

Sarcasm Detection on Twitter Using Gated Recurrent Units Method

Maulidan¹, Khadafi^{2*}, Huzaeni³

^{1,2,3} Jurusan Teknologi Informasi dan Komputer Politeknik Negeri Lhokseumawe, Jln. B.Aceh Medan Km.280 Buketrata 24301
INDONESIA

*Penulis Korespondensi : mkhadafi@pnl.ac.id

INFORMASI ARTIKEL

Riwayat artikel:

Diajukan pada 15 Mei 25

Direvisi pada 03 Juni 25

Publikasi pada 20 Juni 25

Kata kunci:

Sarkasme

Twitter

Gated Recurrent Unit

Natural Language Processing

Keywords:

Sarcasm

Twitter

Gated Recurrent Unit

Natural Language Processing

ABSTRAK

Sarkasme sering digunakan sebagai alat untuk mencapai efek humor, menyampaikan kritik dengan cara yang menghibur, atau menyampaikan pesan dengan keironisan. Media sosial, khususnya Twitter, menjadi platform yang signifikan untuk ekspresi pendapat dan pertukaran informasi, di mana sarkasme sering kali digunakan sebagai sarana komunikasi. Penelitian ini berfokus pada penggunaan *Gated Recurrent Unit* (GRU), sebuah jenis unit dalam jaringan saraf rekurensial (RNN), untuk mendeteksi kalimat sarkasme di Twitter. Data yang digunakan dalam penelitian ini diambil dari sumber yang sudah dipergunakan dalam pengujian sebelumnya, berjumlah total 10.000 data, yang dibagi menjadi dua kelas: sarkastik dan non-sarkastik. Sebanyak 80% dari data ini digunakan sebagai data latih, sementara 20% sisanya digunakan sebagai data uji. Hasil pengujian menunjukkan bahwa model GRU mampu mencapai akurasi sebesar 73% dengan menggunakan *confusion matrix* dan 74% dengan metode *K-Fold Cross Validation*. Ini menunjukkan kemampuan model dalam mengklasifikasikan kalimat sarkastik dan non-sarkastik secara efektif.

ABSTRACT

Sarcasm is often used as a tool to achieve humorous effects, convey criticism in an entertaining way, or convey messages with irony. Social media, especially Twitter, has become a significant platform for expression and information exchange, where sarcasm is often used as a means of communication. This study focuses on the use of Gated Recurrent Unit (GRU), a type of unit in a recurrent neural network (RNN), to detect sarcastic sentences on Twitter. The data used in this study were taken from sources that had been used in previous tests, totaling 10,000 data, which were divided into two classes: sarcastic and non-sarcastic. As much as 80% of this data was used as training data, while the remaining 20% was used as test data. The test results showed that the GRU model was able to achieve an accuracy of 73% using the confusion matrix and 74% using the K-Fold Cross Validation method. This shows the model's ability to classify sarcastic and non-sarcastic sentences effectively.

1. Pendahuluan

Sarkasme sering digunakan untuk mencapai efek humor, mengkritik dengan cara yang menghibur, atau menyampaikan pesan dengan cara yang bersifat ironis. Sarkasme juga memiliki nilai positif dan negatifnya. Salah satu nilai positif dari sarkasme adalah sebagai humor dan ekspresi kreatif dalam menyampaikan pesan atau mengkritik pesan. Nilai negatif sarkasme salah satunya yaitu menyinggung yang menyebabkan menyakiti perasaan orang lain. Media sosial adalah salah satu platform terbesar bagi orang-

orang mengutarakan pendapatnya dan berbagi informasi. Banyak pemerintah dan organisasi perusahaan menggunakan data ini untuk memahami sentimen masyarakat terhadap produk, film, dan peristiwa politik [1]. Twitter menjadi salah satu tujuan web terbesar bagi orang-orang untuk mengekspresikan pendapat mereka, berbagi pemikiran mereka dan melaporkan peristiwa secara *real-time*. Sepanjang tahun-tahun sebelumnya, konten twitter terus meningkat, sehingga merupakan contoh khas dari hal disebut data besar. Saat ini, menurut situs resminya, twitter memiliki lebih dari 288 juta pengguna aktif, dan lebih dari 500 juta tweet dikirim setiap hari. Banyak perusahaan dan organisasi yang tertarik dengan data ini untuk tujuan mempelajari opini masyarakat terhadap peristiwa politik, produk populer atau film. Namun, karena bahasa informal yang digunakan di twitter dan keterbatasan karakter (yaitu 140 karakter per tweet), cukup sulit untuk memahami pendapat pengguna dan melakukan analisis tersebut [2]. Twitter juga layanan mikro-blog dan jejaring sosial waktu nyata yang telah mendapatkan popularitas luas sejak dekade terakhir [3].

Sarkasme adalah kejadian umum di media sosial dan sulit diidentifikasi dan dianalisis contohnya penggunaan kata-kata pedas yang ditujukan untuk menyakiti hati orang lain. Kata-kata ini berupa cemooh dan ejekan kasar. Sarkasme berarti menyebutkan sesuatu yang bertentangan dengan kebenaran. Ini juga disebut sebagai ironi verbal. Deteksi sarkasme adalah pendekatan yang menentukan apakah pendapat yang diungkapkan pengguna bersifat positif yaitu sarkastik atau tidak yaitu non-sarkastik [4]. Dalam mendeteksi kalimat sarkasme yang memiliki suatu nilai positif atau negatif atau bahkan sering disebut kalimat sarkastik atau non-sarkastik, dapat dikembangkannya suatu aplikasi berbasis model *machine learning* untuk mendeteksi suatu kalimat sarkasme yang ada di twitter dengan menggunakan metode *Gated Recurrent Unit (GRU)* dikarenakan metode GRU dapat memiliki kemampuan generalisasi yang baik, yang berarti model yang dilatih dapat lebih baik menangani data yang belum pernah dilihat sebelumnya. Hal ini dapat bermanfaat dalam mendeteksi sarkasme yang mungkin memiliki berbagai gaya dan jenis ekspresi [5].

Gated Recurrent Unit (GRU) merupakan salah satu jenis unit dalam *Recurrent Neural Network (RNN)* yang digunakan dalam bidang pembelajaran mesin dan pemrosesan bahasa alami. *Recurrent Neural Network (RNN)* menyediakan cara yang elegan untuk menangani data sekuensial yang mewujudkan korelasi antara titik data yang berdekatan dalam urutan [5][6]. Penelitian ini juga termasuk pada kedalam *Natural Language Processing (NLP)* didefinisikan sebagai proses komputasi yang menganalisis unit bahasa alami dalam berbagai tingkat analisis linguistik[7][8]. NLP sebagai bidang penelitian telah memainkan peran utama dalam pengembangan sistem [9][10]. Penelitian ini akan menggunakan metode *Gated Recurrent Units* untuk membuat model pendeteksi kalimat sarkasme. Algoritma *Gated Recurrent Units* dipilih karena mampu dalam memahami dan memproses data berurutan dengan baik. Banyak penelitian yang mendeteksi sarkasme dengan menggunakan metode SVM, RNN, *random forest* dan metode lainnya. Hasil penelitian ini diharapkan dapat membantu pengguna media sosial khususnya pada twitter dalam memahami suatu arti dari kalimat sarkasme.

2. Metode

2.1 Analisis Kebutuhan Sistem

Analisis kebutuhan data pada penelitian ini meliputi data postingan dari pengguna twitter berbahasa inggris. Data tersebut dibagi menjadi dua bagian yaitu data training dan data uji. Data training akan

digunakan untuk mendapatkan model yang bagus dari metode Gated Recurrent Units. Sedangkan data uji akan digunakan untuk menguji model yang sudah di latih.

2.2 Analisa dan Kebutuhan Fungsional

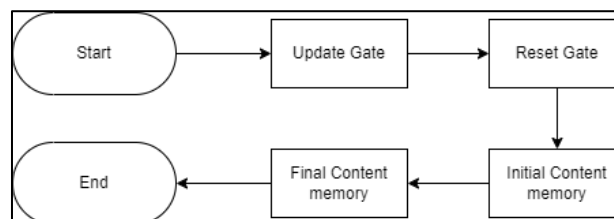
Kebutuhan Kebutuhan fungsional adalah proses yang terjadi pada setiap entitas yang terlibat di dalamnya, dengan adanya analisa kebutuhan fungsional dapat mempermudah suatu proses pengolahan data pada sistem. Beberapa kebutuhan yang diperlukan dalam membuat perancangan pendeteksi sarkasme di twitter yaitu pengguna dapat memasukkan kalimat postingan baru dari twitter untuk dilakukan deteksi oleh sistem. *Preprocessing*, admin dapat melakukan *preprocessing* pada data. *Labelling*, admin dapat melakukan *Labelling* pada data. Klasifikasi, admin dapat melakukan klasifikasi dengan metode *Gated Recurrent Units*. Hasil deteksi, pengguna dapat melihat hasil dari pendeteksi sarkasme

2.3 Pengumpulan Data

Pengumpulan data untuk penelitian ini dilakukan melalui platform Twitter dengan teknik *scraping* dan telah digunakan oleh pihak lain dalam suatu penelitian. Data tersebut yaitu data dari postingan pengguna twitter berbahasa inggris. Data tersebut mencakup 10.000 postingan dari berbagai pengguna Twitter dari tahun ke tahun. Pengumpulan data juga disebut dengan data *mining* yang dimana kegiatan mencari rahasia informasi dari kumpulan data dengan cara atau teknik yang tidak mudah.

2.4 Alur Perancangan model Gated Recurrent units

Proses dimulai dengan membagi data menjadi dua set: data data pengujian dan data pelatihan. Data pelatihan digunakan untuk membangun model, sementara data pengujian digunakan untuk mengevaluasi kinerja model. Selanjutnya, data akan diteruskan dengan *update gate* yang bertujuan mengontrol seberapa besar memori. *reset gate* bertujuan untuk mengatur seberapa banyak dari informasi sebelumnya yang akan dibuang. *initial content memory* bertujuan untuk penggabungan antara informasi lama (memori sebelumnya) dan informasi baru yang diproses dan *final content memory* adalah memori dari langkah waktu sebelumnya yang disimpan dalam jaringan dan digunakan untuk menghitung keadaan memori saat ini. Untuk alur perancangannya dapat dilihat pada Gambar 1



Gambar 1. Alur perancangan Model

2.5 Confusion Matrix

Confusion matrix merupakan cara untuk memvisualisasikan kinerja yang diharapkan dari model prediktif dalam data *mining*. Setiap data dari setiap kelas dalam tabel *confusion matrix* menunjukkan jumlah prediksi yang dibuat untuk mengklasifikasikan kelas sebagai benar atau salah.

Evaluasi dengan *confusion matrix* menghasilkan nilai akurasi, presisi dan *recall* nantinya. Nilai akurasi dalam klasifikasi adalah persentase keakuratan catatan data yang diklasifikasikan dengan benar setelah

pengujian dari hasil klasifikasi. *Trus Positif* (TP), jumlah data kelas yang positif yang terklasifikasi sebagai positif oleh sistem; *Trus Negatif* (TN), jumlah data kelas yang negatif yang terklasifikasi sebagai negatif oleh sistem; *False Positif* (FP), jumlah data kelas negatif yang terklasifikasi sebagai positif oleh sistem; *False Negatif* (FN), jumlah data kelas positif yang terklasifikasi sebagai negatif oleh sistem.

2.6 K-Fold Cross Validation

Pengujian *K-Fold Cross Validation* dilakukan untuk mengevaluasi kinerja model *Gated Recurrent Unit* (GRU) dengan membaginya menjadi beberapa *subset* (atau “*fold*”). Teknik ini membantu memastikan bahwa model yang dilatih dan diuji menggunakan data yang berbeda setiap kali, memberikan evaluasi yang lebih akurat terhadap performa model. Tujuan *cross validation* ini untuk mendapatkan nilai *best validation* dan *best learning result*.

3. Hasil Dan Pembahasan

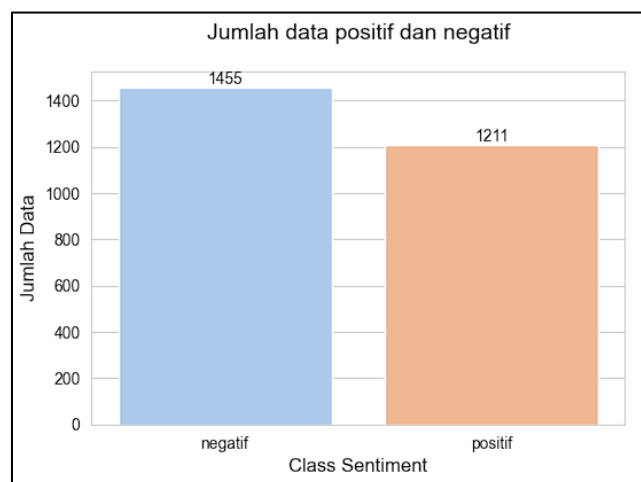
3.1 Dataset

Pada gambar 2, data yang digunakan merupakan *dataset* dari postingan twitter berbahasa inggris yang hanya terdiri dari *headline* dan *is_sarcastic* dengan total sebanyak 10000 data. *Dataset* tersebut akan dibagi menjadi dua bagian, yaitu 80% sebagai data *training* dan 20% sebagai data uji.

	headline	is_sarcastic
1	former versace store clerk sues over secret 'black code' for minority shoppers	0
2	the 'roseanne' revival catches up to our thorny political mood, for better and worse	0
3	mom starting to fear son's web series closest thing she will have to grandchild	1
4	boehner just wants wife to listen, not come up with alternative debt-reduction ideas	1
5	j.k. rowling wishes snape happy birthday in the most magical way	0
6	advancing the world's women	0
7	the fascinating case for eating lab-grown meat	0
8	this ceo will send your kids to school, if you work for his company	0
9	top snake handler leaves sinking huckabee campaign	1
10	friday's morning email: inside trump's presser for the ages	0

Gambar 2. Dataset

3.2 Hasil Labelling



Gambar 3. Hasil Labelling

Menampilkan hasil *labelling* data, hasil *labelling* ini mencakup penandaan kategori pada setiap entri data, seperti positif dan negatif, untuk memudahkan analisis lebih lanjut. Dengan data yang telah dilabeli,

model tersebut dapat dilatih dengan lebih efektif dan hasil lebih akurat.. Berikut hasil *labelling* dapat dilihat pada gambar 3.

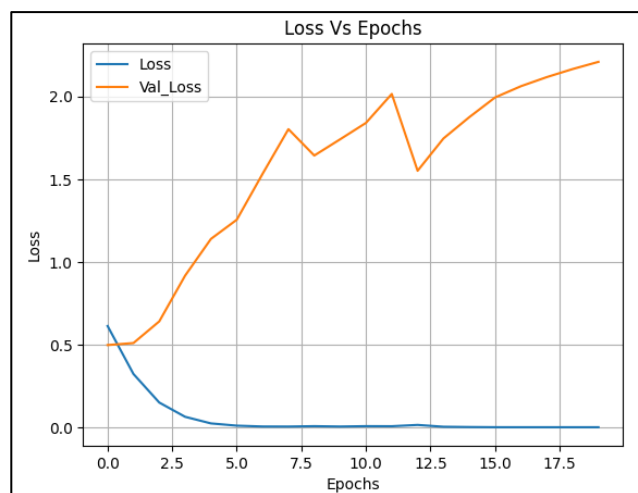
3.3 Hasil Implementasi Metode *Gated Recurrent Units*

Menunjukkan ringkasan arsitektur dari sebuah model pembelajaran mendalam (*deep learning*) yang dibangun dengan menggunakan Keras *Sequential API*. Model ini terdiri dari empat lapisan (*layer*) utama. Lapisan pertama adalah lapisan *embedding* dengan 1.335.450 parameter, yang berfungsi mengonversi input yang merupakan urutan kata ke dalam representasi vektor berdimensi 50. Lapisan kedua adalah lapisan GRU (*Gated Recurrent Unit*) dengan 64 unit dan 22.272 parameter, yang digunakan untuk menangani data urutan atau rangkaian. Lapisan ketiga adalah lapisan *dense* dengan 48 unit dan 3.120 parameter, yang berfungsi untuk melakukan komputasi linier terhadap *input* dari lapisan sebelumnya. Lapisan terakhir adalah lapisan *dense* dengan 2 unit dan 98 parameter, yang kemungkinan besar berfungsi sebagai lapisan *output* dengan dua kelas. Model ini memiliki total 1.360.940 parameter, seluruhnya dapat dilatih (*trainable*), tanpa parameter yang tidak dapat dilatih (*non-trainable*). Total memori yang digunakan oleh model ini adalah sekitar 5,19 MB.. Berikut tampilan halaman keranjang belanja dapat dilihat pada gambar 4.

Layer (type)	Output Shape	Param #
embedding (Embedding)	(None, 100, 50)	1,335,450
gru (GRU)	(None, 64)	22,272
dense (Dense)	(None, 48)	3,120
dense_1 (Dense)	(None, 2)	98

Total params: 1,360,940 (5.19 MB)
 Trainable params: 1,360,940 (5.19 MB)
 Non-trainable params: 0 (0.00 B)

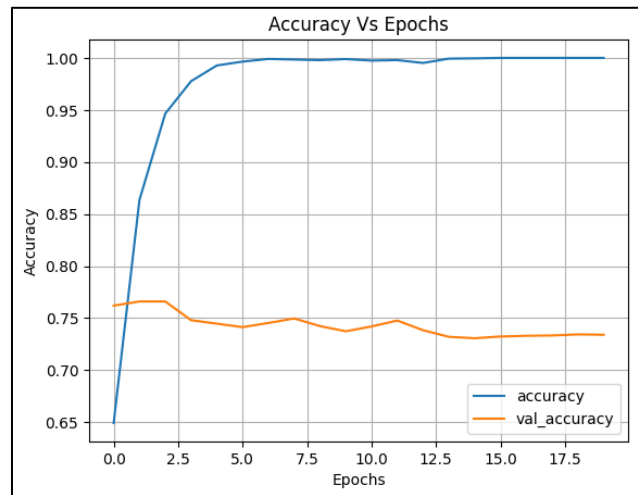
Gambar 4. Hasil Visualisasi Metode *Gated Recurrent Units*



Gambar 5. Hasil Visualisasi *loss* dan *val loss* pada model *Gated Recurrent Units*

Hasil dari visualisasi *loss* dan *val loss* pada model *Gated Recurrent units* dapat dilihat pada gambar 5. Gambar tersebut menunjukkan grafik *loss* pada model *machine learning* selama proses pelatihan. Garis biru menunjukkan *loss* pada data pelatihan, sedangkan garis *orange* menunjukkan *loss* pada data validasi. Terlihat bahwa *loss* pada data pelatihan menurun dengan cepat pada awal pelatihan dan kemudian mencapai *plateau*,

artinya model sudah mulai mempelajari pola yang ada di data pelatihan dengan baik. Namun, *loss* pada data validasi cenderung meningkat seiring berjalannya pelatihan, yang menunjukkan bahwa model mulai mengalami *overfitting*, yaitu model terlalu fokus pada pola yang ada di data pelatihan dan tidak dapat menggeneralisasikan pada data yang baru. Hal ini bisa jadi disebabkan karena model terlalu kompleks atau data pelatihan tidak cukup banyak.



Gambar 6. Hasil visualisasi akurasi pada model *Gated Recurrent Units*

Hasil visualisasi akurasi pada model *Gated Recurrent Units* dapat dilihat pada gambar 6. Garis biru menunjukkan akurasi pada data pelatihan, sedangkan garis oranye menunjukkan akurasi pada data validasi. Dapat dilihat bahwa akurasi model pada data pelatihan meningkat secara signifikan selama iterasi awal pelatihan, kemudian mencapai *plateau*. Akurasi pada data validasi juga meningkat pada awalnya, tetapi kemudian mengalami penurunan. Hal ini menunjukkan bahwa model mulai *overfitting* pada data pelatihan, yaitu model terlalu menyesuaikan diri dengan data pelatihan sehingga tidak dapat generalisasi dengan baik pada data yang belum pernah dilihat sebelumnya.

3.4 Hasil klasifikasi dan Pengujian *Confussion Matrix*

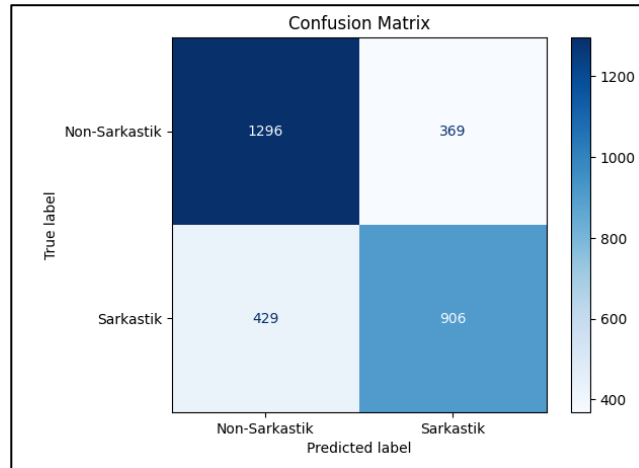
hasil *accuracy* dari *classification report* memperoleh akurasi keseluruhan klasifikasi sebesar 73% menunjukkan bahwa pengklasifikasi tersebut mengklasifikasikan 73% dari total sampel dengan benar.. Berikut hasil pengujian *Confussion Matrix* dapat dilihat pada gambar 7.

Accuracy: 0.73				
Classification Report:				
	precision	recall	f1-score	support
Non-Sarkastik	0.75	0.78	0.76	1665
Sarkastik	0.71	0.68	0.69	1335
accuracy			0.73	3000
macro avg	0.73	0.73	0.73	3000
weighted avg	0.73	0.73	0.73	3000

Gambar 7. Hasil Pengujian *Confussion Matrix*

Berdasarkan hasil evaluasi yang ditampilkan, model klasifikasi tersebut memiliki akurasi sebesar 0,73 atau 73%. Artinya, model tersebut mampu memprediksi dengan benar 73% dari data yang diuji. Selain itu, terlihat bahwa model tersebut memiliki nilai *precision*, *recall*, dan *f1-score* yang baik, dengan nilai masing-

masing sekitar 0.70. Hal ini menunjukkan bahwa model memiliki kemampuan yang baik dalam mengklasifikasikan data dengan benar, mengingat data yang digunakan untuk evaluasi kemungkinan terdiri dari data yang kompleks. Secara keseluruhan, model klasifikasi yang diuji memiliki performa yang cukup baik, terlihat dari nilai akurasi, *precision*, *recall*, dan *f1-score* yang relatif tinggi. Hal ini menunjukkan bahwa model dapat digunakan untuk menyelesaikan masalah klasifikasi dengan tingkat keakuratan yang relatif tinggi.

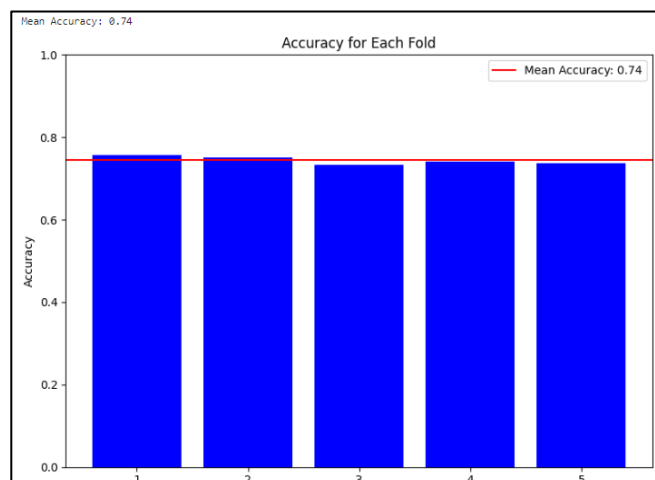


Gambar 8. Hasil Pengujian *Confusion Matrix*

Pada gambar 8, model tersebut memiliki tingkat akurasi yang cukup baik dalam mengklasifikasikan kalimat sarkastik dan non-sarkastik. Model ini mampu mengklasifikasikan 1296 kalimat non-sarkastik dengan benar dan 906 kalimat sarkastik dengan benar. Meskipun terjadi kesalahan klasifikasi sebanyak 369 kali untuk kalimat non-sarkastik dan 429 kali untuk kalimat sarkastik, namun model ini masih dapat memberikan prediksi yang cukup akurat. Kesimpulannya, model ini cukup efektif dalam mengklasifikasikan kalimat sarkastik dan non-sarkastik.

3.5 Hasil klasifikasi dan pengujian *K-Fold Cross Validation*

Hasil akurasi dari klasifikasi dan pengujian menggunakan *K-Fold Cross validation* sebesar 74% dengan pelatihan 5 *Fold*. Hasilnya bisa dilihat pada gambar 9.



Gambar 9. Hasil Pengujian *K-Fold Cross Validation*

Grafik ini menunjukkan akurasi model pada setiap *fold* (lipatan) dari *cross-validation*. Terlihat bahwa akurasi model relatif konsisten di setiap *fold*, dengan nilai akurasi sekitar 0.74. Rata-rata akurasi dari seluruh *fold* adalah 0.74. Grafik ini menunjukkan bahwa model *machine learning* yang digunakan memiliki performa yang stabil dan konsisten di berbagai bagian data. Ini menunjukkan bahwa model tersebut tidak terlalu dipengaruhi oleh variasi data dan dapat digeneralisasikan dengan baik ke data baru.

4. Kesimpulan

Kesimpulan setelah melakukan perancangan dan pengujian pada sistem deteksi sarkasme dengan studi kasus sarkasme inggris menggunakan metode *Gated Recurrent Unit* berbasis website. Hasil evaluasi dari metode GRU atau *Gated Recurrent Units* menggunakan *k-fold cross validation* memperoleh akurasi sebesar 74% dengan menggunakan 5 *fold* dan hasil evaluasi metode menggunakan *confusion matrix* memperoleh akurasi sebesar 73%. Skor F1-score pada data yang sarkastik yaitu sebesar 76% dan pada data yang non-sarkastik 69%. Pada *k-fold cross validation*, setiap *accuracy* dari fold-1 sampai fold-5 yaitu naik turun mulai dari hasil *accuracy* fold-1 yaitu 76% hingga menurun pada fold-2 yaitu sebesar 75%. *Accuracy* menurun kembali pada fold-3 yaitu 73% dan pada fold-4 yang awalnya *accuracy* 73% naik menjadi 74% hingga *accuracy* pada fold-5 pun sebesar 74%. Matrix rata-rata makro dan rata-rata tertimbang menunjukkan bahwa meskipun pengklasifikasi memiliki keseimbangan kinerja yang wajar di semua kelas, masih ada ruang untuk perbaikan.

Referensi

- [1] A. Kumar, V. T. Narapareddy, V. A. Srikanth, A. Malapati, and L. B. M. Neti, "Sarcasm Detection Using Multi-Head Attention Based Bidirectional LSTM," IEEE Access, vol. 8, pp. 6388–6397, 2020, doi: 10.1109/ACCESS.2019.2963630.
- [2] M. Bouazizi and T. Otsuki, "A Pattern-Based Approach for Sarcasm Detection on Twitter," IEEE Access, vol. 4, pp. 5477–5488, 2016, doi: 10.1109/ACCESS.2016.2594194.
- [3] Z. Doshi, S. Nadkarni, K. Ajmera, and N. Shah, "TweeterAnalyzer: Twitter Trend Detection and Visualization," 2017 Int. Conf. Comput. Commun. Control Autom. ICCUBEA 2017, 2017, doi: 10.1109/ICCUBEA.2017.8463951.
- [4] D. M. Ashok, A. Nidhi Ghanshyam, S. S. Salim, D. Burhanuddin Mazahir, and B. S. Thakare, "Sarcasm detection using genetic optimization on LSTM with CNN," 2020 Int. Conf. Emerg. Technol. INCET 2020, pp. 3–6, 2020, doi: 10.1109/INCET49848.2020.9154090.
- [5] T. Ishitaki, R. Obukata, T. Oda, and L. Barolli, "Application of deep recurrent neural networks for prediction of user behavior in tor networks," Proc. - 31st IEEE Int. Conf. Adv. Inf. Netw. Appl. Work. WAINA 2017, pp. 238–243, 2017, doi: 10.1109/WAINA.2017.63.
- [6] S. A. Al-Ghamdi, J. Khabti, and H. S. Al-Khalifa, "Exploring NLP web APIs for building Arabic systems," 2017 12th Int. Conf. Digit. Inf. Manag. ICDIM 2017, vol. 2018-Janua, no. Icdim, pp. 175–178, 2017, doi: 10.1109/ICDIM.2017.8244649.
- [7] G. A. J. Satvika, S. M. Nasution, and R. A. Nugrahaeni, "Determination of the Best Vehicle Pathway with Classification of Data Mining Twitter using K-Nearest Neighbor," 2018 Int. Conf. Inf. Technol. Syst. Innov. ICITSI 2018 - Proc., pp. 72–76, 2018, doi: 10.1109/ICITSI.2018.8695947.
- [8] Jahan, S. S. (2014). Human Resources Information System (HRIS): A theoretical perspective. Journal of Human Resource and Sustainability Studies, 02(02), 33–39. <https://doi.org/10.4236/jhrss.2014.22004>.
- [9] Niu, Z., Yu, Z., Tang, W., Wu, Q., & Reformat, M. (2020). Wind power forecasting using attention-based gated recurrent unit network. Energy, 196, 117081. <https://doi.org/10.1016/j.energy.2020.117081>.
- [10] Sherstinsky, A. (2020). Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network. Physica D Nonlinear Phenomena, 404, 132306. <https://doi.org/10.1016/j.physd.2019.132306>.