JEDUMATIC

e-ISSN 3031-8408

Journal of Education and Informatics Research /Vol.7, No.1 2026

Perbandingan Akurasi Algoritma RNN, LSTM, dan Bidirectional LSTM dalam Klasifikasi SMS

Rio Meisya Resnanda[1], Alifia Zumrotul Nisa'[2], Asdhana Ayu Firdauzy Prameistha[3], Laili Cahyani[4*]

[1,2,3,4] *Pendidikan Informatika, Universitas Trunodjoyo Madura, Kabupaten Bangkalan, Indonesia*

email: [1]rega_meisya@yahoo.com, [2]alifiazn817@gmail.com, [3]asdhana.comel@gmail.com, [4]laili.cahyani@trunojoyo.ac.id

ABSTRAK

Kebutuhan informasi ialah kebutuhan fundamental pada setiap orang untuk melakukan komunikasi. Terdapat beberapa cara untuk melakukan penyaluran informasi, salah satunya ialah melewati pengiriman Short Message Service (SMS). Meningkatnya penggunaan layanan SMS banyak dimanfaatkan oleh pihak-pihak tertentu yang tidak bertanggung jawab. Pesan yang tidak bermanfaat dan merugikan disebut SMS spam atau teks pesan spam. Teks pesan spam yang dikirimkan melalui sms banyak yang mengandung konten ilegal, sehingga menimbulkan rasa tidak nyaman bagi penerima pesan tersebut. Pesan spam yang banyak beredar di Indonesia adalah pesan spam yang didalamnya terdapat pengumuman promosi, pengumuman hadiah, dan diskon toko atau pesan yang tidak memiliki makna yang lainnya. Pada penelitian ini digunakan metode Recurrent Neural Network (RNN), Long Short Term Memory (LSTM), dan Bidirectional Long Short Term Memory (Bi-LSTM) untuk mengklasifikasikan sms spam dan non spam. Data yang digunakan merupakan data SMS berbahasa indonesia. Tujuan dari penelitian ini yaitu membandingkan akurasi pada klasifikasi SMS spam. Hasil dari penelitian ini menunjukkan bahwa model Bi-LSTM mendapatkan nilai akurasi pengujian 82%. Sedangkan model lainnya, LSTM 78% dan RNN 79%. Hal itu menunjukkan bahwa model Bi-LSTM lebih baik daripada model yang lainnya.

Kata Kunci: Deep Learning, Recurrent Neural Network, Long Short Term Memory, Bidirectional LSTM, Deteksi SMS Spam

ABSTRACT

The need for information is a fundamental need for everyone to communicate. There are several ways to disseminate information, one of which is through the delivery of Short Message Service (SMS). The increasing use of SMS services is widely exploited by irresponsible parties. They send messages that are not useful and harmful called SMS spam or spam text messages. Spam text messages sent via SMS contain a lot of illegal content, causing discomfort for the recipient of the message. Spam messages that are widely circulated in Indonesia are spam messages that contain promotional announcements, prize announcements, and store discounts or messages that have no other meaning. In this study, the Recurrent Neural Network (RNN), Long Short Term Memory (LSTM), and Bidirectional Long Short Term Memory (Bi-LSTM) methods were used to classify spam and non-spam SMS. The data used is Indonesian language SMS data. The purpose of this study is to compare the accuracy of spam SMS classification. The results of this study show that the designed Bi-LSTM model achieves better accuracy than other models.

Keywords: Deep Learning, Recurrent Neural Network, Long Short Term Memory, Bidirectional LSTM, Spam SMS detection

PENDAHULUAN

Kebutuhan informasi sangat penting digunakan dalam kehidupan sehari hari untuk melakukan sebuah komukasi atau bertukar kabar jarak jauh. Terdapat banyak cara untuk melakukan sebuah komunikasi slah satunya dengan menggunakan *Short Message Servis* (SMS). SMS adalah data yang menggunakan tipe *asynchronous message* dimana transmisi data SMS dilaksanakan dengan Teknik protokol *store and forward* yang berarti pengirim pesan SMS mengirimkan pesan serta nomor telepon tujuan dan mengirimkannya ke *server* SMS atau SMS-*Center*. SMS-*Center* (SMSC) berkewajiban dalam mengirimkan pesan tersebut ke nomor telepon yang akan dituju. *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 nonspam (Reviantika dkk., 2021). SMS dengan jenis spam cenderung mengganggu kenyamanan pengguna dan biasa berisikan penipuan dan promosi. SMS jenis nonspam 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 (Pratama, 2022).

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 (Rahutomo dkk., 2019). Beberapa penelitian diantaranya dilakukan oleh Alfin Hikmaturokhman pada tahun 2022 melakukan klasifikasi untuk mengidentifikasi SMS spam menggunakan metode *Bidirectional Long Short Term Memory* (Bi-LSTM) (Hikmaturokhman dkk., 2022). Pramunendar pada tahun 2022 juga melakukan analisis sentimen opini publik terkait vaksin *covid-19* dengan menggunakan metode *Recurrent Neural Networks* (RNN) (Pramunendar dkk., 2022).

Selain *Recurrent Neural Networks* (RNN) dan *Bidirectional Long Short Term Memory* (Bi-LSTM), terdapat metode lain yang dapat digunakan untuk klasifikasi teks. Salah satunya adalah *Long-Short Term Memory* atau biasa disingkat dengan LSTM. Winda Kurnia Saripada tahun 2020 melakukan klasifikasi teks artikel berita menggunakan LSTM. Di tahun berikutnya, Agung Nur Hidayat melakukan klasifikasi teks ujaran kebencian terhadap kicauan Twitter menggunakan metode LSTM. Masih di tahun yang sama, Yahyadi dan Latifah juga melakukan klasifikasi tekspada kicauan Twitter terkait kebijakan PPKM menggunakan metode LSTM (Pratama, 2022).

Berdasarkan uraian diatas, disimpulakan bahwa ada banyak metode untuk melakukan klasifikasi terhadap kasus SMS spam. *State of the art* dalam penelitian ini adalah analisa perbandingan tiga model sekaligus yaitu *Recurrent Neural Networks* (RNN), *Long Short Term Memory* (LSTM), dan *Bidirectional Long Short Term Memory* (Bi-LSTM) untuk permasalahan klasifikasi SMS spam. Dataset yang digunakan berpacu pada dataset Github tentang SMS spam yang manjadi masukan untuk ketiga model *deep learning* tersebut. Hasil dari model tersebut dapat diketahui akurasi, *loss* dan *performance* modelnya. Kemudian dari ketiga model tersebut akan dilakukan perbandingan.

METODE

Penelitian ini dilakukan degan membangun model prediksi untuk 2 variabel yaitu teks dan label dengan tahapan sebagai berikut.

1. Tahapan Akuisisi Data SMS

Pada tahapan awal ini data yang didapatkan bersumber dari *website* Github, data yang berhasil diunduh berformat .csv dan berisi sebanyak 1143 baris dan 2 kolom.

2. Tahapan *Preprocessing*

Data teks SMS yang digunakan masih mengandung beberapa noise. Karena itu data tersebut perlu untuk diproses terlebih dahulu. Proses preprocessing dari dataset meliputi:

• Case Folding, merupakan proses mengubah semua huruf kapital menjadi huruf kecil (lower case). Hal ini perlu dilakukan untuk menjadikan teks seragam (Pratama, 2022).

• Filtering, merupakan proses menghilangkan beberapa karakter spesial yang tidak dibutuhkan, seperti tanda baca, angka, dan beberapa simbol lainnya yang tidak diperlukan dalam proses training (Pratama, 2022).

• Tokenization, merupakan proses memotong kalimat menjadi kata per kata, hal ini dilakukan dengan cara memotong suatu kalimat dengan acuan tiap spasi (Pratama, 2022).

3. Tahapan *Word Embbeding*

Setelah melakukan proses *preprocessing* dari dataset, proses selanjutnya adalah *Word Embbeding*. Proses tersebut bertujuan untuk merepresentasikan kata kedalam vektor angka dengan mengaplikasikan Word2Vec. Untuk masukkannya sendiri (*input*) disebut *corpus* dan *output* yang dihasilkan adalah satu set vektor. Word2vec memiliki metode, yaitu *Continous Bag of Word* (CBOW) dan *Skip-gram.* CBOW bertujuan untuk memprediksi kata yang diberikan oleh kata-kata disekitarnya. Metode ini menggunakan *current word* sebagai target dan konteks sebagai input. Berbanding terbalik dengan CBOW, Skip-gram menggunakan *current word* sebagai input untuk memprediksi konteks sebagai target (Isnain dkk., 2022).

4. Tahapan *Data Preparation* Sebelum Proses *Training* dan *Testing*

Setelah melakukan proses *Word Embbeding*, proses selanjutnya adalah *Split-dataset* atau membagi dataset. Dataset dibagi menjadi 2, yaitu 80% data *train* dan 20% data *test*.

5. Tahapan *Training* Setiap Model

Sebelum dilakukan data *training*, maka dibuat 3 permodelan yaitu Metode *Recurrent Neural Networks* (RNN), *Long Short Term Memory* (LSTM), dan *Bidirectional Long Short Term Memory* (Bi-LSTM). Dari model yang dibuat kemudian masing-masing model dilakukan training sebanyak 10 *epoch* dengan *bacth size* berjumlah 32 menggunakan *optimizer* adam. Kinerja model dinilai berdasarkan fungsi *loss* dengan *categorical crossentropy*. Proses *training* dilakukan dengan melatih model menggunakan data *training* sehingga didapatkan model yang diharapkan. Setelah iterasi proses *training* selesai, maka dilakukan validasi untuk menilai kinerja model dari hasil *training*.

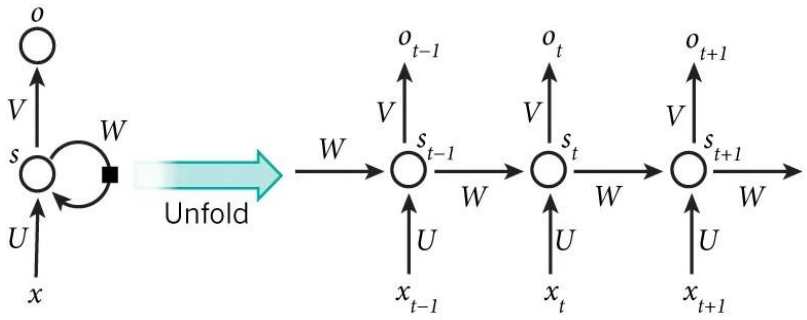
***Recurrent Neural Networks* (RNN)**

Recurrent neural network (RNN) ialah arsitektur jaringan saraf tiruan yang prosesnya dipanggil secara berulang untuk data masukan yang sifatnya sekuensial. RNN merupakan bagian dari *deep learning* karena pemprosesan data dilakukan melalui banyak *layer*. RNN telah mengalami kemajuan yang pesat dan telah merevolusi bidang-bidang seperti pemrosesan bahasa alami (NLP), pengenalan suara, sintesa musik, pemrosesan data finansial seri waktu, analisa deret DNA, analisa video, dan sebagainya. Dalam pengembangan lebih lanjut, RNN banyak digunakan untuk menyelesaikan tugas yang terkait dengan data *time series*, misalnya data ramalan cuaca. Sebagai contoh, cuaca hari ini dapat bergantung pada cuaca hari sebelumnya, jika hari sebelumnya mendung, maka kemungkinan hari ini akan hujan (Prijono, 2018).

Dalam proses *training*, RNN tidak membuang data atau informasi proses sebelumnya. Hal inilah yang membedakan RNN dibandingkan ANN biasa. RNN menyimpan data memori agar mendapatkan pola data yang lebih baik, selanjutnya hasilnya digunakan untuk mendapatkan prediksi yang akurat. Metoda yang digunakna oleh RNN agar dapat menyimpan informasi atau data masa lalu yaitu dengan

cara melakukan *looping*, sehingga secara otomatis informasi masa lalu tetap tersimpan (Karyadi & Santoso, 2022).

Konsep dasar RNN adalah menggunakan algoritma yang dilakukan secara berulang. Hal ini berarti bahwa operasi yang sama dilakukan untuk setiap elemen dari suatu urutan, dengan *output*-nya tergantung pada input saat ini dan operasi sebelumnya. Intinya adalah RNN berfokus pada sifat data, dimana waktu saat sebelumnya atau saat ini (t) mempengaruhi pada waktu berikutnya (t + 1) (Karyadi & Santoso, 2022).



Gambar 1. Proses RNN Saat Perhitungan Waktu di Depannya.

Pada gambar 1. bagian kiri merupakan sebuah diagram sirkuit, dimana kotak hitam menunjukkan *time delay* (penundaan waktu) dari satu *time step*. Terlihat RNN pada posisi yang *unrolled* kearah *full network*. Sedangkan bagian kanan menunjukkan RNN yang telah dibuka (*unrolled* / *unfolded*) menjadi *full network* sehingga urutan sekuensnya menjadi lengkap (Karyadi & Santoso, 2022).

Berikut ini ialah keterangan simbol formula:

- xt adalah *input* pada *time step*. Misalnya, x1 dapat menjadi *one-hot vector* yang sesuai dengan kata kedua dari sebuah kalimat yang sedang diproses.
- st adalah *hidden state* pada setiap *time step* t. *Hidden state* adalah sebagai "*memory*" suatu *network* yang menyimpan hasil perhitungan dan *record* sebelumnya. st dihitung berdasarkan *hidden state* sebelumnya dan berdasarkan *input* pada *current state*: st = f(Uxt + Wst-1). st-1 digunakan untuk menghitung *hidden state* yang pertama, biasanya pada inisialisasi selalu diawali dengan 0 (nol).

ot adalah *output* pada *step* t. Sebagai contoh, apabila kita ingin memprediksi "kata berikutnya" di suatu kalimat, maka ot merupakan vektor probabilitas di seluruh kosakata dalam suatu data yang kita punya.

***Long Short Term Memory* (LSTM)**

Long Short-Term Memory (LSTM) adalah jenis algoritma pembelajaran mesin yang digunakan untuk memproses data seri dan memprediksi hasil berdasarkan data sejarah. LSTM memiliki kemampuan untuk mengingat informasi jangka panjang dan mengabaikan informasi yang tidak relevan, sehingga dapat membuat prediksi yang lebih akurat. LSTM digunakan dalam berbagai aplikasi, seperti pengenalan suara, pemrosesan bahasa alami, dan pemrosesan datateks. Rumus algoritma LSTM (Aprian dkk., 2020) adalah sebagai berikut:

Forget Gate

$ft = \sigma(WfxXt + Wfhht - 1 + WfcCt - 1 + bf)$ (1)

Input Gate

$it = \sigma(WixXt + Wihht - 1 + WicCt - 1 + bi) * tanh(WcxXt + Wchht - 1 + bi)$ (2)

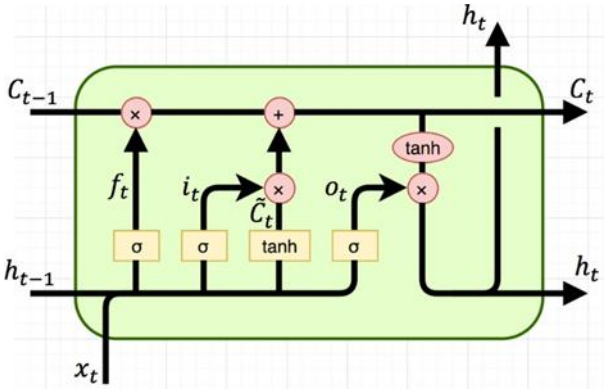
$Ct = ft * Ct - 1 + it$ (3)

Output Gate

$Ot = \sigma(WoxXt + Wohht - 1 + WocCt - 1 + bo)$ (4)

$ht = Ot * tanh(Ct)$ (5)

ft adalah *Forget Gate*, it adalah *Input Gate*, Ct adalah *Update Cell State*, Ot adalah *Output Gate*, ht adalah Nilai *Output*, Xt adalah Nilai *Input*, $ht-1$ adalah Nilai *Output Cell* sebelumnya, $Ct-1$ adalah *Cell State* sebelumnya, b adalah *Bias*, W adalah *Weight*, σ adalah Sigmoid. Arsitektur LSTM (Rizki dkk., 2020) dapat dilihat pada Gambar 2.

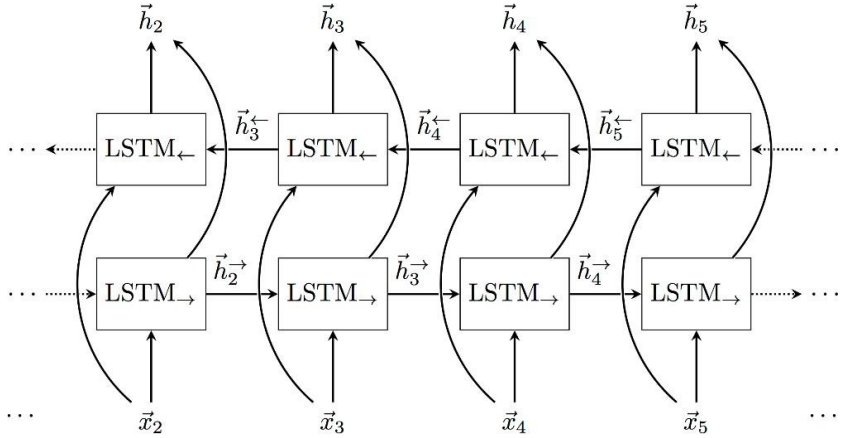


Gambar 2. Arsitektur *Long Short Term Memory*

LSTM memiliki unit-unit *cell* yang membuat informasi baru dan mempertahankan informasi yang sudah ada. *Cell* ini memiliki tiga kontrol lup: *input*, *forget*, dan *output*. Kontrol *input* memperbarui informasi yang disimpan dalam *cell*, kontrol *forget* memungkinkan *cell* untuk memutuskan informasi yang tidak perlu untuk diteruskan, dan kontrol *output* memperbarui informasi yang disimpan dan diteruskan ke *cell* berikutnya, mempertahankan informasi yang relevan dan membuang informasi yang tidak perlu, LSTM dapat mengatasi masalah *vanishing gradient* yang sering terjadi pada RNN. Masalah ini terjadi ketika informasi yang diteruskan dari waktu ke waktu menjadi sangat kecil dan tidak berguna (Azhari dkk., 2023).

***Bidirectional Long Short Term Memory* (Bi-LSTM)**

Bi-LSTM merupakan pengembangan dari model LSTM, dimana struktur Bi-LSTM memiliki 2 lapisan dengan proses yang saling berlawanan arah. Bi-LSTM memiliki keunggulan dalam mempelajari konteks teks atau kalimat. Bi-LSTM sangat cocok untuk menangani masalah semantik dengan kemampuan mengenali pola dalam kalimat dari setiap kata pada teks yang diproses secara sekuensial. Pada Bi-LSTM, lapisan di bawahnya bergerak maju (forward) untuk mempelajari dari kata awal menuju kata terakhir sedangkan lapisan di atasnya bergerak mundur (backward) untuk mempelajari dari kata terakhir menuju kata awal (Aritonang dkk., 2022). Sesuai arstitektur bidirectional LSTM pada gambar 4, bidirectional LSTM memilki 2 lapisan LSTM yang terpisah satu untuk maju dan satu untuk mundur (Alghifari dkk., 2022).



Gambar 3. Arsitektur Bi-LSTM

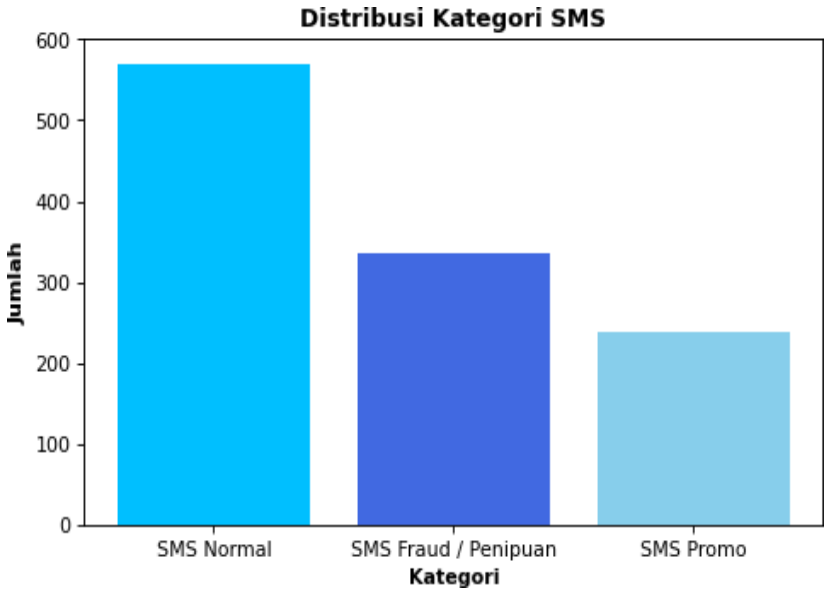
6. Tahapan Evaluasi

Evaluasi dilakukan dengan metode confussion matrix. Parameter yang diukur adalah akurasi, presisi, recall, dan f-measure. Akurasi adalah nilai kemiripan antara nilai prediksi dan nilai sebenarnya. Presisi adalah tingkat kebenaran antara informasi yang diminta pengguna dan yang diberikan oleh sistem. Recall adalah tingkat keberhasilan dari sistem dalam menemukan kembali suatu informasi. F-measure merupakan bentuk parameter gabungan antara recall dan presisi untuk keberhasilan retrival (Isnain dkk., 2022).

HASIL DAN PEMBAHASAN

Bagian ini menjelaskan hasil penelitian dan pembahasan yang telah diperoleh oleh para peneliti. Semua temuan utama dalam studi ini dituliskan dan disertai dengan pembahasan atau diskusi berdasarkan sumber referensi yang relevan. Dijelaskan pula kesamaan dan perbedaan antara pekerjaan penelitian yang dilakukan oleh penulis dengan studi sebelumnya, baik dalam hal data, metode, maupun hasil. Selain itu, dijelaskan pula terkait pencapaian penelitian, apa saja keberhasilan yang dicapai sesuai dengan tujuan penelitian atau kegagalan yang dihadapi beserta penyebabnya. Penulis juga menjelaskan tentang pekerjaan apa yang masih dapat dilakukan untuk penelitian lebih lanjut. Penulisan hasil dan pembahasan dapat dilengkapi dengan tabel atau grafik yang digunakan sebagai referensi dalam penulisan sesuai dengan format yang telah disajikan sebelumnya.

Dari pembahasan diatas dataset yang digunakan diambil dari *website* Github yang berisikan sebanyak 1143 baris dan 2 kolom. Dataset yang digunakan yakni kumpulan data sms yang terdiri dari 3 label yaitu 0, 1, dan 2. Dari ketiga label tersebut kemudian dikelompokkan menjadi sms normal untuk data berlabel 0, sms penipuan untuk data berlabel 1, dan sms promo untuk data berlabel 2. Didapatkan 569 data berlabel sms normal, 335 data sms penipuan, dan 239 data sms promo. Gambar 4 menampilkan persebaran data sms.



Gambar 4. Grafik persebaran data SMS

Sebelum dilakukan pemodelan dan pengujian terhadap dataset, dilakukan tahapan preprocessing terlebih dahulu, diantaranya *case folding, filtering,* dan *tokenization.* Contoh hasil *preprocessing* dapat dilihat pada tabel 1.

Tabel 1. Contoh hasil *preprocessing* pada dataset

Index	*Content*	Hasil *preprocessing*
5	5 HARI LAGI ! EKSTRA Pulsa 50rb dg beli paket internet bulanan di MyTelkomsel utk pembelian pertama sjk 25Aug-25Sept. Cek detail promo di tsel.me/mytsel1	['hari', 'lagi', 'ekstra', 'pulsa', 'rb', 'dg', 'beli', 'paket', 'internet', 'bulanan', 'di', 'mytelkomsel', 'utk', 'pembelian', 'pertama', 'sjk', 'augsept', 'cek', 'detail', 'promo', 'di', 'tselmemytsel']
99	Jangan sampai kalah Bro, games mu bisa terbantu dengan menambah koinmu, beli aja pakai pulsa, cek caranya di http://tsel.me/jajanonline	['jangan', 'sampai', 'kalah', 'bro', 'games', 'mu', 'bisa', 'terbantu', 'dengan', 'menambah', 'koinmu', 'beli', 'aja', 'pakai', 'pulsa', 'cek', 'caranya', 'di']
790	Hbd nama1 , cepet sidang cepet wisuda yaaa aamiiiin	['hbd', 'nama', 'cepet', 'sidang', 'cepet', 'wisuda', 'yaaa', 'aamiiiin']
796	Info Gempa Mag:5.3 SR, 21-Jun-16 21:10:23 WIB, Lok:2.21 LS,99.33 BT (98	['info', 'gempa', 'mag', 'sr', 'jun', 'wib', 'lok', 'ls', 'bt', 'km', 'tenggara', 'kepmentawaisumbar', 'kedlmn', 'km', 'bmkg']
1096	km Tenggara KEP-MENTAWAI-SUMBAR), Kedlmn:34 Km ::BMKG Tp berapa lama lagi ya Pak?	['tp', 'berapa', 'lama', 'lagi', 'ya', 'pak']

Tahap selanjutnya adalah *word embbeding* menggunakan *word2vec* yang bertujuan untuk merepresentasikan kata kedalam vektor angka, kata-kata yang berhubungan dengan 'hadiah' dapat dilihat pada tabel 2.

Tabel 2. *Similar* kata

Similar kata dengan kata hadiah	Nilai
mdptkan	0.9978235960006714
pemenang	0.9975258708000183
undian	0.9973876476287842
terpilih	0.9973334670066833
mendapatkan	0.9970734119415283
simcard	0.9964334964752197
yaris	0.9958063364028931
dri	0.9954462051391602
toyota	0.9951916933059692
selamat	0.9950172901153564

Berdasarkan tabel 2 dapat dilihat bahwa kata hadiah banyak berkaitan dengan kata yang berlabel penipuan seperti pemenang, undian, mendapatkan, dan yaris yang menunjukan bahwa kata hadiah sangat berkaitan dengan label sms penipuan dalam dataset.

Pada penelitian ini seluruh *experiment* dilakukan menggunakan *google colab*. Setelah data melalui tahap *preprocessing*, dan *word embbeding* selanjutnya dataset dibagi menjadi dua, yaitu 80% sebagai data *training* dan 20% sebagai data *testing*. Selanjutnya melakukan pengujian terhadap algoritma yang diusulkan yaitu *Recurrent Neural Networks* (RNN), *Long Short Term Memory* (LSTM), dan *Bidirectional Long Short Term Memory* (Bi-LSTM). Dalam ketiga metode tersebut terdapat nilai

parameter yang telah ditentukan di awal dan dapat mempengaruhi nilai akurasi algoritma, nilai parameter yang digunakan untuk pengujian dapat dilihat pada tabel 3.

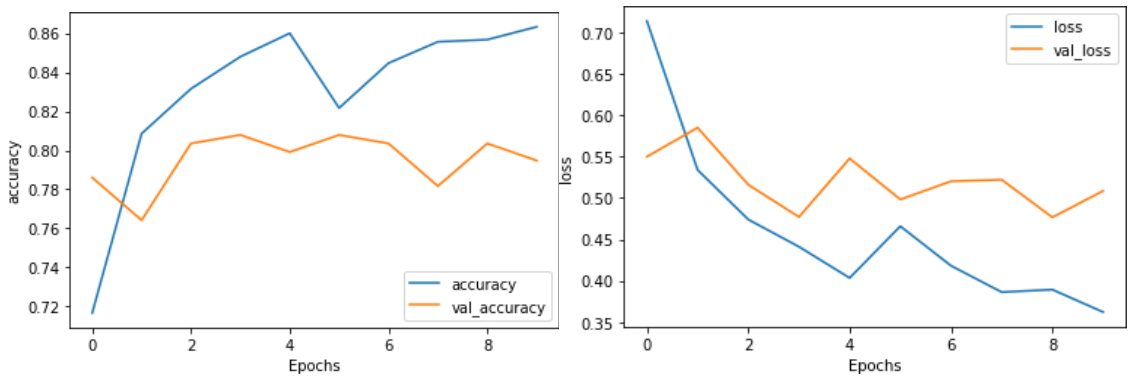
Tabel 3. Parameter yang digunakan

Nama Parameter	Nilai Parameter
Embbeding Size	100
Dropout	0.5
Activation	Softmax
Optimizer	Adam
Batch Size	32

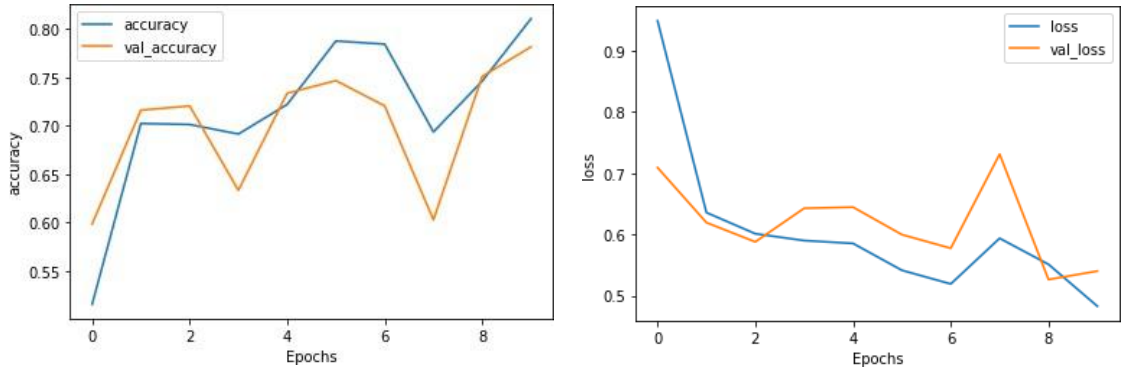
Berdasarkan hasil pengujian antara RNN, LSTM, dan Bi-LSTM pada tabel 4 dapat dilihat hasil metrik evaluasi menunjukan bahwa Bi-LSTM memiliki kinerja lebih handal. Bi-LSTM juga lebih baik jika dibandingkan algoritma lainnya pada tabel 4. Pada gambar 5-7 menunjukkan Bi-LSTM memiliki akurasi pelatihan stabil terus naik setiap epoch-nya dan traning loss yang lebih stabil dari RNN dan LSTM. Sedangkan pada RNN dan LSTM akurasi pelatihan dan training loss cenderung naik turun pada setiap epoch-nya. Akurasi terbaik yang dihasilkan metode Bi-LSTM, yaitu akurasi pengujian sebesar 82%. Berdasarkan pengujian yang telah dilakukan Bi-LSTM memiliki kinerja yang handal dari RNN dan LSTM karena memiliki kelebihan dapat memproses kata 2 arah. BiLSTM juga memilki kelemahan, yaitu membutuhkan data yang banyak serta waktu dan biaya komputasi yang lebih tinggi dari metode lain pada penelitian ini.

Tabel 4. Hasil pengujian RNN, LSTM, dan Bi-LSTM

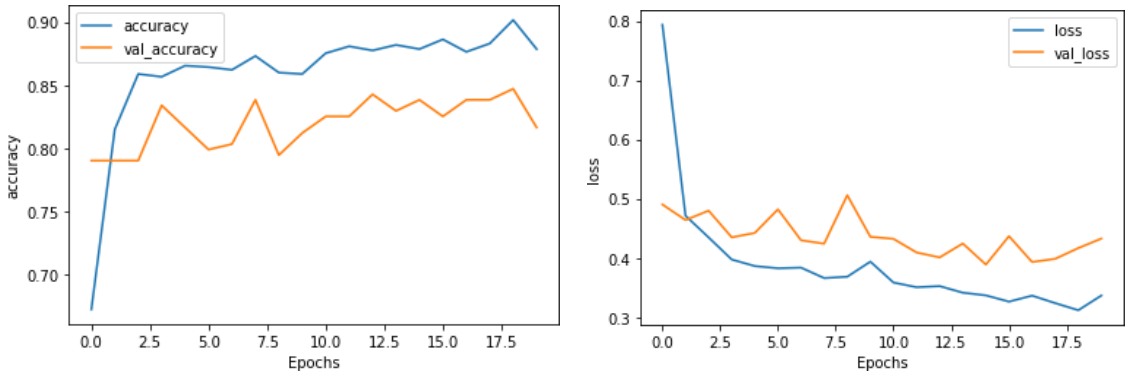
Algoritma	Akurasi	*Training Loss*	*Training Accuracy*	*Validation Loss*
RNN	79%	36%	86%	50%
LSTM	78%	48%	81%	54%
Bi-LSTM	82%	33%	87%	43%



Gambar 5. Grafik hasil pelatihan RNN



Gambar 6. Grafik hasil pelatihan LSTM



Gambar 7. Grafik hasil pelatihan Bi-LSTM

KESIMPULAN

Berdasarkan hasil pengujian yang telah dilakukan, algoritma Bidirectional Long Short Term Memory (Bi-LSTM) telah berhasil diimplementasi untuk kasus klasifikasi terhadap pesan SMS dan terbukti menghasilkan akurasi yang yang lebih baik dari algoritma RNN dan LSTM biasa yang masing-masing menghasilkan akurasi 79% dan 78%, sedangkan Bi-LSTM menghasilkan akurasi terbaik mencapai 82%. Hasil tersebut didapatkan karena pada Bi-LSTM informasi dibaca secara 2 arah sekaligus. Tetapi BiLSTM juga memiliki kelemahan, yaitu membutuhkan dataset yang besar agar dapat menghindari overfitting, dan juga waktu dan biaya komputasi yang dibutuh juga cukup tinggi. Saran untuk penelitian yang berikutnya dapat menghasilkan representasi kata yang lebih banyak dan variatif dengan cara mempertimbangakan pada kombinasi word embedding-nya.

REFERENSI

Alghifari, D. R., Edi, M., & Firmansyah, L. (2022). Implementasi Bidirectional LSTM untuk Analisis Sentimen Terhadap Layanan Grab Indonesia. Jurnal Manajemen Informatika (JAMIKA), 12(2), 89–99.

Aprian, B. A., Azhar, Y., & Nastiti, V. R. S. (2020). Prediksi Pendapatan Kargo Menggunakan Arsitektur Long Short Term Memory. Jurnal Komputer Terapan, 6(2), 148–157.

Aritonang, Y. V., Napitupulu, D. P., Sinaga, M. H., & Amalia, J. (2022). Pengaruh Hyperparameter Pada Fasttext Terhadap Performa Model Deteksi Sarkasme Berbasis Bi-LSTM. Jurnal Teknik Informatika dan Sistem Informasi, 9(3), 2612–2624.

Azhari, Z., Efrizoni, L., Agustin, W., & Yanti, R. (2023). Opinion Mining menggunakan Algoritma Deep Learning untuk Menganalisis Penggunaan Aplikasi Jamsostek Mobile. Indonesian Journal of Computer Science, 12(2), 666–678.

Hikmaturokhman, A., Nafi'ah, H., Larasati, S., Wahyudin, A., Ariprawira, G., & Pramono, S. (2022). Deep Learning Algorithm Models for Spam Identification on Cellular Short Message Service. Journal of Communications, 17(9), 769–776.

Isnain, A. R., Sulistiani, H., Hurohman, B. M., Nurkholis, A., & Styawati. (2022). Analisis Perbandingan Algoritma LSTM dan Naive Bayes untuk Analisis Sentimen. JEPIN ((Jurnal Edukasi dan Penelitian Informatika)), 8(2), 299–303.

Karyadi, Y., & Santoso, H. (2022). Prediksi Kualitas Udara Dengan Metoda LSTM, Bidirectional LSTM, dan GRU. Jurnal Teknik Informatika dan Sistem Informasi, 9(1), 671–684.

Pramunendar, R. A., Prabowo, D. P., & Megantara, R. A. (2022). Metode Recurrent Neural Network (RNN) Dengan Arsitektur Lstm Untuk Analisis Sentimen Opini Publik Terkait Vaksin Covid-19. JURNAL INFORMATIKA UPGRIS, 8(1), 44–48.

Pratama, E. D. (2022). Implementasi Model Long-Short Term Memory (LSTM) pada Klasifikasi Teks Data SMS Spam Berbahasa Indonesia. The Journal on Machine Learning and Computational Intelligence (JMLCI), 1(2), 38–42.

Prijono, B. (2018, April 4). Pengenalan Recurrent Neural Network (RNN) – Bagian 1. indoml.com. https://indoml.com/2018/04/04/pengenalan-rnn-bag-1/

Rahutomo, F., Pratiwi, I. Y. R., & Ramadhani, D. M. (2019). Eksperimen Naïve Bayes Pada Deteksi Berita Hoax Berbahasa Indonesia. JURNAL PENELITIAN KOMUNIKASI DAN OPINI PUBLIK, 23(1), 1–15. https://doi.org/10.33299/jpkop.23.1.1805

Reviantika, F., Azhar, Y., & Marthasari, G. I. (2021). Analisis Klasifikasi SMS Spam Menggunakan Logistic Regression. Jurnal Sistem Cerdas, 4(3), 155–160. https://www.cnbcindonesia.com

Rizki, M., Basuki, S., & Azhar, Y. (2020). Implementasi Deep Learning Menggunakan Arsitektur Long Short Term Memory(LSTM) Untuk Prediksi Curah Hujan Kota Malang. Jurnal Repositor, 2(3), 331–338.