|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **D:\Software Scribus\konservasi.png** | UJM … (…) 20…  **UNNES Journal of Mathematics**  http://journal.unnes.ac.id/sju/index.php/ujm | | | D:\Logo UJM.jpg |  |
| Analisis Sentimen Aplikasi myUNNES menggunakan metode *Long Short-Term Mermory.*  Aditya Pratama Juliyawan\*, Asika Duri, Fatmawati, Lavina Mutia Dewi  Universitas Negeri Semarang, Indonesia  Sekaran, Kecamatan Gunung Pati, Semarang, 50229 | | | | |  |
| Article Info  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Article History:  Recived …….  Accepted …..  Published …..  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Kata kunci:  Analisis sentimen, Long  Short Term Mermory (LSTM),  myUNNES, | | Abstrak  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Aplikasi myUNNES merupakan aplikasi pendukung akademik yang digunakan oleh mahasiswa Universitas Negeri Semarang. Tingkat kepuasan pengguna aplikasi ini bervariasi, tercermin dari ulasan positif, netral, dan negatif yang tersedia. Penelitian ini bertujuan untuk menganalisis sentimen pengguna aplikasi myUNNES dengan menggunakan metode *Long Short Term Mermory (LSTM)* untuk memberikan masukan dalam pengembangan aplikasi. Metode penelitian ini diperoleh dari pengumpulan data ulasan dari Google Play Store melalui *scrapping,* yang kemudian dilakukan *preprocessing* teks seperti *case folding* dan *punctuation removal*. Model LSTM dilatih dengan menggunakan data yang telah diproses, dan evaluasi dilakukan menggunakan *confusion matrix, precision, recall,* serta *F1-score.* Sentimen pengguna diklasifikasikan ke dalam tiga kategori: positif, netral, dan negatif. Hasilnya menunjukkan akurasi model sebesar 64%, dengan perfoma terbaik pada sentimen netral (*F1-score*: 0,74). Model LSTM cukup efektif untuk menganalisis sentimen pengguna aplikasi myUNNES.  How to cite:  Juliyawan, A. P., Duri, A., Fatmawati. & Dewi, L. M. 2024. Analisis Sentimen Aplikasi myUNNES menggunakan metode *Long Short-Term Mermory*. *UNNES Journal of Mathematics*. 7(2):1-5.  © 2020 Universitas Negeri Semarang | | |  |
| \*Correspondence Address  E-mail: [author@domain.com](mailto:author@domain.com) | | | p-ISSN 2252-6943  e-ISSN 2460-5859 | |  |

1. Pendahuluan

Seiring dengan perkembangan teknologi, banyak media sosial dan aplikasi yang diciptakan untuk memudahkan berbagai kegiatan manusia. Aplikasi adalah suatu program di dalam komputer atau *handphone* yang digunakan untuk menjalankan suatu program yang telah dibuat (Dewi dkk., 2021). Banyak aplikasi yang telah diciptakan untuk mendukung kegiatan komunikasi, hiburan, pemerintahan, dan berbagai aspek lainnya. Di samping itu, juga semakin banyak aplikasi yang dirancang untuk mendukung kegiatan akademik dan pembelajaran, sehingga mempermudah dalam mengakses informasi dunia pendidikan.

Salah satu aplikasi di lingkup akademik adalah myUNNES*,* di mana aplikasi ini digunakan oleh seluruh mahasiswa Universitas Negeri Semarang (UNNES). Aplikasi myUNNESmenyediakan beberapa fitur yang menunjang perkuliahan, seperti Jadwal Kuliah, Kalender Akademik, Akses *Hostpot*, Lapor MBKM, dan Pemindaian Kartu, serta fitur layanan untuk mahasiswa. Namun, setiap mahasiswa memiliki tingkat kepuasan yang bervariasi dalam menggunakan aplikasi myUNNES*,* sehingga cukup banyak saran dan keluhan yang diberikan untuk aplikasi ini.

Analisis sentimen merupakan salah satu solusi yang dapat digunakan untuk mengetahui tingkat kepuasan pengguna (Alghifari dkk., 2022). Dengan menerapkan analisis sentimen, maka dapat mengidentifikasi pola atau tren dalam ulasan penggunaan aplikasi. Analisis sentimen adalah proses mengekstraksi, mengolah data maupun memahami suatu data tekstual secara otomatis atau biasa disebut analisis sentimen merupakan analisis yang biasanya mengolah data opini atau suatu pandangan publik yang nantinya akan menghasilkan kesimpulan dari data yang dianalisis tersebut (Erfina & Putra, 2019; Saepulrohman dkk., 2021). Sehingga dapat memungkinkan untuk mendukung pengembangan aplikasi yang lebih baik lagi sesuai dengan kebutuhan dan harapan pengguna.

Analisis sentimen juga merupakan proses klasifikasi yang bertujuan untuk menentukan apakah suatu teks opini bersifat positif, negatif, atau netral. *Deep Learning* merupakan salah satu metode yang dapat digunakan dalam pengimplementasian analisis sentimen (Alghifari dkk., 2022). *Deep learning* merupakan salah satu bidang *Machine Learning* yang memanfaatkan jaringan saraf tiruan untuk implementasi permasalahan dengan *dataset* yang besar (Arsal dkk., 2020). *Deep learning* merupakan metode yang memanfaatkan *artificial neural network* yang berlapis-lapis (Nugroho dkk., 2020). Struktur berlapis ini memungkinkan model untuk belajar fitur-fitur yang lebih kompleks dari data, sehingga meningkatkan akurasi dalam klasifikasi.

Salah satu metode dalam *deep learning* untuk pengklasifikasian adalah *Long Short-Term Memory (LSTM).* *Long Short-Term Memory networks* atau LSTM adalah salah satu jenis arsitektur RNN yang dikenalkan pertama kali oleh Hochreiter & Schmidhuber pada tahun 1997 (Andiyantama dkk., 2021). LSTM merupakan pengembangan dari metode sebelumnya, yaitu *Recurrent Neural Network (RNN)* (Pardede & Pakpahan, 2023). LSTM menggunakan tiga gerbang yaitu gerbang *input*, gerbang *forget* dan gerbang *output*. Dalam LSTM satu komponen *gate* digunakan saat mengontrol informasi yang masuk ke dalam memori yang bertugas memecahkan masalah dari gradien kehilangan dan pembagian (Hermanto dkk., 2021). Hal ini menunjukkan bahwa LSTM dirancang dengan tantangan yang dihadapi oleh model-model sebelumnya, sehingga meningkatkan stabilitas dan ketepatan dalam pelatihan. LSTM memiliki keunggulan data runtun waktu urutan panjang melalui mekanisme *gate* yang mengatur aliran informasi dan mencegah masalah *vanishing gradient* dan *exploding gradient* (Verianto, 2024). Oleh karena itu, LSTM sangat relevan untuk analisis data yang berurutan, seperti dalam pemrosesan analisis sentimen, di mana konteks dan urutan kata sangat penting.

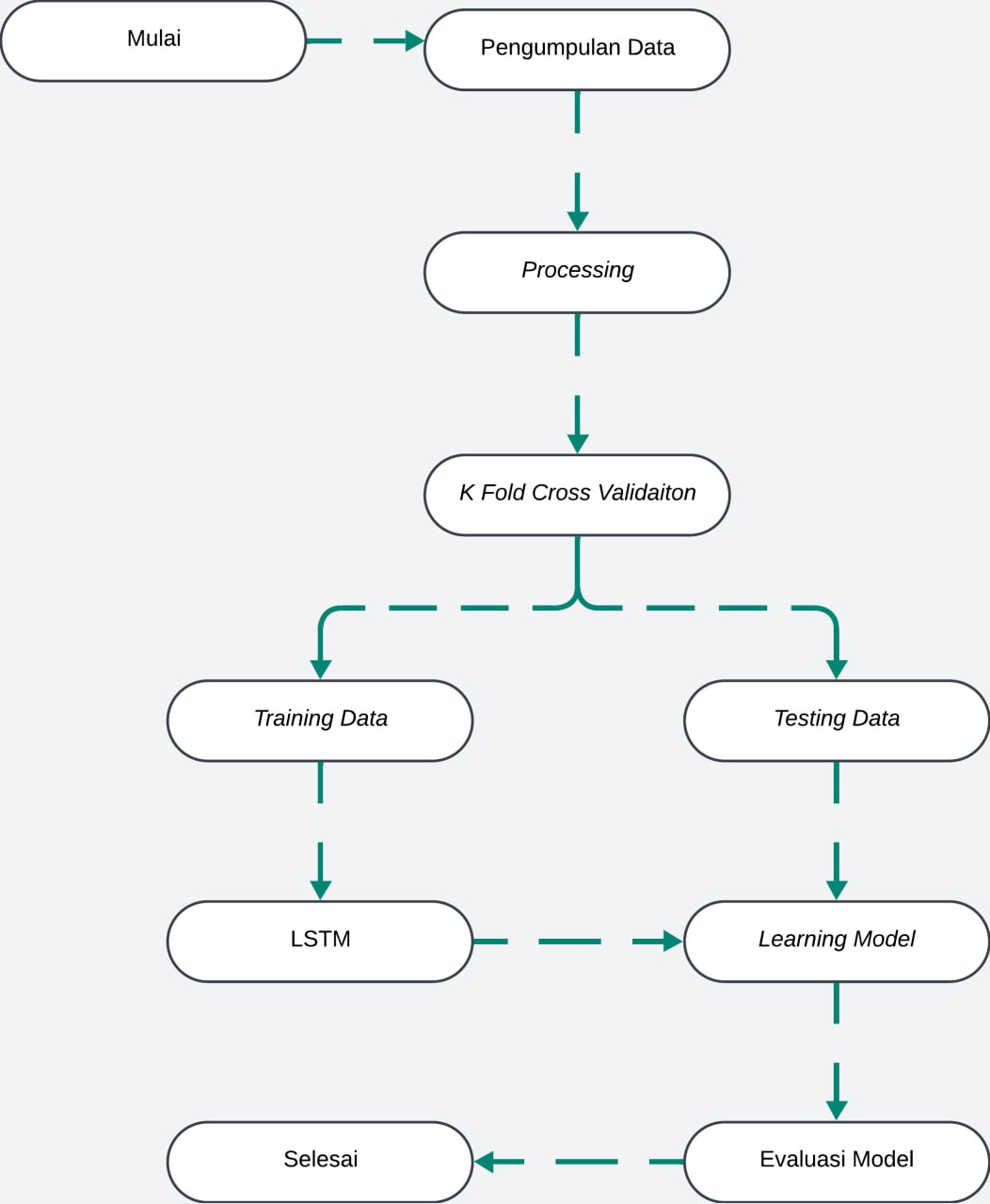
Metode LSTM telah banyak diterapkan pada penelitian sebelumnya, seperti pada penelitian yang dilakukan oleh Amelia & Aminuallah (2023) dalam jurnal “Teks Analisis Sentimen *Chat* Grup Whatsapp Menggunakan *Long Short Term Memory* (LSTM)”. Penelitian ini menganalisis sentimen terhadap salah satu grup Whatsapp untuk dilakukan pemrosesan terhadap isi *chat* dari grup tersebut menggunakan metode LSTM. Di mana hasil penelitian ini didapatkan tingkat akurasi pada uji sebesar 0,76% sedangkan tingkat akurasi pada data *training* yaitu 0,71%, metode ini memiliki hasil akurasi yang lebih baik pada pemrosesan analisis sentimen *chat* grup Whatsapp.

Penelitian lain pernah dilakukan juga oleh Nurvania dkk. (2021), pada jurnal dengan judul “Analisis Sentimen Pada Ulasan di *TripAdvisor* Menggunakan Metode *Long Short-Term Memory* (LSTM)” dengan tujuan mengklasifikasikan ulasan pengunjung tentang pengaruh COVID -19 terhadap tempat wisata di Bali dari *Tripadvisor* dengan LSTM. Penelitian ini menganalisis sentimen yang digunakan dalam paragraf yang berisi kalimat menggunakan bahasa manusia menjadi bahasa komputer. Hasil dari pengujian pada model yang dibangun cukup tinggi yaitu nilai akurasinya sebesar 71,67%.

Oleh karena itu, penelitian ini bertujuan untuk menganalisis sentimen pengguna terhadap aplikasi myUNNES berdasarkan ulasan yang diperoleh dari Google *Play Store* menggunakan metode *Long Short-Term Memory* (LSTM). Hasil analisis sentimen pengguna myUNNES menggunakan LSTM, diharapkan dapat memberikan pengembangan aplikasi berdasarkan data ulasan pengguna.

1. Metode

Dalam penelitian ini, terdapat beberapa tahapan untuk mengklasifikasi teks menggunakan LSTM, yaitu pengumpulan data, *preprocessing, splitting set data,* pelatihan set data, danevaluasi model.



Gambar 1. Diagram Alur Metode Penelitian

* 1. ***Pengumpulan Data***

Pada penelitian ini, data yang digunakan adalah komentar pengguna mengenai aplikasi myUNNES. Dengan mempertimbangkan ulasan yang relevan, data komentar dikumpulkan secara langsung dari platform Google Play Store. Komentar tersebut berupa pengalaman pengguna yang mencangkup beberapa sentimen, seperti positif, netral, dan negatif. Pengumpulan data dilakukan pada tanggal 20 November 2024 melalui *scrapping* pada ulasan yang bersifat publik dan dapat diakses secara umum di Aplikasi *Playstore*. Data yang terkumpul kemudian akan diklasifikasi berdasarkan sentimen (positif, netral, dan negatif) untuk memastikan keakuratan dan relevansi analisis.

* 1. ***Text Preprocessing***

Tahap *text preprocessing* perlu dilakukan sebelum set data diproses ke dalam model klasifikasi menggunakan LSTM. Tahap *Text preprocessing* yang dilakukan adalah *case folding* dan *punctuation removal.*

* + 1. *Case Folding*

Teknik *case folding* adalah pengubahan seluruh karakter pada teks menjadi huruf kecil (*lower case). Case Folding* bertujuan untuk menyeragamkan karakter yang ada pada teks set data sehingga dapat memudahkan dalam proses penghapusan karakter atau istilah-istilah eksklusif yang tidak diinginkan pada penelitian (Pardede & Pakpahan, 2023)*.* Langkah ini dilakukan untuk menghilangkan perbedaan makna yang muncul akibat penggunaan huruf kapital.

|  |  |
| --- | --- |
| Tabel 1. Contoh *Case Folding* | |
| Sebelum *Case Folding* | Sesudah *Case Folding* |
| di Indonesia yg gratis hanya *wifi*, bodo amat pemerintah dg Covid 19 ....lbh pnting ekonomi | di indonesia yg gratis hanya wifi, bodo amat pemerintah dg covid 19 ....lbh pnting ekonomi |

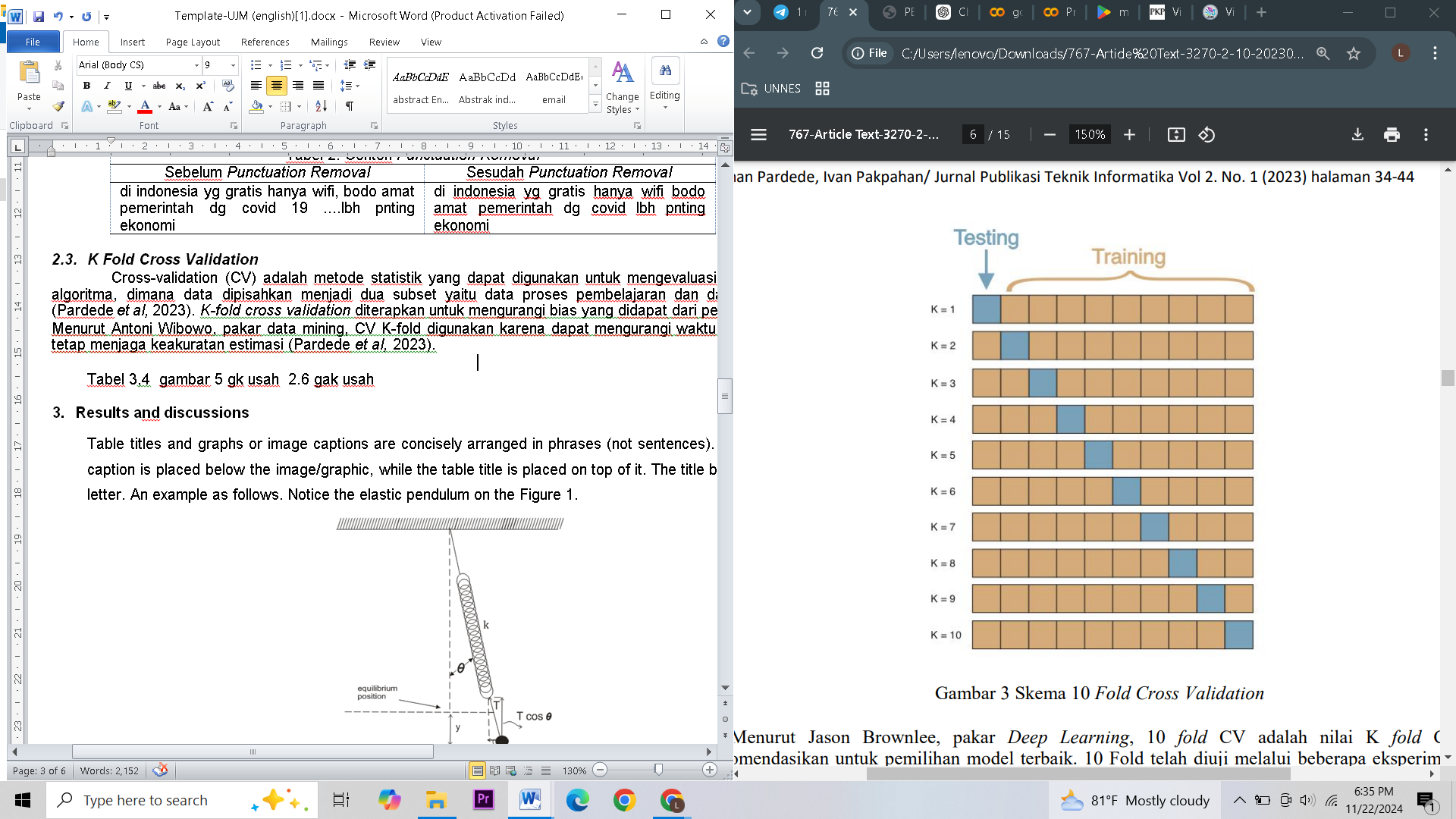
* + 1. *Punctuation Removal*

Teknik *Punctuation Removal* ini digunakan untuk menghapus tanda baca, seperti ! @ # $ % ^ & \* ( ) [ ] { } \ | ; : ’ ” , . < > / ?. Menghapus tanda baca dilakukan karena tanda baca tidak diperhatikan selama proses *training* sehingga dengan melakukan penghapusan tanda baca membuat proses *training* akan menjadi lebih ringan (Pardede & Pakpahan, 2023).

|  |  |
| --- | --- |
| Tabel 2. Contoh *Punctuation Removal* | |
| Sebelum *Punctuation Removal* | Sesudah *Punctuation Removal* |
| di indonesia yg gratis hanya wifi, bodo amat pemerintah dg covid 19 ....lbh pnting ekonomi | di indonesia yg gratis hanya wifi bodo amat pemerintah dg covid lbh pnting ekonomi |

* 1. ***K Fold Cross Validation***

*Cross-validation* (CV) adalah metode statistik yang dapat digunakan untuk mengevaluasi kinerja model atau algoritma, di mana data dipisahkan menjadi dua *subset* yaitu data proses pembelajaran dan data validasi/evaluasi (Pardede & Pakpahan, 2023). *K-fold cross validation* diterapkan untuk mengurangi bias yang didapat dari pengambilan sampel. Menurut Antoni Wibowo, pakar *data* *mining*, CV K-fold digunakan karena dapat mengurangi waktu komputasi dengan tetap menjaga keakuratan estimasi (Pardede & Pakpahan, 2023).



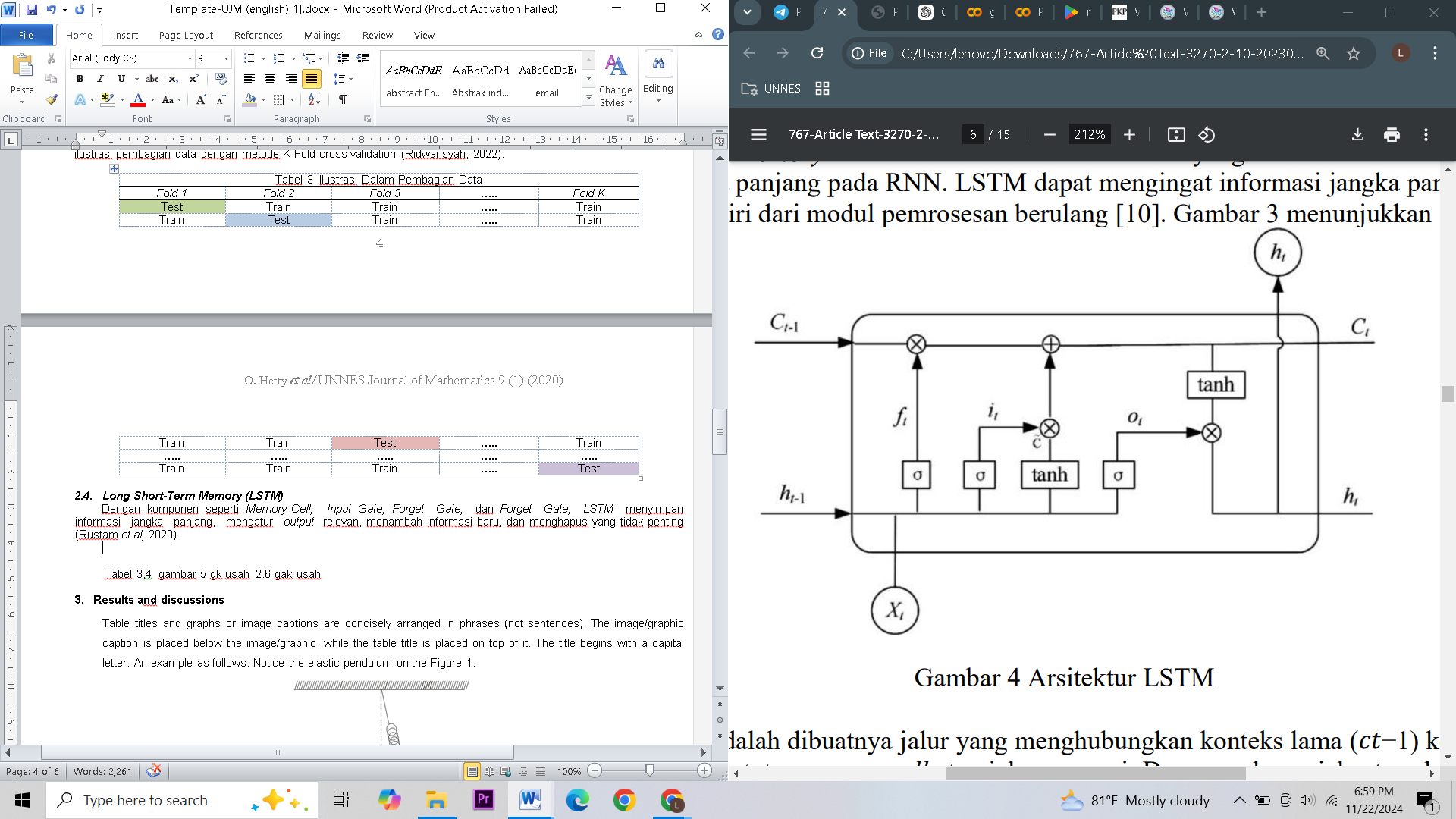
Gambar 2. Skema 10 *Fold Cross Validation*

K yaitu besar angka pemilahan data yang digunakan untuk pembagian *train* dan *test*. Tabel berikut adalah ilustrasi pembagian data dengan metode K-Fold *cross validation* (Ridwansyah, 2022).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tabel 3. Ilustrasi Dalam Pembagian Data | | | | |
| *Fold 1* | *Fold 2* | *Fold 3* | ***…..*** | *Fold K* |
| Test | Train | Train | **…..** | Train |
| Train | Test | Train | **…..** | Train |
| Train | Train | Test | **…..** | Train |
| **…..** | **…..** | **…..** | **…..** | **…..** |
| Train | Train | Train | **…..** | Test |

* 1. ***Long Short-Term Memory (LSTM)***

Dengan komponen seperti *Memory-Cell, Input Gate, Forget Gate,* dan *Forget Gate, LSTM* menyimpan informasi jangka panjang, mengatur *output* relevan, menambah informasi baru, dan menghapus yang tidak penting (Kurniawan & Mustikasari, 2020)



Gambar 3. Arsitektur LSTM

* + 1. ***Forget Gate***

Proses *Forgat Gate* akan menghilangkan informasi yang kurang dibutuhkan atau tidak terlalu memiliki makna terhadap data yang diolah menggunakan fungsi *Singmoid.*

(1)

Di mana:

= *forget gate*

= fungsi *singmoid*

= nilai *weight input* untuk *forget gate*

= nilai *input* pada orde ke t

= nilai *weight output* untuk *forget gate*

= nilai *output/hidden* sebelum orde ke t

= nilai bias pada *forget gate*

* + 1. ***Input Gate***

Pada tahap ini, fungsi aktivasi *Singmoid* digunakan memilah dan menentukan informasi tertentu yang kemudian akan diperbaharui ke *cell state.*

(2)

(3)

Di mana

= *input gate*

= fungsi *Sigmoid*

= nilai *weight input* untuk *input gate*

= nilai *weight output* untuk *input gate*

= nilai *output/hidden* sebelum orde ke t

= nilai *input* pada orde ke t

= nilai bias pada *input gate*

= nilai kandidat baru ke *cell state*

= fungsi Tanh

= nilai *weight input* untuk *cell state*

= nilai *weight output* untuk *cell state*

= nilai bias untuk cell state

* + 1. ***Cell Get***

Perbaharui nilai *cell state* lama (ct – 1) menjadi *cell state* baru (ct)

(4)

Di mana

= *Cell state*

= *forget gate*

= *Cell state* sebelum orde ke t

= *input gate*

= nilai kandidat baru ke *cell state*

* + 1. ***Output Gate***

Kemudian, yang terakhir adalah *output gate*. Dilakukan dengan menjalankan aktivasi *Sigmoid* untuk menghasilkan nilai *output* pada *hidden state* dan menempatkan *cell state* pada Tanh (Pardede & Pakpahan, 2023).

(5)

(6)

Di mana

= *output gate*

= fungsi *singmoid*

= nilai *weight input* untuk *output gate*

= nilai *input* pada orde ke t

= nilai *weight output* untuk *output gate*

= nilai *output/hidden* sebelum orde ke t

= nilai bias pada *output gate*

*=* nilai *output* orde t

= fungsi Tanh

= *Cell state*

* 1. ***Evaluasi Model***

Evaluasi model adalah proses mengukur performa model menggunakan data pengujian untuk menghitung nilai *loss* dan akurasi (Utomo dkk., 2024). Metode yang digunakan untuk mengevaluasi model pada penelitian ini adalah *confusion matrix.* Model ini merepresentasikan hasil klasifikasi menggunakan matriks seperti pada tabel berikut.

|  |  |  |  |
| --- | --- | --- | --- |
| Tabel 4. *Confussion Matrix* | | | |
|  | Prediksi | | |
| Aktual | Positif | Netral | Negatif |
| Positif | TP | FP | FP |
| Netral | FP | TT | FN |
| Negatif | FN | FN | TN |

Penjelasan pada gambar di atas antara lain, TP (*True Positif*), yaitu jumlah ulasan positif yang diklasifikasikan dengan benar sebagai positif. FP (*False Positif*), yaitu ulasan yang seharusnya negatif atau netral tetapi diklasifikasikan salah sebagai positif. TT (*True* *Netral*), yaitu jumlah ulasan netral yang diklasifikasikan dengan benar sebagai netral. FN (*False Netral)*, yaitu ulasan yang seharusnya positif atau netral tetapi diklasifikasikan salah sebagai negatif. Dan Tn (*True Negatif),* yaitu jumlah ulasan negatif yang diklasifikasikan dengan benar sebagai *negative*.

Metode pengujian yang digunakan dalam penelitian ini menggunakan perhitungan yang diformulasikan sebagai berikut

* + 1. *Accurancy*

*Accuranccy* digunakan untuk menghitung seberapa tepat klasifikasi yang telah dibangun sesuai target.

* + 1. *Precision*

*Precision* digunakan untuk menghitung keakuratan antara data target dengan hasil prediksi dari model

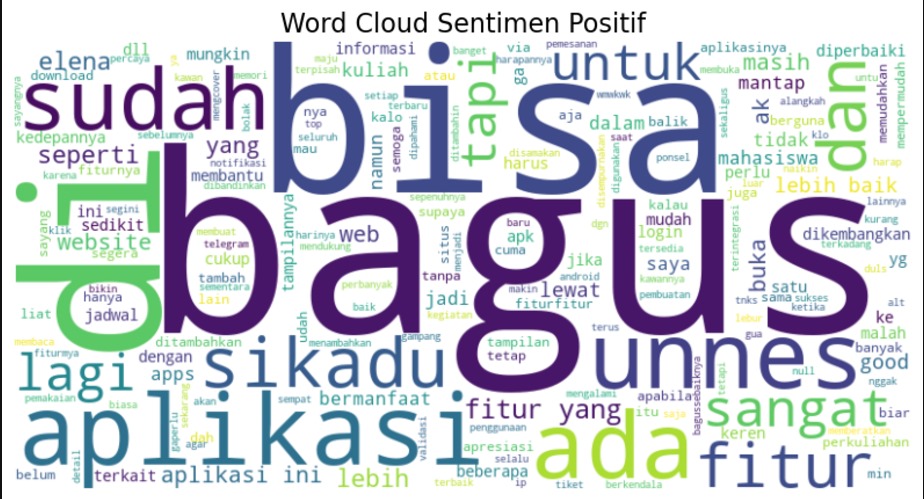
* + 1. *Recall*

*Recall*  digunakan untuk menggambarkan keberhasilan model dalam menentukan kembali informasi.

* + 1. *F Measure*

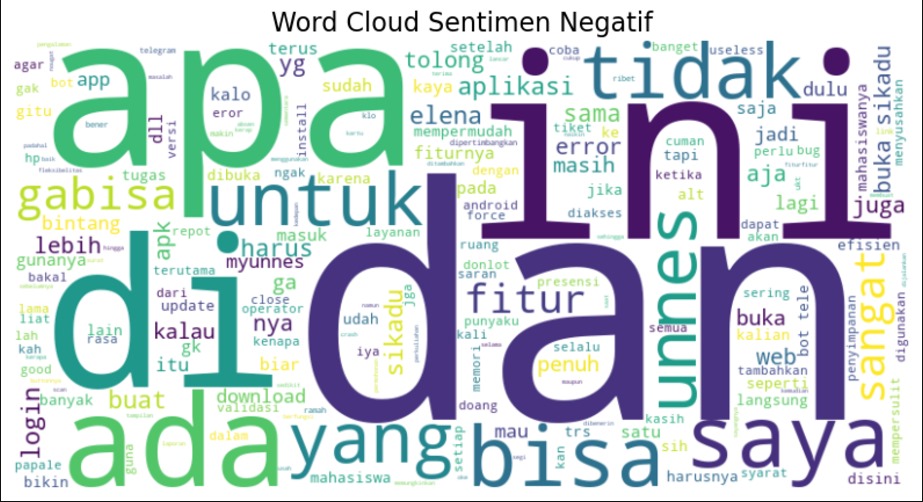
*F Measure* digunakan untuk menggambarkan perbandingan antara *precision* dan *recall.*

1. Hasil dan pembahasan
   1. *Visualisasi Sentimen*



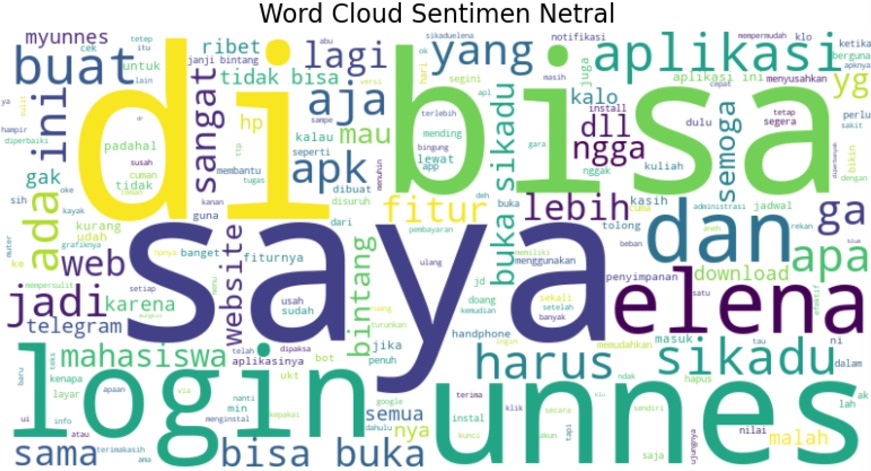
Gambar 4. *Word Cloud* Pada Kelas Positif

Gambar 2 merupakan hasil visualisasi pada kelas Positif, pada hasil visualisasi tersebut menunjukkan adanya kepuasan mahasiswa terhadap kinerja aplikasi myUNNES. Pada visualisasi tersebut terdapat kata bagus, mudah, *good*, keren, dan lain-lain, visualisasi di kelas positif dapat dijadikan acuan untuk pengembang mempertahankan bahkan terus meningkatkan kinerja dari aplikasi myUNNES tersebut, agar para mahasiswa lebih nyaman menggunakannya.

\

Gambar 5. *Word Cloud* Pada Kelas Negatif

Gambar 3 merupakan hasil visualisasi pada kelas negatif. Hasil visualisasi pada kelas negatif menunjukkan adanya ketidakpuasan/kekecewaan mahasiswa terhadap kinerja aplikasi myUNNES. Seperti pada cuitan berikut “Gajelas, gunanya buat apa coba? Bikin penuh penyimpanan doang hadeh. Lain kali tolong lebih efisien ruang. Kasihan org lain yg hpnya ga memadahi buat pasang apk ini kan jd gabisa buka sikadu, elena, dll. Tolong lebih dipikirkan dan dicermati!!!!!!!!”. Visualisasi di kelas negatif ini dapat dijadikan acuan pengembang agar lebih optimal lagi dalam mengembangkan aplikasi.



Gambar 6. *Word Cloud* Pada Kelas Netral

Gambar 4 menunjukkan hasil visualisasi pada kelas netral. Hasil visualisasi ini mencerminkan respons mahasiswa yang tidak menunjukkan perasaan kuat baik positif maupun negatif terhadap kinerja aplikasi myUNNES. Sebagai contoh, terdapat cuitan yang berbunyi, “Aplikasinya sudah bagus tampilannya juga sudah mudah dipahami, tapi fitur yang tersedia sudah ada sebelumnya seperti jadwal sudah ada di telegram. Lebih baik lagi jika menambahkan fitur seperti di Sikadu seperti validasi kuliah itu akan lebih membuat aplikasi ini lebih bermanfaat setiap harinya.”. Meskipun terdapat keluhan, reaksi yang muncul lebih bersifat netral dan bisa dijadikan masukan bagi pengembang untuk mempertimbangkan berbagai faktor dalam pengembangan aplikasi agar lebih baik.

* 1. *Hasil Performa Model LSTM*

Hasil evaluasi kinerja model analisis *sentiment* LSTM yang dilatih dengan *dataset* 213 *reviews,* terhadap data uji yang dilabeli secara manual. Model tersebut mencapai akurasi 64%, untuk lebih jelas akan ditampilkan pada tabel di bawah :

Tabel 5. Performa Model LSTM

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | *Precision* | *Recall* | *F1-Score* | *Support* |
| Negatif | 0.60 | 0.43 | 0.50 | 7 |
| Netral | 0.70 | 0.78 | 0.74 | 18 |
| Positif | 0.50 | 0.50 | 0.50 | 8 |
|  |  |  |  |  |
| Accuracy |  |  | 0.64 | 33 |
| Macro avg | 0.60 | 0.57 | 0.58 | 33 |
| Weight avg | 0.63 | 0.64 | 0.63 | 33 |

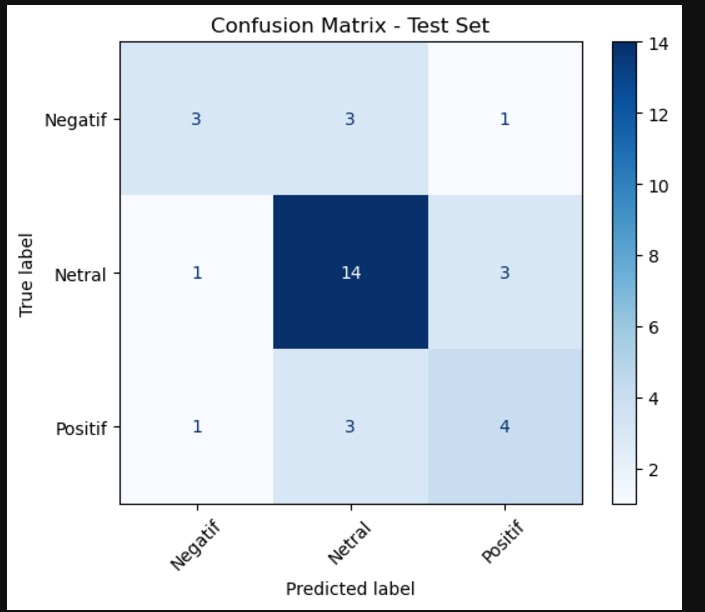
Tabel ini mencakup beberapa metrik yang umum digunakan untuk mengevaluasi kualitas prediksi model pada data uji, yang dalam kasus ini terbagi ke dalam tiga kelas sentimen: Negatif, Netral, dan Positif. *Precision* (Presisi) mengukur berapa banyak prediksi yang benar dari total prediksi yang dikategorikan, ditabel tersebut untuk kelas *negatif,precision* 0.60, artinya 60% dari prediksi negatif model benar-benar negatif. Kelas netral memiliki *Precision* 0.70, artinya 70% dari prediksi netral model benar-benar netral., kelas positif memiliki *Precision* 0.50, artinya 50% dari prediksi positif model benar-benar positif.

*Recall* mengukur seberapa banyak prediksi positif yang benar dibandingkan dengan jumlah data sebenarnya yang seharusnya dikategorikan positif. *Recall* tinggi berarti model mampu mendeteksi sebagian besar kasus positif. Dari tabel tersebut kelas negatif memiliki *Recall* 0.43, artinya hanya 43% dari data yang seharusnya dikategorikan negatif terdeteksi dengan benar oleh model. Kelas netral memiliki *Recall* 0.78, artinya 78% dari data yang seharusnya netral terdeteksi dengan benar oleh model. Kelas positif memiliki Recall 0.50, artinya hanya 50% dari data yang seharusnya positif terdeteksi dengan benar oleh model.

*F1-Score* adalah rata-rata harmonis dari *Precision* dan *Recall*, memberikan gambaran umum tentang keseimbangan antara keduanya. *F1-Score* baik bila model memiliki keseimbangan antara *Precision* dan *Recall*. Nilai *F1-Score* yang tinggi berarti model memiliki kemampuan yang baik dalam mengklasifikasikan kelas secara tepat (baik presisi maupun recall). Pada tabel tersebut kelas negatif memiliki F1-Score 0.50, artinya model memiliki keseimbangan yang rendah antara *precision* dan *recall* pada kelas negatif. Pada kelas netral F1-Score 0.74, artinya model cukup baik dalam mendeteksi dan memprediksi kelas netral. Pada kelas positif F1-Score 0.50, artinya model memiliki keseimbangan yang rendah pada kelas positif. Lalu *support* menunjukkan jumlah sampel yang ada di setiap kelas.

*Macro average* menghitung rata-rata metrik (*precision, recall*, f1-*score*) tanpa memperhitungkan distribusi kelas. *Precision*: 0.60, Recall: 0.57, F1-Score: 0.58 Interpretasi: *Macro average* memberikan gambaran umum performa model dengan memberi bobot yang sama pada setiap kelas, terlepas dari jumlah sampel masing-masing kelas. *Weighted average* menghitung rata-rata metrik berdasarkan *support* (jumlah sampel) dari setiap kelas. Artinya, kelas dengan lebih banyak sampel memiliki pengaruh yang lebih besar terhadap nilai rata-rata. *Precision*: 0.63, *Recall*: 0.64, F1-Score: 0.63 *Weighted average* memberikan gambaran performa model dengan mempertimbangkan distribusi kelas dalam data uji, sehingga kelas yang lebih besar (seperti kelas netral) mempengaruhi nilai rata-rata lebih besar.

* 1. *Confusion Matrix*



Gambar 7. *Confusion Matrix* Model LSTM

Evaluasi silang dilakukan pada 15% data evaluasi dari 32 data yang telah diberi label secara manual. Hasilnya direpresentasikan dalam bentuk *confusion matrix* pada model yang dilatih dengan 214 ulasan aplikasi myUNNES pada data uji yang dapat dilihat pada gambar tersebut. Terdapat 3 ulasan negatif yang diklasifikasikan dengan benar sebagai negatif (*True Negative*/TN), 3 ulasan negatif yang salah di klasifikasikan sebagai netral (*False Netral*/FN), 1 ulasan negatif yang salah diklasifikasikan sebagai positif (*False Positive*/FP), 1 ulasan netral yang salah diklasifikasikan sebagai negatif (*False Negative*/FN), 14 ulasan netral yang benar sebagai netral (*True Nertral*/TN), 3 ulasan netral yang salah diklasifikasikan sebagai positif (*False Positive*/FP), 1 ulasan positif yang salah diklasifikasikan sebagai negatif (*False Negative*/FN), 3 ulasan positif yang salah diklasifikasikan sebagai netral (*False Netral*/FN), dan 4 ulasan positif yang benar sebagai positif (*True Positive*/TP).

* 1. *Perbandingan Sentimen Positif, Netral, Dan Negatif*

Sebuah gambar berisi teks, cuplikan layar, Persegi, diagram

Deskripsi dibuat secara otomatis

Gambar 8. Distribusi kelas sebenarnya dan prediksi

Visualisasi perbandingan antara sentimen positif, netral, dan negatif dibagi menjadi dua bagian yaitu untuk kelas sebenarnya dan untuk kelas prediksi. Perbandingan sentimen pada kelas sebenarnya menunjukkan bahwa sentimen netral memiliki persentase tertinggi sekitar 55%, sedangkan sentimen negatif memiliki persentase sekitar 20%, dan sentimen positif memiliki persentase sekitar 25%. Kemudian perbandingan sentimen pada kelas prediksi menunjukkan bahwa sentimen netral memiliki persentase tertinggi sekitar 60%, sedangkan sentimen positif memiliki persentase sekitar 25% dan sentimen negatif memiliki persentase sekitar 15% .

Dari perbandingan antara kelas sebenarnya dan kelas prediksi menunjukkan bahwa model atau algoritma yang digunakan yaitu LSTM bekerja dengan cukup baik dengan menghasilkan prediksi yang relatif sesuai dengan kelas sebenarnya.

1. Kesimpulan

Berdasarkan hasil evaluasi model analisis sentimen menggunakan LSTM pada ulasan aplikasi myUNNES, model ini mampu mengklasifikasikan sentimen mahasiswa ke dalam tiga kategori, yaitu positif, netral, dan negatif, dengan tingkat akurasi keseluruhan sebesar 64%. Visualisasi sentimen menunjukkan bahwa mahasiswa memberikan respons positif terhadap kinerja aplikasi dengan kata-kata seperti "bagus", "mudah", dan "keren", yang mengindikasikan kepuasan terhadap fitur-fitur yang ada. Namun, pada kelas sentimen negatif, ditemukan keluhan terkait penggunaan ruang penyimpanan dan efisiensi aplikasi yang perlu menjadi perhatian pengembang. Untuk kelas netral, meskipun sebagian mahasiswa memberikan masukan, respons mereka tidak menunjukkan emosi yang kuat, baik positif maupun negatif, sehingga dapat dijadikan bahan pertimbangan untuk pengembangan fitur lebih lanjut.

Performa model pada data uji menunjukkan bahwa model memiliki *precision* terbaik pada kelas netral (0.70) dan performa terendah pada kelas positif dengan *precision* dan *recall* masing-masing sebesar 0.50. Keseimbangan antara *precision* dan *recall* yang diukur melalui F1-score menunjukkan performa terbaik pada kelas netral (0.74) dibandingkan dengan kelas negatif dan positif yang hanya mencapai 0.50. Analisis *macro average* memberikan gambaran bahwa rata-rata performa model cukup baik, meskipun ketidakseimbangan distribusi data memengaruhi hasil evaluasi.

*Confusion matrix* menunjukkan bahwa model cukup andal dalam mengklasifikasikan ulasan netral dengan 14 ulasan diklasifikasikan benar, meskipun terdapat beberapa kesalahan klasifikasi pada ulasan negatif dan positif. Perbandingan sentimen antara kelas sebenarnya dan prediksi menunjukkan bahwa sentimen netral mendominasi, dengan persentase sekitar 55% pada kelas sebenarnya dan 60% pada prediksi, sedangkan sentimen negatif dan positif masing-masing memiliki proporsi lebih kecil. Secara keseluruhan, model LSTM bekerja cukup baik dalam merepresentasikan distribusi sentimen pada data uji, meskipun ada ruang untuk meningkatkan performa, terutama pada kelas sentimen positif dan negatif. Hasil ini dapat dijadikan acuan bagi pengembang aplikasi untuk memperbaiki kinerja aplikasi sesuai dengan kebutuhan pengguna.

Referensi

Nugroho, P. A., Fenriana, I., & Arijanto, R. 2020. IMPLEMENTASI DEEP LEARNING MENGGUNAKAN CONVOLUTIONAL NEURAL NETWORK ( CNN ) PADA EKSPRESI MANUSIA. *JURNAL ALGOR*, *2*(1). <https://jurnal.buddhidharma.ac.id/index.php/algor/index>

Alghifari, D. R., Edi, M., & Firmansyah, L. 2022. Implementasi Bidirectional LSTM untuk Analisis Sentimen Terhadap Layanan Grab Indonesia. *Jurnal Manajemen Informatika (JAMIKA)*, *12*(2), 89–99. <https://doi.org/10.34010/jamika.v12i2.7764>

Amelia, D. S., & Aminuallah, N. C. 2023. *TEKS DAN ANALISIS SENTIMEN PADA CHAT GRUP WHATSAPP MENGGUNAKAN LONG SHORT TERM MEMORY (LSTM)* (Vol. 3, Nomor 2).

Andiyantama, M. Q., Zahira, I., & Irawan, A. 2021. Prediksi Energi Listrik Kincir Angin Berdasarkan Data Kecepatan Angin Menggunakan LSTM. *JITCE (Journal of Information Technology and Computer Engineering)*, *5*(01), 1–7. <https://doi.org/10.25077/jitce.5.01.1-7.2021>

Arsal, M., Wardijono, B. A., & Anggraini, D. 2020. Face Recognition Untuk Akses Pegawai Bank Menggunakan Deep Learning Dengan Metode CNN. *Jurnal Nasional Teknologi dan Sistem Informasi*, *6*(1), 55–63. <https://doi.org/10.25077/teknosi.v6i1.2020.55-63>

Dewi, N. K., Irawan, B. H., Fitry, E., & Putra, A. S. 2021. *Konsep Aplikasi E-Dakwah Untuk Generasi Milenial Jakarta*.

Erfina, A., & Putra, Y. H. 2019. Irony Sentence Detection Techniques Using Fuzzy Historical Classifier. *IOP Conference Series: Materials Science and Engineering*, *662*(6). <https://doi.org/10.1088/1757-899X/662/6/062004>

Hermanto, T. D., Setyanto, A., & Luthfi, E. T. 2021. Algoritma LSTM-CNN untuk Sentimen Klasifikasi dengan Word2vec pada Media Online LSTM-CNN Algorithm for Sentiment Clasification with Word2vec On Online Media. *Citec Journal*, *8*, 2354–5711.

Kurniawan, A. A., & Mustikasari, M. 2020. Implementasi Deep Learning Menggunakan Metode CNN dan LSTM untuk Menentukan Berita Palsu dalam Bahasa Indonesia. *Jurnal Informatika Universitas Pamulang*, *5*(4), 2622–4615. <https://doi.org/10.32493/informatika.v5i4.7760>

Nurvania, J., Jondri, & Lhaksamana, K. M. 2021. *Analisis Sentimen Pada Ulasan di TripAdvisor Menggunakan Metode Long Short-Term Memory (LSTM)*.

Pardede, J., & Pakpahan, I. 2023. *JURNAL PUBLIKASI TEKNIK INFORMATIKA Analisis Sentimen Penanganan Covid-19 Menggunakan Metode Long Short-Term Memory Pada Media Sosial Twitter*.

Ridwansyah, T. 2022. KLIK: Kajian Ilmiah Informatika dan Komputer Implementasi Text Mining Terhadap Analisis Sentimen Masyarakat Dunia Di Twitter Terhadap Kota Medan Menggunakan K-Fold Cross Validation Dan Naïve Bayes Classifier. *Media Online*, *2*(5), 178–185.

Saepulrohman, A., Saepudin, S., & Gustian, D. 2021. Analisis Sentimen Kepuasan Pengguna Aplikasi Whatsapp Menggunakan Algoritma NaÃ¯ve Bayes Dan Support Vector Machine. *is The Best Accounting Information Systems and Information Technology Business Enterprise this is link for OJS usf@*, *6*(2), 91–105. <https://doi.org/10.34010/aisthebest.v6i2.4919>

Utomo, M. W. S., Murti, H. W., Sujatmoko, A. W. I., & Sari, A. P. 2024. DETEKSI SPAM EMAIL MENGGUNAKAN METODE LSTM (LONG SHORT TERM MEMORY). Dalam *Jurnal Mahasiswa Teknik Informatika* (Vol. 8, Nomor 6).

Verianto, E. 2024. Penerapan LSTM Dengan Regularisasi Untuk Mencegah Overfitting Pada Model Prediksi Tingkat Inflasi di Indonesia. *Simkom*, *9*(2), 195–204. <https://doi.org/10.51717/simkom.v9i2.460>