

IMPLEMENTASI ALGORITMA LONG SHORT-TERM MEMORY (LSTM) UNTUK MENDETEKSI UJARAN KEBENCIAN (*HATE SPEECH*) PADA KASUS PILPRES 2019

Aini Suri Talita¹, Aristiawan Wiguna²

^{1,2}Jurusan Teknik Informatika, Fakultas Teknologi Industri, Universitas Gunadarma

Email : ainisuri@staff.gunadarma.ac.id, aristiawan.wiguna@gmail.com

ABSTRAK

Penelitian dengan menggunakan Jaringan Syaraf Tiruan (Artificial Neural Network/ANN) maupun turunannya telah banyak dilakukan, khususnya dalam masalah praktis *data mining*, klasifikasi, *clustering*, ataupun kasus khusus pendeteksian suatu objek. Terdapat beberapa jenis metode yang merupakan tipe khusus dari ANN, misalnya Recurrent Neural Network (RNN). Pada penelitian ini, salah satu arsitektur dari RNN yang biasa digunakan pada masalah *deep learning* yaitu Long Short Term Memory (LSTM) akan diimplementasikan untuk mendeteksi ujaran kebencian (*hate speech*) berkaitan dengan Pemilihan Presiden (Pilpres) 2019. Tahapan pengerjaan dalam penelitian ini dimulai dari studi kepustakaan, pengumpulan data, pra-pemrosesan data, pembuatan model *word2vec*, perancangan model pendeteksi ujaran kebencian, pelatihan model, dan pengujian model. Model ini dilatih dengan menggunakan 950 kalimat dan diuji dengan 190 kalimat dari *dataset* yang bersumber dari media sosial Facebook. Hasil uji coba terbaik dengan 190 kalimat memiliki nilai *recall* 0.7021, yang artinya dari keseluruhan data uji ujaran kebencian, 70.21% benar dideteksi sebagai ujaran kebencian, sedangkan sisanya 29.79% salah dideteksi sebagai bukan ujaran kebencian. Nilai parameter lainnya masih cenderung rendah yang dapat disebabkan oleh kalimat ujaran kebencian yang digunakan mencakup bahasa informal, dan diambil langsung dari media sosial dimana penulisan dari kata-kata seringkali tidak konsisten.

Kata Kunci : Deep Learning, Long Short-Term Memory, RNN, Ujaran Kebencian

ABSTRACT

Researches involving Artificial Neural Network (ANN) or its derivative have been published all around the world, specifically to solve data mining problem, classification, clustering, or detection problems. Recurrent Neural Network is a class of ANN with Long Short Term Memory (LSTM) as its one of the architecture that commonly used in deep learning problems. On this paper, we use LSTM to detect hate speech on social media related with Indonesia President Election on 2019. There are several steps on this research, we start with literature study, data collection, data preprocessing, training step, and testing step. The dataset consist of 950 sentences, while the testing data consist of 190 comments on Facebook. The best model performance was reached with recall value 0.7021, which means that from the whole relevant instances on the testing data, 70.21% were categorized as relevant, on this case as hate speech (HS). The other performance parameter value as in accuracy and precision still quite low due to the testing data that comes directly from social media which highly possible consist of inconsistent choices of words, informal words, or contains grammatically error sentences.

Keyword: Deep Learning, Long Short Term Memory, Hate Speech, RNN

Author Korespondensi (Aini Suri Talita)

Email : ainisuri@staff.gunadarma.ac.id

I. PENDAHULUAN

Etika dalam dunia *online* perlu ditegaskan, mengingat dunia *online* merupakan hal yang sangat familiar dan menyebar di masyarakat Indonesia, maupun dunia. Ketika

terjadi penyebaran berita bohong atau fitnah maupun ujaran kebencian, pada umumnya Internet Service Provider (ISP) menjadi pihak yang dianggap bertanggung jawab, meskipun sebenarnya hal tersebut bukanlah menjadi kewenang ISP namun menjadi tanggung jawab

pihak ketiga yang menyebarkan hal tersebut. Tahun 2019 merupakan tahun politik di Negara Indonesia dimana pada tahun 2019 ini Indonesia menggelar pesta demokrasi atau pemilihan umum untuk memilih wakil rakyat dari DPD, DPRD Kabupaten/Kota, DPRD provinsi, DPR, Presiden dan Wakil Presiden. Pada pemilihan umum ini terdapat berbagai perbedaan dalam pilihan di masyarakat. Semarak pesta demokrasi ini menyebar ke seluruh negeri ini, yang juga melibatkan berbagai media dari yang konvensional hingga ke media sosial. Pada media sosial ini masyarakat dapat mengungkapkan berbagai aspirasi dan dukungannya ke calon wakil rakyat atau pejabat eksekutif yang akan dipilihnya. Tetapi sayangnya dalam proses penyampaian aspirasi itu terdapat berbagai kata-kata yang terkandung ujaran kebencian (*hate speech*) yang seharusnya tidak perlu untuk digunakan.

Semakin maraknya penyebaran ujaran kebencian ini tentu menimbulkan efek negatif baik bagi pengguna media sosial itu sendiri, maupun bagi objek dari ujaran kebencian tersebut. Salah satu cara tradisional yang biasa dilakukan oleh pengguna media sosial untuk mengurangi atau memerangi tindakan ujaran kebencian adalah dengan melaporkan akun atau pihak yang menyebarkan ujaran kebencian tersebut. Ke depannya, dimungkinkan bagi pengelola media sosial untuk secara otomatis menghapus ujaran kebencian yang tersebar di *platform* media sosial yang dikelolanya. Untuk menuju hal ini, tentunya masih diperlukan beberapa tahapan yang harus dilakukan, misalnya tahapan pendeteksian ujaran kebencian. Perlu dibangun suatu mesin yang dapat membedakan antara kalimat normal dengan kalimat yang mengandung ujaran kebencian.

Pada penelitian ini, akan diimplementasikan algoritma Long Short-Term Memory (LSTM) pada kasus ujaran kebencian, khususnya yang berkaitan dengan kasus pemilihan presiden di tahun 2019. Algoritma Long Short-Term Memory [1] merupakan salah satu jenis arsitektur dari Recurrent Neural Network (RNN) yang biasa digunakan pada masalah-masalah yang berkaitan dengan *deep learning*. Terdapat berbagai penelitian yang menerapkan LSTM.

Pada penelitian [2], Long Short Term Memory Recurrent Neural Network digunakan sebagai model prediksi statis untuk memprediksi nilai dari indikator polusi udara dalam deret

waktu. Filter Kalman digunakan sebagai model penyesuaian dinamis (*dynamic adjustment model*).

Pertama, model prediksi statis digunakan untuk memprediksi nilai dari suatu polutan pada pada saat tertentu (misal, saat uji dilakukan), kemudian model penyesuaian dinamis digunakan untuk secara dinamis menyesuaikan nilai prediksi yang diperoleh dari model prediksi statis berdasarkan nilai observasi dari polutan tersebut. Untuk menganalisa kinerja model yang diajukan, penelitian ini menggunakan *mean square error* dari model LSTM-Kalman dan model statis LSTM. Serta digunakan juga nilai *R-squared* (koefisien determinasi) dari model prediksi yang dibangun.

Penelitian selanjutnya yang berkaitan dengan LSTM berhubungan dengan salah satu isu penting pada sistem transportasi cerdas yaitu peramalan kondisi lalu lintas jangka pendek (*short-term traffic forecast*) [3]. Pada penelitian [3], diajukan suatu model yang berdasarkan algoritma LSTM. Penelitian ini juga turut mempertimbangkan data lalu-lintas yang sangat besar dengan struktur hierarki dan mengintegrasikan korelasi *temporal-spatial* pada LSTM untuk menghasilkan peramalan kondisi lalu lintas yang tepat. Model yang dihasilkan juga berdasarkan pada *Internet Of Vehicles* (IOVs) serta analisis korelasi.

Model prediksi trajektori untuk kondisi lalu lintas padat jangka panjang dengan memanfaatkan LSTM diajukan pada [4]. Model LSTM dimodifikasi dengan menambahkan *shortcut connections* dengan tujuan akhir membangun model prediksi trajektori *spatio-temporal*. Percobaan dilakukan dengan menggunakan *dataset* NGSIM I-80 dan US-101. Model ini secara implisit dapat mengukur interaksi dari kendaraan yang “bertetangga” dan diklaim dapat menutupi kekurangan model LSTM klasik dimana LSTM tidak dapat secara simultan mendeskripsikan interaksi spasial antar kendaraan serta relasi sewaktu antara trajektori deret waktu.

Selanjutnya adalah implementasi LSTM di bidang keamanan jaringan. Pengelola jaringan perlu mewaspadaikan akan adanya kemungkinan serangan dari luar atau dalam jaringan, yang berniat mencari kelemahan jaringan, atau mengakses data yang sifatnya rahasia. Pada umumnya *firewall* digunakan untuk mengamankan jaringan. Namun itu saja tidak cukup, untuk lebih mengamankan jaringan

dibangunlah *Intrusion Detection System*. Pada penelitian [5], model LSTM diimplementasikan pada masalah *Intrusion Detection System (IDS)* khususnya menggunakan salah satu *benchmark dataset* yaitu *KDD Cup '99*. Data *KDD Cup '99* yang digunakan adalah versi 10% data dimana terdapat 4 jenis serangan yaitu *Probe*, *Denial of Service (DoS)*, *User to Root (U2R)*, dan *Remote 2 Local (R2L)*. Pada penelitian ini dipilih metode *Random Forest* sebagai metode pemilihan fitur. Data *KDD Cup '99* memiliki lebih dari 40 fitur dan secara keseluruhan terdiri dari lebih dari 400.000 *records* yang akan berakibat pada mahalnya biaya komputasi apabila tidak dilakukan tahapan pemilihan fitur terlebih dahulu. Hasil klasifikasi dengan menggunakan LSTM mencapai keakuratan 98.85%.

Penelitian [6] membahas tentang desain, implementasi, dan evaluasi dari suatu aplikasi *mobile* yang dinamakan *Third-Eye*, yang dapat digunakan pada ponsel untuk memonitor $PM_{2.5}$, yaitu partikel dengan diameter 2.5 mikron atau kurang dari itu. Hal ini umum digunakan pada pengukuran kualitas udara di perkotaan. LSTM dan *Convolutional Neural Network (CNN)* digunakan untuk melatih model inferensia $PM_{2.5}$. Hasil ujicoba menunjukkan *Third-Eye* memiliki akurasi klasifikasi mencapai 81.55%.

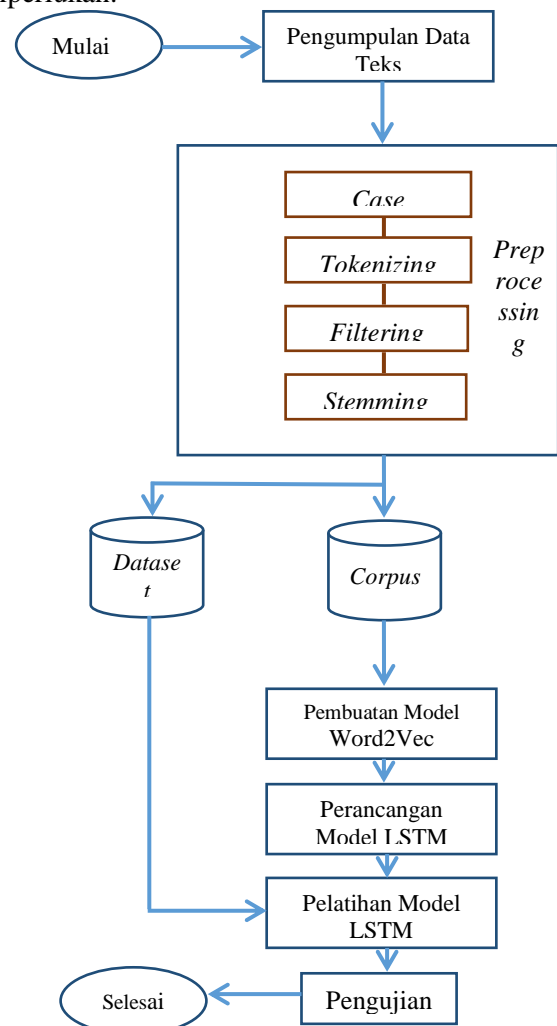
Penelitian [7] mengajukan model *hybrid* yang diklaim sederhana secara komputasi, lebih sederhana dari model fisis, cepat dan efisien yang menggabungkan model prediksi yang bersifat *hybrid data-driven Ensemble Empirical Mode Composition (EEMD)* serta LSTM yang dapat memprediksi suhu harian permukaan tanah (*Daily Land Surface Temperatur/Daily LST*). Model ini dinamakan *EEMD Long Short-Term Memory*. Data yang digunakan adalah data dari *Dongting Lake, Cina* pada 1 January 2014 sampai dengan 31 Desember 2016. Kinerja dari model prediksi diukur dengan parameter *Mean Square Error (MSE)*, *Mean Absolute Error (MAE)*, *Mean Absolute Percentage Error (MAPE)*, *Root Mean Square Error (RMSE)*, Koefisien Korelasi Pearson dan Koefisien Efisiensi Nash-Sutcliffe. Hasil penelitian menunjukkan bahwa model yang diajukan bekerja lebih baik dari 5 model lain dalam memprediksi *Daily LST*, yaitu model yang menggunakan *RNN*, *LSTM*, *Empirical Mode Decomposition (EMD)-RNN*, *EMD-LSTM*, dan *EEMD-RNN*.

Data *corpus* yang digunakan pada penelitian ini berasal dari Universitas Leipzig [8]

dan *dataset* kalimat normal maupun ujaran kebencian berasal dari komentar di media sosial Facebook. Pada penelitian ini, dibatasi *dataset* yang digunakan baik untuk data latih maupun data uji adalah kalimat berbahasa Indonesia baik formal dan nonformal, serta dipilih kalimat yang berhubungan secara spesifik dengan Pilpres Indonesia 2019. Pada penelitian ini bahasa pemrograman yang digunakan adalah *Phyton*.

II. METODOLOGI

Secara umum, tahapan penelitian ini diawali dengan studi literatur, dilanjutkan dengan pengumpulan data untuk membuat *corpus* dan *dataset* yang berasal dari data *corpus* Universitas Leipzig dan data kalimat ujaran kebencian (*Hate Speech/HS*) dan non ujaran kebencian (*Non Hate Speech/NHS*) yang berasal dari Facebook. *Data preprocessing* (pra-pemrosesan data) yang dilakukan terhadap data teks (*corpus* dan *dataset*) adalah pembersihan *stopwords*, *stemming*, dan menghilangkan karakter-karakter yang tidak diperlukan.



Gambar 1. Alur metodologi penelitian

Hasil dari pra-pemrosesan dikumpulkan ke dalam sebuah *file* teks. Selanjutnya, tahapan pembuatan model *Word2Vec*. Teks hasil pra-pemrosesan diubah ke dalam bentuk vektor agar dapat dibaca oleh algoritma Long Short-Term Memory (LSTM). Dilanjutkan dengan merancang model *Long-Short Term Memory* yang sesuai dengan *corpus* dan *dataset* agar model yang dirancang dapat mendeteksi penggunaan ujaran kebencian dengan baik, model dilatih dengan menggunakan *dataset* yang berisi kumpulan kalimat mengandung ujaran kebencian, dan diakhiri dengan pengujian model.

Data *corpus* yang digunakan terdiri dari bermacam data yang bertotal 3 juta artikel. Artikel-artikel diambil dari teks pada *web*, berita *online*, dan lain-lain, kemudian artikel yang diambil dari *website* atau situs-situs berita dan data dari *website* yang dipilih secara acak yang masing-masing memiliki 1 juta artikel. Contoh data yang diambil dari *corpus* dapat dilihat pada Tabel 1.

Tabel 1. Contoh Data *Corpus*

No.	Sumber	Contoh Data
1	Data campuran	Sementara ketiga kelompok kecil itu sedang berusaha menelusuri gerak pasukan Pati, maka para prajurit dan pengawal yang berada di perkemahan tetap bersiaga sepenuhnya untuk menghadapi segala ke?
2	<i>Website</i> atau situs berita	Mungkin bom rakit-an, tapi masih dalam penelitian laboratorium forensik (labfor),” kata Kabid Humas Polda Sulsel, Joko Subroto, Senin (03/07).
3	<i>Website</i> dipilih secara acak	Misalnya ditahun ini ada anggaran Program Pemberdayaan Masyarakat Kelurahan (PPMK) Rp540 juta, kita bisa meningkatkan

		anggaran itu, tetapi tidak juga menguap, dan harus ada pengawasan dan kerjasama dengan pihak yang sukses dalam memberdayakan masyarakat.
--	--	--

Dataset yang digunakan pada penelitian ini diambil dari komentar media sosial Facebook. *Training data* yang digunakan terdiri dari 3 kolom yaitu kolom id, kolom label dan kolom *comment* yang berisi komentar-komentar yang berasal dari Facebook. Kolom label berisi 2 buah kategori yang didefinisikan yaitu *Hate Speech* (HS) dan *Non Hate Speech* (Non_HS). *Training data* berjumlah total 950 komentar. Contoh *training data* dapat dilihat di Tabel 2.

Tabel 2. Contoh *Dataset*

No.	Label	<i>Comment</i>
1	HS	Selamat ya pak, sudah d peralat oleh antek2 politik bapak..Sudah memberikan warisan kebodohan dan kecurangan kepada para penerus bangsa ini..Ini akan menjadi catatan bagi kami masyarakat yg ingin kejujuran dan keadilan
2	HS	Sy sebagai rakyat indonesia, malu punya presiden hasil dari pemilihan yg curang dg merampok dan memanipulasi suara rakyat.
3	HS	jangan pencitraan ,rakyat sudah jenuh dan bosan dengan pencitraan yang tak sesuai dengan citra .
4	Non_HS	Nilai2.pancasila yg perlu di aplikasikan dlm kehidupan di negri ini namun seiringny waktu hanya sebagai slogan, pengakuan, gambaran, dan teriakan saja, entah ada letak salah dimn, ??
5	Non_HS	Kita boleh membanggakan Panutan kita. Tapi gak boleh merendahkan yg lain. Krn semua manusia pasti ada kelebihan dan kekurangan.

Sebelum dilakukan pembuatan model *Word2Vec*, dilakukan pra-pemrosesan terlebih dahulu menggunakan *library Python* yaitu *gensim* [9] pada *corpus* dan *dataset*. *Library gensim* mengandung fungsi untuk melakukan pra-pemrosesan teks yang meliputi:

1. *Case Folding*

Proses *Case Folding* ini bertujuan untuk mengubah huruf dalam teks menjadi huruf standar (huruf kecil). Data yang diterima pada *case folding* hanya huruf 'a' sampai 'z', karakter selain huruf-huruf tersebut dihilangkan dan hanya dianggap delimiter.

2. *Tokenizing*

Setelah melakukan *case folding*, proses selanjutnya adalah data teks dipecah per kata pada tahapan *tokenizing*.

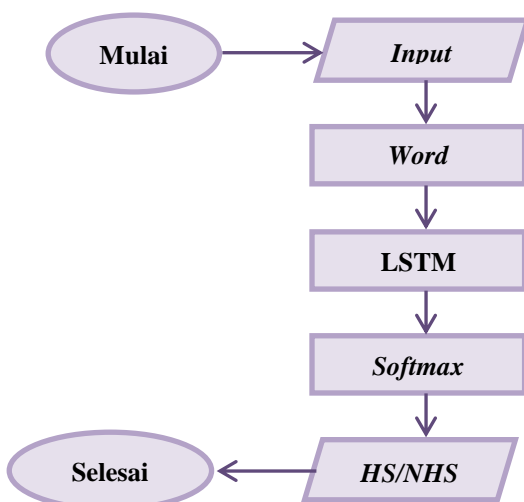
3. *Filtering*

Pada tahap *filtering* dilakukan proses penghapusan kata yang tidak diperlukan atau diluar objek penelitian (*stopwords*). Beberapa kata *stopwords* pada penelitian ini diantaranya adalah 'maka', 'akan', 'yang', 'untuk', 'dan', 'juga', 'dari', 'di', 'ya', serta 'kan'.

4. *Stemming*

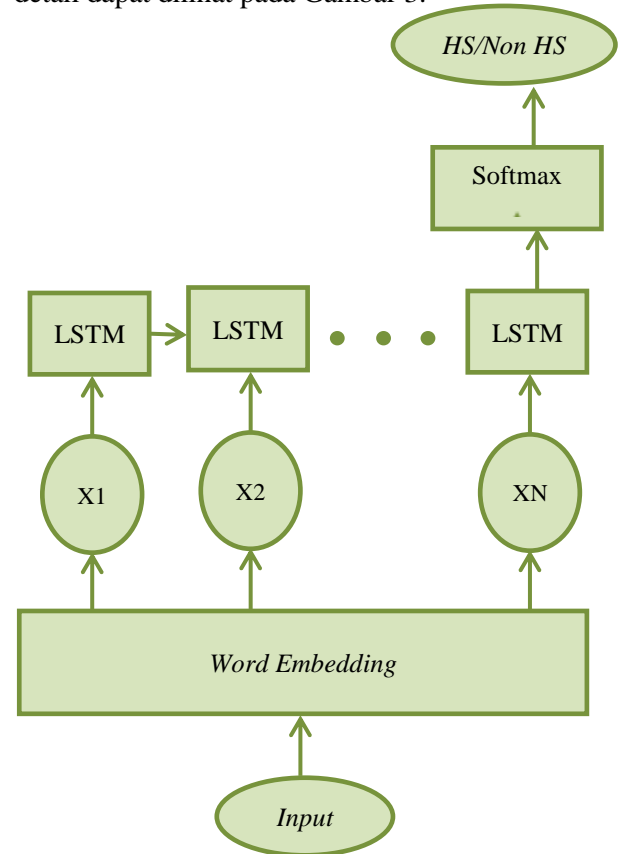
Setelah proses *filtering* dilakukan, proses yang terakhir adalah *stemming* dimana kata yang ada pada data ditransformasi ke kata dasarnya.

Dalam perancangan *Word2Vec*, digunakan *library gensim*. *Library gensim* diperlukan untuk melakukan *import* fungsi *Word2Vec*, dimana fungsi ini digunakan untuk mengubah teks pada "*corpus.txt*" dari kata menjadi vektor untuk dapat dijalankan pada algoritma LSTM dan hasil dari *Word2Vec* disimpan ke model *word2vec_300.model*.



Gambar 2. Gambaran umum LSTM

Pada penelitian ini, LSTM digunakan sebagai algoritma pada program pendeteksi ujaran kebencian, cara kerja dari LSTM secara umum dapat dilihat pada Gambar 2. Algoritma LSTM dipilih karena memiliki memori yang besar, cocok digunakan untuk data berbentuk *sequence*. Data diinput terlebih dahulu sebelum masuk ke algoritma LSTM melalui tahap *word embedding*, algoritma LSTM kemudian memproses hasil dari *word embedding*. Sesudah input pertama selesai diproses, hasil tersebut dimasukkan ke memori, dimana hasil input sebelumnya akan digunakan untuk proses berikutnya dalam memeriksa hubungan antara *input* pertama dengan kedua, ketiga, keempat dan selanjutnya. Hasil proses berikutnya dipengaruhi oleh hasil *input* sebelumnya. *Input* berupa kalimat yang terdiri dari beberapa kata. Cara kerja LSTM secara lebih detail dapat dilihat pada Gambar 3.



Gambar 3. Cara kerja algoritma LSTM

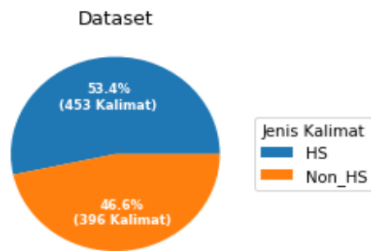
III. HASIL DAN PEMBAHASAN

Dalam proses pelatihan model, dilakukan 100 kali iterasi, model diberi 100 data per satu kali masukan agar penggunaan memori tidak terlalu

besar dan mengurangi frekuensi pembaruan bobot tiap *layer*.

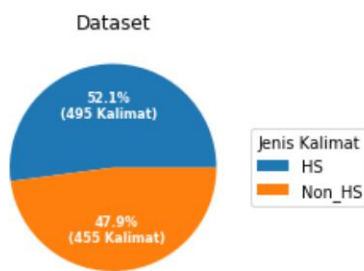
Bobot awal diinisiasi dengan algoritma Xavier [10]. Xavier menginisiasi bobot dari distribusi Gaussian dengan nilai $\mu = 0$ dan selalu membuat nilai variansi tetap setiap kali terdapat sinyal masukan melalui lapisan di dalam model.

Selanjutnya adalah tahapan pengujian model pendeteksian ujaran kebencian. Pengujian ini dilakukan dua kali dengan data latih berjumlah 849 kalimat dimana terdiri dari 453 kalimat ujaran kebencian dan 396 kalimat non ujaran kebencian dan *dataset* untuk pengujian yang kedua ditambah 101 kalimat sehingga berjumlah 950 kalimat dimana terdiri dari 495 kalimat ujaran kebencian dan 455 kalimat non ujaran kebencian. *Dataset* pengujian pertama dapat dilihat pada Gambar 4. dan *dataset* kedua dapat dilihat melalui Gambar 5.



Gambar 4. Kalimat HS dan Non_HS pada tahapan pelatihan untuk pengujian pertama

Pada Tabel 3., diberikan beberapa contoh data uji yang digunakan pada penelitian ini.



Gambar 5. Kalimat HS dan Non_HS pada tahapan pelatihan untuk pengujian kedua

Tabel 3. Data Uji

No.	Kalimat	Jenis
1	Siapa yang mau geser..pak jelaskan kok kamu sumber hoax urus tuh mobil ESEMKA katanya dulu	HS

	ada 1000 orang yg inden..PEMBOHONG !	
2	Mundur dari Jabatan dong... Gak malu apa kampanye pakai duit negara.... Itu duit rakyat, jangan dibuat kampanye pak #SaveUratMalu	HS
3	Belajar saling menghargai karna kita semua manusia. Termasuk diriku..	Non_HS
4	Pilih yg udah nyata kerjanya pak jokowi lah terbaik Salam 2 priode	Non_HS
5	Tolong pak turun kan harGa Pesawat dan jangan naik kan harga minyak. Naik gaji guru 500rb	Non_HS

Dari pengujian data tersebut didapat hasil yang beberapa contohnya diberikan pada Tabel 4.

Tabel 4. Hasil Pengujian

No	Kalimat	Jenis	Hasil Prediksi
1	Siapa yang mau geser..pak jelaskan kok kamu sumber hoax urus tuh mobil ESEMKA katanya dulu ada 1000 orang yg inden..PEMBOHONG !	HS	HS
2	Mundur dari Jabatan dong... Gak malu apa kampanye pakai duit negara.... Itu duit rakyat, jangan dibuat kampanye pak #SaveUratMalu	HS	HS
3	Belajar saling menghargai karna kita semua manusia. Termasuk diriku..	Non_H S	Non_H S
4	Pilih yg udah nyata kerjanya pak jokowi lah terbaik Salam 2 priode	Non_H S	Non_H S
5	Tolong pak turun kan harGa Pesawat dan jangan naik kan harga	Non_H S	Non_H S

minyak. Naik gaji guru 500rb		
------------------------------	--	--

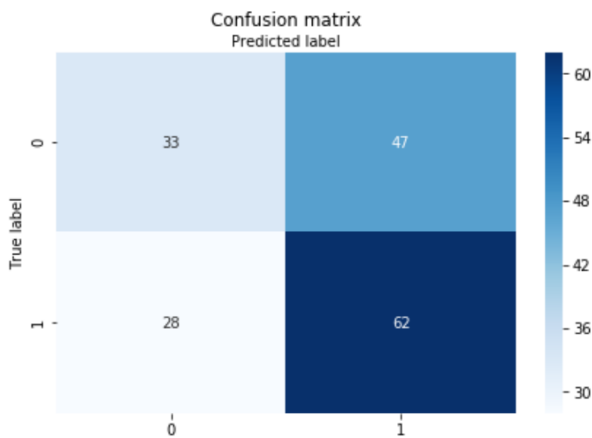
Performa dari model pendeteksi ujaran kebencian diukur dengan menggunakan beberapa parameter yaitu akurasi (*accuracy*), *recall*, dan *precision*. Formula dari ketiga parameter tersebut diberikan pada Persamaan (1-3).

$$Akurasi = \left(\frac{TP+TN}{TP+TN+FP+FN} \right) \times 100\% \quad (1)$$

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

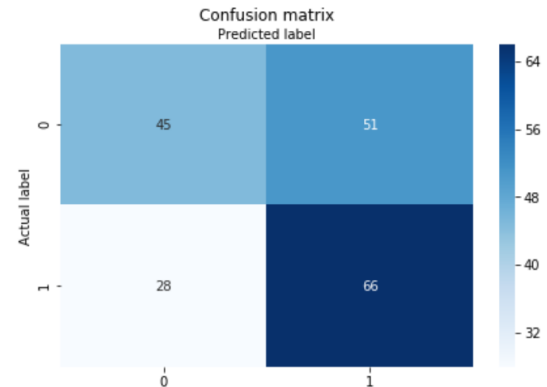
Dengan TP (*True Positive*) memiliki makna banyaknya data uji yang dikategorikan sebagai kelas positif, dalam hal ini *Hate Speech* (HS) dan sebenarnya memang HS. *True Negative* (TN) menyatakan banyaknya data uji yang dikategorikan sebagai kelas negatif, dalam hal ini *Non Hate Speech* (NHS) dan sebenarnya memang NHS. Selanjutnya *False Positif* (FP) menyatakan banyaknya data uji yang dideteksi sebagai positif HS dan sebenarnya masuk dalam kategori NHS, sedangkan *False Negative* (FN) menyatakan banyaknya data uji yang dideteksi sebagai data negatif atau bukan ujaran kebencian (NHS) dimana kenyataannya merupakan ujaran kebencian (HS). Untuk mendeskripsikan secara detail performa model diberikan *confusion matrix* pada Gambar 6. dan 7., untuk pengujian pertama dan kedua.



Gambar 6. *Confusion Matrix* Pengujian Pertama

Dapat dilihat pada Gambar 6., total terdapat total 170 data uji dimana 33 data uji dianggap

NHS dan kenyataannya juga NHS, banyaknya data yang salah dideteksi sebagai ujaran kebencian namun sebenarnya NHS adalah 47, banyaknya data yang dideteksi salah sebagai NHS adalah 28, dan banyaknya data yang dideteksi benar sebagai ujaran kebencian adalah 62. Atau dengan kata lain, nilai TN = 33, FP = 47, FN = 28, dan TP = 62. Sehingga didapat akurasinya adalah 55.88%, nilai *recall*-nya adalah 0.6889, dan nilai *precision*-nya adalah 0.5688.



Gambar 7. *Confusion Matrix* Pengujian Kedua

Dapat dilihat pada Gambar 7., total terdapat total 190 data uji dimana 45 data uji dianggap NHS dan kenyataannya juga NHS, banyaknya data yang salah dideteksi sebagai ujaran kebencian namun sebenarnya NHS adalah 51, banyaknya data yang dideteksi salah sebagai NHS adalah 28, dan banyaknya data yang dideteksi benar sebagai ujaran kebencian adalah 66. Atau dengan kata lain, nilai TN = 45, FP = 51, FN = 28, dan TP = 66. Sehingga didapat akurasinya adalah 58.42%, nilai *recall*-nya adalah 0.7021, dan nilai *precision*-nya adalah 0.5641.

Hasil uji terbaik dengan parameter akurasi adalah pengujian kedua dengan akurasi 58.42%, untuk parameter *recall* juga terdapat peningkatan pada pengujian kedua yaitu menjadi 0.7021, namun terdapat penurunan *precision* dari 0.5688 pada pengujian pertama menjadi 0.5641. Makna dari *recall* adalah proporsi dari data HS yang dideteksi sebagai HS dengan keseluruhan data HS yang ada. Sehingga dari keseluruhan data uji pada pengujian kedua yang termasuk HS, 70.21% dideteksi benar sebagai HS, dan 29.79% salah dideteksi sebagai Non HS.

IV. SIMPULAN DAN SARAN

Berdasarkan hasil penelitian yang dilakukan dengan menggunakan *data testing* 190 kalimat dari 950 kalimat dari *dataset*, algoritma *Long Short Term Memory* sudah cukup baik dalam mendeteksi kalimat ujaran kebencian dengan nilai parameter *recall* mencapai 0.7021. Nilai parameter lainnya masih cenderung rendah yang dapat disebabkan oleh kalimat ujaran kebencian yang digunakan mencakup bahasa informal, dan diambil langsung dari media sosial dimana penulisan dari kata-kata seringkali berubah karena disingkat, banyaknya ejaan yang tidak benar atau tidak konsisten, juga kecenderungan mengganti huruf tertentu sebagai angka sehingga tidak dapat terdeteksi sebagai ujaran kebencian sesuai dengan kriteria pada model yang telah dibangun.

Untuk penelitian selanjutnya, hasil pada penelitian ini berpeluang untuk ditingkatkan dengan menambah jumlah data latih maupun data uji serta khususnya untuk kalimat yang bermakna ambigu, atau berupa bahasa informal dan singkatan yang umum digunakan di media sosial. Penelitian selanjutnya juga dapat menggunakan metode *k-fold cross validation* untuk menentukan akurasi.

V. UCAPAN TERIMA KASIH

Terima kasih kepada segenap pimpinan Universitas Gunadarma yang telah mengajukan dukungan materiil dan immaterial sehingga penelitian ini dapat diselesaikan dengan baik .

REFERENSI

- [1] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, 8, pp. 1735-1780, 1997.
- [2] X. Song, J. Huang and D. Song, "Air Quality Prediction based on LSTM-Kalman Model," in *2019 IEEE 8th Joint International Information Technology and Artificial Intelligence Conference (ITAIC)*, Chongqing, China, 2019, pp. 695-699.
doi: 10.1109/ITAIC.2019.8785751
- [3] Z. Zhao, W. Chen, X. Wu, P. C. Y. Chen and J. Liu, "LSTM network: a deep learning approach for short-term traffic forecast," *IET Intelligent Transport Systems*, vol. 11, no. 2, pp. 68-75, 3 2017.
doi: 10.1049/iet-its.2016.0208
- [4] S. Dai, L. Li and Z. Li, "Modeling Vehicle Interactions via Modified LSTM Models for Trajectory Prediction," *IEEE Access*, vol. 7, pp. 38287-38296, 2019.
doi: 10.1109/ACCESS.2019.2907000
- [5] W. Anani and J. Samarabandu, "Comparison of Recurrent Neural Network Algorithms for Intrusion Detection Based on Predicting Packet Sequences," in *2018 IEEE Canadian Conference on Electrical & Computer Engineering (CCECE)*, Quebec City, QC, 2018, pp. 1-4.
doi: 10.1109/CCECE.2018.8447793
- [6] L. Liang, L. Wu, Z. Yu, M. Huadong, and Z. Cheng, "Third-Eye: A Mobilephone-Enabled Crowdsensing System for Air Quality Monitoring," *ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, New York, USA, March 2018, Article No. 20 20:1 – 20:26.
- [7] Z. Xike, Z. Qiuwen, Z. Gui, N. Zhiping, G. Zifan, and Q. Huafei, "A Novel Hybrid Data-Driven Model for Daily Land Surface Temperature Forecasting Using Long Short-Term Memory Neural Network Based on Ensemble Empirical Mode Decomposition," *International Journal of Environmental Research and Public Health*, vol. 15, 5, May 2018.
1032.DOI:10.3390/ijerph15051032
- [8] <http://wortschatz.uni-leipzig.de/en/download>, diunduh pada 24 juni 2019
- [9] B. Srinivasa-Desikan, *Natural Language Processing and Computational Linguistics: A practical guide to text analysis with Python, Gensim, spaCy, and Keras*. Birmingham UK: Packt Publishing, 2018.
- [10] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *The 13th International Conference on Artificial Intelligence and Statistics (AISTATS)*, Sardinia, Italy, 2010, pp. 249-256.