

Penerapan Model *Word2Vec* dan LSTM dalam Analisis Sentimen Ulasan Pengguna *Mobile legends*

Application of Word2Vec and LSTM Models in Sentiment Analysis of Mobile Legends User Reviews

¹Nur Faizal Basri*, ²Ema Utami

^{1,2}Informatika Program Magister, Universitas Amikom Yogyakarta

^{1,2}Jl. Ring Road Utara, Ngringin, Condongcatur, Kec. Depok, Kabupaten Sleman, Daerah Istimewa Yogyakarta 55281, Indonesia

*e-mail: nurfaizal@students.amikom.ac.id

(received: 10 February 2025, revised: 18 February 2025, accepted: 19 February 2025)

Abstrak

Analisis sentimen menjadi aspek penting dalam memahami opini pengguna terhadap suatu produk atau layanan, termasuk dalam industri game. Penelitian ini mengimplementasikan kombinasi model Word2Vec dan Long Short-Term Memory (LSTM) untuk menganalisis sentimen ulasan pengguna game Mobile Legends yang diperoleh dari Google Play Store. Dataset yang digunakan mencakup 100.000 ulasan yang telah melalui tahap preprocessing seperti pembersihan teks, tokenisasi, dan penghapusan stopwords. Model Word2Vec digunakan untuk merepresentasikan teks dalam bentuk vektor numerik, sedangkan LSTM digunakan untuk memprediksi sentimen ulasan. Hasil evaluasi menunjukkan bahwa model ini mencapai akurasi sebesar 87,88% yang mengindikasikan efektivitas metode ini dalam mengklasifikasikan sentimen pengguna. Analisis lebih lanjut menunjukkan bahwa mayoritas ulasan pengguna bersentimen positif, dengan kata-kata seperti "bagus", "seru", dan "mantap" yang sering muncul dalam word cloud. Penelitian ini dapat memberikan wawasan bagi pengembang game dalam memahami opini pengguna serta menjadi referensi dalam penerapan deep learning untuk analisis sentimen di industri game.

Kata kunci: analisis sentimen, word2vec, LSTM, mobile legends, deep learning

Abstract

Sentiment analysis has become an important aspect of understanding user opinions regarding a product or service, including in the gaming industry. This study implements a combination of Word2Vec and Long Short-Term Memory (LSTM) models to analyze the sentiment of user reviews for the game Mobile Legends, obtained from the Google Play Store. The dataset used comprises 100,000 reviews that have undergone preprocessing stages such as text cleaning, tokenization, and stopword removal. The Word2Vec model is employed to represent the text in the form of numerical vectors, while LSTM is used to predict the sentiment of the reviews. Evaluation results indicate that this model achieves an accuracy of 87.88%, demonstrating the effectiveness of this method in classifying user sentiment. Further analysis reveals that the majority of user reviews are positive, with words such as "good," "exciting," and "awesome" frequently appearing in the word cloud. This research can provide insights for game developers in understanding user opinions and serve as a reference for the application of deep learning in sentiment analysis within the gaming industry.

Keywords: sentiment analysis, word2vec, LSTM, mobile legends, deep learning

1 Pendahuluan

Perkembangan industri game mobile telah mengalami peningkatan yang pesat dalam beberapa tahun terakhir, didorong oleh kemajuan teknologi dan penetrasi perangkat mobile yang semakin luas. Mobile gaming telah menjadi salah satu segmen terbesar dalam industri hiburan digital, dengan jumlah pemain yang terus bertambah dari berbagai kalangan [1]. Salah satu game yang mendominasi pasar adalah *Mobile legends: Bang Bang* (MLBB), sebuah permainan berbasis Multiplayer Online Battle Arena (MOBA) yang dikembangkan oleh Moonton. Game ini telah menarik jutaan pengguna

<http://sistemasi.ftik.unisi.ac.id>

aktif setiap harinya, yang memberikan berbagai ulasan dan feedback melalui platform seperti Google Play Store dan App Store [2].

Ulasan yang diberikan oleh pengguna mencerminkan pengalaman mereka dalam bermain game, baik dari segi mekanisme permainan, kualitas grafis, stabilitas server, hingga kebijakan pengembang terkait fitur dan monetisasi. Informasi ini sangat berharga bagi pengembang dalam meningkatkan kualitas permainan serta menyesuaikan strategi pengembangan produk agar lebih sesuai dengan kebutuhan pengguna [3]. Namun, jumlah ulasan yang sangat besar dan bervariasi dalam bahasa alami membuat analisis manual menjadi tidak efektif dan memakan banyak waktu. Oleh karena itu, diperlukan pendekatan yang lebih sistematis dan otomatis dalam memahami opini pengguna melalui teknik analisis sentimen [4].

Analisis sentimen adalah teknik dalam bidang Natural Language Processing (NLP) yang digunakan untuk mengidentifikasi dan mengkategorikan opini yang diekspresikan dalam teks [5]. Teknik ini memungkinkan pemrosesan data tekstual dalam jumlah besar untuk menggali wawasan berharga terkait kepuasan pengguna terhadap suatu produk atau layanan. Dengan meningkatnya jumlah data ulasan yang tersedia, pendekatan yang lebih efektif dan akurat dalam mengolah data tersebut menjadi kebutuhan utama dalam analisis sentimen. Dalam konteks industri game, analisis sentimen dapat membantu pengembang memahami reaksi komunitas terhadap pembaruan game, memperbaiki bug yang sering dikeluhkan, serta mengidentifikasi faktor yang meningkatkan atau menurunkan loyalitas pemain [6].

Seiring dengan perkembangan teknologi kecerdasan buatan, metode deep learning telah menjadi pendekatan unggulan dalam berbagai tugas NLP, termasuk analisis sentimen. Dibandingkan dengan metode tradisional seperti Naïve Bayes, Support Vector Machine (SVM), atau pendekatan berbasis lexicon, deep learning mampu menangkap pola dan hubungan kompleks dalam teks dengan lebih baik [7]. Salah satu model deep learning yang banyak digunakan dalam pemrosesan sekuensial adalah Long Short-Term Memory (LSTM), sebuah jenis Recurrent Neural Network (RNN) yang dirancang untuk menangani masalah long-term dependencies dalam teks [8]. Model LSTM mampu menangkap hubungan kontekstual dalam teks yang lebih panjang, sehingga memberikan hasil yang lebih baik dalam analisis sentimen dibandingkan model tradisional seperti Naïve Bayes dan Support Vector Machine (SVM) [9].

Selain model deep learning, representasi teks yang digunakan dalam analisis sentimen juga berperan penting dalam meningkatkan akurasi model. Salah satu pendekatan yang populer dalam pembelajaran representasi teks adalah Word2Vec, sebuah metode embedding kata yang mengubah teks menjadi vektor numerik dengan mempertimbangkan hubungan semantik antar kata [10]. Word2Vec dapat menghasilkan representasi kata yang lebih kaya dibandingkan metode tradisional seperti one-hot encoding atau TF-IDF, sehingga meningkatkan pemahaman model terhadap makna teks [11] [12]. Kombinasi antara Word2Vec dan LSTM telah terbukti efektif dalam berbagai penelitian sebelumnya dalam meningkatkan performa analisis sentimen, khususnya dalam menangkap hubungan semantik yang lebih kompleks dalam teks ulasan [9] [13].

Meskipun banyak penelitian yang telah menggunakan kombinasi Word2Vec dan LSTM dalam analisis sentimen, masih terdapat beberapa tantangan yang perlu diatasi. Salah satu tantangan utama adalah menangani aspek kontekstual dalam teks, terutama dalam bahasa informal atau slang yang sering digunakan dalam ulasan pengguna game [14]. Selain itu, pelatihan model deep learning memerlukan komputasi yang tinggi dan sering kali membutuhkan optimasi lebih lanjut untuk meningkatkan efisiensi serta mengurangi overfitting. Oleh karena itu, penelitian ini bertujuan untuk menerapkan model LSTM dengan representasi Word2Vec dalam analisis sentimen ulasan pengguna *Mobile legends* guna mengeksplorasi efektivitas metode ini dalam mengklasifikasikan opini pengguna.

Penelitian ini bertujuan untuk mengembangkan metode analisis sentimen yang lebih akurat dengan menggabungkan Word2Vec dan LSTM. Word2Vec dipilih karena mampu menangkap hubungan semantik antar kata secara lebih efektif dibandingkan pendekatan tradisional seperti *one-hot encoding* atau *TF-IDF*, yang hanya mempertimbangkan frekuensi kata tanpa memahami konteks. Sementara itu, LSTM unggul dalam menangani data sekuensial dengan mempertimbangkan ketergantungan temporal dalam teks, menjadikannya lebih efektif dibandingkan metode klasik seperti *Naïve Bayes* atau *Support Vector Machine* (SVM), yang tidak mempertimbangkan urutan kata. Dengan kombinasi ini, diharapkan dapat diperoleh wawasan yang lebih mendalam tentang opini

<http://sistemasi.ftik.unisi.ac.id>

pengguna terhadap *Mobile legends*, serta memberikan rekomendasi bagi pengembang dalam meningkatkan kualitas permainan.

2 Tinjauan Literatur

Pada penelitian Muhammad [13] mengkaji analisis sentimen ulasan hotel dalam bahasa Indonesia menggunakan kombinasi model Word2Vec dan Long Short-Term Memory (LSTM). Dengan mengumpulkan 2500 ulasan dari situs Traveloka, penelitian ini menemukan bahwa kombinasi parameter terbaik untuk Word2Vec adalah arsitektur Skip-gram, metode evaluasi Hierarchical Softmax, dan dimensi vektor 300. Sementara itu, parameter terbaik untuk LSTM adalah nilai dropout 0,2, pooling rata-rata, dan laju pembelajaran 0,001. Hasilnya menunjukkan akurasi sebesar 85,96%, yang menunjukkan bahwa kombinasi model ini efektif dalam mengklasifikasikan sentimen ulasan hotel dalam bahasa Indonesia.

Kemudian penelitian Shyahrin [15] menggunakan kombinasi model Long Short-Term Memory (LSTM) dan Word2Vec dalam analisis sentimen. Dengan mengumpulkan 5000 ulasan dari Google Playstore, penelitian ini menemukan bahwa kombinasi parameter terbaik untuk Word2Vec adalah variasi skip-gram dengan akurasi sebesar 88,20%, sedangkan variasi CBOW menghasilkan akurasi sebesar 74,20%. Hasil ini menunjukkan bahwa model LSTM dengan ekstraksi fitur Word2Vec variasi skip-gram lebih efektif dalam mengklasifikasikan sentimen ulasan aplikasi Ferizy.

Perbandingan model machine learning pada penelitian Aakash [16] untuk membandingkan kinerja berbagai model machine learning dalam analisis sentimen, mengevaluasi metrik kinerja seperti akurasi, presisi, recall, dan F1-Score, serta memahami kelebihan dan kelemahan masing-masing model. Hasil penelitian menunjukkan bahwa model LSTM dan GRU memiliki presisi dan recall yang tinggi, sementara SVM, Random Forest, dan Naive Bayes juga menunjukkan kinerja yang baik. Nilai akurasi tertinggi dicapai oleh model GRU dengan 86,3%, sementara SVM dan Logistic Regression mencapai 84%. Kelemahan penelitian ini termasuk waktu pelatihan yang lama untuk model Random Forest dan trade-off antara waktu pelatihan dan prediksi untuk model KNN.

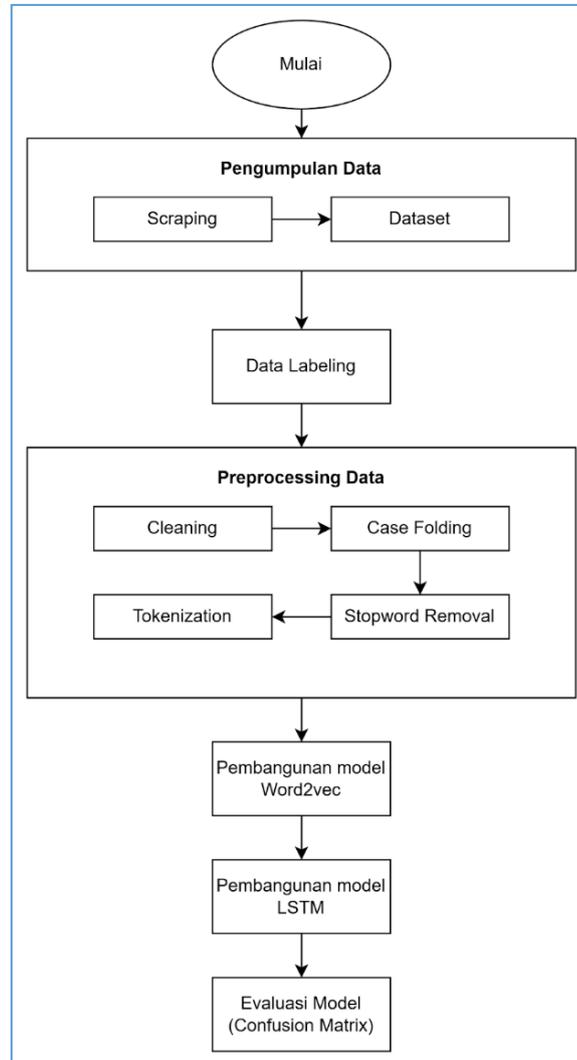
Selanjutnya penelitian Nurdy [17] mengkaji analisis sentimen ulasan game Stumble Guys di Google Play Store menggunakan algoritma Naïve Bayes. Dengan mengumpulkan 1.500 ulasan, penelitian ini menemukan bahwa model Naïve Bayes mencapai akurasi sebesar 86%, dengan precision, recall, dan f1 score masing-masing sebesar 86%. Hasil ini menunjukkan bahwa algoritma Naïve Bayes efektif dalam mengklasifikasikan sentimen ulasan game Stumble Guys, yang mencerminkan berbagai pendapat pengguna tentang game tersebut.

Pada penelitian W. Widayat [9] melakukan analisis sentimen ulasan film menggunakan kombinasi model Word2Vec dan Long Short-Term Memory (LSTM). Dengan menggunakan dataset yang terdiri dari 25.000 ulasan film, penelitian ini menemukan bahwa metode Skip-Gram pada Word2Vec dengan dimensi vektor 100 menghasilkan akurasi terbaik sebesar 88,17%, sedangkan metode CBOW dengan dimensi vektor 200 menghasilkan akurasi sebesar 87,68%. Hasil ini menunjukkan bahwa model LSTM dengan ekstraksi fitur Word2Vec efektif dalam mengklasifikasikan sentimen ulasan film.

Penelitian Chandradev [18] mengkaji analisis sentimen ulasan hotel di Indonesia menggunakan metode deep learning BERT (Bidirectional Encoder Representations from Transformers). Penelitian ini bertujuan untuk membantu pengusaha hotel memahami sentimen pengunjung melalui ulasan mereka, yang dapat digunakan untuk pengambilan keputusan terkait layanan dan aspek bisnis. Model SmallBERT yang digunakan dalam penelitian ini dilatih dengan dataset 515 ribu ulasan hotel selama lima epoch dan mencapai akurasi sebesar 91,40%, precision 90,51%, recall 90,51%, dan F1 score 90,51%. Hasil ini menunjukkan bahwa model SmallBERT efektif dalam mengklasifikasikan sentimen ulasan hotel di Indonesia. Visualisasi hasil analisis menunjukkan bahwa mayoritas ulasan memiliki sentimen positif, yang dapat mempengaruhi peningkatan kunjungan wisata ke wilayah tersebut.

3 Metode Penelitian

Pada penelitian ini, tahapan-tahapan yang akan dilakukan meliputi pengumpulan data, data labeling, preprocessing data, pembangunan model Word2vec, pembangunan model LSTM, dan evaluasi model. Pada Gambar 1 menampilkan diagram tahapan pelaksanaan penelitian.



Gambar 1. Tahapan penelitian

3.1 Pengumpulan Data

Data penelitian ini berasal dari ulasan pengguna *Mobile legends* di Google Play Store, mencakup teks ulasan, skor bintang (1-5), dan tanggal ulasan. Sebanyak 100.000 ulasan dikumpulkan secara acak dari berbagai periode menggunakan web scraping atau API resmi. Pengambilan data dilakukan dengan menentukan ID aplikasi *com.mobile.legends*, menyaring ulasan berbahasa Indonesia, dan menyortir berdasarkan yang terbaru. Teknik paginasi dengan *continuation token* memastikan variasi waktu. Data kemudian dikonversi ke *DataFrame* menggunakan *pandas* dan disimpan dalam file CSV untuk analisis lebih lanjut.

3.2 Pelabelan Data

Penelitian ini melakukan pelabelan data berdasarkan skor bintang ulasan pengguna. Ulasan dengan rating 1 dan 2 dianggap sentimen negatif, rating 4 dan 5 sebagai sentimen positif, sementara rating 3 dianggap netral dan tidak digunakan dalam pemodelan. Keputusan untuk mengecualikan ulasan netral bertujuan agar klasifikasi sentimen lebih jelas, meskipun ini dapat memengaruhi distribusi data dan mengurangi kompleksitas model. Meskipun demikian, model menjadi lebih fokus mempelajari pola sentimen positif dan negatif, menghindari ambiguitas dari ulasan netral.

3.3 Preprocessing Data

Preprocessing merupakan tahap untuk meningkatkan kualitas data yang digunakan dalam penelitian ini. Preprocessing data yang dilakukan adalah cleaning, case folding, Stopword Removal, dan Tokenisasi.

Pada tahap pembersihan data (cleaning), dilakukan penghapusan karakter non-alfabet dari teks menggunakan regular expression (regex) untuk memastikan bahwa hanya karakter huruf yang dipertimbangkan dalam analisis. Berikut contoh cleaning dapat dilihat pada Tabel 1.

Tabel 1. Contoh hasil cleaning

Teks Asli	Setelah Cleaning
Game MOBA nomor 1 nih boss 👍	Game MOBA nomor 1 nih boss
Tolong kasih tim yang adil dan seimbang selalu dapet tim yang ga bener tapi lawan nya jago 🙄👤	Tolong kasih tim yang adil dan seimbang selalu dapet tim yang ga bener tapi lawan nya jago

Selanjutnya, tahap case folding diterapkan untuk mengubah seluruh teks menjadi huruf kecil, sehingga menghilangkan perbedaan antara huruf besar dan kecil yang tidak relevan. Berikut contoh tahap case folding dapat dilihat pada Tabel 2.

Tabel 2. Contoh hasil case folding

Cleaning	Setelah Case Folding
Game MOBA nomor 1 nih boss	game moba nomor 1 nih boss
Tolong kasih tim yang adil dan seimbang selalu dapet tim yang ga bener tapi lawan nya jago	tolong kasih tim yang adil dan seimbang selalu dapet tim yang ga bener tapi lawan nya jago

Penghapusan stopwords dilakukan untuk menghilangkan kata-kata yang sering muncul tetapi tidak memberikan makna signifikan, seperti kata sambung dan preposisi (misalnya "yang", "dan", "atau"). Berikut contoh tahap stopwords removal dapat dilihat pada Tabel 3.

Tabel 3. Contoh hasil stopwords removal

Case Folding	Setelah Stopword removal
game moba nomor 1 nih boss	game moba nomor 1 boss
tolong kasih tim yang adil dan seimbang selalu dapet tim yang ga bener tapi lawan nya jago	tolong kasih tim adil seimbang dapet tim ga bener lawan jago

Terakhir, proses tokenisasi memecah teks menjadi unit-unit kata yang lebih kecil, memungkinkan analisis yang lebih terfokus pada kata-kata penting dalam setiap ulasan. Berikut contoh tahap tokenization dapat dilihat pada Tabel 4.

Tabel 4. Contoh hasil Tokenization

Stopword removal	Setelah Tokenization
game moba nomor 1 boss	["game", "moba", "nomor", "1", "boss"]
tolong kasih tim adil seimbang dapet tim ga bener lawan jago	["tolong", "kasih", "tim", "adil", "seimbang", "dapet", "tim", "ga", "bener", "lawan", "jago"]

Semua tahapan ini bertujuan untuk mempersiapkan data ulasan agar siap dianalisis dalam proses selanjutnya.

3.4 Pemodelan Word2vec

Penelitian ini menggunakan Word2Vec untuk mengubah kata dalam ulasan menjadi representasi vektor numerik agar lebih mudah diproses oleh model pembelajaran mesin. Teknik ini menangkap hubungan semantik antar kata berdasarkan konteks penggunaannya dalam teks.

Metode *Skip-gram* dipilih dibandingkan *Continuous Bag of Words* (CBOW) karena lebih unggul dalam menangani dataset besar dan distribusi kata yang tidak seimbang. *Skip-gram* memprediksi kata-kata di sekitar (*context words*) berdasarkan kata target (*center word*), sedangkan CBOW melakukan sebaliknya. Dalam analisis ulasan pengguna, *Skip-gram* lebih efektif dalam mengenali kata-kata jarang muncul (*low-frequency words*), yang sering ditemukan dalam ulasan game dengan bahasa informal. Selain itu, metode ini lebih fleksibel dalam memahami hubungan semantik, terutama dalam teks yang mengandung istilah khas komunitas game. Oleh karena itu, *Skip-gram* dipilih untuk menghasilkan representasi kata yang lebih kaya dan akurat guna mendukung analisis sentimen.

3.5 *Pemodelan LSTM*

Penelitian ini menggunakan model Long Short-Term Memory (LSTM) untuk menganalisis pola dan konteks dalam ulasan pengguna. LSTM, sebagai jenis Recurrent Neural Network (RNN), dirancang untuk menangani ketergantungan jangka panjang dalam teks, sehingga lebih efektif dalam analisis sentimen. Implementasi model dilakukan dengan *TensorFlow* dan *Keras* karena kemudahannya dalam membangun arsitektur jaringan saraf serta efisiensinya dalam pemrosesan data skala besar.

Beberapa parameter utama telah ditentukan untuk meningkatkan performa model. LSTM menggunakan 100 unit untuk menangkap pola dalam data tanpa meningkatkan kompleksitas berlebihan. *Dropout* sebesar 0.2 diterapkan untuk mengurangi risiko *overfitting*. Model dioptimalkan dengan *Adam optimizer* karena kemampuannya menyesuaikan *learning rate* secara dinamis, mempercepat dan menstabilkan konvergensi dibandingkan dengan *Stochastic Gradient Descent* (SGD). Pelatihan dilakukan selama 20 *epochs* dengan *batch size* 64 untuk menjaga keseimbangan antara waktu pelatihan dan performa model. Dengan konfigurasi ini, model diharapkan mencapai akurasi tinggi tanpa mengalami *overfitting* atau *underfitting*.

3.6 *Evaluasi Model*

Evaluasi dilakukan untuk mengukur kinerja model dalam memprediksi sentimen ulasan berdasarkan data uji. Model yang telah dilatih diuji dengan data yang belum pernah dilihat sebelumnya untuk memastikan kemampuannya dalam menggeneralisasi, bukan sekadar menghafal data latih.

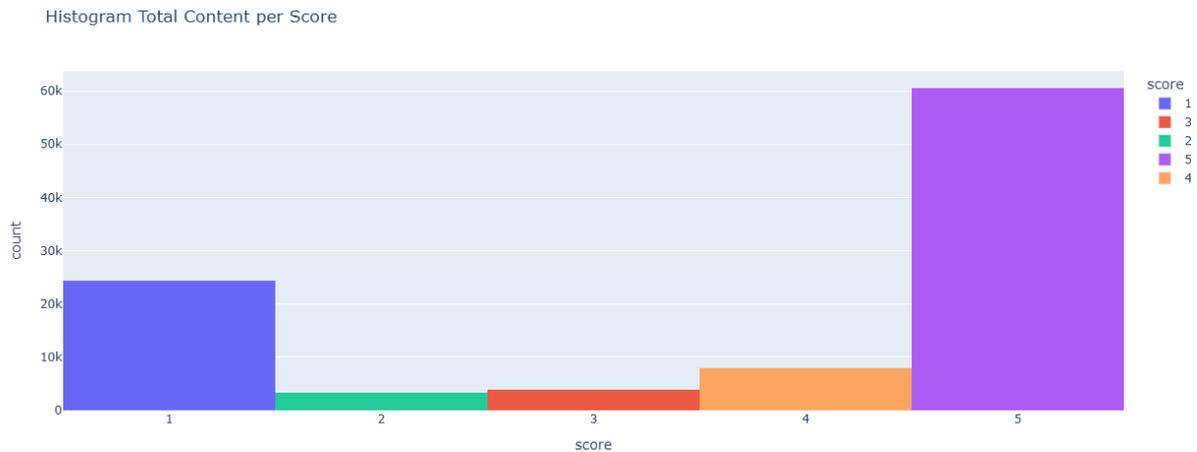
Metrik evaluasi yang digunakan meliputi akurasi, presisi, recall, dan F1-score untuk menilai sejauh mana model dapat mengklasifikasikan sentimen dengan benar. Selain itu, hasil evaluasi dibandingkan dengan metode lain seperti *Support Vector Machine* (SVM) dan *Naïve Bayes* guna memahami apakah model LSTM memberikan performa lebih baik dibandingkan teknik analisis teks yang telah terbukti efektif.

4 Hasil dan Pembahasan

Pada bagian ini, peneliti akan memaparkan hasil dan pembahasan dari penelitian yang dilakukan, yang mencakup tahapan Exploratory Data Analysis (EDA) untuk memahami karakteristik data, preprocessing untuk membersihkan dan mempersiapkan data, pemodelan Word2Vec untuk menghasilkan representasi vektor kata, pemodelan Long Short-Term Memory (LSTM) untuk memprediksi pola dalam teks, hasil pemodelan Word2Vec dan LSTM, serta perbandingan hasil pengujian dengan model Naive Bayes dan Support Vector Machine (SVM).

4.1 *Exploraty Data Analysis*

Pada bagian ini, dilakukan analisis eksploratif terhadap dataset untuk memahami struktur, pola, dan hubungan yang terdapat dalam data. EDA merupakan langkah awal yang penting dalam proses analisis data, yang bertujuan untuk memperoleh wawasan awal sebelum melakukan modelisasi lebih lanjut.



Gambar 2. Histogram score ulasan

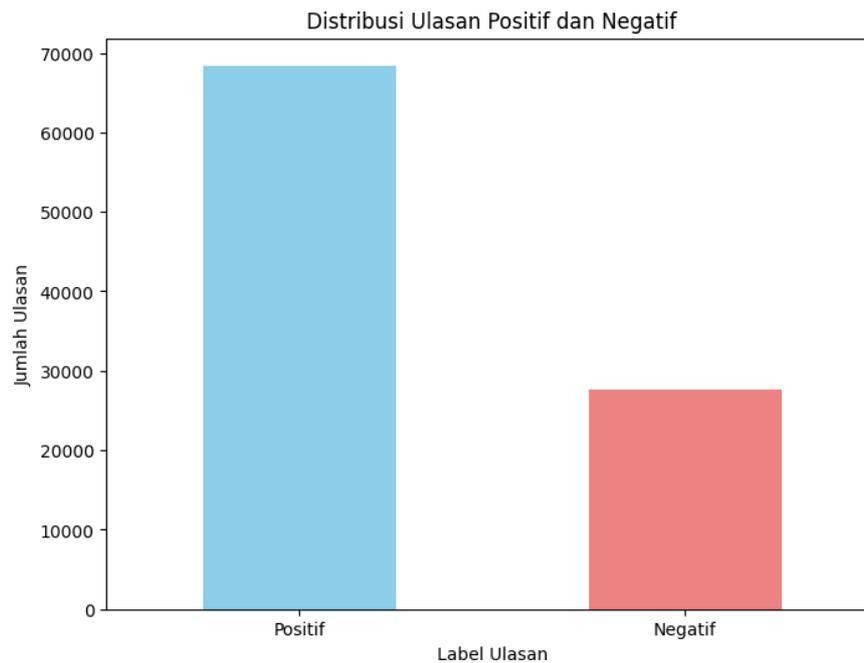
Pada Gambar 2 merupakan Histogram yang menunjukkan distribusi skor ulasan game *Mobile legends*, dengan sumbu x mewakili skor (1-5) dan sumbu y mewakili jumlah ulasan (0-70 ribu). Warna berbeda digunakan untuk setiap skor: biru (1), hijau (2), merah (3), oranye (4), dan ungu (5). Mayoritas ulasan memiliki skor 5 (lebih dari 60 ribu), diikuti oleh skor 1 (sekitar 25 ribu). Skor 2, 3, dan 4 memiliki jumlah ulasan yang jauh lebih rendah. Distribusi ini menunjukkan bahwa sebagian besar pengguna memberikan penilaian sangat tinggi atau sangat rendah, dengan sedikit ulasan yang memberikan skor sedang.



Gambar 3. Grafik Jumlah ulasan per hari

Selanjutnya Gambar 3 merupakan grafik garis yang menunjukkan jumlah ulasan game *Mobile legends* per hari. Sumbu x pada grafik ini mewakili tanggal, mulai dari 8 Desember 2024 hingga 19 Januari 2025, sementara sumbu y mewakili jumlah ulasan, dengan rentang dari 0 hingga 16 ribu. Grafik ini menunjukkan penurunan tajam dalam jumlah ulasan dari sekitar 16 ribu pada 8 Desember 2024 hingga hampir 0 pada 29 Desember 2024, dan tetap pada 0 hingga 19 Januari 2025. Penurunan ini mungkin mencerminkan perubahan dalam aktivitas pengguna atau periode waktu tertentu yang mempengaruhi jumlah ulasan yang diberikan.

Analisis sentimen disini menggunakan metode labeling berbasis aturan menunjukkan distribusi label yang signifikan dalam dataset ulasan game *Mobile legends*. Dengan menggunakan aturan jika skor < 3 dianggap negatif dan skor > 3 dianggap positif, Dataset ini mencakup 68.418 sentimen positif dan 27.656 sentimen negatif. Berikut pada Gambar 4 menampilkan grafik bar ulasan positif dan negatif.



Gambar 4. Distribusi ulasan positif dan negatif

Pada Gambar 4 Grafik menunjukkan bahwa jumlah ulasan positif (68.418) jauh lebih tinggi dibandingkan dengan ulasan negatif (27.656). Ini mengindikasikan bahwa mayoritas pengguna memberikan ulasan yang positif terhadap game *Mobile legends*. Dengan demikian, gambar tersebut memberikan gambaran visual tentang distribusi sentimen pengguna terhadap game *Mobile legends*, dengan mayoritas ulasan cenderung positif berdasarkan aturan yang diterapkan.

4.2 Preprocessing

Setelah melalui tahapan sebelumnya, data ulasan siap di *preprocessing* untuk analisis lebih lanjut. Pada tahap *cleaning*, karakter non-alfabet yang tidak relevan dihapus menggunakan *regular expression* (regex), memastikan hanya huruf yang dipertimbangkan. Selanjutnya, *case folding* diterapkan dengan mengubah seluruh teks menjadi huruf kecil untuk menghilangkan perbedaan kapitalisasi yang tidak relevan. *Stopwords* juga dihapus guna mengeliminasi kata-kata umum seperti "dan" atau "atau" yang tidak memberikan informasi signifikan. Terakhir, proses tokenisasi memisahkan teks menjadi unit kata lebih kecil, memungkinkan analisis yang lebih mendalam terhadap kata-kata penting. Dengan tahapan ini, data ulasan telah dipersiapkan secara optimal untuk analisis yang lebih efisien dan terfokus, seperti yang ditampilkan pada Gambar 5.

```
content \
0 Gawe paling rusak ,giliran musuh pada jago ,se...
1 Buat skin buat smua hero merata bosan itu2 mul...
2                               game stres
3           Game rusak dapet tim jelek mulu
4           GAME SEKARAT BENTAR LAGI MATI!!!!

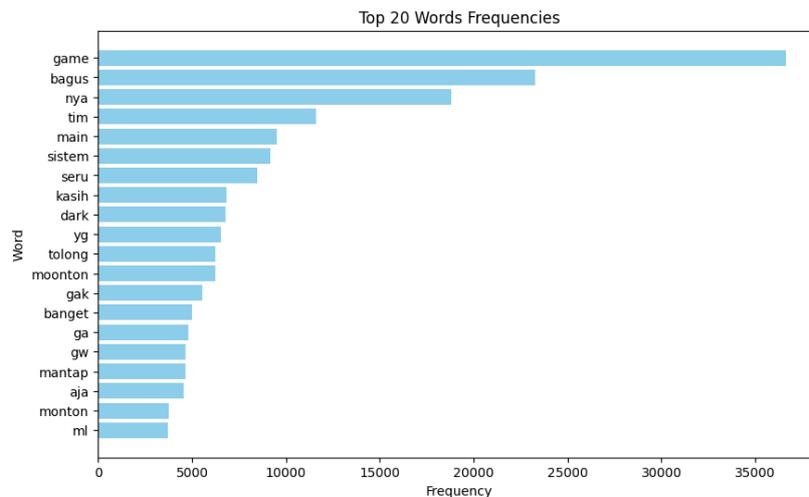
preprocessed_review
0 [gawe, rusak, giliran, musuh, jago, gua, dikas...
1 [skin, smua, hero, merata, bosan, mulu, heronya]
2                               [game, stres]
3           [game, rusak, dapet, tim, jelek, mulu]
4           [game, sekarat, bentar, mati]
```

Gambar 5. Preprocessing data ulasan

Gambar 5 menampilkan hasil *preprocessing* data ulasan game *Mobile legends*. Ulasan asli seperti "Gawe paling rusak, giliran musuh pada jago, se..." dan "game stres" telah melalui beberapa <http://sistemasi.ftik.unisi.ac.id>

tahap pemrosesan, termasuk pembersihan data untuk menghapus karakter non-alfabet, *case folding* untuk mengubah teks menjadi huruf kecil, penghapusan *stopwords* untuk menghilangkan kata-kata tidak signifikan, serta tokenisasi untuk memisahkan teks menjadi unit kata lebih kecil.

Hasil *preprocessing* menunjukkan daftar kata yang lebih bersih dan relevan, seperti [*gave*, *rusak*, *giliran*, *musuh*, *jago*] dan [*game*, *stres*]. Proses ini mengubah data mentah menjadi format yang lebih terstruktur, sehingga memungkinkan analisis yang lebih fokus pada kata-kata kunci yang informatif.

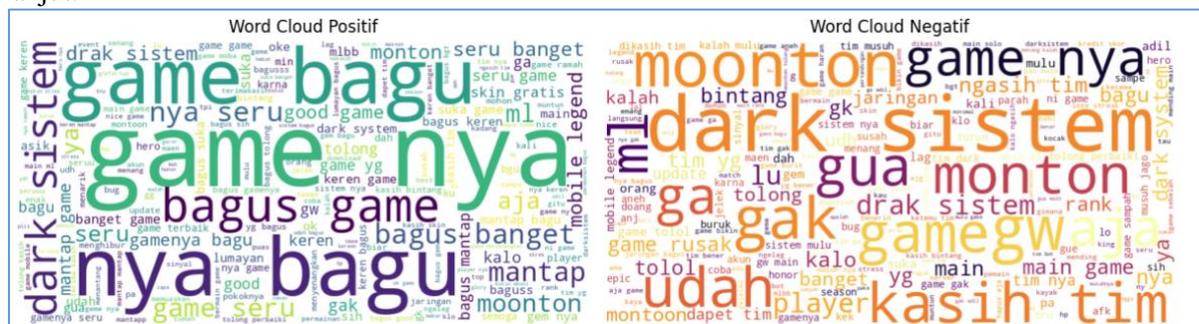


Gambar 6. 20 kata paling umum

Gambar 6 menampilkan distribusi frekuensi 20 kata paling umum dalam dataset ulasan pengguna terhadap game *Mobile legends*, yang divisualisasikan melalui grafik batang horizontal. Sumbu vertikal merepresentasikan kata-kata yang paling sering muncul, sementara sumbu horizontal menunjukkan jumlah kemunculannya.

Kata dengan frekuensi tertinggi adalah "game", yang muncul lebih dari 30.000 kali, diikuti oleh kata-kata lain seperti "bagus", "nya", "tim", "main", "sistem", "seru", "kasih", "dark", "yg", "tolong", "moonton", "gak", "banget", "ga", "gw", "mantap", "aja", "monton", dan "ml".

Keberadaan kata seperti "seru" dan "monoton" menunjukkan adanya sentimen positif dan negatif dalam ulasan pengguna. Sementara itu, kemunculan kata "ml" yang cukup signifikan mengindikasikan bahwa banyak ulasan secara eksplisit menyebutkan *Mobile legends*. Hasil analisis ini memberikan wawasan awal mengenai pola kemunculan kata dalam ulasan pengguna, yang dapat digunakan untuk memahami aspek-aspek utama yang sering dibahas dalam analisis sentimen lebih lanjut.



Gambar 7. Work cloud positif dan negatif

Selanjutnya Gambar 7 menampilkan *word cloud* yang menggambarkan distribusi kata-kata paling sering muncul dalam ulasan positif dan negatif terhadap *Mobile legends*. Visualisasi ini memberikan wawasan tentang aspek yang dihargai maupun dikeluhkan oleh pengguna.

Pada *word cloud* ulasan positif, kata-kata seperti "game", "bagus", "nya", "seru", dan "sistem" mendominasi. Hal ini menunjukkan bahwa banyak ulasan positif berfokus pada pengalaman bermain yang menyenangkan, kualitas permainan, serta fitur yang disukai oleh pengguna.

Sebaliknya, dalam *word cloud* ulasan negatif, kata-kata seperti "moonton", "dark", "sistem", "gak", dan "gua" sering muncul. Ini mengindikasikan bahwa keluhan utama terkait dengan sistem permainan, kebijakan pengembang, serta pengalaman bermain yang dianggap kurang memuaskan.

Hasil analisis ini memberikan pemahaman lebih mendalam mengenai persepsi pengguna terhadap *Mobile legends*. Informasi ini dapat dimanfaatkan dalam pengolahan sentimen berbasis NLP untuk mengevaluasi faktor-faktor yang mempengaruhi kepuasan pemain.

4.3 Pemodelan Word2vec

Penelitian ini menggunakan model *Word2Vec* untuk merepresentasikan kata-kata dalam ulasan pengguna *Mobile legends* ke dalam bentuk vektor numerik yang bermakna. *Word2Vec* merupakan teknik *word embedding* yang dapat menangkap hubungan semantik antar kata berdasarkan konteks penggunaannya dalam teks.

Sebelum membangun model *Word2Vec*, dilakukan proses tokenisasi dan pembersihan teks agar hanya kata-kata yang relevan digunakan dalam pelatihan model. Data yang telah diproses kemudian dikonversi ke dalam format daftar kata yang menjadi input bagi model *Word2Vec*.

Tabel 5. Tuning parameter model word2vec

Vector Size	Window	Min Count	Workers	Sg
100	5	5	4	1

Pada Tabel 5 dilakukan tuning parameter pada model *Word2Vec* untuk menghasilkan representasi kata yang optimal dalam analisis sentimen ulasan pengguna terhadap game *Mobile legends*. Model ini menggunakan *vector size* 100, yang cukup untuk menjaga hubungan semantik antar kata tanpa kehilangan terlalu banyak informasi. Parameter *window* ditetapkan sebesar 5, sehingga kata-kata dalam rentang lima kata dari kata target digunakan sebagai konteks. Untuk menghindari kata yang jarang muncul dan kurang signifikan, hanya kata yang muncul setidaknya lima kali (*min count* = 5) yang dimasukkan ke dalam model.

Proses pelatihan dipercepat dengan penggunaan empat *workers* atau *threads*. Model ini menggunakan metode *Skip-gram* (*sg* = 1) karena lebih efektif dalam menangkap hubungan semantik pada dataset besar serta lebih baik dalam mengenali kata-kata yang jarang muncul dibandingkan *Continuous Bag-of-Words (CBOW)*. Dengan konfigurasi ini, model *Word2Vec* diharapkan mampu menghasilkan representasi kata yang lebih akurat dan relevan untuk analisis sentimen.

```
Vektor dari kata 'permainan':
[ 0.05674804 -0.15711255 -0.07214909 0.14982821 -0.21375974 -0.50911105
 0.4346589 0.46626693 0.02638291 -0.32550555 0.04580178 -0.13332394
-0.28358525 -0.34790027 0.135715 0.264435 0.06112507 -0.06913309
 0.2315996 -0.41840324 -0.02119483 0.2825973 0.07308306 0.370937
 0.24709341 0.23031367 -0.11656094 0.19901286 -0.42176718 0.22731738
 0.22639129 -0.01447267 0.3049782 -0.37114364 -0.08060271 0.16971348
 0.15140724 -0.08901774 0.04098324 -0.5689394 -0.49808437 0.05568173
-0.22837332 0.00838695 0.56130713 -0.13756293 -0.23357904 0.09757728
-0.03096035 0.07219376 0.2602032 -0.07557776 -0.6194091 -0.15374331
 0.45628 -0.16688944 0.21356928 0.05738363 -0.3230444 0.6649496
-0.14851834 -0.37606862 0.32668695 -0.33022475 0.01562446 0.1683979
 0.02310284 0.24814482 -0.50989616 0.43787014 0.01626746 0.1340143
 0.0930692 0.06973626 0.12928021 -0.16206995 0.15178658 -0.19668184
 0.16042852 -0.06612018 -0.08962081 0.24212572 -0.23070118 0.44512987
-0.55656004 -0.36952698 0.10271475 0.23632306 0.06092165 -0.04870175
 0.2943026 0.5152526 0.02387745 0.3815591 0.69155234 0.375517
 0.0813994 -0.1108216 0.22739421 0.16272545]
```

Gambar 8. Contoh hasil vektor kata 'Permainan'

Gambar 8 menunjukkan hasil vektor representasi dari kata "permainan" yang dihasilkan oleh model *Word2Vec* setelah dilatih menggunakan ulasan pengguna terhadap game *Mobile legends*. Dalam *Word2Vec*, setiap kata direpresentasikan sebagai vektor dalam ruang berdimensi tinggi. Pada penelitian ini, vektor yang dihasilkan memiliki 100 dimensi, yang berarti setiap kata dalam model diwakili oleh 100 nilai numerik (*floating-point values*). Nilai-nilai ini menentukan posisi kata

"permainan" dalam ruang vektor berdasarkan hubungan semantik dan sintaksisnya dengan kata-kata lain dalam dataset.

Vektor ini terbentuk berdasarkan konteks kata dalam kalimat yang digunakan selama pelatihan. Kata yang memiliki makna atau penggunaan serupa dengan "permainan" akan memiliki representasi yang dekat dalam ruang vektor (*vector space*), seperti "game", "pertandingan", atau "hiburan". Sebaliknya, kata-kata yang tidak memiliki hubungan semantik yang kuat akan berada pada posisi yang lebih jauh. Dengan pendekatan ini, model dapat memahami hubungan antar kata dalam analisis sentimen, bukan sekadar membandingkan kata secara leksikal. Hal ini membantu model dalam mengenali pola makna yang lebih kompleks dalam ulasan pengguna.

4.4 Pemodelan LSTM

Pada tahap awal analisis sentimen, skor ulasan dikategorikan ke dalam label sentimen yang sesuai untuk pemodelan. Penelitian ini menggunakan aturan klasifikasi biner, di mana skor 1 dan 2 dikategorikan sebagai sentimen negatif (label = 0), sedangkan skor 4 dan 5 dikategorikan sebagai sentimen positif (label = 1). Skor 3 dianggap sebagai netral dan tidak digunakan dalam pemodelan. Proses konversi ini dilakukan dengan menerapkan fungsi *categorize_review* pada kolom skor menggunakan metode *.apply()* dari Pandas. Setelah itu, label yang dihasilkan dikonversi menjadi bentuk *one-hot encoding* menggunakan fungsi *to_categorical* dari Keras agar sesuai untuk model klasifikasi biner.

Selanjutnya, teks ulasan dikonversi ke bentuk numerik agar dapat diproses oleh model *deep learning*. Model *Word2Vec* yang telah dilatih sebelumnya digunakan untuk menyediakan daftar kata unik (*vocabulary*) beserta vektornya. Langkah pertama adalah ekstraksi kosakata menggunakan *w2v_model.wv.index_to_key*, yang menghasilkan daftar kata unik. Setiap kata kemudian diberikan indeks unik untuk membentuk *word_index*, yaitu *dictionary* yang memetakan kata ke indeks numerik. Ulasan dikonversi menjadi daftar indeks kata dengan mencocokkan setiap kata dalam ulasan terhadap *word_index*, yang dilakukan dengan fungsi *text_to_sequence*. Proses ini menghasilkan daftar indeks kata yang sesuai dengan kosakata dari *Word2Vec*.

Karena panjang ulasan dalam dataset bervariasi, dilakukan langkah *padding sequence* untuk standarisasi panjang input. Agar model *LSTM* dapat memproses data dalam *batch* dengan ukuran yang seragam, setiap *sequence* dipadatkan menjadi panjang yang sama. Dalam penelitian ini, panjang maksimal (*max_sequence_length*) ditentukan sebagai 100 kata, sehingga setiap ulasan dipotong atau dipadatkan menjadi panjang 100 token. *Padding* dilakukan menggunakan fungsi *pad_sequences* dari Keras, yang menambahkan nol (0) pada awal *sequence* untuk menyamakan panjangnya.

Terakhir, dataset dibagi menjadi dua bagian dengan rasio 80:20, di mana 80% data digunakan sebagai data latih (*X_train, y_train*), sementara 20% sisanya digunakan sebagai data uji (*X_test, y_test*). Pembagian ini bertujuan untuk menguji kinerja model pada data yang belum pernah dilihat sebelumnya, sehingga dapat mengevaluasi kemampuan generalisasi model terhadap data baru. Berikut pada Tabel 6 merupakan tuning parameter untuk model LSTM.

Tabel 6. Tuning parameter model LSTM

Jumlah Unit LSTM	Epoch	Batch Size	Dropout	Optimizer
100	20	64	0.2	Adam

Setelah data dikonversi ke dalam bentuk numerik, model *Long Short-Term Memory* (LSTM) dibangun untuk melakukan analisis sentimen. LSTM merupakan arsitektur *Recurrent Neural Network* (RNN) yang efektif dalam menangani data sekuensial, seperti teks. Model ini dikembangkan menggunakan API *Sequential* dari Keras, dimulai dengan *Embedding Layer* yang berfungsi mengonversi indeks kata menjadi representasi vektor berdasarkan model *Word2Vec*. Matriks *embedding* dibuat dengan menyalin vektor kata yang telah dilatih dari *Word2Vec* ke dalam array, sehingga model dapat memanfaatkan representasi kata yang sudah dipelajari sebelumnya. *Embedding* ini dikonfigurasi agar tidak berubah selama pelatihan untuk menjaga informasi yang telah diperoleh dari model *Word2Vec*.

Untuk mengurangi *overfitting*, diterapkan teknik *Spatial Dropout*, yang menghilangkan informasi dari beberapa unit dalam *embedding layer* selama pelatihan. Lapisan berikutnya adalah *LSTM Layer*, yang berfungsi menangkap pola dalam urutan kata. Ukuran unit LSTM ditetapkan sebanyak 100, dengan *dropout* dan *recurrent dropout* masing-masing sebesar 0.2 untuk mengurangi risiko *overfitting*

<http://sistemasi.ftik.unisi.ac.id>

pada unit memori berulang. Lapisan terakhir adalah *Fully Connected Layer* dengan dua neuron, sesuai dengan jumlah kelas dalam klasifikasi (negatif dan positif). Fungsi aktivasi yang digunakan adalah *softmax*, karena output yang dihasilkan berbentuk probabilitas untuk masing-masing kelas.

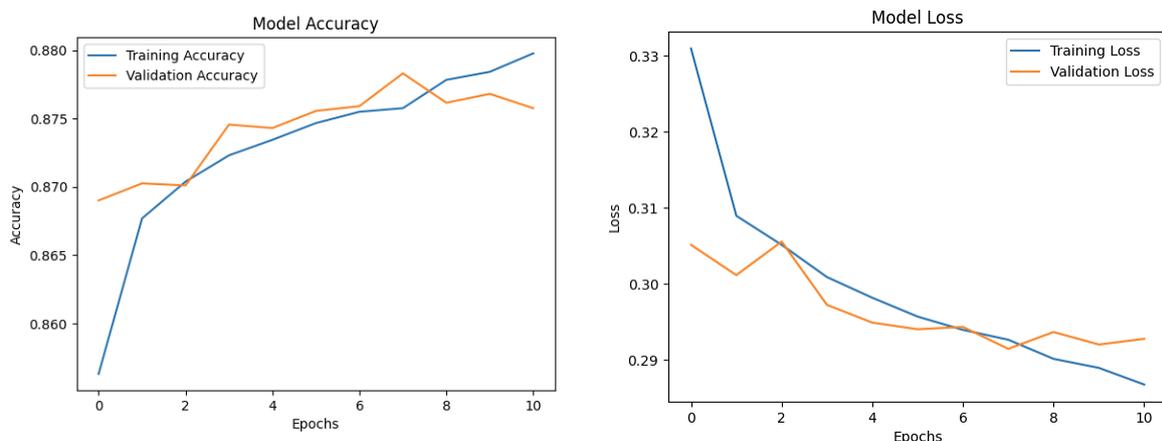
Setelah arsitektur model selesai, model dikompilasi menggunakan *categorical_crossentropy* sebagai fungsi *loss*, karena klasifikasi dilakukan pada dua kelas dengan *one-hot encoding*. Optimizer yang digunakan adalah *Adam*, karena algoritma ini dapat menyesuaikan *learning rate* secara dinamis, sehingga banyak digunakan dalam aplikasi NLP. Untuk mengukur performa model, digunakan metrik akurasi, yang menghitung persentase prediksi yang benar.

Model dilatih dengan konfigurasi tertentu, di mana jumlah *epoch* ditetapkan sebanyak 20. Untuk mencegah *overfitting*, proses pelatihan dapat dihentikan lebih awal menggunakan teknik *Early Stopping* jika tidak ada peningkatan performa. *Batch size* yang digunakan adalah 64, yang menentukan jumlah sampel yang diproses dalam satu iterasi. Selain itu, sekitar 20% dari dataset dialokasikan sebagai data validasi untuk memantau performa model selama pelatihan.

4.5 Hasil Pemodelan Word2vec dan LSTM

Model Word2Vec dan LSTM telah diterapkan pada dataset ulasan game *mobile legends* sebanyak 100.000 data dengan rasio pembagian 80:20 untuk pelatihan dan pengujian. Pendekatan ini menggabungkan keunggulan Word2Vec dalam representasi kata yang kaya dengan kemampuan LSTM dalam menangani urutan data yang panjang dan kompleks. Setelah proses pelatihan model selesai, evaluasi dilakukan untuk mengukur kinerja model menggunakan beberapa metrik utama seperti akurasi, precision, recall, dan f1-score.

Hasil metrik evaluasi model meliputi akurasi sebesar 87,88%, precision sebesar 91,33%, recall sebesar 92,21%, dan f1-score sebesar 91,77%. Hasil ini menunjukkan bahwa model memiliki performa yang baik, terutama dalam mengklasifikasikan kelas 1, dengan nilai precision, recall, dan f1-score yang tinggi. Evaluasi ini menandakan bahwa model tersebut efektif dalam mengidentifikasi kelas positif dan memberikan hasil yang cukup akurat.



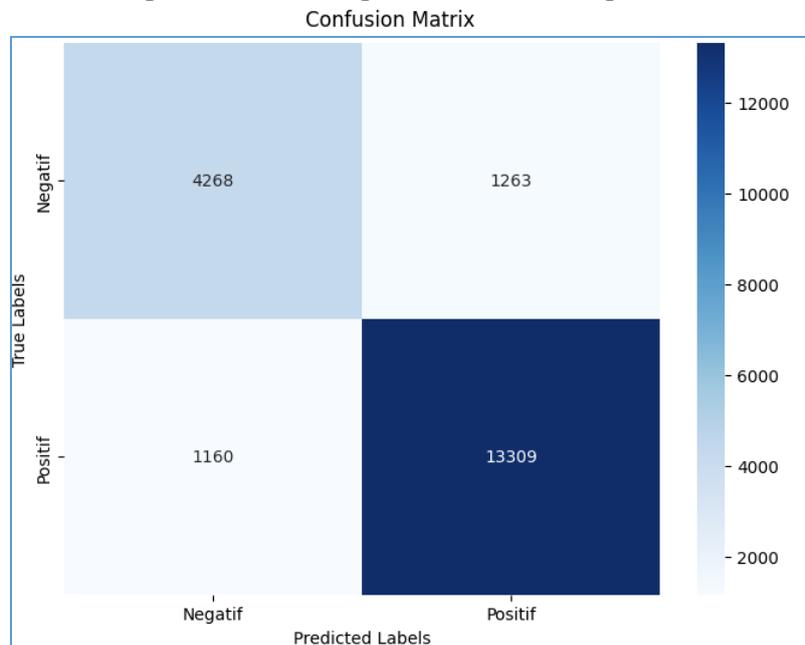
Gambar 9. Grafik akurasi dan loss model

Gambar 9 menunjukkan grafik akurasi model selama proses pelatihan. Sumbu x merepresentasikan jumlah epoch, sedangkan sumbu y menunjukkan tingkat akurasi. Terdapat dua garis: satu untuk akurasi pada dataset pelatihan dan satu lagi pada dataset validasi. Grafik ini menunjukkan bahwa akurasi pada dataset pelatihan meningkat secara bertahap dari sekitar 0,86 pada epoch pertama hingga mencapai 0,88 pada epoch ke-10. Akurasi pada dataset validasi juga mengalami tren peningkatan serupa, dimulai dari 0,87 dan mencapai 0,875 pada akhir epoch. Peningkatan yang konsisten ini menunjukkan bahwa model mampu belajar pola sentimen dengan baik dan semakin akurat dalam melakukan prediksi.

Seperti grafik akurasi, Grafik nilai loss selama proses pelatihan, sumbu x menunjukkan jumlah epoch, sementara sumbu y menunjukkan nilai loss. Terdapat dua garis yang masing-masing merepresentasikan loss pada dataset pelatihan dan dataset validasi. Pada epoch pertama, nilai loss dataset pelatihan dimulai dari sekitar 0,33 dan secara bertahap menurun hingga 0,285 pada epoch ke-10. Nilai loss pada dataset validasi juga menunjukkan tren penurunan, dimulai dari 0,305 dan mencapai sekitar 0,29 pada akhir epoch. Penurunan nilai loss ini menunjukkan bahwa model semakin

baik dalam memahami pola sentimen, dengan kesalahan prediksi yang semakin kecil seiring bertambahnya jumlah epoch.

Secara keseluruhan, kedua grafik ini mengindikasikan bahwa model mengalami peningkatan akurasi dan penurunan loss secara stabil selama pelatihan. Tidak terdapat indikasi overfitting yang signifikan, karena akurasi dan loss pada dataset validasi tidak mengalami lonjakan atau stagnasi yang mencolok. Hal ini menunjukkan bahwa model memiliki kemampuan generalisasi yang baik, mampu mengadaptasi pola dari data pelatihan, dan tetap efektif dalam memprediksi data baru



Gambar 10. Confusion matrix

Kemudian pada Gambar 10 merupakan Confusion matrix yang digunakan untuk menilai kinerja model klasifikasi. Matriks ini menunjukkan jumlah prediksi yang benar dan salah untuk dua kelas, yaitu "Negatif" dan "Positif". Elemen-elemen yang ada dalam matriks tersebut adalah True Negatives (TN) sebesar 4268, yang berarti data negatif diprediksi dengan benar sebagai negatif; False Positives (FP) sebesar 1263, yaitu data negatif yang diprediksi secara salah sebagai positif; False Negatives (FN) sebesar 1160, yang menunjukkan data positif yang diprediksi secara salah sebagai negatif; dan True Positives (TP) sebesar 13309, yang berarti data positif diprediksi dengan benar sebagai positif. Confusion matrix ini membantu kita memahami seberapa baik model dalam memprediksi kelas-kelas tersebut. Nilai tinggi untuk True Positives (13309) menunjukkan bahwa model cukup baik dalam memprediksi kelas positif. Namun, adanya False Positives (1263) dan False Negatives (1160) menunjukkan bahwa masih ada beberapa kesalahan prediksi. Dari matriks ini, kita bisa menghitung metrik evaluasi seperti akurasi, presisi, dan recall untuk memahami kinerja model lebih lanjut.

Prediksi yang Salah			
	Review	True Label	Predicted Label
5112	bagus bgth	0	1
10075	gimana ini montonn tiap hari ditrol mulu, maen game buat seneng' eh ini malah bikin emosi, kali' dapet tim tuh jago minimal ngerti maen lah kaga jago juga	0	1
7278	game yang jelek bayak dak sistem tim yang ga bener	0	1
2473	Di kasih tim yang gak bisa main terus sedangkan musuh jago terus	0	1
5098	GAME TAIIII G PERNAH DIKASIH MENANG	1	0
843	hapus player tolol nya lah cape gw bngst	0	1
2020	Game party. Solo tim ngetroll	1	0
19075	game ini menemani aku sejak dari kecil perubahannya gk jauh ²	1	0
7836	Lumayan lah keren semua apa lagi ada karakter baru pokoknya seru megisi waktu	0	1
5015	pas maen tiba tiba ngebug balek ke lobby	0	1

Gambar 11. Sample prediksi yang salah

Kesalahan klasifikasi dalam ulasan negatif dapat disebabkan oleh beberapa faktor. Pertama, bahasa tidak formal dan slang membuat model sulit mengenali ulasan sebagai negatif. Contohnya, pada Gambar 11 ulasan seperti "*bagus bgth*" dan "*game yang jelek bayak dak sistem yang ga bener*" menggunakan bahasa yang tidak standar dan emosional.

Kemudian model mungkin kesulitan memahami konteks penuh dari kalimat panjang yang tidak beraturan, seperti dalam ulasan "*gimana ini montonn tiap hari ditrol mulu, maen game buat seneng' eh ini malah bikin emosi, kali' dapet tim tuh jago minimal ngerti maen lah kaga jago juga*".

Dan juga beberapa ulasan mengandung kata-kata positif yang sebenarnya digunakan dalam konteks negatif, seperti "*bagus bgth*" yang sebenarnya sarkastik. Kesalahan ini menunjukkan bahwa model perlu ditingkatkan untuk menangani variasi bahasa dan konteks. Peningkatan dapat dilakukan dengan melatih model menggunakan dataset yang lebih beragam atau menggunakan teknik pemrosesan bahasa alami yang lebih canggih.

4.6 Perbandingan Hasil Pengujian

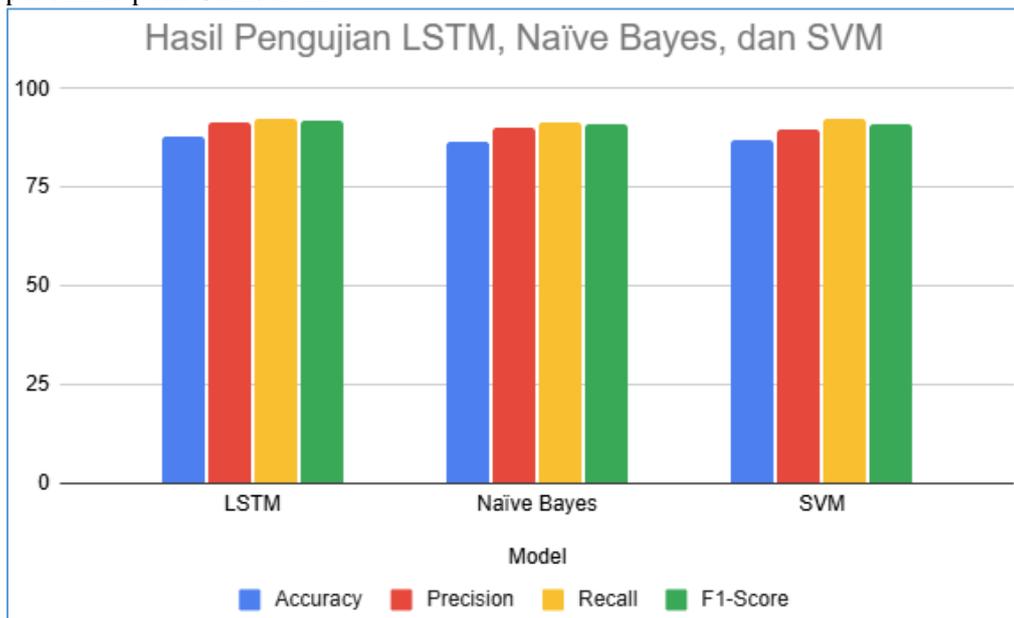
Pada bagian ini, peneliti menguji model yang dibangun antara LSTM, Naïve Bayes, dan SVM. Pengujian yang dilakukan mendapatkan nilai akurasi, presisi, recall, dan f1-score. Berikut merupakan table perbandingan hasil pengujiannya.

Tabel 7. Tuning parameter model LSTM

Model	Akurasi	Presisi	Recall	F1-Score
LSTM	87,88%	91,33%	92,21%	91,77%
Naive bayes	86,68%	90%	91,57%	90,78%
SVM	87,09%	89,79%	92,48%	91,12%

Berdasarkan pada Tabel 7 yaitu hasil pengujian yang dilakukan, LSTM menunjukkan performa terbaik di antara ketiga model, dengan akurasi 87,88%, precision 91,33%, recall 92,21%, dan F1-score 91,77%. Hal ini menunjukkan bahwa LSTM sangat baik dalam mengklasifikasikan sentimen ulasan pengguna *Mobile legends*, dengan kemampuan yang tinggi dalam menemukan ulasan positif (recall) dan menghindari kesalahan dalam prediksi positif (precision). Naïve Bayes, meskipun sedikit lebih sederhana, menghasilkan akurasi 86,68%, dengan precision 90% dan recall 91,57%, yang menunjukkan bahwa model ini cukup efektif, meskipun sedikit tertinggal dibandingkan LSTM dalam hal keseimbangan antara precision dan recall. Sedangkan SVM memiliki akurasi 87,09%, sedikit lebih rendah daripada LSTM, namun dengan recall yang sangat tinggi yaitu 92,48%, menunjukkan kemampuannya dalam menangkap hampir semua ulasan positif yang ada, meskipun ada sedikit penurunan dalam precision (89,79%). Secara keseluruhan, LSTM adalah model yang paling unggul, diikuti oleh SVM yang sangat baik dalam recall, sementara Naïve Bayes tetap memberikan hasil yang solid meskipun sedikit lebih rendah performanya. LSTM paling ideal jika mencari keseimbangan antara precision, recall, dan F1-score, sedangkan SVM cocok untuk menangkap lebih banyak ulasan positif meskipun dengan sedikit kompromi pada precision. Naïve Bayes bisa menjadi pilihan jika

kecepatan dan kesederhanaan menjadi prioritas utama. Perbandingan hasil pengujian dalam bentuk grafik dapat dilihat pada Gambar 12.



Gambar 12. Grafik hasil perbandingan LSTM, naïve bayes, dan SVM

Pada Gambar 12 Grafik ini menunjukkan hasil pengujian tiga model yaitu LSTM, Naïve Bayes, dan SVM berdasarkan empat metrik evaluasi: Akurasi, Precision, Recall, dan F1-Score. Secara umum, ketiga model memiliki performa yang cukup baik dengan skor di atas 80% untuk semua metrik. LSTM dan Naïve Bayes memiliki skor yang hampir seragam di semua metrik, sementara SVM sedikit lebih rendah dalam akurasi tetapi masih kompetitif dalam Precision, Recall, dan F1-Score. Naïve Bayes tampaknya memiliki performa paling konsisten di antara ketiganya, sedangkan LSTM dan SVM menunjukkan sedikit variasi. Hal ini menunjukkan bahwa ketiga model ini dapat digunakan secara efektif, tetapi pemilihan model terbaik tergantung pada kebutuhan spesifik dari tugas klasifikasi yang dilakukan.

5 Kesimpulan

Penelitian ini menunjukkan bahwa kombinasi Word2Vec dan LSTM efektif dalam analisis sentimen ulasan pengguna Mobile Legends, dengan akurasi 87,88% serta keseimbangan precision dan recall yang baik. LSTM terbukti lebih unggul dibandingkan Naïve Bayes dan SVM dalam klasifikasi sentimen. Meskipun model memiliki performa yang baik, terdapat keterbatasan seperti ketidakseimbangan dataset dan kesulitan menangani bahasa informal. Untuk penelitian selanjutnya, disarankan eksplorasi model NLP lebih canggih seperti BERT serta penambahan kategori sentimen netral agar analisis lebih akurat dan representatif. Penelitian ini dapat memberikan wawasan bagi pengembang game dalam memahami opini pengguna serta menawarkan pendekatan deep learning untuk meningkatkan pengalaman bermain dan kepuasan pemain Mobile Legends.

Referensi

- [1] R. N. Rahmadiani, A. H. Nasution, and M. F. Sugihartanto, "Analisis Faktor-Faktor yang mempengaruhi Keputusan Pembelian Virtual Item dalam *Online Mobile Game* (Studi Kasus: *Mobile Legends Bang Bang*)," *Jurnal Teknis ITS*, 2021.
- [2] F. D. Oktaviansyah and R. P. Tutiasri, "Keputusan Konsumen dalam Pembelian *Virtual Items* pada *Game Online Mobile legends: Bang Bang*," *Dawatuna: Journal of Communication and Islamic Broadcasting*, 2023, doi: 10.47467/dawatuna.v4i2.3655.
- [3] F. I. Wibowo and A. Febriandirza, "Analisis Sentimen Ulasan Pengguna *Game Pubg* di *Google Play Store* menggunakan *Algoritma Naïve Bayes*," *Jurnal Sistem Komputer dan Informatika (JSON)*, vol. 5, no. 3, p. 590, Mar. 2024, doi: 10.30865/json.v5i3.7264.

- [4] B. Setiawan, K. A. Baihaqi, E. Nurlaelasari, and H. H. Handayani, “Analisis Sentimen Ulasan Aplikasi Identitas Kependudukan Digital menggunakan *Algoritma Logistic Regression* dan *K-Nearest Neighbor*,” *Technology and Science (BITS)*, vol. 6, no. 1, pp. 533–540, 2024, doi: 10.47065/bits.v6i1.5389.
- [5] B. Ramadhani and R. R. Suryono, “Komparasi Algoritma *Naïve Bayes* dan *Logistic Regression* untuk Analisis Sentimen *Metaverse*,” *Jurnal Media Informatika Budidarma*, 2024, doi: 10.30865/mib.v8i2.7458.
- [6] S. Mujilahwati and M. A. Ubaydillah, “Analisis Sentimen berbasis Aspek pada Ulasan *Game Mobile Legends* dengan Pendekatan *Decision Tree* untuk Evaluasi Pengalaman Pengguna,” *Jurnal Pendidikan dan Teknologi Indonesia*, vol. 4, no. 11, pp. 325–333, Dec. 2024, doi: 10.52436/1.jpti.465.
- [7] L. Zhang, S. Wang, and B. Liu, “*Deep Learning for Sentiment Analysis: A Survey*,” 2018.
- [8] R. K. Yadav, L. Jiao, M. Goodwin, and O. C. Granmo, “*Positionless Aspect based Sentiment Analysis using Attention Mechanism[Formula Presented]*,” *Knowl Based Syst*, vol. 226, Aug. 2021, doi: 10.1016/j.knsys.2021.107136.
- [9] W. Widayat, “Analisis Sentimen *Movie Review* menggunakan *Word2Vec* dan Metode *LSTM Deep Learning*,” *Jurnal Media Informatika Budidarma*, vol. 5, no. 3, p. 1018, Jul. 2021, doi: 10.30865/mib.v5i3.3111.
- [10] N. A. Dirfas and V. R. S. Nastiti, “Perbandingan Kinerja *Pre-Trained Word Embedding* pada Klasifikasi Sentimen Ulasan Produk Tokopedia dengan *Long Short-Term Memory (LSTM)*,” *Technology and Science (BITS)*, vol. 6, no. 2, 2024, doi: 10.47065/bits.v6i2.5634.
- [11] G. D. Gennaro, A. Buonanno, and F. A. N. Palmieri, “*Considerations about Learning Word2Vec*,” *Journal of Supercomputing*, vol. 77, no. 11, pp. 12320–12335, Nov. 2021, doi: 10.1007/s11227-021-03743-2.
- [12] F. W. Kurniawan and W. Maharani, “Analisis Sentimen *Twitter Bahasa Indonesia* dengan *Word2Vec*,” 2020, [Online]. Available: <https://code.google.com>
- [13] P. F. Muhammad, R. Kusumaningrum, and A. Wibowo, “*Sentiment Analysis using Word2vec and Long Short-Term Memory (LSTM) for Indonesian Hotel Reviews*,” in *Procedia Computer Science*, Elsevier B.V., 2021, pp. 728–735. doi: 10.1016/j.procs.2021.01.061.
- [14] R. Sharma, S. Shrivastava, S.K. Singh, A. Kumar, S. Saxena, and R.K. Singh, “*Deep-Abppred: Identifying Antibacterial Peptides in Protein Sequences using Bidirectional LSTM with Word2vec*,” *Brief Bioinform*, vol. 22, no. 5, Sep. 2021, doi: 10.1093/bib/bbab065.
- [15] M. V. Shyahrin, Y. Sibaroni, and D. Puspendari, “Penerapan Metode *Long Short-Term Memory* dan *Word2Vec* dalam Analisis Sentimen Ulasan pada Aplikasi *Ferizy LSTM* dan *Word2Vec Application for Sentiment Analysis of Reviews on Ferizy*,” *Jurnal Buana Informatika*, vol. 22, no. 4, pp. 833–842, 2023.
- [16] Aakash, S. Gupta, and A. Noliya, “*URL-based Sentiment Analysis of Product Reviews using LSTM and GRU*,” *Procedia Comput Sci*, vol. 235, pp. 1814–1823, 2024, doi: 10.1016/j.procs.2024.04.172.
- [17] A. H. Nurdy, A. Rahim, and Arbansyah, “Analisis Sentimen Ulasan *Game Stumble Guys* pada *Playstore* menggunakan Algoritma *Naïve Bayes*,” *Teknika*, vol. 13, no. 3, pp. 388–395, Sep. 2024, doi: 10.34148/teknika.v13i3.993.
- [18] V. Chandradev, I. M. A. D. Suarjaya, and I. P. A. Bayupati, “Analisis Sentimen Review Hotel menggunakan Metode *Deep Learning BERT*,” *Jurnal Buana Informatika*, vol. 14, no. 2, pp. 107-116, 2023.