

Implementasi Model *Long-Short Term Memory* (LSTM) pada Klasifikasi Teks Data SMS Spam Berbahasa Indonesia

Erico Dwi Pratama
Mathematics Departement, Universitas
Negeri Surabaya, Indonesia
erico.18061@mhs.unesa.ac.id

Abstract— Di Tahun 2022, tercatat penggunaan internet masyarakat Indonesia sebesar 73.7% dari total populasi untuk aktivitas sehari-hari. Internet telah membawa kemudahan bagi masyarakat dalam media sosial dan komunikasi. Ekspektasi yang tercipta ketika semakin berkembangnya internet dan aplikasi perpesanan adalah tidak lagi dibutuhkan penggunaan nomor telepon dan SMS. Fakta mengenai hal ini justru berbalik dimana penggunaan internet dan aplikasi perpesanan saat ini tidak menghilangkan penggunaan SMS di Indonesia. Fenomena spam dalam SMS dapat terjadi karena beberapa faktor. Salah satu faktor tersebut adalah kurangnya wawasan masyarakat dalam membedakan pesan yang berupa sebuah spam atau tidak. Klasifikasi teks menggunakan metode *Long-Short Term Memory* (LSTM) dilakukan untuk mengklasifikasi data SMS ke dalam tiga kategori, yaitu promo, penipuan dan normal. Untuk melakukan klasifikasi teks perlu melibatkan beberapa proses, yaitu: *Data Collection*, *Data Preprocessing*, *Word Representation*, *Classification*, dan *Evaluation/Testing*. Data terdiri dari 1143 SMS yang telah diberi label serta menggunakan *framework tensorflow* dan *keras* dalam penelitian ini. Dengan menggunakan algoritma LSTM, model penelitian yang dibentuk mencapai akurasi sebesar 94%. Metode LSTM menghasilkan hasil yang lebih baik dari kedua metode pembandingnya. Model dengan metode LSTM menghasilkan nilai *accuracy* 27% lebih baik dibandingkan dengan metode *Naive Bayes* dan 24% lebih baik dari metode KNN.

Keywords— *Klasifikasi Teks, SMS, Spam, Long-Short Term Memory*.

I. PENDAHULUAN

Di Tahun 2022, tercatat penggunaan internet masyarakat Indonesia sebesar 73.7% dari total populasi untuk aktivitas sehari-hari[1]. Hal ini mengindikasikan besarnya penggunaan internet pada saat ini. Masyarakat semakin terbuka dengan kehadiran internet dan menjadikan internet sebagai kebutuhan. Masyarakat dapat mengakses berbagai informasi sesuai kebutuhannya hanya dengan *browsing* melalui situs web yang tersedia[2]. Tidak hanya digunakan untuk pada kegiatan atau sektor tertentu saja, internet telah banyak membantu masyarakat dalam memajukan berbagai hal. Mulai dari sektor industri makanan, pakaian, media informasi, media sosial dan media komunikasi.

Internet telah membawa kemudahan bagi masyarakat dalam media sosial dan komunikasi. Dengan berkembangnya teknologi internet dan perangkat elektronik, pertukaran informasi digital juga semakin cepat. Hal ini diperkuat dengan bermuncunya beragam aplikasi perpesanan berbasis internet yang bernama *messenger*. Ekspektasi yang tercipta ketika

semakin berkembangnya internet dan aplikasi perpesanan adalah tidak lagi dibutuhkan penggunaan nomor telepon dan SMS. Fakta mengenai hal ini justru berbalik dimana penggunaan internet dan yang berkembang pesat saat ini tidak menghilangkan penggunaan SMS di Indonesia. Nomor telepon tetap digunakan dalam pembuatan akun seperti *email* dan sosial media, termasuk aplikasi perpesanan. Sehingga hal ini tidak sepenuhnya menghentikan penggunaan nomor telepon dan SMS.

Short Message Service atau biasa disingkat SMS adalah salah satu layanan bagian pesan dari perusahaan komunikasi yang ada pada telepon seluler. SMS sendiri dibagi menjadi beberapa jenis. Menurut Reviantika, dalam artikelnya mendeskripsikan SMS terbagi menjadi dua jenis, spam dan non spam[3]. SMS dengan jenis spam cenderung mengganggu kenyamanan pengguna dan biasa berisikan penipuan dan promosi. SMS jenis non spam adalah pesan-pesan formal yang berisikan informasi penting. Fenomena spam dalam SMS dapat terjadi karena beberapa faktor. Salah satu faktor tersebut adalah lemahnya sistem keamanan dalam SMS. Berbeda dengan aplikasi perpesanan yang secara kontinu mengembangkan keamanan pada aplikasinya, seperti pesan yang terenkripsi secara *end-to-end*, atau teknologi lainnya. Hal lainnya adalah kurangnya wawasan masyarakat dalam membedakan pesan yang berupa sebuah spam atau tidak.

Adapun salah satu cara yang dapat menangani kurangnya wawasan masyarakat dalam membedakan pesan yang berupa sebuah spam atau tidak. Klasifikasi merupakan sebuah proses untuk mencari model dengan cara mengkategorikan sehingga dapat memprediksi kelas yang tidak mempunyai label[4]. Beberapa penelitian diantaranya dilakukan oleh Budi Indarto pada tahun 2016 mengenai klasifikasi SMS dengan menggunakan metode *Naive Bayes*[5]. Ulviyana pada tahun 2020 juga melakukan klasifikasi teks pada *email* spam menggunakan metode KNN[6].

Selain *Naive Bayes* dan KNN, terdapat metode lain yang dapat digunakan untuk klasifikasi teks. Salah satunya adalah *Long-Short Term Memory* atau biasa disingkat dengan LSTM[7]. Winda Kurnia Sari pada tahun 2020 melakukan klasifikasi teks artikel berita menggunakan LSTM[8]. Di tahun berikutnya, Agung Nur Hidayat melakukan klasifikasi teks ujaran kebencian terhadap kicauan Twitter menggunakan metode LSTM[9]. Masih di tahun yang sama, Yahyadi dan Latifah juga melakukan klasifikasi teks pada kicauan Twitter terkait kebijakan PPKM menggunakan metode LSTM[10].

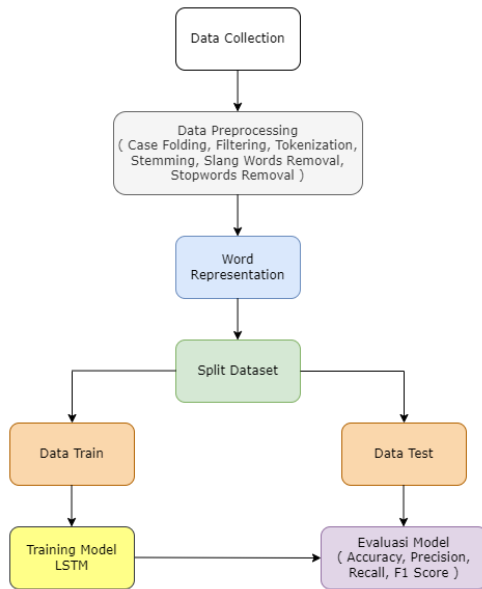
Berdasarkan uraian diatas, maka penting dilakukan klasifikasi teks terhadap kasus SMS spam dengan

menggunakan metode LSTM. Penelitian ini diharapkan dapat mendeteksi SMS spam yang terdiri dari promosi maupun penipuan serta memberikan informasi, edukasi, atau *awareness* kepada masyarakat. Adapun batasan pada penelitian ini adalah data yang digunakan merupakan *tweet* pengguna Twitter berjumlah 1143 data yang telah diberi label serta *framework* yang digunakan adalah *tensorflow* dan *keras*.

II. METODE

A. Klasifikasi Teks

Dalam penelitian ini, untuk melakukan klasifikasi teks perlu melibatkan beberapa proses, yaitu: *Data Collection*, *Data Preprocessing*, *Word Representation*, *Classification*, dan *Evaluation/Testing*.



Gambar 1. Proses Klasifikasi Teks

Langkah pertama yang dilakukan dalam klasifikasi teks adalah *Data Collection*. Data berupa kumpulan teks SMS dan diperoleh dari situs *open source*.

Langkah selanjutnya adalah *Data Preprocessing / Cleaning of Dataset*. Data teks SMS yang digunakan masih mengandung beberapa *noise*. Karena itu data tersebut perlu untuk diproses terlebih dahulu. Proses *Cleaning of Dataset* meliputi:

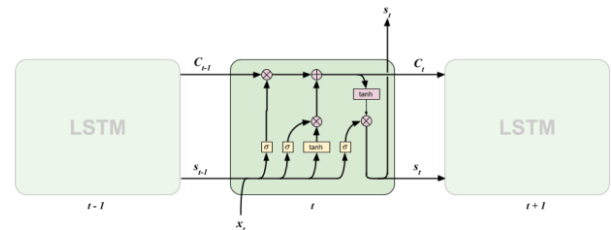
- *Case Folding*, merupakan proses mengubah semua huruf kapital menjadi huruf kecil (*lower case*). Hal ini perlu dilakukan untuk menjadikan teks seragam.
- *Filtering*, merupakan proses menghilangkan beberapa karakter spesial yang tidak dibutuhkan, seperti tanda baca, angka, dan beberapa simbol lainnya yang tidak diperlukan dalam proses *training*.
- *Tokenization*, merupakan proses memotong kalimat menjadi kata per kata, hal ini dilakukan dengan cara memotong suatu kalimat dengan acuan tiap spasi.
- *Stemming*, merupakan proses menghapus imbuhan pada suatu kata sehingga menjadi kata dasar.
- *Slang Words Removal*, merupakan proses menghapus setiap kosakata gaul (*slang*).

- *Stopwords Removal*, merupakan proses menghapus kata-kata yang tidak memiliki makna atau informasi penting dalam proses *training*.

Setelah melakukan proses *Cleaning of Dataset*, proses selanjutnya adalah *Word Representation*. Proses tersebut bertujuan untuk mengonversi setiap kata menjadi vektor bilangan real dengan mengaplikasikan *Trainable Embedding Layer* berbasis Keras. *Trainable Embedding Layer* akan menginisiasi secara acak nilai bobot dan diperbaharui selama proses *training* model dengan menggunakan algoritma *back-propagation*. Data yang telah dikonversi memiliki dimensi, yaitu 1143×20 .

Setelah melakukan proses *Word Representation*, proses selanjutnya adalah *Split-dataset* atau membagi *dataset*. *Dataset* dibagi menjadi 2, yaitu 80% data *train* dan 20% data *test*. Untuk melakukan klasifikasi teks pada suatu SMS, algoritma yang digunakan adalah LSTM.

Jaringan *Long Short Term Memory* – biasanya hanya disebut “LSTM” – adalah salah satu variasi *Recurrent Neural Networks* dengan kemampuan memori yang diperluas secara eksplisit cocok untuk menangani dependensi jangka panjang[11]. Jaringan LSTM diusulkan oleh peneliti Jerman Sepp Hochreiter dan Juergen Schmidhuber pada tahun 1997 sebagai solusi untuk masalah gradien yang hilang[12]. Arsitektur keseluruhan LSTM ditunjukkan pada Gambar 3.



Gambar 2. Gerbang LSTM

LSTM berisi informasi dari konteks dalam sel yang terjaga keamanannya. Sel mengontrol data yang akan ditulis, disimpan, dibaca, dan dihapus menggunakan *forget gate*, *input* dan *output* yang diimplementasikan dengan perkalian elemen oleh sigmoid. *Forget gate* mempelajari bobot yang mengontrol laju peluruhan nilai disimpan dalam sel memori. Misalnya ketika *input gate* dan *output gate* mati, dan *forget gate* tidak menyebabkan pembusukan, sel memori mempertahankan nilainya dari waktu ke waktu menyebabkan gradien kesalahan tetap konstan selama *backpropagation*. Hal ini memungkinkan model untuk mengingat informasi untuk waktu yang lebih lama. Secara matematis setiap langkah dapat dijelaskan sebagai berikut:

Pada langkah pertama, lapisan *forget gate* memutuskan fitur yang akan dihapus dari status sel dengan melihat h_{t-1} dan input baru x_t .

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \quad (1)$$

Pada langkah kedua, memutuskan informasi yang disimpan dalam status sel dilakukan dalam dua langkah. Lapisan *output gate* itu yang merupakan lapisan sigmoid menetapkan nilai-nilai yang akan diperbarui. Kemudian lapisan tanh menghasilkan vektor nilai kandidat baru \tilde{c}_t .

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \quad (2)$$

$$c_t = \tanh(W_c[h_{t-1}, x_t] + b_c) \quad (3)$$

Status sel lama C_{t-1} diperbarui ke sel baru C_t menjumlahkan *output* dari fungsi lapisan *forget gate* f_t dan $i_t \times \tilde{C}_t$.

$$c_t = f_t \times c_{t-1} + i_t \times \tilde{C}_t$$

Keluaran ditentukan dalam dua langkah – Pertama, lapisan sigmoid memutuskan bagian-bagian sel yang akan dikeluarkan. Produk dari status sel baru C_t melalui \tanh dan *output* dari gerbang sigmoid menghasilkan h_t bagian yang ditentukan secara selektif.

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \quad (4)$$

$$h_t = o_t \times \tanh(c_t) \quad (5)$$

Penyetelan dan optimasi *hyperparameters* adalah tugas yang sulit dan eksperimental. Pelatihan model LSTM mahal dalam hal memori dan daya komputasi[13].

B. Metrik Evaluasi

Evaluation metrics adalah parameter yang digunakan untuk mengukur kualitas model atau algoritma pembelajaran mesin[14]. *Evaluation metrics* yang digunakan pada penelitian ini yaitu, *Accuracy*, *Precision*, *Recall*, dan juga *F1-Score*. Masing-masing *evaluation metrics* tersebut dirumuskan sebagai berikut:

Accuracy adalah nilai perbandingan prediksi *True Positive* (TP) dan *True Negatif* (TN) dengan jumlah keseluruhan data.

$$Accuracy = \frac{TP+TN}{(TP+FN)+(FP+TN)} \quad (6)$$

Precision adalah nilai perbandingan prediksi *True Positive* (TP) dengan banyaknya data yang diprediksi positif.

$$Precision = \frac{TP}{TP+FP} \quad (7)$$

Recall adalah nilai perbandingan prediksi *True Positive* (TP) dengan banyaknya data yang benar positif.

$$Recall = \frac{TP}{TP+FN} \quad (8)$$

Terdapat perbedaan diantara *precision* dan *recall*, dimana *precision* mempunyai variabel *False Positive* (FP) sedangkan *recall* mempunyai variabel *False Negative* (FN).

Selanjutnya, *F1-Score* adalah nilai perbandingan rata-rata *precision* dan *recall* yang dibobotkan.

$$F1\ Score = \frac{2TP}{2TP+FN+FP} \quad (9)$$

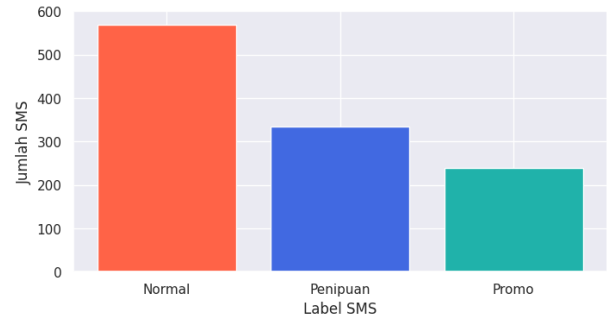
III. HASIL DAN PEMBAHASAN

A. Dataset

Dataset yang dipakai pada penelitian ini adalah *open-source dataset* yang diperoleh peneliti dari <https://github.com> dengan judul SMS Spam oleh Afif Akbar Iskandar pada tahun 2021[15]. Data tersebut merupakan kumpulan SMS yang telah diberi label dan dibersihkan dari informasi pribadi milik

pengguna. SMS terdiri dari tiga label berbeda, yaitu normal, penipuan dan promo.

Berikut adalah salah satu contoh dari SMS normal, “Pelanggan Yth. Update informasi tarif terbaru anda di tsel.me/as atau hubungi 188 (Rp 300/call).”. Untuk SMS penipuan, berikut adalah salah satu contoh, “PT.MTRONIK Berkah isi ulang pulsa m.tronik no. anda m-dptkan hadiah ke-2 dengan pin pemenang(25e477r) u/info klik; www.undianmtronik75.blogspot.com”. Untuk SMS promo, berikut adalah salah satu contoh, “Januari pokoknya mau sama sapapun kemanapun harus liburan fix aamiin”.

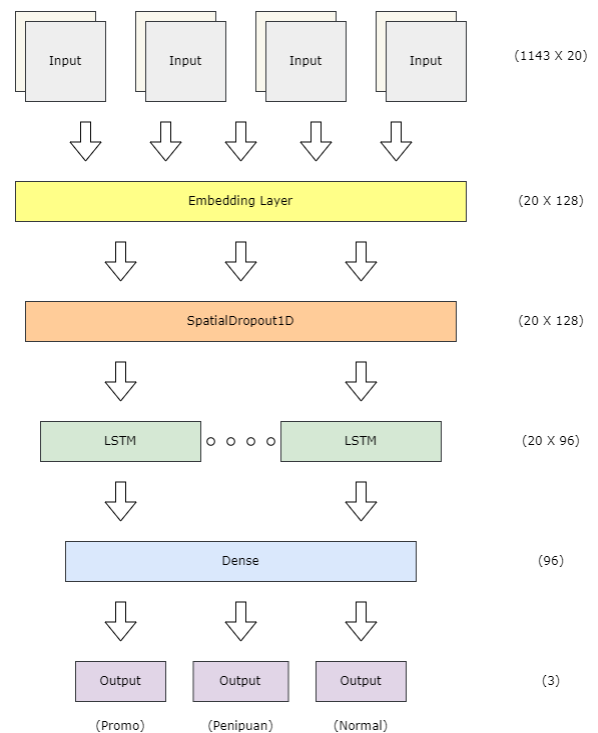


Gambar 3. Distribusi Jumlah SMS

Berdasarkan Gambar 3, SMS dengan label normal mempunyai data sebanyak 569, SMS dengan label penipuan mempunyai data sebanyak 335 dan SMS dengan label promo sebanyak 239. Jumlah dari keseluruhan data SMS berlabel adalah 1143.

B. Hasil Pengujian Model

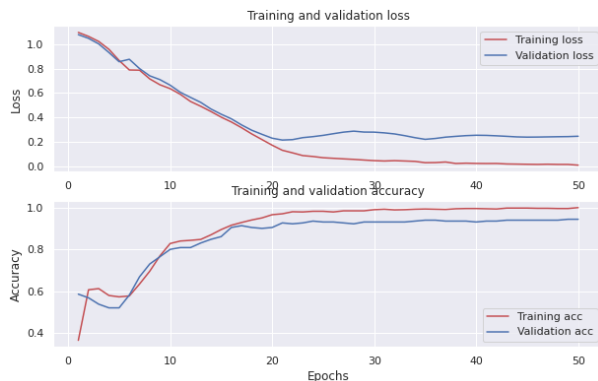
Pada penelitian ini, peneliti mengklasifikasikan SMS menjadi tiga kelas, yaitu normal, penipuan dan promo. Algoritma yang digunakan untuk penelitian tersebut adalah LSTM.



Gambar 4. Arsitektur Model LSTM

Gambar 4 menunjukkan arsitektur model yang digunakan. Pelatihan model menggunakan panjang *sequence* 20. Model tersebut dilatih dengan menggunakan nilai *hyperparameter* sebagai berikut; *max_epoch* 50, *batch_size* 512, *early stopping monitor* = *val_loss*, *patience* = 1, *dropout* = 0.2, dan LSTM unit 96.

Pada sesi pengujian dari arsitektur model LSTM dengan *hyperparameter* yang telah ditetapkan, didapatkan hasil berupa grafik sebagai berikut:



Gambar 5. Grafik Training dan Validation

Dari hasil pengujian ini didapatkan model ketika nilai *validation loss* sebesar 0.2444. Nilai *accuracy* yang didapat dari hasil pengujian terhadap *data test* adalah 94.32%.

C. Perbandingan Hasil dengan Metode Lain

Metode LSTM menghasilkan hasil yang lebih baik dari kedua metode yang digunakan sebagai pembandingan, *Naive Bayes* dan KNN. *Evaluation metrics* masing-masing metode tersebut mempunyai nilai *Precision*, *Recall*, dan *F1-Score* yang ditampilkan pada Tabel 1 berikut.

Tabel 1. Evaluation Metrics LSTM

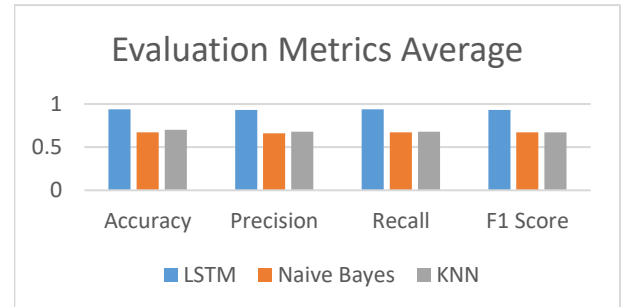
Metode	Kelas	<i>Precision</i>	<i>Recall</i>	<i>F1-Score</i>
LSTM	Promo	97%	98%	97%
	Penipuan	97%	90%	94%
	Normal	85%	94%	89%
KNN	Promo	79%	86%	82%
	Penipuan	73%	56%	63%
	Normal	52%	62%	57%
Naive Bayes	Promo	68%	66%	67%
	Penipuan	67%	68%	68%
	Normal	64%	66%	65%

Dari Tabel 1 dapat diketahui bahwa untuk metode LSTM pada kelas promo mempunyai nilai *precision* 97%, *recall* 98% dan *F1-score* 97%. Pada kelas penipuan mempunyai nilai *precision* 97%, *recall* 90% dan *F1-score* 94%. Pada kelas normal mempunyai nilai *precision* 85%, *recall* 94% dan *F1-score* 89%.

Untuk metode KNN pada kelas promo mempunyai nilai *precision* 79%, *recall* 86% dan *F1-score* 82%. Pada kelas penipuan mempunyai nilai *precision* 73%, *recall* 56% dan *F1-score* 63%. Pada kelas normal mempunyai nilai *precision* 52%, *recall* 62% dan *F1-score* 57%.

Untuk metode *Naive Bayes* pada kelas promo mempunyai nilai *precision* 68%, *recall* 66% dan *F1-score* 67%. Pada kelas penipuan mempunyai nilai *precision* 67%, *recall* 68% dan *F1-score* 68%. Pada kelas normal mempunyai nilai *precision* 64%, *recall* 66% dan *F1-score* 65%.

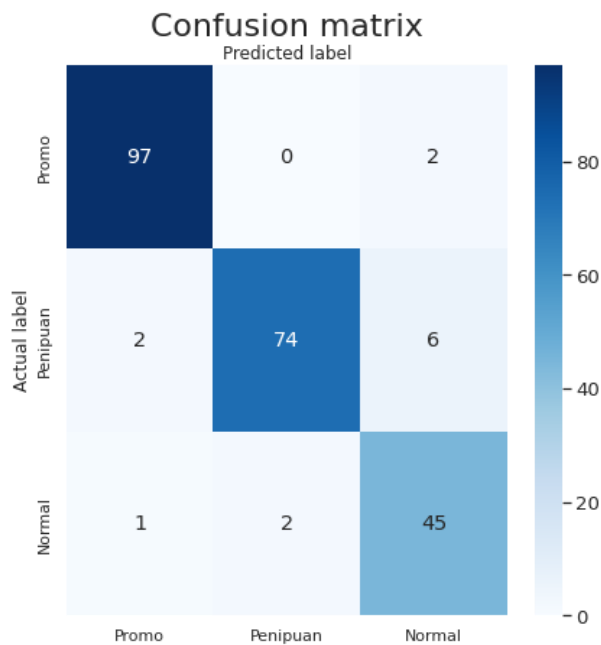
Hasil skor perbandingan rata-rata *evaluation metrics* setiap metode ditampilkan pada Gambar 6 berikut.



Gambar 6. Perbandingan rata-rata *evaluation metrics* setiap model

Dari Gambar 6 dapat diketahui bahwa kinerja metode LSTM pada kasus klasifikasi teks jauh lebih baik daripada metode *Naive Bayes* dan KNN dengan masing-masing *accuracy* yang dihasilkan adalah LSTM 94%, *Naive Bayes* 67% dan KNN 70%. Metode LSTM menghasilkan hasil yang lebih baik dari kedua metode pembandingnya. Model dengan metode LSTM menghasilkan nilai *accuracy* 27% lebih baik dibandingkan dengan metode *Naive Bayes* dan 24% lebih baik dari metode KNN. Hal ini dikarenakan beberapa faktor yang memengaruhi perbedaan dari nilai *accuracy* masing-masing metode. Metode *Naive Bayes* tidak memperhitungkan faktor urutan kata, sedangkan pada metode LSTM hal ini memberikan pengaruh yang signifikan pada hasil akhir prediksi. Metode KNN mempunyai keterbatasan dalam hal memori, sedangkan pada metode LSTM mempunyai memori yang cukup untuk memberikan hasil akhir prediksi yang lebih baik.

Penelitian ini juga menganalisis hasil *confusion matrix* atau matriks konfusi dari metode LSTM. Hasil *confusion matrix* ditampilkan pada Gambar 7 berikut.



Gambar 7. Confusion matrix LSTM

Dari Gambar 7 dapat diketahui bahwa, metode LSTM dapat memprediksi benar kelas promo sebanyak 97 dari 99 total data kelas promo. Untuk kelas penipuan metode LSTM dapat memprediksi benar sebanyak 74 dari 82 total data kelas penipuan. Untuk kelas normal metode LSTM dapat memprediksi benar sebanyak 45 dari 48 total data kelas normal.

IV. KESIMPULAN

Berdasarkan hasil penelitian yang dilakukan klasifikasi teks pada SMS spam berbahasa Indonesia, metode LSTM menghasilkan nilai *accuracy* sebesar 94%. Metode LSTM menghasilkan hasil yang lebih baik dari kedua metode pembandingnya. Model dengan metode LSTM menghasilkan nilai *accuracy* 27% lebih baik dibandingkan dengan metode *Naive Bayes* dan 24% lebih baik dari metode KNN. Dari hasil yang diperoleh terdapat beberapa hal yang perlu dianjurkan untuk pengembangan lanjutan dari penelitian ini. Peneliti menyarankan bahwa langkah *Data Preprocessing* memainkan peran yang sangat penting dalam proses klasifikasi teks, di mana tata cara yang dilakukan harus menyesuaikan dengan karakteristik dari dataset tersebut. Selain itu, peneliti berharap pengembangan dari penelitian ini dapat menerapkan pengaturan *hyperparameter* yang sesuai sehingga dapat diperoleh hasil akurasi yang lebih baik.

REFERENSI

- [1] D. Bayu, "Jumlah Pengguna Internet di Indonesia," *dataindonesia.id*, 2022. <https://dataindonesia.id/digital/detail/pengguna-internet-di-indonesia-capai-205-juta-pada-2022>
- [2] T. Setiadi, "Prediksi Perkembangan Jaringan Internet 2022," *sistem-komputer-s1.stekom.ac.id*, 2022. <http://sistem-komputer-s1.stekom.ac.id/informasi/baca/Prediksi-Perkembangan-Jaringan-Internet-2022/fee055b039d2e9ce3396eeeb5d34a5659678326>
- [3] F. Reviatika, Y. Azhar, and G. I. Marthasari, "Analisis Klasifikasi SMS Spam Menggunakan Logistic Regression," *APIC*, 2021. [Online]. Available: <https://www.cnbcindonesia.com>
- [4] F. Rahutomo, I. Y. R. Pratiwi, and D. M. Ramadhani, "Eksperimen Naïve Bayes Pada Deteksi Berita Hoax Berbahasa Indonesia," *J. Penelit. Komun. Dan Opini Publik*, vol. 23, no. 1, 2019, doi: 10.33299/jpkop.23.1.1805.
- [5] B. Indarto, "Klasifikasi Sms Spam Dengan Metode Naive Bayes Classifier Untuk Menyaring Pesan Melalui Selular," *J. Telemat. MKOM*, vol. 8, no. 2, pp. 167–172, 2016.
- [6] Ulviyana, "KLASIFIKASI SPAM EMAIL MENGGUNAKAN ALGORITMA K-NEAREST NEIGHBOR DAN CHI SQUARE," 2020, [Online]. Available: https://repository.unsri.ac.id/35599/0Ahttps://repository.unsri.ac.id/35599/3/RAMA_56201_09011281520102_0003047905_01_front_ref.pdf
- [7] M. Z. Rahman, Y. A. Sari, and N. Yudistira, "Analisis Sentimen Tweet COVID-19 menggunakan Word Embedding dan Metode Long Short-Term Memory (LSTM)," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 5, no. 11, pp. 5120–5127, 2021, [Online]. Available: <http://j-ptiik.ub.ac.id>
- [8] Winda Kurnia Sari, "Klasifikasi Teks Menggunakan Long Short-Term Memory Dengan Fitur Word Embedding Glove Dan Word2Vec," 2020.
- [9] A. N. Hidayat, "DEEP LEARNING UNTUK KLASIFIKASI TEKS HATE SPEECH MENGGUNAKAN LONG SHORT TERM MEMORY," 2020.
- [10] F. Yahyadi, Adi; Latifah, "ANALISIS SENTIMEN TWITTER TERHADAP KEBIJAKAN PPKM DI TENGAH PANDEMI COVID-19," *J. Inf. Syst. Applied, Manag. Account. Res.*, vol. 6, no. 2, pp. 464–471, 2022, doi: 10.52362/jisamar.v6i2.791.
- [11] D. Li and J. Qian, "Text sentiment analysis based on long short-term memory," *2016 1st IEEE Int. Conf. Comput. Commun. Internet, ICCCI 2016*, pp. 471–475, 2016, doi: 10.1109/CCCI.2016.7778967.
- [12] X. H. Le, H. V. Ho, G. Lee, and S. Jung, "Application of Long Short-Term Memory (LSTM) neural network for flood forecasting," *Water (Switzerland)*, vol. 11, no. 7, 2019, doi: 10.3390/w11071387.
- [13] S. Patil, V. M. Mudaliar, P. Kamat, and S. Gite, "LSTM based Ensemble Network to enhance the learning of long-term dependencies in chatbot," *Int. J. Simul. Multidiscip. Des. Optim.*, vol. 11, 2020, doi: 10.1051/smdo/2020019.
- [14] "Evaluation Metrics," *deeppai.org*. <https://deeppai.org/machine-learning-glossary-and-terms/evaluation-metrics>
- [15] A. A. Iskandar, "SMS Spam," *Github.com*, 2021. https://github.com/afifai/klasifikasi_sms_spam/tree/main/data