

Penerapan Metode LSTM Pada Sistem Klasifikasi Komentar Publik Yang Termasuk Jenis Pelanggaran Undang-Undang ITE

Nadaa Qur'atul 'Ain*¹, Bambang Pramono², Asa Hari Wibowo³

^{1,2,3}Jurusan Teknik Informatika, Fakultas Teknik, Universitas Halu Oleo

e-mail: *¹nadaaqraturain34@gmail.com, ²bambang.pramono@uho.ac.id, ³asa.hari@uho.ac.id

Abstrak

Twitter merupakan media sosial yang menempatkan Indonesia sebagai negara dengan pengguna ke-5 terbanyak di dunia. Besarnya jumlah pengguna *Twitter* menjadikan media ini sebagai salah satu sarana untuk memperoleh simpati dan dukungan publik terhadap seseorang guna memenangkan suatu kontes perebutan kekuasaan (pemilihan presiden/pilpres). Adanya ambisi yang besar untuk memenangkan bakal calon dukungannya memicu oknum-oknum tertentu melakukan usaha-usaha untuk menjatuhkan pesaingnya lewat media sosial *Twitter* yang kegiatan-kegiatan tersebut merupakan suatu pelanggaran hukum. Penelitian ini bertujuan untuk mengembangkan sebuah sistem klasifikasi yang mampu mengklasifikasikan komentar publik (*tweet*) ke dalam jenis-jenis pelanggaran Undang - Undang No. 19 Tahun 2016 tentang perubahan atas Undang - Undang Nomor 11 Tahun 2008 tentang Informasi dan Transaksi Elektronik (ITE) menggunakan metode klasifikasi *Long-Short Term Memory*. Berdasarkan hasil penelitian diperoleh *accuracy* sebesar 80%, *precision* sebesar 53,3% dan *recall* sebesar 58%, artinya metode *Long-Short Term Memory* ini cukup dapat digunakan untuk melakukan klasifikasi komentar publik (*tweet*) kedalam pelanggaran Undang Undang Informasi dan Transaksi Elektronik (ITE).

Kata kunci— Klasifikasi, *Text Mining*, LSTM

Abstract

Twitter is a social media that places Indonesia as the country with the 5th largest users in the world. The large number of twitter users makes this media one of the means to gain public sympathy and support for someone to win the presidential election. The ambition to win the candidate they support triggers certain individuals to make efforts to bring down their competitors through social media *Twitter*, which activities are a violation of the law. The purpose of this research is to build a classification system that is able to classify public comments (*tweets*) into types of violations of law using the *Long-Short Term Memory* classification method. Based on the research results obtained *accuracy* of 80%, *precision* of 53,3% and *recall* of 58%, meaning that the *Long-Short Term Memory* method can be used to classify public comments (*tweets*) into violations of the Electronic Information and Transactions Act (ITE).

Keywords— *Classification*, *Text Mining*, *Long Short Term Memory*

1. PENDAHULUAN

Jumlah pengguna internet di Indonesia terus mengalami pertumbuhan. Perubahan

kebiasaan masyarakat dalam beraktivitas pasca pandemi Covid-19 merupakan salah satu faktor utama melonjaknya jumlah pengguna internet. Berdasarkan laporan Profil Internet

Indonesia 2022, dari total 210 juta orang pengguna internet diperoleh 90 persen atau sekitar 190 juta merupakan pengguna aktif media sosial [1].

Pada media sosial *Twitter* penggunaanya berasal dari berbagai latar belakang, karakter dan opini yang berbeda-beda [2]. Akibatnya seringkali opini atau komentar dari suatu pengguna bersinggungan dengan pengguna lainnya, sehingga tidak jarang mengakibatkan pertikaian yang sadar ataupun tidak sadar dapat mencederai hak dari pengguna lain yang dilindungi oleh hukum. Apalagi pada masa dimana suatu negara sedang atau akan mengadakan pemilihan umum Presiden (pilpres).

Dalam sejarah perhelatan pemilihan Presiden di Indonesia yang dipilih secara langsung oleh rakyat, selalu terdapat lebih dari satu kandidat bakal calon Presiden yang diusung oleh suatu partai politik yang dipercaya mampu untuk memenangkan hati masyarakat. Akibatnya lahir pendukung-pendukung yang berambisi besar untuk memenangkan bakal calon yang didukung. Adanya ambisi ini memicu munculnya oknum-oknum yang berusaha untuk menjatuhkan kredibilitas bakal calon pesaingnya dengan menyebarkan kebencian, isu-isu mengenai Suku, Agama, dan Ras (SARA), *hoax*, dan lain sebagainya melalui media sosial yang kegiatan-kegiatan tersebut merupakan pelanggaran dalam Undang-Undang No. 19 Tahun 2016 tentang perubahan atas Undang-Undang Nomor 11 Tahun 2008 mengenai Informasi dan Transaksi Elektronik (ITE).

Dilansir dari data yang dirilis oleh *Twitter*, terdapat lebih dari 124 juta *tweet* yang membahas mengenai pemilu 2019 lewat tagar #Pemilu2019 (23 September 2018 - 24 April 2019). Seiring dengan bertambahnya jumlah pengguna *Twitter*, besarnya jumlah data ini diperkirakan akan semakin meningkat pada pemilihan umum selanjutnya yang direncanakan akan dilaksanakan di tahun 2024.

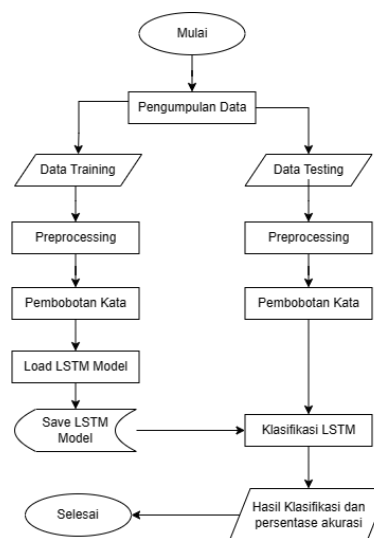
Penelitian terkait berjudul “Penerapan Metode *Long-Short Term Memory* untuk Klasifikasi Pada *Hate Speech*” oleh Kholifatullah & Prihanto (2023) yang bertujuan mendeteksi *hate speech* ke dalam 12 label dengan pembagian 80% data latih dan 20% data uji, hasilnya menunjukkan akurasi pengklasifikasian dengan metode LSTM

sebesar 87.10% [3]. Selain itu, penelitian yang berjudul “Klasifikasi Nama Paket Pengadaan Menggunakan *Long Short-Term Memory* (LSTM) Pada Data Pengadaan” oleh Fajri & Syaiful (2022) yang bertujuan untuk mengklasifikasikan data pengadaan ke dalam 4 kategori paket serta melakukan komparasi metode LSTM dan Naïve Bayes, hasilnya diperoleh akurasi metode LSTM unggul dengan akurasi sebesar 90,25% [4]. Penelitian lain berjudul “Analisis Sentimen Pembangunan *Mass Rapid Transit* (MRT) Jakarta Pada *Twitter* Menggunakan Metode *Improved K-Nearest Neighbor*” oleh Aksan, dkk. (2022) yang berujuan menganalisis polaritas masyarakat terhadap pembangunan MRT di Jakarta, hasilnya diperoleh akurasi klasifikasi sebesar 88,09% [5].

Berdasarkan permasalahan diatas, dibutuhkan sistem yang secara otomatis dapat melakukan klasifikasi *tweet* yang merupakan pelanggaran Undang-Undang ITE. Adapun otomatisasi dapat diwujudkan dengan menggunakan algoritma *Long-Short Term Memory* (LSTM). Penelitian ini di harapkan dapat membantu mewujudkan Masyarakat yang bijak dalam bermedia sosial.

2. METODE PENELITIAN

Gambar 1 menunjukkan tahapan penelitian yang dimulai dengan pengumpulan data, *preprocessing*, pembobotan kata, klasifikasi dengan metode *Long-Short Term Memory* (LSTM), serta perhitungan akurasi menggunakan *confusion matrix*.



Gambar 1 Metode Penelitian

2.1 Pengumpulan Data

Data dikumpulkan dengan cara *scrapping* menggunakan *tweet harvest*. Data yang dikumpulkan merupakan *tweet* berbahasa Indonesia mengenai “pemilihan presiden 2024”. *Tweet* yang telah dikumpulkan selanjutnya dilakukan *labelling* manual dan validasi label oleh validator. Lalu kemudian dibagi menjadi 80% data latih dan 20% data uji. Adapun label/kategori yang digunakan serta contoh *labelling manual* dapat dilihat pada Tabel 1.

Tabel 1 Label/Kategori Pelanggaran

Label	Contoh <i>Tweet</i>	Aturan Hukum
Ujaran Kebencian (UK)	Pengamat bego Yg ngomong ganjar ngawur Umr jateng aja kurang 2jt sok sok an bilang gaji 30jt	Pasal 45A ayat (2) Undang-Undang No. 19 Tahun 2016
Pencemaran Nama Baik (PNM)	@KangManto123 Ganjar bapak bokep dan prostitusi indonesia https://t.co/9Ime7cyLPT	Pasal 45 ayat (3) Undang-Undang No. 19 Tahun 2016
Biasa (B)	@OposisiCerdas Ya sdh merapat ke Ganjar atau Prabowo saja, tegas gak usah wacana	-

Dari hasil *labelling* manual diperoleh jumlah data berdasarkan kategori pelanggaran seperti pada Tabel 2.

Tabel 2 Komposisi Data

Data Latih (80%)	UK	37
	PNM	249
B	244	
Data Uji (20%)	UK	10
	PNM	62
	B	58
Total		660

2.2 Preprocessing

Preprocessing adalah tahapan persiapan teks untuk dapat di proses [6]. Terdapat beberapa tahapan dalam *preprocessing*, diantaranya:

a) *Cleaning*

Tahapan ini bertujuan untuk membersihkan atau menghilangkan karakter yang tidak diperlukan pada teks [6]. Tabel 3 merupakan contoh *cleaning*.

Tabel 3 Tahapan *Cleaning*

Teks Awal	<i>Cleaning</i>
@OposisiCerdas Ya sdh merapat ke Ganjar atau Prabowo saja, tegas gak usah wacana	Ya sdh merapat ke Ganjar atau Prabowo saja tegas gak usah wacana

b) *Case Folding*

Tahapan ini bertujuan untuk mengubah seluruh huruf dalam teks menjadi huruf kecil (*lowercase*) [6]. Tabel 4 merupakan contoh *case folding*.

Tabel 4 Tahapan *Case Folding*

Teks Awal	<i>Case Folding</i>
Ya sdh merapat ke Ganjar atau Prabowo saja tegas gak usah wacana	ya sdh merapat ke ganjar atau prabowo saja tegas gak usah wacana

c) *Normalization*

Tahapan ini bertujuan untuk mengubah dan memperbaiki kata ke dalam kata yang memiliki arti berdasarkan Kamus Besar Bahasa Indonesia (KBBI) [6]. Tabel 5 merupakan contoh *normalization*.

Tabel 5 Tahapan *Normalization*

Teks Awal	<i>Normalization</i>
ya sdh merapat ke ganjar atau prabowo saja tegas gak usah wacana	iya sudah merapat ke ganjar atau prabowo saja tegas tidak usah wacana

d) *Stemming*

Tahapan ini bertujuan untuk mengubah bentuk kata menjadi kata dasar atau tahap mencari root dari tiap kata [7]. Tabel 6 merupakan contoh *stemming*.

Tabel 6 Tahapan *Stemming*

Teks Awal	<i>Stemming</i>
iya sudah merapat ke ganjar atau prabowo saja tegas tidak usah wacana	iya sudah rapat ke ganjar atau prabowo saja tegas tidak usah wacana

e) *Filtering/Stopword*

Tahapan ini bertujuan untuk memilih kata yang dianggap penting dengan menghapus kata-kata yang dianggap tidak penting (*stopword*) [7]. Tabel 7 merupakan contoh *filtering*.

Tabel 7 Tahapan *Filtering*

Teks Awal	<i>Filtering</i>
iya sudah rapat ke ganjar atau prabowo saja tegas tidak usah wacana	sudah rapat ganjar prabowo saja tegas tidak usah wacana

f) *Tokenizing*

Tahapan ini bertujuan untuk memotong kalimat sehingga menjadi kata-kata Tunggal [6]. Tabel 8 merupakan contoh *tokenizing*.

Tabel 8 Tahapan *Tokenizing*

Teks Awal	<i>Tokenizing</i>
sudah rapat ganjar prabowo saja tegas tidak usah wacana	'sudah', 'rapat', 'ganjar', 'prabowo', 'saja', 'tegas', 'tidak', 'usah', 'wacana'

2.3 Pembobotan Kata (*Word Embedding*)

Data hasil *preprocessing* selanjutnya harus terlebih dahulu diubah ke dalam bentuk numerik atau vektor-vektor yang merepresentasikan setiap kata atau token sehingga selanjutnya dapat diproses atau dilakukan perhitungan secara matematik. Tahapan mengkonversi setiap kata ke dalam bentuk numerik inilah yang disebut *word embedding* [8].

Word embedding yang digunakan pada penelitian ini adalah *Global Vector for Word Representation* (GloVe) yang merepresntasikan kata ke menjadi *word embedding* yang secara khusus dapat sinonim/analogi kata (semantik kata), serta pengenalan entitas kata [9].

2.4 *Recurrent Neural Networks* (RNN)

Recurrent Neural Networks (RNN) merupakan jenis arsitektur yang lain dari *Artificial Neural Networks* (ANN) yang dikembangkan secara khusus untuk dapat menangani atau memproses data yang bersambung atau berurutan (*sequential data*) seperti data *time series* (deretan data-data historis), teks (*sequence* huruf, sub-kata, kata, atau kalimat), suara, atau video [10].

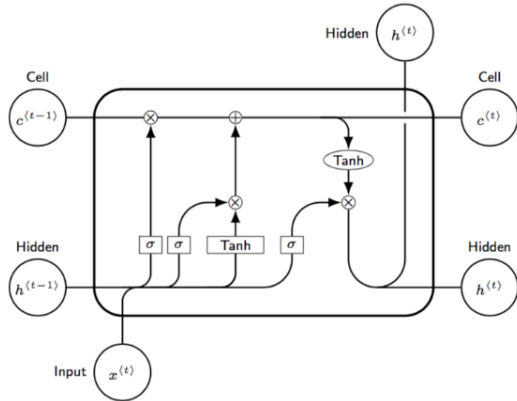
2.5 *Backpropagation Through Time* (BPTT)

Backpropagation adalah teknik yang digunakan untuk menghitung *gradient* dari parameter *neural network*, metode ini melewati jaringan dengan cara berjalan mundur, dari lapisan *output* menuju lapisan *input* berdasarkan aturan rantai (chain rule) yang dipelajari pada kalkulus. *Backpropagation* menyimpan turunan parsial yang dibutuhkan ketika menghitung gradien terhadap beberapa parameter. Dalam proses pelatihan (*training*) *deep learning*, *forward propagation* dan *backpropagation* saling berhubungan satu sama lain.

Backpropagation Through Time (BPTT) adalah sebuah teknik untuk memperbarui parameter yang terdapat pada algoritma jenis RNN dan LSTM.

2.6 *Long-Short Term Memory*

Long Short-Term Memory (LSTM) merupakan algoritma hasil pengembangan dari *Recurrent Neural Network* (RNN) yang diperkenalkan oleh Sepp Hochreiter dan Jürgen Schmidhuber pada tahun 1997. LSTM dibuat untuk menangani masalah ketidakmampuan dalam menyimpan informasi ketika proses *learning* apabila terlalu banyak informasi yang harus disimpan akibat *vanishing* atau *exploding gradient* pada RNN [10]. Adapun arsitektur dari LSTM seperti yang ditunjukkan pada Gambar 2.



Gambar 2 Arsitektur LSTM [11]

LSTM memiliki tiga *gates* atau gerbang yang masing-masing memiliki peran [10], diantaranya:

1) *Forget gate*

Pada *forget gate* ditentukan informasi yang akan dibuang dari *cell state* sebelumnya dengan menggunakan fungsi *sigmoid*. Formalasinya terdapat pada Persamaan (1).

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

2) *Input Gate*

Pada input gate ditentukan informasi yang akan ditambahkan ke *cell state*.

Formulasi untuk *input gate* dapat dilihat pada Persamaan (2), dilanjutkan dengan Persamaan (3), dan Persamaan (4) untuk memperoleh hasil perhitungan *cell state*.

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

$$C'_t = \tanh(w_c \cdot [h_{t-1}, x_t] + b_c) \quad (3)$$

$$C_t = f_t * C_{t-1} + i_t * C'_t \quad (4)$$

3) *Output Gate*

Pada *output gate* ditentukan *output* dari unit LSTM dengan menggunakan Persamaan (5) dan Persamaan (6).

$$o_t = \sigma(w_o \cdot [h_{t-1}, x_t] + b_o) \quad (5)$$

$$h_t = o_t * \tanh(C_t) \quad (6)$$

3. HASIL DAN PEMBAHASAN

Pada penelitian ini, sistem dibangun dengan menggunakan *Django python* dan *library tensorflow*. Gambar 3 merupakan *source code* dari implementasi metode *long-short term memory* yang sebelumnya telah terlebih dahulu dilakukan proses *preprocessing*, *word embedding* serta *padding* pada masing-masing data latih (*data training*) maupun data uji (*data testing*) untuk memperoleh panjang data yang sama.

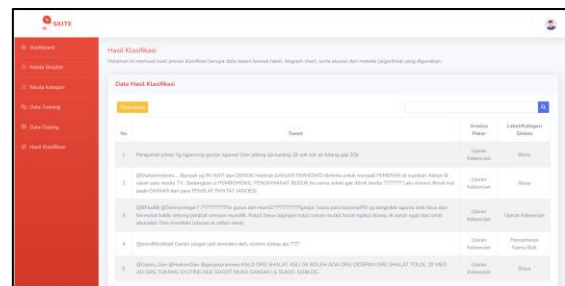
```
# Model
model = Sequential()
model.add(Embedding(input_dim=vocab_size, output_dim=128,
                    input_length=padded_data.shape[1]))
model.add(LSTM(64))
model.add(Dense(len(set(labels)), activation='softmax'))
model.compile(optimizer='adam', loss='categorical_crossentropy',
              metrics=['accuracy'])

# Train Model
model.fit(padded_data, one_hot_labels, epochs=100, batch_size=32,
        validation_data=(padded_data_testing, one_hot_labels_testing))
```

Gambar 3 Source Code Implementasi LSTM

Dari *source code* pada Gambar 3, data latih (*padded_data*) dan kategori/label pelanggarannya (*one_hot_label*) akan terlebih dahulu dilakukan pembelajaran (*training*) untuk mendapatkan bobot atau model LSTM menggunakan algoritma BPTT (*Backpropagation Through Time*).

Bobot yang di dapatkan selanjutnya akan digunakan untuk klasifikasi atau memperoleh kategori dari setiap komentar publik yang terdapat pada data uji. Hasil klasifikasi data uji oleh sistem ditunjukkan pada Gambar 4 dan Gambar 5.



Gambar 4 Interface Hasil Klasifikasi



Gambar 5 Interface Hasil Klasifikasi (2)

Dari Gambar 4 dapat dilihat beberapa hasil klasifikasi seperti pada Tabel 9.

Tabel 9 Hasil Klasifikasi

Komentar Publik (Tweet)	Label Pakar <i>actual</i>	Label Sistem <i>predicted</i>
Pengamat bego Yg ngomong ganjar ngawur Umr jateng aja kurang 2jt sok sok an bilang gaji 30jt	UK	UK
Hati-hati jika Anies memimpin negara ini. Bisa-bisa jadi sarang teroris!! https://t.co/ccYhWZKFEO	UK	UK
@BangPakatDayak Tukang bual ni pasangan Prabowo & amp anies	PNB	PNB
Anies manusia munafik, ngaku merakyat tapi kemana mana pake jet pribadi https://t.co/24BITJsQwz	PNB	B
@tempodotco Ini biar di pilih kahh wkwk... tetep gua mah sapa pun wakilnya.... tetep gue ga ke lain hati sama pakde ganjar	B	B

Hasil klasifikasi data uji selanjutnya dilakukan pengujian dengan menggunakan *matrix confusion* untuk memperoleh performa sistem. *Confusion matrix* adalah sebuah matriks yang memuat data hasil klasifikasi yang dilakukan oleh sistem klasifikasi, baik secara aktual maupun prediktif [12].

Berdasarkan hasil klasifikasi dari 130 data uji, terbentuk matriks *confusion* seperti pada Tabel 10.

Tabel 10 Confusion Matrix

Analisis Pakar (<i>actual</i>)	Klasifikasi Sistem (<i>predicted</i>)		
	PNM	UK	B
PNM	51	0	7
UK	7	0	3
B	8	0	54

Dari Tabel 10 diperoleh nilai nilai *true positive* (TP), *true negative* (TN), *false positive* (FP), *false negative* (FN). Dimana TP merupakan jumlah data dengan label hasil klasifikasi yang sama oleh pakar (validator) dan sistem, FP merupakan jumlah data yang oleh pakar diklasifikasikan berlabel 'B' namun sistem diklasifikasikan sebagai 'A', serta FN merupakan jumlah data yang oleh pakar diklasifikasikan sebagai 'A' namun sistem mengklasifikasikan sebagai 'B'. Nilai TP, FP dan FN untuk masing-masing label/kategori dapat dilihat pada Tabel 11.

Tabel 11 TP, FP, FN

	Klasifikasi Sistem (<i>predicted</i>)		
	PNM	UK	B
TP	51	0	54
FP	7 + 8 = 15	0 + 0 = 0	7 + 3 = 10
FN	0 + 7 = 7	7 + 3 = 10	8 + 0 = 8

Sehingga nilai *accuracy*, *precision*, *recall* adalah sebagai berikut.

1) *Accuracy*

Accuracy menunjukkan kedekatan hasil klasifikasi dengan nilai sesungguhnya.

$$Accuracy = \frac{\sum TP}{Jumlah\ Data}$$

$$Accuracy = \frac{51 + 0 + 54}{130} \times 100\% = 80\%$$

2) *Precision* dan *recall*

Untuk *precision* dan *recall* akan dihitung hasil klasifikasi dari setiap label/kategori. Berikut adalah nilai *precision* dan *recall* untuk label "Ujaran Kebencian".

$$Precision = \frac{TP}{FP + TP} = \frac{0}{0 + 0} = \frac{0}{0} = 0\%$$

$$Recall = \frac{TP}{FN + TP} = \frac{0}{10 + 0} = \frac{0}{10} = 0\%$$

Berikut merupakan *precision* dan *recall* untuk kategori "Pencemaran Nama Baik".

$$Precision = \frac{TP}{FP + TP} = \frac{51}{15 + 51} = 77\%$$

$$Recall = \frac{TP}{FN + TP} = \frac{51}{7 + 51} = 87\%$$

Berikut merupakan *precision* dan *recall* untuk kategori “Biasa”.

$$Precision = \frac{TP}{FP + TP} = \frac{54}{10 + 54} = 84\%$$

$$Recall = \frac{TP}{FN + TP} = \frac{54}{8 + 54} = 87\%$$

Sehingga total *precision* dan total *recall* adalah sebagai berikut.

$$Precision_{total} = \frac{77\% + 0\% + 84\%}{3} \\ = \frac{1,61}{3} \times 100\% = 53,3\%$$

$$Recall_{total} = \frac{87\% + 0\% + 87\%}{3} \\ = \frac{1,74}{3} \times 100\% = 58\%$$

Berdasarkan hasil perhitungan performa model klasifikasi dengan menggunakan *confusion matrix* diperoleh total *accuracy* yang cukup baik sebesar 80%, total *precision* sebesar 53,3% dan total *recall* sebesar 58% artinya metode *Long-Short Term Memory* cukup baik dalam melakukan klasifikasi komentar publik (*Tweet*) kedalam kategori pelanggaran-pelanggaran Undang-Undang No. 19 Tahun 2016 tentang perubahan atas Undang-Undang Nomor 11 Tahun 2008 mengenai Informasi dan Transaksi Elektronik (ITE) yang dalam penelitian ini berfokus pada topik pembahasan mengenai “Pemilihan Presiden 2024”.

Berdasarkan hasil perhitungan performa tersebut juga diketahui bahwa masih terdapat beberapa kesalahan klasifikasi. Hasil perhitungan *recall* dan *precision* pada kategori/label “Pencemaran Nama Baik” dan “Biasa” sudah cukup baik dikisaran 77% sampai 87%, hanya saja pada kategori “Ujaran Kebencian” nilai *precision* dan *recall* sebesar 0% artinya seluruh atau sebagian besar komentar publik (*tweet*) yang oleh Analis Pakar (Validator) diklasifikasikan sebagai “Ujaran Kebencian” oleh sistem tidak satupun diklasifikasikan ke dalam kategori/label yang sama.

Kesalahan klasifikasi oleh sistem dapat disebabkan oleh banyak faktor. Pada penelitian ini faktor yang mempengaruhi hasil klasifikasi yaitu adanya ketimpangan atau data yang jumlahnya tidak seimbang pada masing-

masing kategori/label (Tabel 2) sehingga sistem kurang dapat mempelajari dengan maksimal pola dari suatu kategori tertentu yang dalam penelitian ini adalah data dengan label “Ujaran Kebencian”. Faktor lain yang mempengaruhi yaitu berdasarkan hasil klasifikasi sistem yang ditunjukkan oleh Tabel 11, data yang berlabel “Ujaran Kebencian” sebagian besar (7 dari 10) diklasifikasikan oleh sistem sebagai “Pencemaran Nama Baik”, hal ini dikarenakan terdapat komentar-komentar publik (*tweet*) yang didalamnya mengandung dua jenis pelanggaran yaitu dapat dikategorikan sebagai “Ujaran Kebencian” dan/atau “Pencemaran Nama Baik”, sedangkan oleh sistem diklasifikasikan hanya ke salah satu label/kategori.

Selain itu, algoritma *Long-Short Term Memory* juga merupakan sebuah algoritma pembelajaran yang dalam proses klasifikasinya dipengaruhi oleh banyak parameter seperti jumlah *epoch*, *batch size*, *optimizer*, jenis *word embedding*, serta jenis fungsi aktivasi yang digunakan. Jumlah data latih dan data uji juga dapat mempengaruhi pemahaman sistem dalam memprediksi hasil klasifikasi, semakin banyak jumlah data latih dan data uji yang digunakan maka sistem akan semakin akurat dalam mengklasifikasikan data.

4. KESIMPULAN

Kesimpulan yang diperoleh berdasarkan hasil penelitian yang telah dilakukan didapati metode *Long-Short Term Memori* (LSTM) dapat diterapkan dalam sistem klasifikasi komentar publik (*tweet*) yang termasuk jenis pelanggaran undang-undang informasi dan transaksi elektronik (ITE) dengan pembagian data 80% data *training* (data latih) dan 20% data *testing* (data uji) menghasilkan nilai akurasi cukup baik yaitu sebesar 80% dengan nilai total *precision* sebesar 53,3%, dan total *recall* sebesar 58%.

5. SARAN

Pada penelitian lanjutan disarankan untuk memperbanyak jumlah data dengan komposisi yang seimbang disetiap kategori/label dengan topik komentar publik yang lebih luas. Selain itu disarankan untuk

untuk menggunakan algoritma atau parameter lain yang mendukung klasifikasi multilabel.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada Affirudin Mathara's Law Firm yang telah bersedia menjadi validator atau mitra dalam penelitian ini.

DAFTAR PUSTAKA

- [1] N. Habibie, "210 Juta Orang Gunakan Internet pada 2022, Pengguna Aktif Medsos 191,4 Juta." Accessed: Mar. 01, 2023. [Online]. Available: <https://www.merdeka.com/peristiwa/210-juta-orang-gunakan-internet-pada-2022-pengguna-aktif-medsos-1914-juta.html>
- [2] S. R. I. Rezeki, Y. Restiviani, and R. Zahara, "Penggunaan sosial media twitter dalam komunikasi organisasi," *Journal of Islamic and Law Studies*, vol. 04, no. 02, pp. 63–78, 2020.
- [3] B. A. H. Kholifatullah and A. Prihanto, "Penerapan Metode Long Short Term Memory Untuk Klasifikasi Pada Hate Speech," *Journal of Informatics and Computer Science*, vol. 04, no. 3, pp. 292–297, 2023.
- [4] F. N. Fajri and Syaiful, "Klasifikasi Nama Paket Pengadaan Menggunakan Long Short-Term Memory (LSTM) Pada Data Pengadaan," *Journal Building of Informatics, Technology and Science (BITS)*, vol. 4, no. 3, pp. 1625–1633, 2022.
- [5] C. Aksan, B. Pramono, and A. M. Sajiah, "Analisis Sentimen Pembangunan Mass Rapid Transit (MRT) Jakarta Pada Twitter Menggunakan Metode Improved K-Nearest Neighbor," *Jurnal semanTIK*, vol. 8, no. 1, pp. 53–60, Jun. 2022.
- [6] D. Rifaldi, A. Fadlil, and Herman, "Teknik Preprocessing Pada Text Mining Menggunakan Data Tweet Mental Health," *Jurnal Pendidikan Teknologi Informasi*, vol. 3, no. 2, pp. 161–171, 2023.
- [7] A. T. Jaka, "Preprocessing Text untuk Meminimalisir Kata yang Tidak Berarti dalam Proses Text Mining," *Jurnal Informatika Universitas PGRI Semarang*, vol. 1, no. 1, pp. 1–9, 2015.
- [8] M. D. Rhman, A. Djunaidy, and F. Mahananto, "Penerapan Weighted Word Embedding pada Pengklasifikasian Teks Berbasis Recurrent Neural Network untuk Layanan Pengaduan Perusahaan Transportasi," *Jurnal Sains dan Seni ITS*, vol. 10, no. 1, pp. 1–6, 2021.
- [9] D. Tiyasya Putra and E. Budi Setiawan, "Sentiment Analysis on Social Media with Glove Using Combination CNN and RoBERTa," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 10, no. 3, pp. 457–563, 2023.
- [10] Suyanto, K. N. Ramadhani, and S. Mandala, "Deep Learning : Modernisasi Machine Learning Untuk Big Data," Bandung: Informatika, 2019.
- [11] J. Qiu, B. Wang, and C. Zhou, "Forecasting stock prices with long-short term memory neural network based on attention mechanism," *Journal Public Library of Science (PLoS ONE)*, vol. 15, no. 1, pp. 1–15, Jan. 2020.
- [12] D. Normawati and S. A. Prayogi, "Implementasi Naïve Bayes Classifier Dan Confusion Matrix Pada Analisis Sentimen Berbasis Teks Pada Twitter," *Jurnal Sains Komputer & Informatika (J-SAKTI)*, vol. 5, no. 2, pp. 697–711, 2021.