407-1116-1-PB (1).pdf

WORD COUNT 4506
TIME SUBMITTED 30-MAR-2023 12:43AM
PAPER ID 98145514

RECURRENT NEURAL NETWORK (RNN) DENGAN *LONG SHORT TERM MEMORY* (LSTM) UNTUK ANALISIS SENTIMEN DATA INSTAGRAM

Rudy Ca[illegible]adi[1], Ariesta Damayanti[2*], Dede Aryadani[3]
[1)]Teknologi Rekayasa Multimedia Poltek Negeri Media Kreatif Jakarta
Jl. Srengseng Sawah, Jagakarsa, Jakarta Selatan
[2,3)]Teknik Informatika STMIK AKAKOM
Jl. Raya Janti 143 Karangjambe Yogyakarta
e-mail: iest_ayanthi@gmail.com[2)]

ABSTRAK

Media sosial merupakan ruang publik baru untuk menyalurkan pendapat dan gagasan. Media sosial seperti Instagram telah dimanfaatkan oleh STMIK AKAKOM Yogyakarta untuk memberikan informasi terkait instansi sekolah tinggi tersebut. Informasi-informasi tersebut nantinya dapat menjadi topik pembicaraan dan hal menarik untuk dibahas oleh masyarakat. Respon atau tanggapan masyarakat terhadap konten Instagram STMIK AKAKOM Yogyakarta tersebut tentunya sangat beragam. Oleh karena itu, peneliti mencoba untuk menganalisa komentar yang membicarakan tentang konten dari Instagram STMIK AKAKOM Yogyakarta.

Analisis sentimen dilakukan dengan menggunakan metode Recurrent Neural Network (RNN) dengan Long Short Term Memory (LSTM). Komentar akan diidentifikasi apakah komentar memiliki sentimen positif, netral, atau negatif. Penelitian ini menggunakan data sebanyak 1.473 data yang diperoleh dari hasil crawling.

Hasil dari penelitian ini adalah sebuah sistem yang dapat mengklasifikasi sentimen. Tingkat akurasi pengujian yang didapatkan sebesar 65% dan tingkat akurasi penerapan yang didapatkan sebesar 79,46%. Beberapa kendala dalam proses analisis sentimen adalah data komentar yang jumlahnya tidak seimbang (imbalanced dataset) sehingga perlu dilakukan beberapa langkah tambahan dalam menyiapkan data untuk pelatihan model.

.

***Kata Kunci**: analisis sentimen, deep learning, RNN, LSTM, instagram*

ABSTRACT

The mechanism for conveying aspirations in STMIK AKAKOM by academicians and how the responses given have not been able to run well. This happens because the media used to convey these aspirations are only through suggestion boxes or by conveying them directly to the parties concerned. The follow-up given by the related parties was felt to still take a long time. In addition, the delivery of information in the STMIMK AKAKOM environment is still done manually (posted on the bulletin board) so that the information is not conveyed quickly and completely because of limited coverage.

Therefore, the application of aspirations and information is needed to accommodate the aspirations and dissemination of information by utilizing Firebase technology. Some features used in making this application are Firebase Firestore to make data updates in real time and the application can be used offline. Firebase Cloud Messaging is used to create push notifications. This application is divided into two parts. The web-based application functions to manage information and aspirations used by the admin of STMIK AKAKOM and the Android-based application functions to receive information and send aspirations to be used by STMIK AKAKOM students.

The results of this study are the application of aspirations and information by utilizing Firebase technology. From the testing that has been done, it is obtained that the application can function as an application to manage student reports and aspirations.

***Keywords**: android, aspirations, cloud messaging, firebase, report*

I. PENDAHULUAN

Perkembangan teknologi internet dan aplikasinya telah berkembang pesat dan memberi dampak di seluruh dunia. Salah satu teknologi yang memberi dampak tersebut adalah layanan media sosial dimana media sosial kini telah menjadi ruang publik baru. Hal ini dianggap karena media sosial dapat digunakan untuk menyalurkan ide, pendapat, gagasan, bahkan perasaan. Selain itu, masyarakat juga bisa memberikan tanggapan sehingga akan tercipta interaksi. Oleh karena itu media sosial banyak dimanfaatkan oleh Perguruan Tinggi untuk

memberikan informasi seputar Perguruan Tinggi tersebut seperti, salah satunya, Sekolah Tinggi Manajemen Informatika dan Komputer AKAKOM Yogyakarta (STMIK AKAKOM Yogyakarta).

Menurut data dari Statista pada bulan Juli 2019, tercatat bahwa Indonesia menempati posisi keempat di dunia dengan jumlah pengguna aktif Instagram sekitar 59 juta orang (statista.com, 2019). Instagram telah dimanfaatkan oleh STMIK AKAKOM Yogyakarta untuk memberikan informasi seputar perguruan tinggi tersebut sejak tahun 2016. Hal ini dikarenakan Instagram merupakan salah satu media sosial dengan pengguna terbanyak di antara beberapa media sosial yang ada (wearesocial.com, 2019). Dengan adanya Instagram, STMIK AKAKOM Yogyakarta dapat mempublikasikan berbagai macam informasi yang terkait dengan instansi sekolah tinggi tersebut. Informasi-informasi tersebut nantinya dapat menjadi topik pembicaraan dan hal menarik untuk dibahas oleh masyarakat. Respon atau tanggapan masyarakat terhadap konten Instagram STMIK AKAKOM Yogyakarta tersebut tentunya sangat beragam. Ada yang memberikan respon positif namun ada juga yang memberikan respon negatif. Analisis sentimen dapat dimanfaatkan untuk mengetahui respon masyarakat terhadap informasi yang diberikan oleh STMIK AKAKOM Yogyakarta melalui Instagram.

Analisis sentimen merupakan proses untuk melakukan klasifikasi apakah sebuah tulisan memiliki emosi positif, netral, atau negatif. Penggunaan secara umum dari teknologi ini adalah untuk menemukan bagaimana perasaan seseorang pada suatu topik yang dibicarakan (lexalytics.com, 2018). Tulisan akan diklasifikasikan ke dalam kelas positif apabila informasi yang disampaikan bersifat baik dan setuju terhadap suatu hal. Sebaliknya, tulisan akan diklasifikasikan ke dalam kelas negatif apabila informasi yang disampaikan tidak baik dan tidak setuju.

Metode Recurrent Neural Network (RNN) dengan Long Short Term Memory (LSTM) merupakan salah satu model deep learning yang dapat digunakan untuk melakukan klasifikasi sentimen. Metode ini dapat memproses data secara sekuensial seperti teks, suara, dan video. Araque (2017) juga melakukan penelitian denga mengimplementasikan metode *Long Short Term Memory* (LSTM) untuk melakukan analisis sentimen pada *tweet* berbahasa Spanyol. Peneliti mengimplementasikan dua tipe fitur yang berbeda, *word embedding* dan *sentiment lexicon values*. Hasil yang didapat mengindikasikan bahwa kombinasi dua fitur ini menambah performa analisis sentimen. Hassan (2017) melakukan penelitian dengan mengimplementasikan metode *Long Short Term Memory* (LSTM) untuk melakukan analisis sentimen pada situs IMDB. Peneliti menggunakan metode *Long Short Term Memory* (LSTM) berbasis word2vec. Penelitian tersebut menghasilkan hasil yang lebih akurat untuk melakukan analisis sentimen.

Irfangi (2019) melakukan penelitian dengan mengimplementasikan metode *Naïve Bayes Classifier* untuk melakukan analisis sentimen terhadap trasportasi online di Indonesia pada media Twitter. Hasil uji akurasi pengujian 109 data, dihasilkan nilai akurasi sebesar 84%. Oni (2019) melakukan penelitian dengan mengimplementasikan metode *Reccurent Neural Network* (RNN) dengan *Long Short Term Memory* (LSTM) untuk melakukan analisis sentimen terhadap tempat wisata di Yogyakarta pada media Twitter. Sistem mampu melakukan klasifikasi dengan tingkat akurasi 95.98% melalui *library* dan 70% melalui *form* klasifikasi. Namun, sistem belum mampu melakukan praproses untuk kata-kata singkatan dan slang.

Septian (2019) melakukan penelitian dengan mengimplementasikan metode *Naïve Bayes Classifier* untuk melakukan analisis sentimen pada media Twitter milik Divisi Humas Polri dimana *tweet* yang akan dianalisis diklasifiasikan menjadi tiga topik: kegiatan polisi, layanan masyarakat dan komentar masyarakat. Hasil akurasi pengujian klasifikasi pada sistem ini adalah 86%.

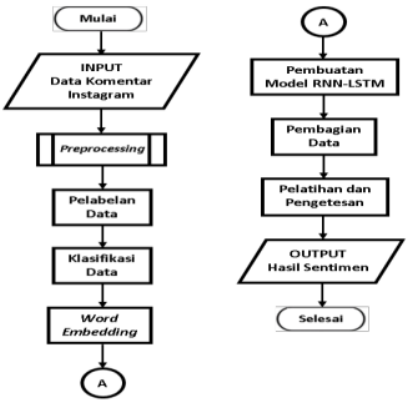
II. METODE

Data yang akan digunakan pada penelitian ini diambil dari Instagram dengan metode Crawler menggunakan instagram-crawler milik Hua-Ying Tsai.

Langkah-langkah dalamproses pengambilan data adalah sebagai berikut.

1. Mengunduh dan memasang chromedriver ke dalam berkas ./inscrawler/bin/chromedriver
2. Memasang Selenium dengan perintah pip install –r requirements.txt
3. Menyalin berkas secret.py dengan perintah cp inscrawler/secret.py dist inscrawler/secret.py
4. Memasukkan kata kunci berupa nama akun Instagram yang akan diambil data komentarnya
5. Selanjutnya, data yang didapat akan disimpan dalam bentuk JSON

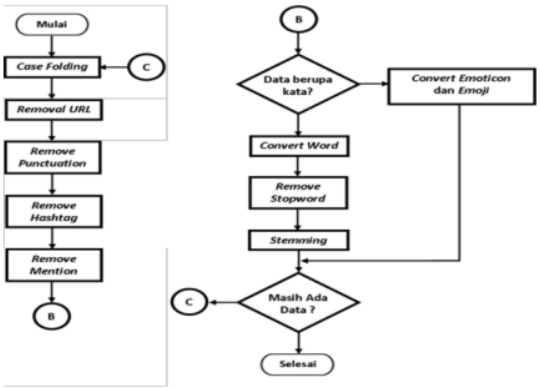
Gambar 1 merupakan gambaran bagaimana sistem bekerja. Data komentar Instagram dari hasil *crawling* yang disimpan dalam *file* JSON akan diproses dalam sistem. Data tersebut akan dimuat oleh sistem untuk dilakukan proses selanjutnya.



Gambar 1. *Flowchart System*

Gambar 3.2 merupakan penjabaran proses pada tahapan *preprocessing*. Tahap *preprocessing* merupakan tahap pembersihan data. Data yang diambil dari Instagram merupakan data kotor yang harus dibersihkan sebelum dilakukan pemrosesan. Tahap *preprocessing* yang akan dilakukan pada penelitian ini adalah sebagai berikut :

a. *Case Folding*, bertujuan membuat semua teks menjadi huruf kecil.
b. *Removal* URL. URL yang terdapat pada data komentar Instagram membuat data tidak efektif dan tidak memiliki arti. Untuk itu perlu adanya penghapusan URL tersebut. Kemunculan alamat web atau URL ini disebabkan karena banyaknya pengguna mempromosikan sebuah produk pada situs mereka sehingga pengguna lain langsung bisa masuk pada halaman web yang dimaksud.
c. *Remove Punctuation*, bertujuan menghapus semua karakter *non alphabet* misalnya simbol, spasi, dan lain – lain.
d. *Remove Hashtag*. *Hashtag* merupakan suatu penunjuk sebuah kata yang dibicarakan oleh sesama pengguna Instagram yang memiliki simbol "#". Biasanya akan digunakan sebagai judul topik pembicaraan dan juga berfungsi sebagai pengelompokan terhadap percakapan yang berhubungan dengan kata yang diberi simbol *hashtag*. Proses ini juga dapat dikategorikan antara penting dan tidak penting, dapat dilakukan ataupun tidak dilakukan proses *Remove Hashtag*.
e. *Remove Mention*. Pada Instagram, simbol "@" digunakan untuk menunjuk atau mengajak teman berkomunikasi langsung. Pada suatu analisis sentimen, nama pengguna tidak diperhatikan sehingga perlu dihapus.
f. *Convert Word*. Saat ini penggunaan bahasa gaul mengakibatkan penggunaan Bahasa Indonesia tidak baku. Sehingga *convert word* diperlukan untuk mengkonversi kata yang tidak baku menjadi kata baku yang sesuai dengan KBBI.
g. *Convert Emoticon* dan *Emoji*. Seringnya ekspresi diungkapkan dengan sebuah gambar (*emoji*) atau simbol karakter *keyboard* (*emoticon*) menyebabkan perlu adanya pengkonversian ke dalam bentuk *string* yang dapat diartikan maknanya.
h. *Remove Stopword*. *Stopword* diproses pada sebuah kalimat jika mengandung kata – kata yang sering keluar dan di anggap tidak penting seperti waktu, penghubung, dan lain sebagainya sehingga perlu dilakukan penghapusan. Untuk melakukan proses penghapusan kata ini diperlukan sebuah data atau daftar kata yang diinginkan untuk dihapus.
i. *Stemming*, digunakan untuk mendapatkan kata dasar dari suatu kata. Hal ini dilakukan untuk menormalisasi kata.



Gambar 2. *Flowchart Preprocessing*

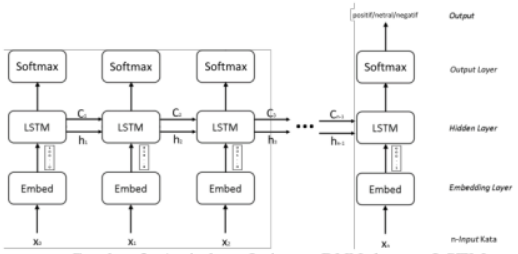
Hasil untuk semua proses pada tahap *preprocessing* yang dilakukan dapat digambarkan pada Tabel I.

Tabel I
Hasil Tahap *Preprocessing*

Proses	Komentar Asli	Hasil Praproses
Case Folding	Mau kemana nih... Seru banget...keluar kampus... Saya sebagai alumni ndak pernah kayak gini...	mau kemana nih... seru banget...keluar kampus... saya sebagai alumni ndak pernah kayak gini...
Removal URL	Tetap semangat ya kak...yuk ikutan pelatihan BLK Surakarta Gratis Lho...! Info : www.blksurakarta.com atau http://bit.ly/wa_blk	Tetap semangat ya kak...yuk ikutan pelatihan BLK Surakarta Gratis Lho...! Info :
Proses	Komentar Asli	Hasil Praproses
Remove Punctuation	Yeay akhimya akakom sosial-isasi di Magelang juga.... Apa sy yg baru tau ya?? hehe.. Biar ada temen dr Magelang... @stmikakakom	Yeay akhirnya akakom sosialisasi di Magelang juga Apa sy yg baru tau ya hehe Biar ada temen dr Magelang @stmikakakom
Remove Hashtag	Alhamdulillah.. semangat terus maju terus dek, moga lebih banyak lagi chalange yang bisa kita jawab. #stmikakakom #stmikakakomyogyakarta	Alhamdulillah.. semangat terus maju terus dek, moga lebih banyak lagi chalange yang bisa kita jawab. stmikakakom stmikakakomyogya-karta
Remove Mention	@zulfakar1998 undang yg lain njung sebelum di lock Kolom komentar nya wkkw	zulfakar1998 undang yg lain njung sebelum di lock Kolom komentar nya wkkw
Convert Word	Yeay akhimya akakom sosial-isasi di Magelang juga.... Apa sy yg baru tau ya?? hehe.. Biar ada temen dr Magelang... @stmikakakom	Yeay akhirnya akakom sosialisasi di Magelang juga.... Apa saya yang baru tau ya?? hehe.. Biar ada temen dari Magelang... @stmikakakom
Convert Emoticon	😍😍😍😍	smiling face with hearteyes smiling face with hearteyes smiling face with hearteyes
Remove Stopword	Alhamdulillah.. semangat terus maju terus dek, moga lebih banyak lagi chalange yang bisa kita jawab. #stmikakakom #stmikakakomyogyakarta	Alhamdulillah.. semangat terus maju terus, moga lebih banyak chalange bisa jawab. #stmikakakom #stmikakakomyogyakarta

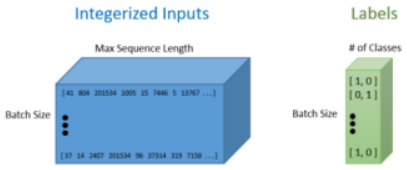
Stemming	Mau kemana nih... Seru banget...keluar kampus... Saya sebagai alumni ndak pernah kayak gini...	Mau mana nih... Seru banget...luar kampus... Saya sebagai alumni tidak pernah kayak gini...

Setelah dilakukan *preprocessing*, langkah selanjutnya adalah pelabelan data. Pelabelan data ini dilakukan untuk membuat data pelatihan. Proses ini akan dilakukan dengan menggunakan *library* Python TextBlob. Setiap data komentar akan diberi label kelas positif, netral atau negatif. Langkah selanjutnya adalah pengklasifikasian data komentar. Data komentar akan diklasifikasi menjadi tiga kelas: Berita, Inspiratif dan IT. *Recurrent Neural Network* tidak dapat melakukan perhitungan pada data yang berbentuk teks. Data teks harus dikonversi ke bentuk vektor. *Layer embedding* merupakan salah satu *layer* yang digunakan dalam pembuatan *word embedding*. *Layer* ini akan mengonversi setiap kalimat dengan hubungan antar katanya ke dalam sebuah matriks.



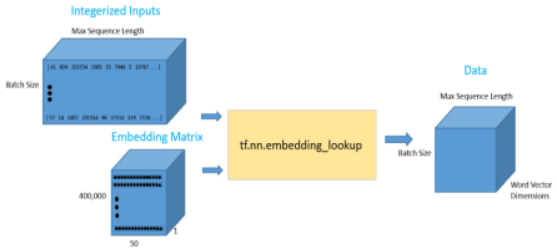
Gambar 3. Arsitektur Jaringan RNN dengan LSTM

Gambar 3 merupakan arsitektur jaringan RNN dengan LSTM yang akan dibuat. Tahap ini merupakan pendefinisian model Keras. Terdapat beberapa parameter yang akan didefinisikan seperti ukuran *batch*, jumlah unit LSTM, jumlah kelas *output* dan nilai maksimal iterasi pelatihan. Hal yang perlu dilakukan adalah mendefinisikan dua *placeholder*. Masing-masing *placeholder* ini berfungsi untuk menampung *input* untuk jaringan dan lainnya untuk menampung label.



Gambar 4. Pembuatan *Batch*

Selanjutnya, fungsi *lookup* akan dilakukan pada data *input* dalam *placeholder* dan vektor hasil proses *embedding*. Fungsi tersebut menghasilkan 3D tensor dengan ukuran sesuai dengan ukuran *batch* dan panjang maksimal dari sekuensial serta dimensi vektor kalimat. Hasil dari *word embeddings* kemudian dimasukkan ke dalam *layer* LSTM, di mana *output* dari *layer* ini diteruskan ke *layer output* dan sel LSTM untuk kata berikutnya di dalam urutan. Jumlah unit LSTM akan ditentukan berdasarkan jumlah kata yang menjadi masukkan pada jaringan ini. Proses ini dapat dilihat pada Gambar 4 dan Gambar 5.



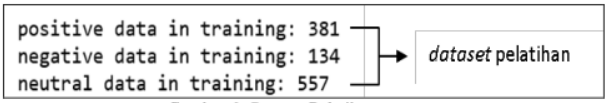
Gambar 5. *Embedding Lookup*

Tahap pembagian data merupakan tahapan untuk membagi *dataset* menjadi data pelatihan dan data pengujian. Dalam tahapan ini akan diterapkan metode *Simple Random Sampling* untuk menentukan data pelatihan dan data pengujian. *Library* Python scikit-learn akan digunakan untuk melakukan *Simple Random Sampling*. Prinsip Pareto akan digunakan untuk menentukan ukuran pembagian *dataset*, yaitu 80% data sebagai data pelatihan dan 20% data sebagai data pengujian. Tahap pelatihan merupakan tahapan untuk melatih jaringan agar dapat mengenali pola. Data pelatihan merupakan bagian dari 80% dari *dataset* penelitian ini. Tujuan utama dari tahap ini adalah mendapatkan model dengan akurasi yang baik.

RNN memiliki kesulitan dalam mempelajari ketergantungan jarak jauh (*long-range dependencies*) yang merupakan interaksi antar kata yang terpisah beberapa bagian. Jaringan akan melupakan kata-kata yang sudah diproses sehingga akan kehilangan konteks tentang apa yang sudah dipelajari. Selain itu, ledakan (*exploding*) dan kelenyapan (*vanishing*) gradien dapat terjadi pada arsitektur ini yang menyebabkan gagalnya proses pelatihan. Peran LSTM dalam hal ini adalah menentukan bagian penting dari t-30 yang akan disimpan dalam sel. Pengujian dilakukan untuk mengetahui keakuratan proses pelatihan. Model yang dihasilkan pada proses pelatihan akan digunakan kembali pada proses ini. Data pengujian merupakan bagian dari 20% dari *dataset* penelitian ini. Proses pengujian akan menghasilkan keakurasian yang dapat digunakan untuk evaluasi proses pelatihan.

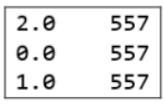
III. HASIL

Data yang telah dimasukkan ke dalam *dataframe* akan dibagi menjadi 3 *dataset*: data_neutral (*dataset* komentar netral, label angka = 1), data_negative (*dataset* komentar negatif, label angka = 0) dan data_positive (*dataset* komentar positif, label angka = 2). Kemudian ketiga *dataset* tersebut akan dimasukkan ke dalam dua *dataset*: dataset train (dengan ukuran data 80% dari masing – masing *dataset*) sebagai dataset pelatihan model LSTM dan dataset test (dengan ukuran data 20% dari masing – masing *dataset*) sebagai *dataset* pengujian model LSTM.



Gambar 6. *Dataset* Pelatihan

Pada Gambar 6 terlihat bahwa ketiga *dataset* pembentuk *dataset* pelatihan (dataset pelatihan komentar positif, negatif dan netral) memiliki jumlah data yang tidak seimbang (*imbalance*). *Dataset* komentar netral termasuk ke dalam kategori ***dataset majority*** (*dataset* dengan jumlah data paling banyak), *dataset* komentar negatif termasuk ke dalam kategori ***dataset minority*** (*dataset* dengan jumlah data paling sedikit) dan *dataset* komentar positif memiliki jumlah data diantara *dataset majority* dan *minority*. Dikarenakan ketidakseimbangan jumlah data ini, maka perlu dilakukan proses *random oversampling* untuk dataset pelatihan komentar positf dan komentar negatif guna menyeimbangkan jumlah data dari ketiga dataset tersebut.***Random Oversampling*** merupakan proses penambahan komentar baru untuk *dataset minority* (dalam penelitian ini yaitu *dataset* pelatihan komentar positif dan negatif) sehingga jumlah data komentarnya sama dengan *dataset majority* (dalam penelitian ini yaitu *dataset* pelatihan komentar netral). Penambahan komentar yang dimaksud di sini adalah dengan menggandakan contoh komentar dari *dataset minority*. Perhitungan bias dan bias1 bertujuan untuk menghitung bobot *dataset* baru untuk *dataset* pelatihan komentar positif dan komentar negatif yang telah melalui proses *random oversampling*, yang akan digunakan saat proses pelatihan model LSTM. Setelah melalui proses *random oversampling*, *dataset minority* akan digabung dengan *dataset majority* dan membentuk data pelatihan baru dengan nama data_upsampled.



Gambar 7. *Dataset* Pelatihan Setelah Proses *Random Oversampling*

Data dari *dataset* pelatihan setelah melalui proses *random oversampling*, yang digunakan untuk pelatihan model LSTM, dibagi dalam bentuk kata dengan *tokenizer*. Token kata ini akan disimpan dan digunakan untuk proses vektorisasi kata, pelatihan dan pengujian agar kata – kata dapat dikenali oleh sistem. Jumlah kata yang dapat disimpan sebanyak 30.000 token.

Setelah melakukan proses *tokenizer*, langkah selanjutnya adalah mengubah token kata ke bentuk vektor agar data dapat diproses oleh sistem. Proses ini memanfaatkan fungsi text_to_sequence dari Keras. Fungsi ini melakukan vektorisasi korpus teks dengan mengubah setiap teks menjadi sekuensi bilangan. Contoh bentuk *one-hot encoded vectors* dapat dilihat pada Pembuatan model pada penelitian ini menggunakan metode sekuensial. Pada model ini, *layer* jaringan didefinisikan satu persatu. Penelitian ini menggunakan beberapa *layer* seperti:

1. *Embedding* : berfungsi untuk mengkonstruksi vektorisasi kata. *Layer* ini didesain untuk dapat mengenali 30.000 token kata (max_fatures=30000), dapat menerima masukan berupa sekuensi kata dengan ukuran maksimal 20 kata (input_length=X_train.shape[1]=20) serta sebagai keluarannya adalah array yang berisi 20 kata dimana setiap kata memiliki ukuran ruang vektor sebesar 20 dimensi (embed_dim=20). Hasil keluaran dari *layer* ini akan seperti pada Lampiran 3.
2. *Spatial Dropout 1D* : berfungsi untuk mencegah *overfitting* pada proses pelatihan. Tabel 4.7 menunjukkan hasil pengujian untuk mengetahui pengaruh nilai *dropout* (nilai probabilitas yang menetukan apakah neuron akan dikeluarkan dari jaringan/tidak diikutsertakan pada setiap tahap pelatihan) terhadap tingkat akurasi pengujian model LSTM yang dibuat dengan jumlah neuron yang akan digunakan pada *hidden layer* LSTM sebanyak 126 neuron pada 10 *epoch* pelatihan.

Tabel II.
Pengujian 1 - Nilai *Dropout* dengan Neuron LSTM = 126 dan *Epoch* = 10

Percobaan ke	Neuron LSTM	*Epoch*	*Dropout*	Akurasi Pelatihan	Akurasi Pengujian
1	126	10	0,1	98,26%	64,04%
2	126	10	0,2	98,62%	67,42%
3	126	10	0,3	96,35%	67,04%
4	126	10	0,4	95,27%	66,67%
5	126	10	0,5	90,48%	54,68%

Berdasarkan Tabel II dengan menggunakan 126 neuron pada *hidden layer* LSTM, 10 *epoch* dan nilai *dropout* sebesar 0,2 (Percobaan ke-2) akan didapat nilai akurasi pengujian optimal sebesar 67,42%. Sehingga nilai *dropout* yang akan digunakan pada *layer Spatial Dropout 1D* dan LSTM adalah 0,2. Hasil *output* dari *layer* ini akan berbentuk sama seperti hasil *output* dari *Embedding layer*.

3. LSTM : merupakan *layer* inti pada penelitian ini. Tabel 4.8 menunjukkan hasil pengujian untuk mengetahui pengaruh jumlah neuron pada *hidden layer* LSTM terhadap tingkat akurasi pengujian model LSTM yang dibuat dengan nilai *dropout* sebesar 0,2 (yang didapat dari Pengujian 1 pada Tabel 4.7) pada 10 *epoch* pelatihan.

Tabel III.
Pengujian 2 - Jumlah Neuron LSTM dengan Nilai *Dropout* = 0,2 & *Epoch* = 10

Percobaan ke	**Neuron LSTM**	***Epoch***	***Dropout***	**Akurasi Pelatihan**	**Akurasi Pengujian**
1	126	10	0,2	97,55%	67,04%
2	144	10	0,2	98,86%	67,42%
3	162	10	0,2	98,62%	67,80%
4	180	10	0,2	98,32%	66,97%
5	198	10	0,2	98,56%	68,55%
6	216	10	0,2	97,61%	67,79%
7	234	10	0,2	98,92%	67,54%
8	252	10	0,2	98,65%	65,05%
9	270	10	0,2	96,54%	66,30%
10	288	10	0,2	97,50%	65,45%

Berdasarkan Tabel III dengan menggunakan 198 neuron pada *hidden layer* LSTM, 10 *epoch* dan nilai *dropout* sebesar 0,2 (Percobaan ke-5) akan didapat nilai akurasi pengujian optimal sebesar 68,55%. Sehingga jumlah neuron yang akan digunakan pada *hidden layer* LSTM sebanyak 198 neuron. Hasil keluaran dari *layer* ini akan seperti pada Lampiran 5.

4. *Dense* : merupakan *layer* yang merepresentasikan jumlah kelas. Pada penelitian ini kelas yang digunakan berjumlah 3 yaitu negatif, netral dan positif. *Layer* ini menggunakan fungsi aktivasi *softmax*.

Setelah didapat nilai optimal untuk parameter *dropout* sebesar 0,2 (Pengujian 1, Tabel 4.7) dan jumlah neuron yang akan digunakan pada *hidden layer* LSTM sebanyak 198 neuron (Pengujian 2, Tabel 4.8), maka pengujian selanjutnya adalah pengujian jumlah *epoch* (*hyperparameter* yang menentukan berapa kali algoritma pembelajaran akan bekerja di seluruh *set* data pelatihan) terhadap tingkat akurasi pengujian model LSTM yang dibuat.

Tabel IV
Pengujian 3 - Jumlah *Epoch* dengan Jumlah Neuron LSTM = 198 & *Dropout* = 0.2

Percobaan ke	Neuron LSTM	*Epoch*	*Dropout*	Akurasi Pelatihan	Akurasi Pengujian
1	198	5	0,2	67,27%	49,06%
2	198	10	0,2	97,37%	68,16%
3	198	15	0,2	97,79%	68,04%
4	198	20	0,2	98,56%	67,65%

Berdasarkan Tabel IV dengan menggunakan 198 neuron pada *hidden layer* LSTM, 10 *epoch* dan nilai *dropout* sebesar 0,2 (Percobaan ke-2) akan didapat nilai akurasi pengujian optimal sebesar 68,16%. Sehingga jumlah *epoch* yang akan digunakan sebanyak 10 *epoch*. Pada proses pelatihan, metode *Adaptive Moment Estimation* (ADAM) digunakan untuk mengoptimasi proses pelatihan. Fungsi ini bekerja dengan meminimalisir *loss* pada setiap pelatihan. Pengimplemantasian metode ini disediakan oleh Keras.

IV. PEMBAHASAN

Tahap pengujian dilakukan untuk mengukur tingkat akurasi, *precision*, *recall* dan *F1-score* model. Penggunaan alat ukur selain tingkat akurasi ini dikarenakan penggunaan *imbalance dataset* untuk melatih model LSTM dapat mempengaruhi tingkat akurasi model saat memprediksi sentimen suatu komentar. Data yang digunakan untuk pengujian pada penelitian ini sebanyak 267 data dengan rincian sebagai berikut.

```
positive data in test: 95
negative data in test: 33
neutral data in test: 139
```
→ *dataset* pengujian

Gambar 8. *Dataset* Pengujian

Berdasarkan hasil dari pengujian tersebut, tingkat keakurasian, *precision*, *recall* dan *F1-score* akan diketahui untuk mengukur seberapa efektif proses pelatihan. *Confusion matrix* juga digunakan untuk mengetahui seberapa besar tingkat akurasi model LSTM yang dibuat. Tabel 4.10 merupakan fungsi pengujian dalam program Python.
Hasil keluaran dari model LSTM pada penelitian ini adalah label sentimen (0/negatif, 1/netral atau 2/positif) dari masukkan berupa komentar Instagram STMIK AKAKOM Yogyakarta yang telah dianalisis sentimen oleh model LSTM. Tabel 4.11 merupakan fungsi pengklasifikasi sentimen dalam program Python.

Pengujian aplikasi dilakukan untuk menguji fungsi-fungsi dari aplikasi yang telah dibuat untuk mencari kesalahan/*bug*. Penelitian ini melakukan pengujian menggunakan teknik *blakbox*. *Blackbox testing* adalah pengujian yang didasarkan pada detail aplikasi seperti tampilan aplikasi, fungsi-fungsi yang ada pada aplikasi, dan kesesuaian alur fungsi dengan bisnis proses yang telah dirancang sebelumnya. Pengujian ini tidak melihat dan menguji kode program. Hasil pengujian dapat dilihat pada tabel IV.

TABEL IV. HASIL PENGUJIAN APLIKASI ANDROID

No	Fungsi yang diuji	Kondisi	*Output* yang diharapkan	*Output* yang dihasilkan	Status pengujian
1	Masuk aplikasi	NIM dan *password* benar	Sukses masuk aplikasi	Sukses masuk aplikasi	Valid
		NIM dan *password* salah maupun kosong	Gagal masuk aplikasi dan menampilkan pesan kesalahan	Gagal masuk aplikasi dan menampilkan pesan kesalahan	Valid
2	Tampil informasi	Pilih menu informasi	Menerima *update* dalam waktu milidetik	Menerima *update* data dalam waktu kurang dari satu detik	Valid
3	Tampil aspirasi	Pilih menu aspirasi	Menerima *update* dalam waktu milidetik	Menerima *update* data dalam waktu kurang dari satu detik	Valid
4	Tambah aspirasi	Teks dan gambar terisi	Bisa terkirim	Bisa terkirim	Valid
		Gambar diisi dan keterangan kosong	Tidak bisa dikirim dan muncul pesan untuk menggisikan keterangan	Tidak bisa dikirim dan muncul pesan untuk menggisikan keterangan	Valid
5	Ubah aspirasi	Belum ada tanggapan	Dapat diubah	Dapat diubah	Valid
		Ada tanggapan	Tidak dapat diubah	Tidak dapat diubah	Valid
6	Tampil tanggapan aspirasi	Buka detail aspirasi	Tampil tanggapan	Tampil tanggapan	Valid

No	Fungsi yang diuji	Kondisi	*Output* yang di-harapkan	*Output* yang dihasilkan	Status pen-ujian
7	*Push Notification*	Terdapat *update* informasi baru	Tampil	Tampil	Valid
		Terdapat tanggapan terhadap aspirasi	Tampil	Tampil	Valid
		Terdapat perubahan progres aspirasi	Tampil	Tampil	Valid
		Pengguna keluar dari aplikasi	Tidak Tampil	Tidak Tampil	Valid
8	Ubah status tampil aspirasi	Pilih menu *Hide*	Tidak tampil pada menu aspirasi	Tidak tampil pada menu aspirasi	Valid
		Pilih menu *Unhide*	Tampil pada menu aspirasi	Tampil pada menu aspirasi	Valid
9	Melihat daftar notifikasi	Pilih menu notifikasi	Tampilkan notifikasi yang belum dibaca	Tampilkan notifikasi yang belum dibaca	Valid
10	Keluar aplikasi	Pilih menu *log out*	Keluar dari aplikasi	Keluar dari aplikasi	Valid
11	Dapat digunakan *offline*	Tidak tersambung internet	Data yang pernah dibuka sebelumnya tetap dapat diakses.	Data yang pernah dibuka sebelumnya tetap dapat diakses.	Valid

Tabel IV menunjukkan bahwa semua hasil pengujian bernilai valid. Berdasarkan hasil pengujian ini maka dapat disimpulkan bahwa aplikasi dapat berjalan baik dan sesuai dengan rancangan sebelumnya. Pembuatan aplikasi aspirasi dan informasi dengan memanfaatkan teknologi Firebase ini menggunakan fitur Firebase Firestore untuk basis data, Firebase Storage digunakan untuk penyimpanan gambar, Firebase Cloud Messaging untuk mengirimkan *push notifications*, Firebase Cloud Functions digunakan untuk mendeteksi setiap ada perubahan pada koleksi dan Firebase Hosting yang digunakan untuk meng-*hosting* aplikasi web yang digunakan oleh admin serta Firebase Authentication yang digunakan untuk autentikasi email dan kata kunci saat masuk di web admin.

Penggunaan Firebase Firestore memungkinkan semua perangkat yang terhubung akan menerima *update* dalam waktu milidetik, sehingga mahasiswa dapat menerima informasi secara *realtime*. Aplikasi dapat dijalankan secara *offline* dan ketika perangkat terhubung dengan internet maka secara otomatis data akan terbaharui.

V. SIMPULAN DAN SARAN

Berdasarkan dari rangkaian proses perancangan hingga implementasi sistem, kesimpulan yang diperoleh antara lain:

1. Pelatihan pada penelitian ini menghasilkan model dengan tingkat akurasi 0,9737 dan *loss* 0,0986 pada 10 *epoch*.
2. Pada saat pengujian, model memiliki tingkat akurasi sebesar 65% dengan nilai rata-rata *macro precision* sebesar 0,59, nilai rata-rata *macro recall* sebesar 0,62 dan nilai rata-rata *macro F1-score* sebesar 0,60; serta nilai rata-rata terbobot *precision* sebesar 0,67, nilai rata-rata terbobot *recall* sebesar 0,65 dan nilai rata-rata terbobot *F1-score* sebesar 0,65.
3. Pada saat penerapan, model memiliki tingkat akurasi sebesar 79,46%.
4. Berdasarkan data dari bulan Mei 2016 sampai dengan Oktober 2019, akun Instagram STMIK AKAKOM Yogyakarta mendapatkan tanggapan netral-positif dengan total komentar netral sebesar 696 komentar (52% dari komentar keseluruhan) dan total komentar positif sebesar 476 komentar (36% dari komentar keseluruhan). Dan sisanya (12% atau sebanyak 167 komentar) merupakan komentar negatif.
5. Konten Berita mendapatkan tanggapan positif paling besar dengan total komentar positif sebanyak 321 komentar. Dengan berfokus pada Konten Berita diharapkan dapat meningkatkan minat masyarakat untuk mengunjungi akun Instagram STMIK AKAKOM Yogyakarta.
6. Sistem belum mampu melakukan klasifikasi komentar *spam*.
7. Perlu menyeimbangkan jumlah data pada setiap komentar kelas positif, netral dan negatif untuk mengatasi masalah *imbalanced dataset*.

REFERENSI

[1] KBBI, "Kamus Besar Bahasa Indonesia (KBBI)", 2016, available at: http://kbbi.web.id/pusat, diakses 7 April 2018.

[2] .BAPPENAS, "Laporan Kajian Manajemen Pengaduan Masyarakat Dalam Pelayanan Publik". Jakarta: Bappenas. 2010.

[3] Riduansyah," Implementasi Google Cloud Messaging Sebagai Media Informasi Berbasis Android (Studi Kasus Informasi Umum Dan Jadwal Kuliah Di STMIK AKAKOM Yogyakarta)", STMIK AKAKOM, Yogyakarta, 2017.

[4] Sumardi, R. Pradipta, "Aplikasi Mobile Notification Informasi Perkuliahan Berbasis Android", STMIK AKAKOM, Yogyakarta.2017.

[5] Prayoga, Febrian, 'Perancangan Prototype Aplikasi Pengumuman Kelas Menggunakan Teknologi Firebase Cloud Message Pada Android", Universitas Stya Wacana, Salatiga, 2016.

[6] Nurzam, Fariz D, dkk, "Rancang Bangun Aplikasi Media Laporan Aspirasi dengan Firebase Cloud Messaging", Seminar Nasional Teknologi Informasi dan Multimedia 2017, 2017,5(1):37-42.

[7] Aminanto, Agus, "Sistem Pengaduan Dan Penyebaran Informasi Tersegmentasi Berbasis Web Dan Android", Universitas Muhammadiyah Yogyakarta, Yogyakarta, 2017

[8] Firebase, https://firebase.google.com/, diakses 7 April 2018

407-1116-1-PB (1).pdf

ORIGINALITY REPORT

24%
SIMILARITY INDEX

PRIMARY SOURCES

1	garuda.ristekbrin.go.id Internet	239 words —	5%
2	fti.uajy.ac.id Internet	136 words —	3%
3	jurnal.amikom.ac.id Internet	85 words —	2%
4	id.123dok.com Internet	79 words —	2%
5	alvindayu.com Internet	67 words —	2%
6	www.semanticscholar.org Internet	55 words —	1%
7	Rafli A. Nugraha, Hilal H Nuha. "Footstep Recognition Using Feedforward Neural Network", 2022 IEEE International Conference on Cybernetics and Computational Intelligence (CyberneticsCom), 2022 Crossref	45 words —	1%
8	ejournal.uin-suska.ac.id Internet	43 words —	1%
9	doku.pub Internet		

		36 words — 1%
10	www.scribd.com Internet	36 words — 1%
11	ojs.uajy.ac.id Internet	31 words — 1%
12	repository.teknokrat.ac.id Internet	26 words — 1%
13	www.researchgate.net Internet	19 words — < 1%
14	repository.bsi.ac.id Internet	18 words — < 1%
15	Oke Dwiraswati, Kemal Nazaruddin Siregar. "ANALISIS SENTIMEN PADA TWITTER TERHADAP PENGGUNAAN ANTIBIOTIK DI INDONESIA DENGAN NAIVE BAYES CLASSIFIER", Media Informasi, 2019 Crossref	12 words — < 1%
16	repository.uin-suska.ac.id Internet	12 words — < 1%
17	www.baka-tsuki.org Internet	12 words — < 1%
18	eprints.utdi.ac.id Internet	11 words — < 1%
19	repository.umy.ac.id Internet	11 words — < 1%
20	repository.unmuha.ac.id Internet	

		11 words — < 1%
21	eprints.uns.ac.id Internet	10 words — < 1%
22	media.neliti.com Internet	10 words — < 1%
23	www.coursehero.com Internet	10 words — < 1%
24	ojs.stmikpringsewu.ac.id Internet	9 words — < 1%
25	text-id.123dok.com Internet	9 words — < 1%
26	etheses.iainponorogo.ac.id Internet	8 words — < 1%
27	repository.uksw.edu Internet	8 words — < 1%
28	www.beritasatu.com Internet	8 words — < 1%
29	zombiedoc.com Internet	8 words — < 1%
30	jurnal.fikom.umi.ac.id Internet	7 words — < 1%

EXCLUDE QUOTES OFF EXCLUDE SOURCES OFF