

Penerapan Metode Long Short-Term Memory dan Word2Vec dalam Analisis Sentimen Ulasan pada Aplikasi Ferizy

LSTM and Word2Vec Application for Sentiment Analysis of Reviews on Ferizy

Mega Vebika Shyahrin¹, Yuliant Sibaroni², Diyas Puspendari³

^{1,2,3}Informatika, Telkom University

E-mail: ¹megavebika@students.telkomuniversity.ac.id, ²yuliant@telkomuniversity.ac.id,
³diyaspuspendari@telkomuniversity.ac.id

Abstrak

Transportasi merupakan hal yang penting bagi masyarakat dalam mobilitas sehari-hari. Karena memiliki peranan penting dan dapat memudahkan kehidupan masyarakat, pemerintah mulai mengoptimalkan pembangunan sarana transportasi dan memulai inovasi digital, salah satunya pada moda transportasi laut. Perseroan Terbatas Angkutan Sungai, Danau, dan Penyeberangan Indonesia (PT ASDP) meluncurkan aplikasi Ferizy pada Google Playstore. Dalam inovasi ini, sentimen masyarakat dapat membantu untuk mengetahui kepuasan, kekurangan, saran, dan kritik. Terkait hal tersebut maka diperlukan analisis sentimen untuk memahami maksud ulasan. Analisis ini mengekstrak data ulasan lalu mengolah data tekstual secara otomatis untuk mendapatkan makna sentimen yang terkandung dalam ulasan. Penelitian ini mengimplementasikan klasifikasi Long Short-Term Memory (LSTM) dan ekstraksi fitur Word2Vec variasi skip-gram serta CBOW pada dataset ulasan aplikasi Ferizy. Hasil pengujian dari model menghasilkan nilai akurasi sebesar 88,20% untuk variasi skip-gram dan 74,20% untuk variasi CBOW.

Kata kunci: Analisis Sentimen, Long Short-Term Memory, Word2Vec

Abstract

Transportation is important for the community in their daily mobility. Because of its crucial role and its ability to facilitate people's lives, the government has started to optimize the development of transportation facilities and initiated digital innovations, including in maritime transportation. PT ASDP (Indonesia's River, Lake, and Ferry Transportation) launched the Ferizy application on Google Playstore. In this innovation, sentiments from the public can help identify satisfaction, shortcomings, suggestions, and criticisms. Therefore, sentiment analysis is needed to understand the meaning behind reviews. This analysis extracted review data and automatically processed textual data to obtain sentiments contained in a review. This research implemented the Long Short-Term Memory (LSTM) classification and Word2Vec skip-gram variation feature extraction as well as CBOW on the Ferizy application review dataset. The test results of the model yielded an accuracy score of 88.20% for the skip-gram variation and 74.20% for CBOW variation.

Keywords: Sentiment Analysis, Long Short-Term Memory, Word2Vec

1. PENDAHULUAN

Transportasi memegang peranan penting dalam mendukung pergerakan banyak sektor dalam kehidupan masyarakat. Karena memudahkan kehidupan masyarakat, transportasi juga mempunyai kepentingan yang sama baik moda darat, udara, maupun laut. Oleh karena itu, peningkatan dalam sarana dan prasarana adalah hal yang harus diperhatikan dengan baik tak terkecuali pada transportasi laut.

Oleh sebab itu, pemerintah mulai mengoptimalkan pembangunan sarana transportasi salah satunya adalah transportasi laut, seperti yang dilakukan oleh PT ASDP dengan merilis

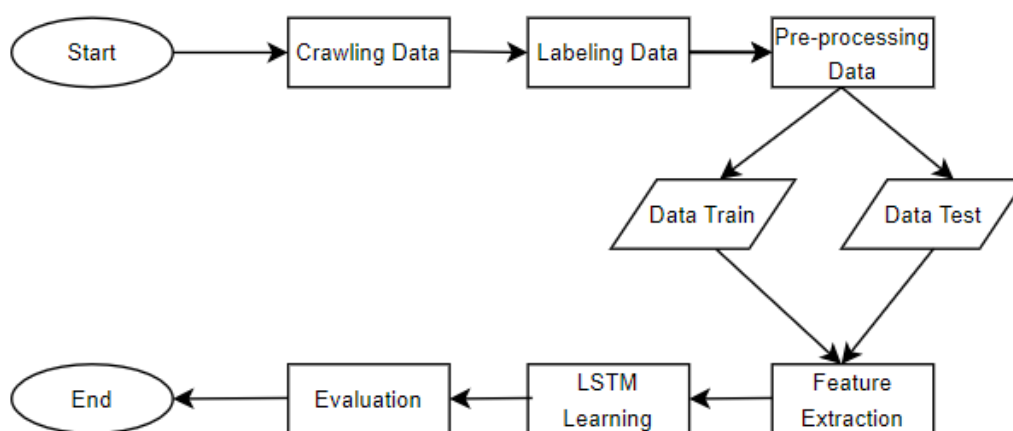
aplikasi Ferizy pada 25 Juli 2020. Dengan tujuan utama adalah digitalisasi, Ferizy memiliki fungsi utama untuk memudahkan masyarakat dalam memesan tiket kapal Ferry dimanapun dan kapanpun. Jika sebelumnya masyarakat melakukan pembelian tiket kapal dengan cara membeli langsung di loket menggunakan kartu e-tol, maka dengan adanya aplikasi Ferizy masyarakat dapat memesan tiket lewat aplikasi. Dengan berbagai kemudahan yang ditawarkan oleh PT ASDP dalam upaya mengoptimalkan pembangunan sarana transportasi, maka ulasan dari para pengguna merupakan suatu hal yang penting. Dari banyaknya ulasan yang diberikan oleh masyarakat, kita dapat mengetahui secara tingkat kepuasan, saran, maupun kritik. Hal tersebut biasa disebut dengan sentimen.

Analisis sentimen adalah metode untuk memahami, mengekstrak data opini, yang secara otomatis mengolah data teks untuk mendapatkan makna sentimen yang terkandung dalam opini [2]. Pada analisis sentimen, ulasan akan dibagi menjadi ulasan positif dan ulasan negatif sehingga pihak Ferizy dapat mengetahui respons masyarakat terhadap suatu pelayanan atau produk. [3]. Dalam penelitian ini, metode yang digunakan dalam analisis sentimen adalah Long Short-Term Memory (LSTM). Pemilihan LSTM didasarkan pada penelitian sebelumnya [4] yang membandingkan metode LSTM dengan naïve bayes classifier dalam analisis sentimen kebijakan *New Normal*, dengan hasil nilai akurasi, presisi, dan *recall* sebesar 82% untuk metode naïve bayes classifier dan 83.33% untuk metode LSTM. Penelitian ini [5] membandingkan metode LSTM, *Convolutional Neural Network* (CNN), dan *Artificial Neural Network* (ANN) dengan hasil akurasi sebesar sebesar 94,78%, 92,98%, dan 92,66%. Pada beberapa penelitian juga digunakan Word2Vec sebagai metode *word embedding* yang dikembangkan oleh Mikolov dalam pengimplementasian metode LSTM, seperti pada penelitian [6] dengan hasil akurasi paling sedikit 85.86%. Penelitian selanjutnya mengenai *review* hotel di Indonesia menggunakan LSTM dan Word2Vec [7] menghasilkan nilai akurasi sebesar 85.96%.

Berdasarkan uraian di atas maka dilakukanlah penelitian analisis sentimen ulasan pada aplikasi Ferizy menggunakan metode *Long Short-Term Memory* (LSTM) karena LSTM telah terbukti bekerja dengan baik dalam menganalisis sentimen pada teks [8] dan Word2Vec untuk merepresentasikan kata menjadi sebuah vektor[9] dengan tujuan hasil yang diperoleh dapat membantu dalam peningkatan kualitas layanan transportasi umum Ferizy dan diharapkan masyarakat dapat merasakan hasil dari tujuan yang menjadi landasan peluncuran aplikasi Ferizy.

2. METODE PENELITIAN

Alur proses perancangan sistem terdiri dari pengumpulan data, persiapan dataset (*preprocessing* dan *pretraining* Word2Vec), pelabelan data, proses klasifikasi menggunakan LSTM, serta evaluasi dapat dilihat pada gambar 1.



Gambar 1. Sistem yang Dirancang

2.1 Crawling Data

Dalam penelitian ini, data yang digunakan adalah ulasan aplikasi Ferizy yang tersedia di *Google Playstore*. Proses pengumpulan data dilakukan menggunakan Google Collab dengan memanfaatkan bahasa python. Google Collab akan memproses dengan cara menginputkan id aplikasi yang bisa ditemukan pada google play versi web. *Dataset* yang diambil merupakan penilaian pengguna terhitung dari tanggal 25 Juni 2020 sampai dengan 28 Mei 2023 dengan jumlah total adalah 5.000 yang dibagi menjadi data positif dan negatif.

2.2 Labeling Data

Dataset yang telah terkumpul selanjutnya akan melalui proses *labeling*. Proses pelabelan dilakukan secara manual. Data akan dikelompokkan dalam 2 klasifikasi, yaitu label positif dan negatif. Data yang termasuk dalam label positif adalah data dengan ulasan kepuasaan atau saran yang membangun terhadap layanan aplikasi Ferizy dan tidak memiliki kendala yang menghambat penggunaan aplikasi. Sedangkan data yang masuk dalam kelompok label negatif adalah ulasan yang berisi keluhan mengenai pelayanan maupun dalam penggunaan aplikasi.

2.3 Pre-processing Data

Dataset yang telah terkumpul selanjutnya dikenai proses *labeling*. Palabelan dilakukan secara manual. Data akan dikelompokkan sesuai label, yaitu label positif dan label negatif. Data yang termasuk dalam ulasan netral akan dimasukkan ke label positif.

Pada tahap ini akan dilakukan *pre-processing* pada data yang telah terkumpul agar dapat diproses oleh model. Berikut ini tahap *pre-processing* yang dilakukan.

1) Case Folding

Tahap ini dilakukan untuk mengonversi huruf kapital menjadi huruf kecil semua atau lowercase[16]. Ulasan diseragamkan menjadi huruf kecil semua untuk memudahkan proses selanjutnya.

2) Cleaning

Pada tahapan ini dilakukan proses menghapus data ulasan yang tidak relevan, seperti simbol, angka, dan spasi yang berlebihan[17].

Tabel 1. Tanda dan Simbol

| Nama | Simbol |
|---------------|--------|
| Titik | . |
| Koma | , |
| Tanda tanya | ? |
| Tanda seru | ! |
| Tanda petik | “ ” |
| Tanda elipsis | ... |
| Tanda kurung | () |

3) Stopword Removal

Tahap ini dilakukan untuk menghapus kata-kata yang sering muncul namun tidak penting atau tidak memiliki makna[18]. Proses *stopword removal* dilakukan dengan memanfaatkan *library* Sastrawi. Berikut ini adalah contoh dari *stopword* yang digunakan dalam penelitian ini.

Tabel 2. *Stopword*

| <i>Stopword</i> | | | | | |
|-----------------|----------|--------|--------|--------|--------|
| Ada | Bahkan | Disini | Kalau | Malah | Sampai |
| Adalah | Bahwa | Dong | Kami | Masih | Sejak |
| Adanya | Begitu | Dulu | Kalian | Naik | Supaya |
| Agak | Biasa | Ikut | Karena | Nah | Tadi |
| Agar | Boleh | Itu | Ke | Nanti | Tambah |
| Akan | Dan | Ini | Kita | Para | Terus |
| Akhir | Dapat | Jadi | Lagi | Perlu | Tiap |
| Amat | Dari | Jauh | Lebih | Saat | Untuk |
| Bagi | Daripada | Juga | Lewat | Sama | Usah |
| Bagian | Dengan | Justru | Luar | Sangat | Yang |

4) *Stemming*

Stemming merupakan tahap mengubah kata berimbuhan menjadi kata dasar. Tahap *stemming* ini dilakukan dengan menggunakan *library Sastrawi* yang terdapat pada python.

Tabel 3. *Stemming*

| Kalimat | <i>Stemming</i> |
|---|-------------------------------|
| pembatalan pengubahan jadwal parah biayanya | batal ubah jadwal parah biaya |

5) *Tokenization*

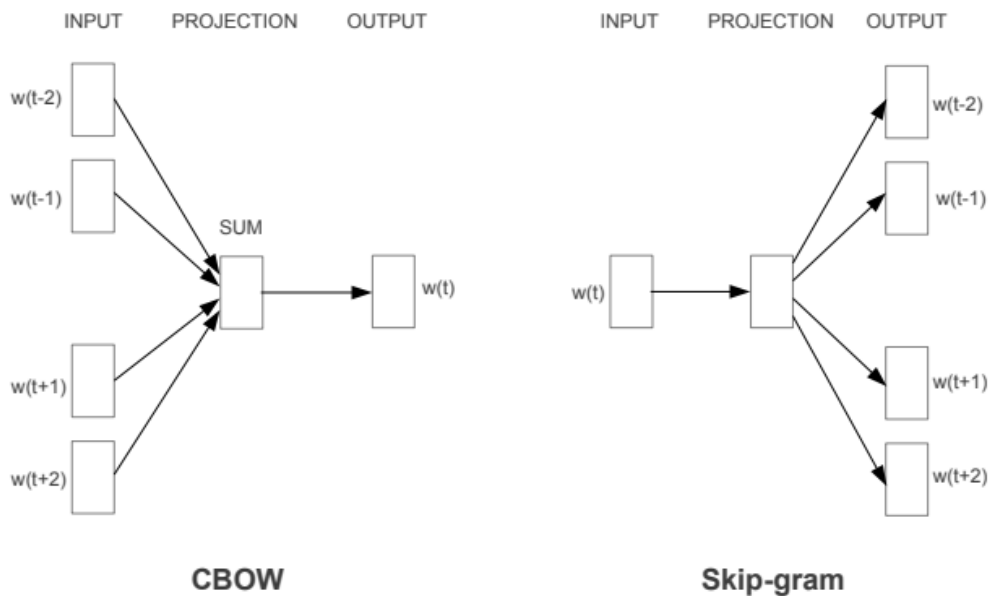
Tahap *tokenization* digunakan untuk mengubah setiap teks pada data ulasan menjadi rangkaian kata atau token.

2.4 *Feature Extraction*

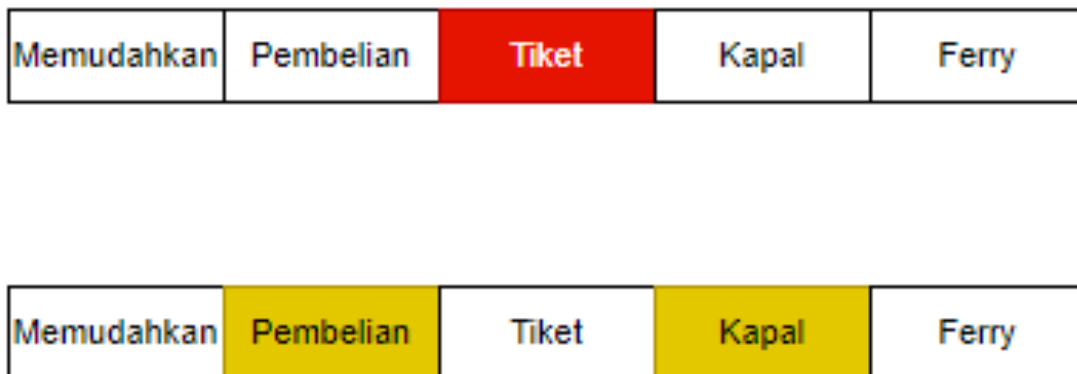
Proses selanjutnya setelah *pre-processing*, yaitu melakukan pembentukan *word vector* menggunakan Word2Vec, yang memiliki kemampuan membaca dan mengolah teks ukuran besar dan mengubah setiap kata menjadi vektor. Word2Vec dapat memahami sintaks 11 dan memaknai bahasa alami lalu merepresentasikan setiap kata dengan vektor.

Word2vec adalah sebuah metode yang digunakan untuk merepresentasikan setiap kata pada teks dokumen ke dalam sebuah vektor dengan N dimensi. Pada penelitian ini penulis menggunakan Word2Vec variasi skip-gram dan continuous bag of word (CBOW). Kedua variasi ini memiliki cara kerja yang berbeda seperti yang tampak pada gambar 2.

Cara kerja *skip-gram* adalah dengan memprediksi konteks dari target kata yang dimiliki sedangkan cara kerja CBOW memprediksi target kata berdasarkan konteks yang ada[16]. Contoh dari cara kerja tersebut akan dijelaskan dibawah ini.



Gambar 2. Arsitektur CBOW dan Skip-gram



Gambar 3. Target Kata dan Konteks

Pada gambar gambar 3, dijelaskan contoh kalimat yang digunakan sebagai input pada model skip-gram dan CBOW. Kata pada kotak merah adalah target kata sedangkan kata pada kotak kuning merupakan konteks yang ada.

a. *Skip-gram*

Skip-gram bertugas memprediksi konteks berdasarkan kata target yang dimiliki. Kalimat yang digunakan sebagai contoh adalah “Memudahkan Pembelian Tiket Kapal Ferry”. Kemudian kalimat tersebut akan diubah menjadi *one-hot encoded vectors*. Jika kata target yang dimiliki adalah “Tiket” maka menjadi $[0, 0, 1, 0, 0]$. Setelah itu skip-gram akan memprediksi konteks dengan hasil kata “Pembelian” dan “Kapal” adalah outputnya.

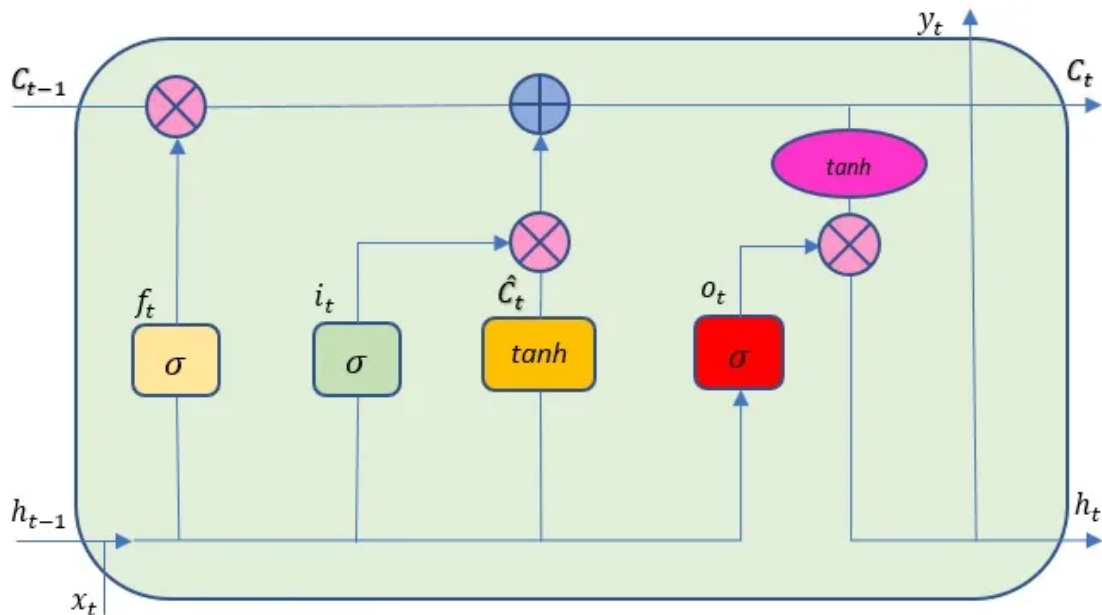
b. CBOW

CBOW memiliki cara kerja memprediksi kata target berdasarkan konteks yang dimiliki. Kata “Pembelian” dan “Kapal” yang merupakan konteks akan diubah ke *one-hot encoded vectors* menjadi $[0, 1, 0, 0, 0]$ dan $[0, 0, 0, 1, 0]$. Output dari CBOW adalah kata “Tiket” yang merupakan kata target.

Pada penelitian ini dilakukan perbandingan antara model variasi skip-gram dan CBOW dengan memanfaatkan *library Gensim*. Variasi skip-gram dan CBOW akan digunakan dengan implementasi window sebesar 5 dan dimensi sebesar 100.

2.5 Classification

Setelah proses pelabelan selesai, selanjutnya akan dilakukan klasifikasi menggunakan metode LSTM yang telah dijelaskan pada bab di atas. Metode LSTM digunakan karena menggunakan blok *memory-cell*, meliputi *input gate*, *forget gate* dan *output gate* untuk mengganti lapisan RNN supaya bisa mengatasi masalah *vanishing gradient* pada RNN [19]. Pada metode LSTM meskipun ada jarak antarteks, analisis masih bisa dilakukan. Hal ini terjadi karena pada LSTM terdapat memori untuk menyimpan informasi sebelumnya dan dapat disimpan untuk



Gambar 4. Arsitektur LSTM

waktu lama[20]. Berikut ini adalah arsitektur dari LSTM.

Berdasarkan gambar di atas, pada proses X_t dilakukan input vektor lalu fungsi sigmoid akan memutuskan untuk menghilangkan informasi yang tidak diperlukan. Hasil dari proses aktivasi sigmoid ini menghasilkan *output forget gate* dengan nilai antara 1 dan 0. Nilai 1 berarti semua informasi diteruskan, sedangkan nilai 0 berarti tidak ada informasi yang diteruskan [2]. Pada proses kedua akan dilakukan aktivasi oleh sigmoid dan tanh untuk selanjutnya kedua hasil tersebut akan dikalikan. Selanjutnya untuk proses ketiga dilakukan perkalian antara nilai *cell gate*, *forget gate*, dan nilai yang dihasilkan dari proses kedua. Proses yang terakhir, yaitu sigmoid akan

memutuskan nilai dari *output gate* untuk akhirnya dikalikan dengan nilai yang dibuat oleh tanh. Berikut ini adalah perhitungan dari setiap gatenya:

- 1) *Forget Gate*

$$f_t = \sigma(W_f \times [x_t + h_{t-1}] + b_f)$$
- 2) *Input Gate*

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

$$\tilde{C}_t = \tanh(W_C \times [x_t + h_{t-1}] + b_C)$$
- 3) *Cell state*

$$C_t = f_t \times C_{t-1} + i_t \times \tilde{C}_t$$
- 4) *Output Gate*

$$o_t = \sigma(W_o \times [x_t + h_{t-1}] + b_o)$$

$$h_t = o_t \times \tanh(C_t)$$

2.6 Evaluation

Untuk mengukur *accuracy*, *precision*, *recall*, dan *f measure* digunakan *confusion matrix*, seperti terlihat pada tabel 4.

Tabel 4. *Confussion Matrix*

| Kelas Sebenarnya | Kelas Prediksi | |
|------------------|---------------------|---------------------|
| | Positif | Negatif |
| Positif | TP (True Positive) | TN (True Negative) |
| Negatif | FP (False Positive) | FN (False Negative) |

Berdasarkan tabel di atas, untuk mengukur kinerja klasifikasi diharuskan menghitung *accuracy*, *precision*, dan *recall*, *f-measure*. Berikut ini adalah rumus-rumus tersebut.

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

$$Precision = \frac{TP}{TP + FP}$$

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

$$F1 - score = 2x \frac{Precision \times Recall}{Precision + Recall}$$

3. HASIL DAN PEMBAHASAN

Proses evaluasi dilakukan dengan menggunakan data ulasan dari aplikasi Ferizy. Evaluasi dilakukan menggunakan K-fold Cross Validation. Dengan membandingkan 2 variasi Word2Vec, yaitu *skip-gram* dan CBOW.

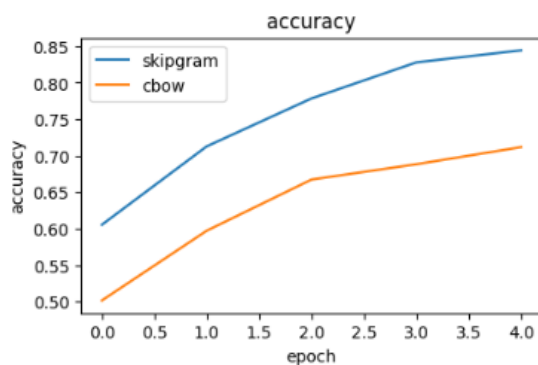
3.1 Hasil Pengujian

Klasifikasi pada aplikasi Ferizy dilakukan dengan menggunakan metode LSTM dan ekstraksi fitur Word2vec variasi skip-gram dan CBOW. Besar dimensi *word vector* yang digunakan pada kedua variasi Word2Vec adalah 100. Pada pengujian ini penulis menggunakan K-fold dengan iterasi sebanyak 10, batch size sebesar 128, dan epoch sebanyak 5 untuk mengetahui nilai akurasi model yang telah dibangun. Perbandingan data *train* dan data *test* yang digunakan adalah 90:10 dengan *random state* = 42.

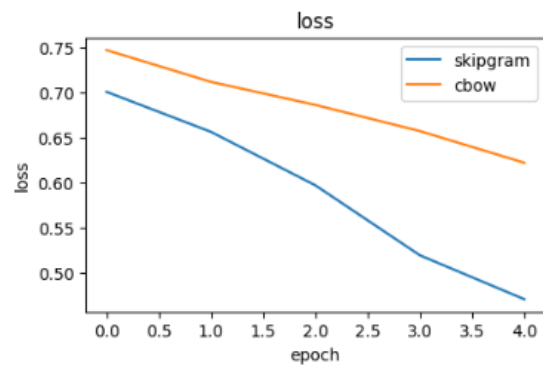
Tabel 5. Hasil Pengujian

| Iterasi ke- | Word2Vec | |
|-------------|------------------|---------------|
| | <i>Skip-gram</i> | CBOW |
| 1 | 89,11% | 70,67% |
| 2 | 86,22% | 75,33% |
| 3 | 86,44% | 81,11% |
| 4 | 87,33% | 74,22% |
| 5 | 90,44% | 80,00% |
| 6 | 86,44% | 69,56% |
| 7 | 88,89% | 77,78% |
| 8 | 87,33% | 72,89% |
| 9 | 86,22% | 72,89% |
| 10 | 89,56% | 73,78% |
| Average | 87.79% | 74,83% |

Berdasarkan hasil pada tabel 8 ditemukan bahwa akurasi tertinggi Word2Vec variasi *skip-gram* didapatkan pada k-fold 5 dengan nilai 90,44% sedangkan pada variasi CBOW nilai akurasi tertinggi terdapat pada k-fold 3 dengan nilai 81,11%. Dengan nilai rata-rata kedua variasi tersebut adalah sebesar 87,79% dan 74,83%. Hasil tersebut menunjukkan Word2Vec variasi *skip-gram* bekerja lebih baik pada data yang ada daripada CBOW yang cenderung lebih rendah.



Gambar 5. Akurasi Skip-gram dan CBOW



Gambar 6. Loss Skip-gram dan CBOW

Hal tersebut juga dapat diamati melalui perbandingan pada gambar 4 dan 5 mengenai grafik nilai akurasi dan loss dari kedua variasi. Nilai akurasi skip-gram dan CBOW semakin tinggi selaras dengan semakin banyaknya epoch. Serta nilai akurasi yang stabil di setiap epoch menunjukkan bahwa kedua model bekerja dengan baik dalam memberi prediksi pada dataset yang ada.

4. KESIMPULAN DAN SARAN

Aplikasi Ferizy merupakan inovasi baru oleh PT ASDP dalam rangka digitalisasi. Sejak diluncurkan pada tahun 2020 pengguna aplikasi tersebut makin bertambah seiring waktu begitu juga dengan ulasan yang diterima pada laman Google Playstore. Dengan menggunakan metode klasifikasi LSTM dan melakukan perbandingan antara 2 variasi pada ekstraksi fitur Word2Vec, dilakukanlah analisis sentimen terhadap ulasan aplikasi Ferizy di Google playstore. Berdasarkan hasil pengujian yang dilakukan pada 5.000 data ulasan aplikasi Ferizy, ekstraksi fitur Word2Vec variasi *skip-gram* memperoleh nilai akurasi sebesar 88,20% lebih tinggi daripada dengan variasi CBOW yang memperoleh akurasi sebesar 74,20%. Pada penelitian berikutnya, diharapkan data

yang digunakan lebih banyak dan menambahkan proses normalisasi pada tahapan *pre-processing* untuk menghasilkan dataset yang lebih optimal pada proses selanjutnya

DAFTAR PUSTAKA

- [1] Ludeman, L. C.. 1987. *Fundamental of Digital Signal Processing*. Singapore : John Wiley & Sons, Inc.
- [2] Faadilah, A. (2020, March 5). Institutional Repository UIN Syarif Hidayatullah Jakarta: Analisis sentimen pada ulasan aplikasi tokopedia di google play store menggunakan metode long short term memory. <https://repository.uinjkt.ac.id/dspace/handle/123456789/50432>.
- [3] E. M. Sipayung, H. Maharani, and I. Zefanya, "Perancangan Sistem Analisis Sentimen Komentar Pelanggan Menggunakan Metode Naive Bayes Classifier," *J. Sist. Inf.*, vol. 8, no. 1, pp. 958–965, 2016.
- [4] Isnain, A.R., Sulistiani, H., Hurohman, B.M., Nurkholis, A. and Styawati, S. (2022). Analisis Perbandingan Algoritma LSTM dan Naive Bayes untuk Analisis Sentimen. *JEPIN (Jurnal Edukasi dan Penelitian Informatika)*, [online] 8(2), pp.299–303. doi:10.26418/jp.v8i2.54704S.
- [5] NUGRAHA, T.I. and Goeiranto, L. (2022). Perbandingan Metode Deep Learning Dalam Analisa Sentimen Ulasan Aplikasi Jakarta Kini (JAKI). [online] repository.mercubuana.ac.id. Available at: <https://repository.mercubuana.ac.id/70536/>.
- [6] Arinata, B. G. I. (2022, October 10). Analisis Sentimen Pada Edom STMIK Primakara Menggunakan Metode Long Short Term Memory | Arinata | *Jurnal Teknologi Informasi dan Komputer*. <https://jurnal.undhirabali.ac.id/index.php/jutik/article/view/2086T>.
- [7] Thomas, M., & C.A, L. (2018b). Sentimental analysis using recurrent neural network. *International Journal of Engineering & Technology*, 7(2.27), 88. <https://doi.org/10.14419/ijet.v7i2.27.12635>.
- [8] Dr. G. S. N. Murthy, Shanmukha Rao Allu, Bhargavi Andhavarapu, and Mounika Bagadi, Mounika Belusonti, "Text based Sentiment Analysis using LSTM," *Int. J. Eng. Res.*, vol. V9, no. 05, May 2020.
- [9] Nurrohmat, M. A., A. Sn, 2019, Sentiment Analysis of Novel Review Using Long ShortTerm Memory Method, *IJCCS (Indonesian Journal of Computing and Cybernetics Systems*, No. 3, Vol. 13, Hal. 209–218.
- [10] Evasaria Magdalena Sipayung, Herastia Maharani, & Ivan Zefanya. (2016a). Perancangan Sistem Analisis Sentimen Komentar Pelanggan Menggunakan Metode Naive Bayes Classifier. *Jurnal Sistem Informasi Universitas Sriwijaya*, 8(1), 131705. <https://doi.org/10.36706/jsi.v8i1.3250>.
- [11] Hermanto, D. T., Setyanto, A., & Luthfi, E. T. (2021). Algoritma LSTM-CNN untuk Binary Klasifikasi dengan Word2vec pada Media Online. *Creative Information Technology Journal*, 8(1), 64. <https://doi.org/10.24076/citec.2021v8i1.264>.
- [12] Jovita Nurvania. (2021). Analisis Sentimen Pada Ulasan di TripAdvisor Menggunakan Metode Long Short-Term Memory (LSTM). *EProceedings of Engineering*, 8(4).
- [13] Widayat, W. (2021). Analisis Sentimen Movie Review menggunakan Word2Vec dan metode LSTM Deep Learning. *JURNAL MEDIA INFORMATIKA BUDIDARMA*, 5(3), 1018. <https://doi.org/10.30865/mib.v5i3.3111>.
- [14] Subowo, E. (2022). Algoritma Bidirectional Long Short Term Memory untuk Analisis Sentimen Berbasis Aspek pada Aplikasi Belanja Online dengan Cicilan | *JURNAL FASILKOM*. ejurnal.umri.ac.id, [online] XII(II). Available at: <https://ejurnal.umri.ac.id/index.php/JIK/article/view/3759>.
- [15] Ailiyya, S. (2020). Analisis sentimen berbasis aspek pada ulasan aplikasi tokopedia menggunakan support vector machine. repository.uinjkt.ac.id, [online] xiv. Available at: <https://repository.uinjkt.ac.id/dspace/handle/123456789/51680>.
- [16] Tryantono, Y. S. (2021). Analisis sentimen terhadap ulasan film menggunakan Word2Vec

- dan SVM. eProceedings of Engineering, 8(4).
<https://openlibrarypublications.telkomuniversity.ac.id/index.php/engineering/article/view/15241>
- [17] Aufa, R. N., Prasetiyowati, S. S., & Sibaroni, Y. (2023). The Effect of Feature Weighting on Sentiment Analysis TikTok Application Using The RNN Classification. *Building of Informatics, Technology and Science (BITS)*, 5(1). <https://doi.org/10.47065/bits.v5i1.3597>
- [18] Putri, P. (2013). Implementasi Metode Improved K-Nearest Neighbor pada Analisis Sentimen Twitter Berbahasa Indonesia. <http://repository.ub.ac.id/145872/>
- [19] Afidah, D. I., Dairoh, D., Handayani, S. F., Pratiwi, R. W., & Sari, S. I. (2022). Sentimen Ulasan Destinasi Wisata pulau Bali Menggunakan bidirectional long short term memory. *MATRIK : Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer*, 21(3), 607-618. <https://doi.org/10.30812/matrik.v21i3.1402>.
- [20] Alghifari, D. R. (2022, September 24). Implementasi Bidirectional LSTM untuk Analisis Sentimen Terhadap Layanan Grab Indonesia | *Jurnal Manajemen Informatika (JAMIKA)*. <https://ojs.unikom.ac.id/index.php/jamika/article/view/7764>