PETIR: Jurnal Pengkajian dan Penerapan Teknik Informatika
Vol. 15, No. 2, September 2022, P-ISSN 1978-9262, E-ISSN 2655-5018
DOI: https://doi.org/10.33322/petir.v15i2.1197

Sentiment Analysis Terhadap Tulisan Mengenai Universitas Pertamina Di Media Sosial Twitter

M. Rizky Widyayulianto[1]; Meredita Susanty[1)]; Ade Irawan[1]*

1. Program Studi Ilmu Komputer, Fakultas Sains dan Komputer, Universitas Pertamina, DKI Jakarta 12220, Indonesia

[*)]Email: meredita.susanty@universitaspertamina.ac.id

Received: 24 Desember 2020 | Accepted: 24 November 2022 | Published: 28 November 2022

ABSTRACT

Nowadays, branding is not only applied to business sectors. Many universities have implemented branding to maintain the university's brand image so that it can attract prospective students. Sentiment analysis is one of the activities to monitor the university's brand image in the community. This research applies a machine learning approach to conduct a social media analysis of Twitter. The dataset consists of 4323 tweets containing the word "Universitas Pertamina". Each tweet is then grouped into three different classes based on the tweet's sentiment, namely negative, neutral and positive. This study uses three types of machine learning models: Convolutional Neural Network (CNN), Long Short Term Memory (LSTM) and CNN-LSTM. The procedure for the three models uses the stratified 5-fold cross-validation technique. Model performance is then measured using a learning curve and measuring parameters such as accuracy, weighted-f1 and balanced accuracy. The CNN-LSTM model records the highest accuracy value, 76.92% and the highest weighted-f1 with 76.78%. Meanwhile, the LSTM model gets the highest balanced accuracy value with 66.05%.

Keywords: *sentiment analysis, machine learning, social media, neural network*

ABSTRAK

Konsep branding saat ini tidak hanya digunakan di sektor bisnis saja. Banyak universitas yang telah menerapkan branding untuk menjaga brand image dari universitas tersebut sehingga dapat menarik minat calon mahasiswa. Sentiment analysis adalah salah satu kegiatan yang dapat dilakukan untuk memantau brand image universitas di mata masyarakat. Penelitian ini menerapkan konsep machine learning untuk melakukan sentiment analysis di media sosial Twitter. Dataset yang digunakan terdiri dari 4323 tweet berbahasa Indonesia yang mengandung kata "Universitas Pertamina". Setiap tweet kemudian dikelompokkan ke 3 kelas berbeda berdasarkan sentimen tweet tersebut, yaitu negatif, netral dan positif. Penelitian ini menggunakan 3 jenis model machine learning, yaitu Convolutional Neural Network (CNN), Long Short Term Memory (LSTM) dan CNN-LSTM. Proses training ketiga model tersebut menggunakan teknik stratified 5-fold cross validation. Performa model kemudian diukur dengan menggunakan learning curve dan mengukur parameter seperti accuracy, weightedf1 dan balanced accuracy. Model CNN-LSTM mendapatkan nilai accuracy tertinggi dengan 76.92% dan weighted-f1 tertinggi dengan 76.78%. Sementara model LSTM mendapatkan nilai balanced accuracy tertinggi dengan 66.05%.

Kata kunci: *sentiment analysis, machine learning, media sosial, jaringan saraf tiruan*

Vol. 15, No. 2, September 2022, P-ISSN 1978-9262, E-ISSN 2655-5018
DOI: https://doi.org/10.33322/petir.v15i2.1197

1. PENDAHULUAN

Setiap konsumen memiliki arti atau definisi masing - masing terhadap produk atau merek yang mereka konsumsi atau gunakan, dimana persepsi ini berhubungan langsung dengan memori dari konsumen terhadap produk atau brand tersebut. Persepsi ini disebut juga sebagai *brand image* [1]. *Brand image* merupakan bagian penting dari *brand equity*, yaitu persepsi konsumen terhadap merek tersebut. Apabila sebuah merek dapat mengontrol persepsi dan sikap konsumen mengenai merek mereka, maka secara tidak langsung merek tersebut dapat mempengaruhi konsumen tersebut untuk membeli produk yang dibuat oleh merek tersebut, dan meningkatkan penjualan dari perusahaan yang bersangkutan [2].

Konsep *brand image* tidak hanya diterapkan pada untuk meningkatkan kinerja perusahaan di sektor bisnis. Beberapa lembaga yang bergerak di sektor publik, seperti kepolisian kota London (*London metropolitan police*), Gereja Katolik Roma, hingga beberapa universitas juga mulai melakukan kegiatan *branding* [3]. Kegiatan *branding* yang dilakukan oleh universitas, atau disebut juga *University branding*, adalah kegiatan yang dilakukan untuk memberikan identitas untuk universitas dan alat untuk menyampaikan aspek - aspek yang dimiliki oleh universitas, seperti kualitas pendidikan, sejarah, dan budaya kepada publik [4]. layaknya kegiatan *branding* yang dilakukan perusahaan untuk meningkatkan penjualan, *university branding* dilakukan oleh universitas mendapat keunggulan kompetitif dari universitas lainnya. Keunggulan ini nantinya dapat menjadi pertimbangan oleh para calon mahasiswa yang sedang mencari universitas untuk melanjutkan pendidikannya.

Selain melakukan *branding*, *brand monitoring*, yaitu pemantauan mengenai apa yang diperbincangkan oleh orang - orang mengenai instansi tersebut, juga perlu dilakukan untuk mengetahui apakah *brand image* instansi sesuai dengan target *branding* yang dilaksanakan. Universitas yang membanggakan kualitas mahasiswa yang dimiliki tentu tidak ingin melihat banyak orang membicarakan mengenai perilaku mahasiswa yang tidak sesuai dengan aturan atau moral yang ada, yang dapat berujung pada rusaknya *brand image* universitas itu sendiri.

Brand monitoring dapat dilakukan melalui media sosial. Jumlah pengguna media sosial di seluruh dunia pada tahun 2019 mencapai 3,484 miliar orang, atau sekitar 45% dari keseluruhan populasi [5]. Setiap harinya, pengguna Twitter membuat 500 juta *tweet*, atau sekitar 6000 *tweet* per detik [5]. Dengan memanfaatkan media sosial, instansi dapat menjangkau banyak orang sekaligus. Namun banyaknya data yang ada di media sosial berarti kegiatan *brand monitoring* di media sosial membutuhkan sumber daya yang tidak sedikit. Hal ini dapat diatasi dengan menggunakan teknologi, yaitu *machine learning* [6]. Teknologi *machine learning* memungkinkan komputer untuk mempelajari tugas baru tanpa perlu diprogram secara detail, dalam artian komputer dapat menentukan sendiri bagaimana tugas tersebut akan dilakukan. Salah satu contoh penerapan dari *machine learning* adalah model *sentiment analysis* yang dapat membaca mengklasifikasikan kalimat berdasarkan polaritasnya. *Sentiment Analysis* adalah studi komputasional terhadap opini, sentimen, maupun emosi yang tertulis di dalam teks [7]. Beberapa ahli menyebutkan bahwa *sentiment analysis* berbeda dengan *opinion mining*, dimana *opinion mining* berfungsi untuk menganalisis dan mengekstraksi opini penulis terhadap sebuah objek, sedangkan *sentiment analysis* digunakan untuk mengidentifikasi dan menganalisis sentimen yang ada pada sebuah teks [8]. Terdapat beberapa pendekatan untuk mengimplementasikan model *sentiment analysis* seperti *Long Short Term Memory* (LSTM) [9], *Convolutional Neural Network* (CNN) [10], *Naive Bayes*, *Linear Regression*, *Support Vector Machines* (SVM), dan *Deep Learning*. Beberapa penelitian sebelumnya melakukan *sentiment analysis* terhadap teks berbahasa Indonesia dengan menggunakan pendekatan *logistic regression*, *support vector machine*, dan *random forest* [11] serta *naive-bayes*, SVM dan *maximum entropy* [12].

Vol. 15, No. 2, September 2022, P-ISSN 1978-9262, E-ISSN 2655-5018
DOI: https://doi.org/10.33322/petir.v15i2.1197

Penelitian ini mencoba pendekatan *neural network* [13,14] (menggunakan arsitektur LSTM, CNN, dan gabungan keduanya) untuk menginterpretasikan tulisan di sosial media menjadi tulisan yang positif, negatif atau netral. Penelitian bertujuan membandingkan model-model tersebut dan mencari model manakah yang memiliki performa paling baik untuk digunakan berdasarkan nilai akurasinya dalam melakukan *sentiment analysis.*

2. METODE PENELITIAN

Penelitian ini mengikuti tahapan yang dijelaskan pada gambar Gambar 1 yang terdiri dari pengumpulan data yang digunakan, pra-pengolahan data, pemodelan yang dibangun dengan menggunakan teknik *deep learning*, serta validasi dan pengujian terhadap model yang dibuat.

Gambar 1. Metodologi Penelitian

2.1. Pengumpulan Data

Data yang dibutuhkan pada tahap ini adalah tweet yang membahas Universitas Pertamina. Cuitan yang dipakai adalah yang mengandung kata "Universitas Pertamina", "Univ Pertamina", atau cuitan yang dikirimkan kepada akun @UnivPertamina. Cuitan yang digunakan dibuat pada tanggal 10 April 2016 hingga 4 Mei 2020. Pengambilan data dilakukan dengan menggunakan library GetOldTweets3[1] dari Python, dimana library ini dapat digunakan untuk mengambil cuitan dan menyimpan informasi dari cuitan tersebut seperti nama user, isi cuitan, hingga tanggal cuitan tersebut dibuat [15]. Jenis *sentiment analysis* yang dilakukan adalah *aspect-based sentiment analysis*. Setiap cuitan akan diberikan label sesuai dengan sentimen yang terkandung dari teks tersebut, dengan label bernilai 0 untuk sentimen negatif, 1 untuk sentimen netral, dan 2 untuk sentimen positif. Proses pemberian label dilakukan secara manual, dimana pemberian label sentimen tidak memperhitungkan kalimat dengan makna sindiran dan/atau ironi.

2.2. Pra-Pengolahan Data

Data teks yang telah diperoleh kemudian diolah agar data tersebut dapat diproses oleh model *machine learning*. Pengolahan data juga dilakukan untuk mengurangi dimensi dari data itu sendiri, sehingga model dapat memproses data lebih cepat. Pengurangan dimensi dilakukan dengan menghapus bagian teks yang dinilai tidak berpengaruh dalam proses *sentiment analysis*.

Tahapan pra-pengolahan data yang dilakukan pada penelitian ini adalah sebagai berikut:

1. *Data cleaning*, yaitu menghapus bagian - bagian teks yang tidak diperlukan untuk sentiment analysis seperti username (diawali dengan '@'), topic (diawali dengan '#'), alamat email, alamat situs, angka, spasi yang berlebihan, tanda baca, dan mengubah semua huruf menjadi huruf kecil.
2. Normalisasi, proses untuk merubah kata - kata dengan ejaan yang tidak baku dalam Bahasa Indonesia menjadi kata baku. Proses ini dilakukan juga untuk mengurangi dimensi dari data
3. *Stemming*, proses untuk mengubah kata berimbuhan menjadi kata dasar.
4. *Stopword removal*. *Stopword* adalah kata - kata yang umumnya sering muncul pada, contohnya kata - kata penghubung seperti "dan", "atau". "di". Kata - kata ini perlu dihapus

Vol. 15, No. 2, September 2022, P-ISSN 1978-9262, E-ISSN 2655-5018
DOI: https://doi.org/10.33322/petir.v15i2.1197

karena pada umumnya kata-kata ini tidak memiliki informasi tambahan untuk proses klasifikasi. Daftar kata *stopwords* berasal dari library Sastrawi[2]

5. Menghapus cuitan yang sama (duplikat).
6. *One-hot encoding*, proses mengubah data kategorikal kedalam bentuk kolom untuk diproses oleh model *machine learning*.

2.3. Pemodelan

Pada penelitian ini ada tiga arsitektur *neural network* yang dibandingkan, yaitu LSTM, CNN, dan CNN-LSTM. Model LSTM dipilih karena model ini umumnya digunakan untuk proses klasifikasi teks dikarenakan kemampuannya untuk mengolah deretan kata [16]. Sementara model CNN dipilih karena walaupun lebih umum digunakan untuk proses klasifikasi gambar, model CNN dapat menghasilkan performa yang baik dalam klasifikasi teks [15]. Sementara model CNN-LSTM merupakan gabungan dari kedua model tersebut dan diketahui dapat menghasilkan performa yang lebih baik dari model CNN maupun model LSTM [17].

Setiap model menggunakan *embedding layer* sebagai *input layer*, dimana masukan cuitan yang diterima akan diproses dengan metode *word embedding*. Selain itu, metode *one-hot encoding* juga digunakan untuk memproses kolom label, sehingga nantinya akan ada 3 kolom label yaitu "negatif", "netral", dan "positif".

Karena label yang digunakan sebagai *output* bersifat kategorikal, dimana ada lebih dari 2 kemungkinan *output*, maka *loss function* yang digunakan pada proses training adalah *categorical crossentropy*. Proses training (*epoch*) dilakukan sebanyak 200 kali, dimana angka 200 merupakan angka yang dipilih secara acak, sebanyak 20 kali untuk mengetahui jumlah *epoch* maksimal ketika nilai *validation/test loss* tidak menurun (N). Selama 20 percobaan, jumlah *epoch* ditambahkan sebanyak 30 jika nilai *validation/test loss* tidak pernah naik untuk mengantisipasi *underfitting*. Pada percobaan selanjutnya digunakan fungsi *early stopping* pada parameter *validation loss*, dimana apabila nilai *validation loss* tidak mengalami penurunan dalam N *epoch* (nilai yang telah didapatkan dari percobaan sebelumnya), maka proses training akan dihentikan. Jumlah *epoch* yang digunakan pada percobaan ke-21 dan selanjutnya adalah jumlah maksimal *epoch* yang didapatkan dari 10 percobaan selanjutnya karena N bisa terjadi pada epoch yang berbeda pada setiap percobaan. Setiap model kemudian akan di-*training* dengan menggunakan teknik *stratified k-fold cross validation* untuk melihat akurasi dan *loss* dari dataset *training* dan *validation*. *Stratified k-fold* dipilih untuk memastikan perbandingan antar kelas pada setiap bagian *fold* tetap sama [18].

2.4. Evaluasi

Performa setiap model yang kemudian akan dievaluasi dengan mengukur *accuracy*, *balanced accuracy*, dan *weighted f1* [19] yang didapat dengan menggunakan validasi *stratified k-fold* serta perbandingan *learning curve* [20,21] antar model. Jika dataset yang digunakan tidak berimbang, dimana ada satu kelas yang jumlah datanya jauh lebih banyak dari kelas lainnya, *accuracy* kurang tepat untuk dijadikan pengukur karena dapat menyebabkan kesalahan dalam interpretasi [15]. Untuk mengantisipasi hal itu, digunakan parameter tambahan dalam penelitian ini, yaitu *weighted f1-score*, dan *balanced accuracy*. *Accuracy*, *balanced accuracy*, dan *weighted f1* memiliki nilai terendah 0 dan nilai tertinggi 1. *Learning curve* digunakan untuk melihat apakah model mengalami *overfit* [16] maupun *underfit* [16] serta memberikan informasi apabila dataset yang digunakan tidak representatif.

Vol. 15, No. 2, September 2022, P-ISSN 1978-9262, E-ISSN 2655-5018
DOI: https://doi.org/10.33322/petir.v15i2.1197

3. HASIL DAN PEMBAHASAN

Dataset yang digunakan berjumlah 4323 cuitan. Setiap cuitan rata - rata memiliki panjang 114 karakter dan 16 kata. Secara keseluruhan dataset terdiri dari 70.348 kata yang tersusun dari 15.522 kata yang berbeda.

Proses pemberian label pada tweet dilakukan dengan pertimbangan [22] sebagai berikut:

- Negatif jika cuitan berisi kritik, ketidakpuasan, kegagalan, mempertanyakan validasi/kompetensi.
- Positif jika cuitan berupa dukungan, kekaguman, pujian, informasi kesuksesan.
- Netral jika cuitan tidak mengandung kedua hal diatas.

Contoh cuitan dan label yang diberikan ditunjukkan pada Tabel 1. Dengan menggunakan pertimbangan tersebut, didapatkan cuitan dengan label netral mendominasi sebanyak 3169 (73.3%), sementara cuitan dengan label negatif berjumlah 638 (14.8%), dan cuitan dengan label positif berjumlah 516 (11.9%).

Tabel 1. Contoh cuitan dan label yang diberikan

Isi Cuitan	Sentimen
Lebih mahal univ pertamina.ggg	Negatif
@UnivPertamina kaa untuk pendaftaran ulang bagi maba up kapan yaa?	Netral
Dosen Universitas Pertamina Terima Hibah Penelitian Rp 345 Juta http://po.st/0ab5lP via @po_st @UnivPertamina	Positif

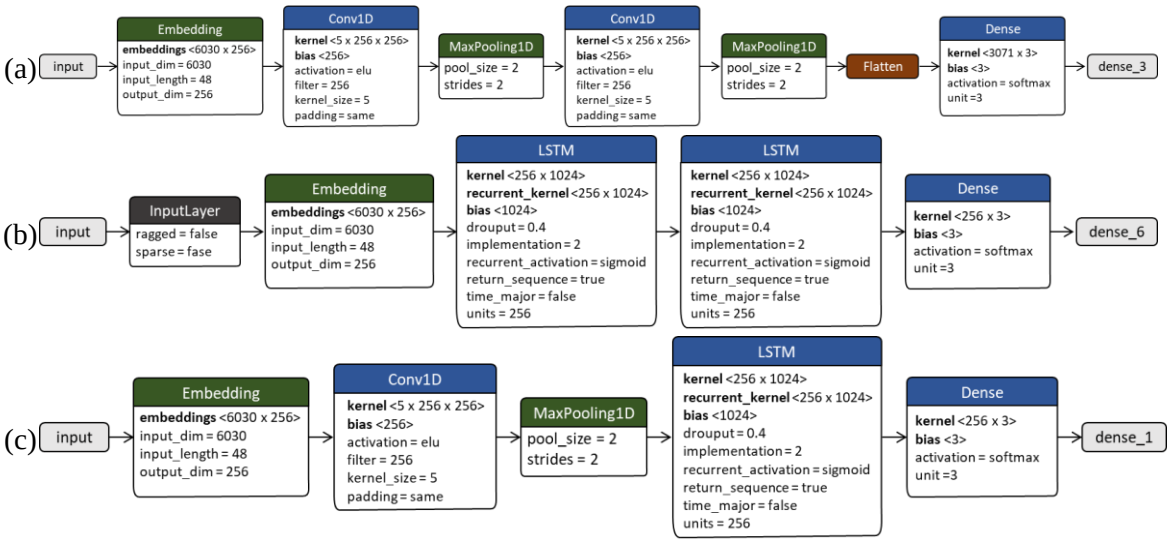
Pra-pengolahan data dilakukan terhadap data yang sudah diberi label. Data yang sudah diberi label kemudian dilakukan proses *cleaning* menggunakan *regular expression*, normalisasi menggunakan *regular expression* dan *lexicon* Bahasa Indonesia sehari - hari [23], *stemming*, dan *stop removal*. Contoh proses ini ditunjukkan ada Tabel 2. Setelah tahapan pra-pengolagan data, jumlag dataset berkurang menjadi 4.269 cuitan, dengan pembagian 634 label negatif, 3.120 data dengan label netral, dan 515 data memiliki label positif.

Tabel 2. Ilustrasi Tahapan Pra-Pengolahan Data

Tahapan	Contoh Cuitan
-	Penjual: Keluhkan Sistem Pembayaran Digital di Kantin Univ Pertamina
Cleaning	penjual keluhkan sistem pembayaran digital di kantin univ pertamina
Normalisasi	penjual keluhkan sistem pembayaran digital di kantin universitas pertamina
Stemming	jual keluh sistem bayar digital di kantin universitas pertamina
Stopword Removal	jual keluh sistem bayar digital kantin universitas pertamina

Karena ketidakseimbangan pada jumlah data di masing - masing label, setiap model akan menggunakan parameter *class weight* pada masing - masing label. Nilai *class weight* ditentukan berdasarkan jumlah data pada setiap label. Perhitungan nilai *class weight* dilakukan dengan memanfaatkan fungsi `compute_class_weight` dari library `Scikit Learn`. Hasil *class weight* untuk masing-masing kelas adalah negatif 2.244, netral 0.456, dan positif 2.763. Semakin tinggi nilai *class weight* menunjukkan semakin sedikit jumlah data pada label tersebut. Setiap model menggunakan *embedding layer* untuk memproses masukan berupa cuitan yang telah diproses dengan teknik *word embedding*. Semua model dijalankan dengan *optimizer* RMSProp, *epoch* 200, *batch size* 64, dan *class weight* sesuai dengan nilai *class weight* yang telah didapatkan sebelumnya. Proses *training* dilakukan dengan *stratified k-fold cross validation*, dimana nilai K yang digunakan adalah

5, yang merupakan nilai *default* dari fungsi. Pada setiap *fold*, data yang digunakan untuk *training* dan *validation* memiliki perbandingan jumlah 8 : 2, dimana 80% data di setiap *fold* digunakan untuk proses *training*, sedangkan 20% data digunakan untuk proses *validation*.



Gambar 2. Arsitektur Model (a) CNN (b) LSTM (c) CNN-LSTM

Model CNN yang ditunjukkan pada Gambar 2 (a) menggunakan *embedding layer* sebagai *input layer*. Model ini menggunakan 2 *layer* Conv1D dan 2 *layer* MaxPooling1D untuk memproses *input* yang diterima oleh *input layer*. Setiap *layer* Conv1D terdiri dari 256 *hidden node.* Model LSTM pada Gambar 2 (b) terdiri dari 2 *layer* LSTM dengan 256 *node* pada setiap *layer*. Untuk mencegah *overfit*, digunakan *dropout* pada *layer* LSTM dengan nilai 0.4. Model CNN-LSTM menggabungkan model CNN dan LSTM seperti pada Gambar 2 (c).

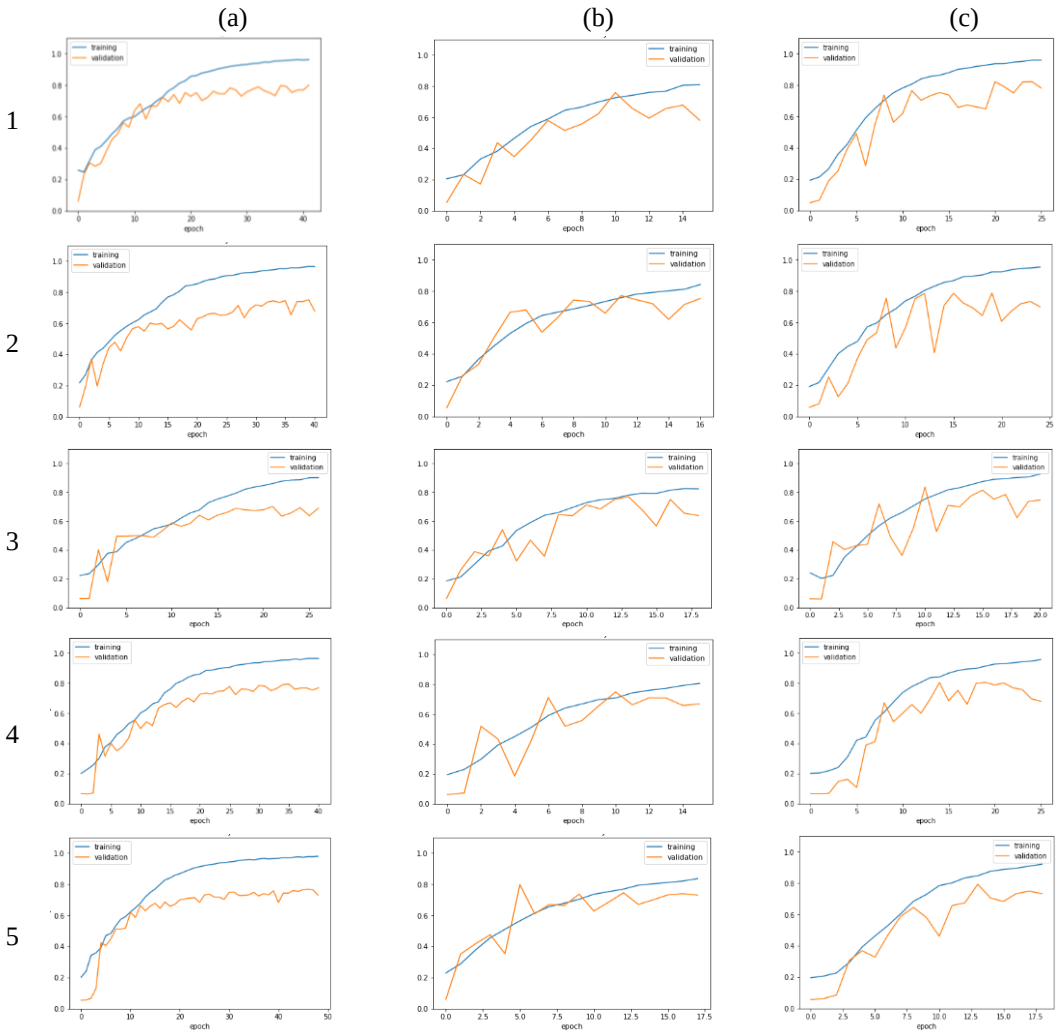
Learning curve pada Gambar 3 dan 4 menunjukkan setiap proses *training* dengan 5-fold cross validation. *Early stopping* (proses *training* dihentikan lebih awal sebelum nilai epoch mencapai 200) terjadi baik pada semua model karenakan performa model tidak mengalami peningkatan setelah melewati 5 epoch. Pada Tabel 3 dapat dilihat bahwa rata - rata proses *training* dihentikan setelah melewati 40 epoch pada model CNN, 17 epoch pada model LSTM dan 23 epoch pada model CNN-LSTM.

Learning curve yang menggambarkan *loss* pada Gambar 4 menunjukkan perbedaan yang cukup besar antara kurva *training* dan *validation* untuk ketiga model, lebih detail pada Tabel 3 dapat dilihat perbedaan sebesar rata-rata perbedaan sebesar 0,57978 antara *training loss* dan *validation loss* untuk model CNN dan 0,39742 untuk model LSTM dan 0,71924 untuk model CNN-LSTM. Hal ini menunjukkan bahwa dataset yang digunakan dalam proses *training* kurang representatif untuk masalah yang ingin diselesaikan. Tidak seperti *learning curve* pada model CNN dan model LSTM, *learning curve* untuk *validation accuracy* maupun *validation loss* pada model CNN-LSTM terlihat lebih fluktuatif dibandingkan dengan *learning curve* pada model CNN maupun LSTM. Hal ini dapat berarti dataset yang digunakan sebagai dataset untuk proses *validation* kurang representatif untuk digunakan pada model CNN-LSTM.

Hasil evaluasi model pada Tabel 4 menunjukkan bahwa aik model CNN, LSTM maupun CNN-LSTM mendapat nilai *balanced accuracy* yang lebih rendah daripada *accuracy*, yang berarti terdapat kelas yang dapat diprediksi lebih baik daripada kelas lainnya di semua model. Hal ini dibuktikan dengan melihat *f1 score* dari masing - masing kelas. Dari nilai *f1 score* pada masing -

Vol. 15, No. 2, September 2022, P-ISSN 1978-9262, E-ISSN 2655-5018
DOI: https://doi.org/10.33322/petir.v15i2.1197

masing kelas, terlihat bahwa performa ketiga model untuk mengenali data dengan kelas netral jauh lebih baik daripada data dengan kelas negatif maupun positif. Hal ini disebabkan karena kelas netral merupakan data mayoritas dengan 73% data pada dataset adalah data dengan kelas netral.



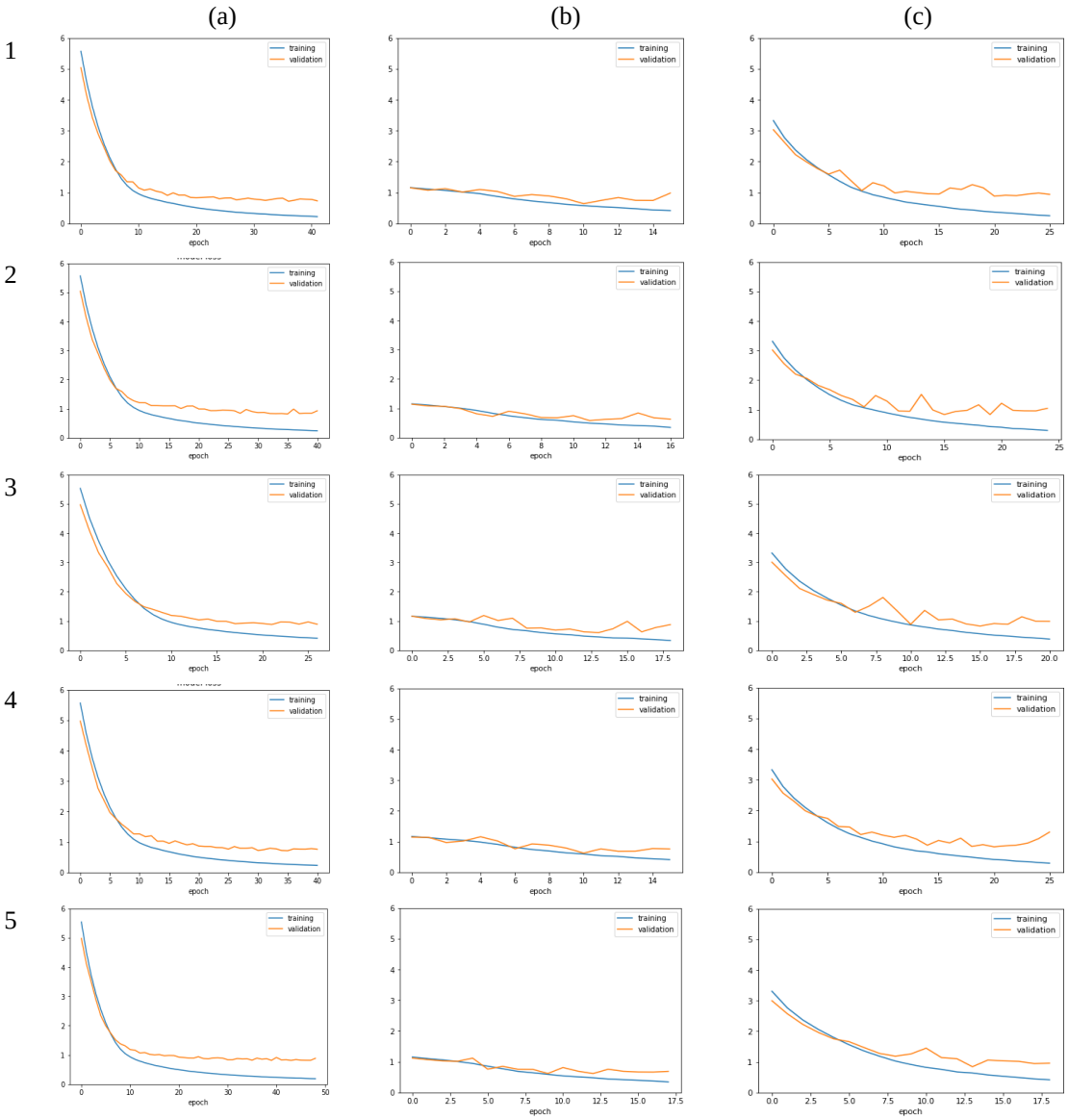
Gambar 3. *Training Accuracy* tiap *round* (a) CNN (b) LSTM (c) CNN-LSTM

Untuk mengantisipasi ketidakseimbangan pada dataset, setiap model telah menggunakan parameter *class weight* untuk memberikan bobot lebih pada data dengan label yang berjumlah sedikit. Namun dapat dilihat pada Tabel 4 performa setiap model pada kelas dengan label yang berjumlah sedikit masih kurang memadai. Dapat dilihat bahwa jumlah data pada masing- masing label sangat mempengaruhi performa model, dimana pada label netral nilai *f1-score* semua model mencapai 83.4%. Banyaknya data pada dataset memungkinkan model untuk lebih mengenali pola yang terdapat pada data dan melakukan generalisasi. Hal itu terlihat pada akurasi label negatif yang rata - rata hanya mencapai f1-score 60% dan label positif yang mendapat f1-score di bawah 45%.

Diantara ketiga model yang digunakan, model CNN-LSTM memiliki nilai *weighted f1* dan *accuracy* yang paling tinggi dan model LSTM memiliki nilai *balance accuracy* paling tinggi. M-LSTM memiliki nilai *f1 score* yang paling tinggi dibanding 2 model lainnya untuk semua kelas. Hal ini menunjukkan CNN-LSTM memiliki performa lebih baik untuk data yang tidak seimbang. Karena

Vol. 15, No. 2, September 2022, P-ISSN 1978-9262, E-ISSN 2655-5018
DOI: https://doi.org/10.33322/petir.v15i2.1197

itu, model CNN-LSTM merupakan model yang paling unggul diantara model lainnya untuk digunakan dalam melakukan analisis sentiment.



Gambar 4. *Training Loss* tiap *round* (a) CNN (b) LSTM (c) CNN-LSTM

Tabel 3. Hasil *training* model CNN dengan *5-fold cross validation*

Model	Evaluation Metrics	1	2	3	4	5	Mean
CNN	Epoch	42	41	27	41	49	40
	Training loss	0.2203	0.2416	0.4095	0.2300	0.1872	0.25772
	Training Accuracy	0.9619	0.9638	0.9008	0.9641	0.9788	0.95388
	Validation loss	0.7295	0.9270	0.8896	0.7581	0.8833	0.8375
	Validation accuracy	0.7994	0.6794	0.6896	0.7687	0.7295	0.73332

Vol. 15, No. 2, September 2022, P-ISSN 1978-9262, E-ISSN 2655-5018
DOI: https://doi.org/10.33322/petir.v15i2.1197

LSTM	Epoch	16	17	19	16	18	17.2
	Training loss	0.415	0.3481	0.3332	0.4119	0.3442	0.37048
	Training Accuracy	0.8086	0.8419	0.8232	0.8049	0.8353	0.82278
	Validation loss	0.981	0.6322	0.8744	0.7606	0.6863	0.7869
	Validation accuracy	0.5813	0.7526	0.6384	0.6676	0.7295	0.67388
CNN-LSTM	Epoch	26	25	21	26	19	23.4
	Training loss	0.2507	0.3005	0.3804	0.2900	0.4173	0.32778
	Training Accuracy	0.9590	0.9531	0.9253	0.9557	0.9198	0.94258
	Validation loss	0.9400	1.0472	0.9875	1.3021	0.9583	1.04702
	Validation accuracy	0.7804	0.6999	0.7452	0.6794	0.7325	0.72748

Tabel 4. Hasil Evaluasi Model

Model	f1	*Weighted f1*	*Accuracy*	*Balanced Accuracy*
CNN	(+):0,584 (-):0,814 (n):0,416	0,73	0,72	0,63
LSTM	(+):0,572 (-):0,814 (n):0,454	0,73	0,72	0,66
CNN-LSTM	(+):0,644 (-):0,841 (n):0,454	0,77	0,77	0,65

4. KESIMPULAN DAN SARAN

Pengolahan dan pengelompokan data kedalam sentiment tertentu yang dilakukan dengan pendekatan *neural network* menggunakan 3 model berbeda, CNN, LSTM dan CNN-LSTM menunjukkan bahwa model CNN-LSTM unggul dibanding dua model lainnya. Hasil penelitian menunjukkan meskipun sudah menggunakan *class weight* untuk mengatasi ketidakseimbangan pada dataset, hasil training dan validasi setiap model cenderung memiliki performa yang lebih baik untuk data dengan label yang berjumlah banyak. Selain itu dataset yang digunakan juga belum optimal di semua model, dimana *learning curve* di semua model menunjukkan bahwa dataset yang digunakan pada saat proses *training* tidak representatif secara statistik, sehingga model tidak dapat melakukan proses generalisasi data dengan baik. Pada model LSTM dan CNN-LSTM, *learning curve* yang dihasilkan menunjukkan bahwa dataset yang digunakan saat proses *validation* kurang representatif, sehingga kurva yang dihasilkan cenderung fluktuatif. Hal ini dapat diakibatkan karena proses pembelajaran yang dilakukan oleh model *deep learning* seperti CNN, LSTM maupun CNN-LSTM membutuhkan data yang jauh lebih banyak jika dibandingkan dengan model *machine learning* lainnya. Agar model bisa lebih optimal dibutuhkan dataset yang lebih banyak dengan jumlah yang seimbang antar labelnya.

DAFTAR PUSTAKA

[1] S. Gensler, F. Völckner, M. Egger, K. Fischbach, dan D. Schoder, "Listen to Your Customers: Insights into Brand Image Using Online Consumer-Generated Product Reviews," International Journal of Electronic Commerce, vol. 20, no. 1, hlm. 112–141, Agu 2015, doi: 10.1080/10864415.2016.1061792.

Vol. 15, No. 2, September 2022, P-ISSN 1978-9262, E-ISSN 2655-5018
DOI: https://doi.org/10.33322/petir.v15i2.1197

[2] Y. Zhang, "The Impact of Brand Image on Consumer Behavior: A Literature Review," Open Journal of Business and Management, vol. 3, no. 1, hlm. 58–62, 2015, doi: 10.4236/ojbm.2015.31006.

[3] D. Marsh dan P. Fawcett, "Branding, politics and democracy," Policy Studies, vol. 32, no. 5, hlm. 515–530, Sep 2011, doi: 10.1080/01442872.2011.586498.

[4] L. Zhang and S. Zhao, "City branding and the Olympic effect: A case study of Beijing", Cities, vol. 26, no. 5, pp. 245-254, 2009. Available: 10.1016/j.cities.2009.05.002.

[5] S. Lambert, "Number of Social Media Users in 2020: Demographics & Predictions - Financesonline.com", Financesonline.com. [Daring]. Tersedia: https://financesonline.com/number-of-social-media-users/. [Diakses: 23- May- 2020].

[6] T. Mitchell, Machine learning. New York: McGraw Hill, 2017.

[7] B. Liu, "Sentiment Analysis and Subjectivity", Handbook of Natural Language Processing, 2nd ed., 2010.

[8] W. Medhat, A. Hassan, dan H. Korashy, "Sentiment analysis algorithms and applications: A survey," Ain Shams Engineering Journal, vol. 5, no. 4, hlm. 1093–1113, Des 2014, doi: 10.1016/j.asej.2014.04.011.K. Gurney, An Introduction to Neural Networks, 1st ed. Taylor & Francis, Inc., 1997.

[9] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," in Neural Computation, vol. 9, no. 8, pp. 1735-1780, 15 Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.

[10] M. Valueva, N. Nagornov, P. Lyakhov, G. Valuev and N. Chervyakov, "Application of the residue number system to reduce hardware costs of the convolutional neural network implementation", Mathematics and Computers in Simulation, vol. 177, pp. 232-243, 2020. Tersedia: 10.1016/j.matcom.2020.04.031.

[11]M. S. Saputri, R. Mahendra, dan M. Adriani, "Emotion Classification on Indonesian Twitter Dataset," dalam 2018 International Conference on Asian Language Processing (IALP), 2018, doi: 10.1109/ialp.2018.8629262.

[12] B. H. Iswanto dan V. Poerwoto, "Sentiment analysis on Bahasa Indonesia tweets using Unibigram models and machine learning techniques," IOP Conference Series: Materials Science and Engineering, vol. 434, hlm. 12255, Des 2018.

[13] K. Gurney, An Introduction to Neural Networks, 1st ed. Taylor & Francis, Inc., 1997.

[14] W. S. McCulloch dan W. Pitts, "A logical calculus of the ideas immanent in nervous activity," The Bulletin of Mathematical Biophysics, vol. 5, no. 4, hlm. 115–133, Des 1943, doi: 10.1007/bf02478259.

[15] Y. Kim, "Convolutional Neural Networks for Sentence Classification," dalam Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2014, doi: 10.3115/v1/d14-1181.

[16] A. Burkov, The hundred-page machine learning book. 2019.

[17] C. Zhou, C. Sun, Z. Liu, F.C.M. Lau, "A C-LSTM Neural Network for Text Classification", 2015. [Daring]. Tersedia: arXiv:cond-mat/0303149. arXiv:1511.08630v2

[18] G. Varoquaux, L. Buitinck, G. Louppe, O. Grisel, F. Pedregosa and A. Mueller, "Scikit-learn", GetMobile: Mobile Computing and Communications, vol. 19, no. 1, pp. 29-33, 2015. Available: 10.1145/2786984.2786995.

[19] J.S. Akosa, "Predictive Accuracy: A Misleading Performance Measure for Highly Imbalanced

Vol. 15, No. 2, September 2022, P-ISSN 1978-9262, E-ISSN 2655-5018
DOI: https://doi.org/10.33322/petir.v15i2.1197

Data", Proceedings of the SAS Global Forum, 2017.

[20] M. Anzanello and F. Fogliatto, "Learning curve models and applications: Literature review and research directions", International Journal of Industrial Ergonomics, vol. 41, no. 5, pp. 573-583, 2011. Tersedia: 10.1016/j.ergon.2011.05.001.

[21] J. Brownlee, "How to use Learning Curves to Diagnose Machine Learning Model Performance", Machine Learning Mastery, 2019. [Daring]. Tersedia: https://machinelearningmastery.com/learning-curves-for-diagnosing-machine-learning-model-performance/. [Diakses: 04- Jun- 2020].

[22] S. Mohammad, “A Practical Guide to Sentiment Annotation: Challenges and Solutions,” dalam Proceedings of the 7th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, 2016, doi: 10.18653/v1/w16-0429.

[23] N. Aliyah Salsabila, Y. Ardhito Winatmoko, A. Akbar Septiandri, dan A. Jamal, “Colloquial Indonesian Lexicon,” dalam 2018 International Conference on Asian Language Processing (IALP), 2018, doi: 10.1109/ialp.2018.8629151.