. Rallis, and I. Georgoulas, “Multi-Class Confusion Matrix Reduction method and its application on Net Promoter Score classification problem,” *ACM Int. Conf. Proceeding Ser.*, no. Cx, pp. 412–419, 2021, doi: 10.1145/3453892.3461323.
- [18] D. Ricardo and G. Gasim, “Perbandingan Akurasi Pengenalan Jenis Beras dengan Algoritma Propagasi Balik pada Beberapa Resolusi Kamera,” *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 3, no. 2, pp. 131–140, 2019, doi: 10.29207/resti.v3i2.894.
- [19] S. Clara, D. L. Prianto, R. Al Habsi, and E. F. Lumbantobing, “Implementasi Seleksi Fitur Pada Algoritma Klasifikasi Machine Learning Untuk Prediksi Penghasilan Pada Adult Income Dataset,” no. April, pp. 741–747, 2021.
- [20] M. Y. Aldean, M. D. Hilmawan, R. Indriyati, and J. Lasama, “Analisa Relevansi Tweet terhadap Hashtag dengan Metode Logistic Regression,” *Centive*, vol. 2, no. 1, pp. 32–38, 2019.