BAB II
TINJAUAN PUSTAKA

2.1 Studi Literatur

Penelitian-penelitian sebelumnya telah menunjukkan perkembangan signifikan dalam penerapan metode *machine learning* dan *deep learning* untuk tugas klasifikasi berbasis data teks. Dalam konteks machine learning, algoritma seperti *Support Vector Machine* (SVM) dan *Random Forest* masih digunakan sebagai pembanding dalam pengujian performa model, meskipun cenderung memiliki keterbatasan dalam menangani data sekuensial seperti teks alami [3]. Sebagai contoh, dalam penelitian klasifikasi aktivitas manusia, performa SVM dan *Random Forest* lebih rendah dibandingkan LSTM, sedangkan LSTM berhasil mencapai akurasi 95%, *precision* 96%, *recall* 95%, dan *F1-score* 95% [3].

Sementara itu, pendekatan *deep learning* dengan arsitektur *Long Short-Term Memory* (LSTM) telah banyak digunakan karena kemampuannya dalam menangkap konteks dan dependensi jangka panjang dalam data teks [2][3][4][5][6][7]. Model LSTM terbukti efektif untuk berbagai tugas klasifikasi, seperti deteksi pelanggaran hukum dalam komentar media sosial dengan akurasi sebesar 80%, precision 53,3%, dan recall 58% [2], klasifikasi aktivitas manusia berbasis sensor dengan akurasi 95% [3], deteksi clickbait pada judul berita dengan akurasi 79%[4], serta deteksi ujaran kebencian dalam *tweet* berbahasa Indonesia dengan akurasi hingga 74% [5]. Selain itu, klasifikasi teks berita menggunakan LSTM menghasilkan *F1-score* sebesar 90,87% [6], dan untuk analisis sentimen pada berita ekonomi di media online, LSTM mencatatkan akurasi sebesar 62% [7].

Penggunaan teknik balancing data seperti *Synthetic Minority Oversampling Technique* (SMOTE) juga terbukti membantu meningkatkan performa klasifikasi dengan mengatasi ketidakseimbangan kelas dalam dataset, sebagaimana ditunjukkan dalam

penelitian klasifikasi aktivitas manusia [3]. Di sisi lain, teknik *preprocessing* seperti *moving average* diterapkan untuk mereduksi dimensi data tanpa menghilangkan informasi penting pada data sensor [3], sedangkan representasi kata menggunakan Word2Vec banyak dimanfaatkan dalam klasifikasi sentimen pada berita ekonomi untuk menyajikan makna semantik dari kata ke dalam vektor numerik yang dapat diproses oleh *model deep learning* [7].

Selain LSTM murni, beberapa studi juga mengeksplorasi kombinasi arsitektur, seperti CNN-LSTM dan LSTM-CNN, yang bertujuan untuk menggabungkan keunggulan CNN dalam mengekstraksi fitur lokal dari teks dan keunggulan LSTM dalam memahami konteks sekuensial [6][7]. Dalam penelitian klasifikasi berita, arsitektur Convolutional LSTM (C-LSTM) menunjukkan performa unggul dengan F1-score sebesar 93,27%, dibandingkan CNN (89,85%) dan LSTM (90,87%) [6]. Sementara itu, pada penelitian analisis sentimen, arsitektur CNN-LSTM mencatatkan akurasi terbaik sebesar 74%, mengungguli LSTM-CNN (65%) dan LSTM murni (62%) [7].

Secara keseluruhan, studi-studi tersebut menunjukkan bahwa metode LSTM, baik secara mandiri maupun dalam bentuk kombinasi, merupakan pendekatan yang menjanjikan dalam pemrosesan bahasa alami dan klasifikasi teks dalam berbagai konteks, terutama dalam bahasa Indonesia. Hasil-hasil penelitian yang konsisten menunjukkan bahwa LSTM mampu memberikan performa yang kompetitif, bahkan unggul, dibandingkan metode lainnya, sehingga memberikan landasan yang kuat untuk pemanfaatannya dalam penelitian ini.

Tabel 2.1 Keaslian Penelitian

No	Judul penelitian	Nama Penulis	Tahun Publikasi	Hasil Penelitian	Perbandingan Penelitian
1	Klasifikasi Aktivitas Manusia *Menggunakan Metode Long Short-Term Memory*	Latansa Nury Izza Afida, Fitra Abdurrachman Bachtiar, Imam Cholissodin	2023	Akurasi LSTM (dataset sekunder): 97% Akurasi LSTM (dataset primer): 95% Akurasi SVM: 96,33% Akurasi Random Forest: 76,89%	Penelitian ini menggunakan data sensor, sedangkan penelitian saya menggunakan data teks naratif kecelakaan. Keduanya sama-sama menggunakan LSTM.
2	Klasifikasi Judul Berita Clickbait menggunakan RNN-LSTM	Widi Afandi, Satria Nur Saputro, Andini Mulia Kusumaningrum, Hikari Ardiansyah, Muhammad Hilmi Kafabi, Sudianto	2022	Akurasi RNN-LSTM: 79%	Penelitian ini fokus pada judul berita singkat, sedangkan penelitian saya menggunakan deskripsi kecelakaan yang lebih panjang. Arsitektur LSTM serupa namun fitur dan konteks berbeda.

No	Judul penelitian	Nama Penulis	Tahun Publikasi	Hasil Penelitian	Perbandingan Penelitian
3	Penerapan Metode LSTM Pada Sistem Klasifikasi Komentar Publik Yang Termasuk Jenis Pelanggaran Undang-Undang ITE	Nadaa Qur'atul 'Ain, Bambang Pramono, Asa Hari Wibowo	2024	Akurasi LSTM: 80% Precision: 53,3% Recall: 58%	Sama-sama klasifikasi teks dengan LSTM, namun fokus dan label berbeda. Penelitian saya lebih fokus pada klasifikasi tingkat keparahan dari teks deskripsi kecelakaan kerja.
4	Penerapan *Convolutional Long Short-Term Memory* untuk Klasifikasi Teks Berita Bahasa Indonesia	Yudi Widhiyasana, Transmissia Semiawan, Ilham Gibran Achmad Mudzakir, Muhammad Randi Noor	2021	Metode: C-LSTM F1-Score: 92,37%	Penelitian ini menggunakan kombinasi CNN dan LSTM, sedangkan saya menggunakan LSTM murni dengan GloVe *embedding*. Data yang digunakan juga berbeda: berita vs deskripsi kecelakaan.

No	Judul penelitian	Nama Penulis	Tahun Publikasi	Hasil Penelitian	Perbandingan Penelitian
5	Penerapan *Long Short-Term Memory* dalam Mengklasifikasi Jenis Ujaran Kebencian pada *Tweet* Bahasa Indonesia	Ni Putu Sintia Watia, Cokorda Pramarthab	2022	Precision: 74% Recall: 77% F1-Score: 75%	Fokus pada klasifikasi ujaran kebencian, sama-sama menggunakan LSTM, namun dataset dan tujuan klasifikasi berbeda.
6	Algoritma LSTM-CNN untuk Sentimen Klasifikasi dengan Word2vec pada Media Online	Dedi Tri Hermanto, Arief Setyanto, Emha Taufiq Luthfi	2021	Model terbaik: CNN-LSTM Akurasi: 62% Recall: 76% Precision: 66%	Sama-sama NLP dan deep learning, tapi embedding yang digunakan Word2Vec, bukan GloVe, dan tugas klasifikasi adalah sentimen, bukan keparahan kecelakaan.

2.2 Dasar Teori

2.2.1 Kecelakaan kerja

Kecelakaan kerja menurut OHSAS 18001:2007, didefinisikan sebagai kejadian yang berhubungan dengan pekerjaan yang dapat menyebabkan cedera atau kesakitan (tergantung dari keparahannya), kejadian kematian (*fatality*), atau kejadian yang dapat menyebabkan kematian[8].

Menurut Heinrich, kecelakaan adalah suatu kejadian yang tidak direncanakan dan tidak terkendali, di mana tindakan atau reaksi dari suatu objek, zat, orang, atau radiasi mengakibatkan cedera pribadi atau kemungkinan terjadinya cedera tersebut[9].

2.2.2 Klasifikasi Tingkat Keparahan Kecelakaan Kerja

Untuk menilai dan mengelola risiko secara efektif, klasifikasi tingkat keparahan kecelakaan sangat penting dilakukan. Berdasarkan pedoman dari *Occupational Safety and Health Administration* (OSHA) yang tercantum dalam *Field Operations Manual* (FOM) Chapter 6, yaitu dengan hanya mempertimbangkan dampak atau akibat terburuk dari kecelakaan (*severity*), tanpa mempertimbangkan probabilitas atau kemungkinan terjadinya kecelakaan tersebut. Probabilitas hanya digunakan dalam konteks penilaian risiko atau penentuan besaran penalti oleh OSHA, bukan dalam klasifikasi tingkat keparahan kecelakaan kerja[10]. Dengan demikian, pendekatan yang digunakan dalam penelitian ini sudah sesuai dengan standar OSHA dan praktik internasional dalam penilaian tingkat keparahan kecelakaan kerja.

Dalam konteks tingkat keparahan, OSHA membagi kecelakaan kerja ke dalam tiga kategori utama, yaitu:

1. *Low Severity* (Ringan)

   Kategori ini mencakup cedera atau penyakit ringan yang hanya membutuhkan perawatan medis sederhana.

2. *Medium Severity* (Sedang)

Cedera atau penyakit dalam kategori ini memerlukan perawatan medis lanjutan dan mungkin memerlukan rawat inap dalam jangka waktu tertentu, namun dengan kemungkinan pemulihan penuh tanpa menimbulkan kecacatan permanen.

3. *High Severity* (Berat)

Merupakan cedera atau penyakit serius yang dapat mengakibatkan kematian, cacat tetap, atau gangguan kesehatan kronis yang tidak dapat disembuhkan.

Dalam penelitian ini, klasifikasi tingkat keparahan kecelakaan kerja disederhanakan menjadi tiga kelas, yaitu ***mild*** (ringan), ***moderate*** (sedang), dan ***severe*** (berat). Klasifikasi ini disusun dengan mengacu pada kategori keparahan yang ditetapkan oleh OSHA, sehingga setiap kelas merepresentasikan tingkat dampak kecelakaan terhadap pekerja. Pendekatan ini digunakan sebagai dasar untuk pelatihan dan pengujian model klasifikasi menggunakan metode *Long Short-Term Memory* (LSTM), yang bertujuan untuk mengidentifikasi tingkat keparahan kecelakaan berdasarkan narasi atau deskripsi kejadian.

2.2.3 *Deep Learning*

Deep learning merupakan cabang dari machine learning (pembelajaran mesin) dan *artificial intelligence* (kecerdasan buatan) yang saat ini dianggap sebagai teknologi inti dari Revolusi Industri Keempat (Industry 4.0). Teknologi *deep learning* memanfaatkan banyak lapisan (*multiple layers*) untuk merepresentasikan abstraksi data dan membangun model komputasi yang kompleks. Dengan menggunakan beberapa lapisan ini, *deep learning* mampu mengekstraksi fitur-fitur penting secara otomatis dari data mentah, sehingga memungkinkan sistem untuk belajar dan membuat keputusan tanpa banyak intervensi manusia[11].

2.2.4 *Natural Language Processing*

Natural Language Processing (NLP) merupakan bidang interdisipliner dalam ilmu komputer, linguistik, dan kecerdasan buatan yang bertujuan untuk memungkinkan komputer memahami, menafsirkan, dan menghasilkan bahasa alami secara efektif. NLP mencakup berbagai proses yang memungkinkan mesin untuk memproses data linguistik, baik dalam bentuk teks maupun suara, dengan cara yang menyerupai pemahaman manusia.

Natural Language Processing (NLP) terbagi menjadi dua komponen utama, yaitu *Natural Language Understanding* (NLU) yang berkaitan dengan kemampuan mesin untuk memahami dan menafsirkan makna dari bahasa manusia, serta *Natural Language Generation* (NLG) yang fokus pada kemampuan mesin untuk menghasilkan teks atau ucapan dalam bahasa alami. Beberapa tugas utama dalam NLP mencakup tokenisasi, *part-of-speech tagging, named entity recognition* (NER), parsing, serta penerjemahan mesin (*machine translation*)[10].

Perkembangan teknologi, khususnya dalam bidang pembelajaran mesin (*machine learning*) dan pembelajaran mendalam (*deep learning*), telah mendorong kemajuan signifikan dalam kinerja sistem NLP. Pendekatan berbasis statistik telah menggantikan metode simbolik tradisional karena kemampuannya dalam menangani kompleksitas dan ambiguitas bahasa. Namun, tantangan seperti ketidakjelasan semantik, konteks pragmatis, serta representasi pengetahuan tetap menjadi fokus penelitian lanjutan dalam pengembangan sistem NLP yang lebih cerdas dan adaptif.

2.2.5 GloVe

GloVe (*Global Vectors for Word Representation*) merupakan salah satu metode representasi kata dalam bentuk vektor yang dikembangkan untuk kebutuhan pemrosesan bahasa alami atau *Natural Language Processing* (NLP). Model ini dikembangkan oleh Pennington et al. (2014) dengan pendekatan yang

menggabungkan keunggulan model berbasis statistik global dan model prediktif lokal seperti Word2Vec[12].

Berbeda dengan Word2Vec yang hanya memanfaatkan konteks lokal (*window* kecil dalam sebuah kalimat), GloVe secara eksplisit memanfaatkan informasi statistik global dari korpus melalui *matrix co-occurrence*. Inti dari pendekatan GloVe adalah ide bahwa rasio frekuensi kemunculan kata (co-occurrence) menyimpan informasi semantik penting antar kata[12].

Model GloVe dibangun dengan menyusun sebuah *matriks co-occurrence* X, di mana setiap elemen X_{ij} merepresentasikan seberapa sering kata j muncul dalam konteks kata i. Berdasarkan data ini, GloVe belajar untuk memetakan setiap kata menjadi sebuah vektor dalam ruang berdimensi tinggi yang mampu menangkap hubungan semantik[12].

Informasi semantik antara dua kata dapat ditangkap melalui rasio frekuensi kemunculan mereka dengan kata ketiga. Jika w_i dan w_j adalah dua kata target dan w_k adalah kata konteks, maka:

$$\frac{P_{ik}}{P_{jk}} = \frac{X_{ik} \,/\, X_i}{X_{jk} \,/\, X_j}$$

Rasio tersebut memberikan informasi tentang kedekatan semantik antara w_i dan w_j. Oleh karena itu, fungsi yang dipelajari harus bergantung pada rasio $X_{ik} \,/\, X_{ik}$. Untuk merepresentasikan kata ke dalam vektor w_i dan w_j, GloVe meminimalkan fungsi kerugian berbasis *mean squared error* (MSE) yang telah dimodifikasi sebagai berikut:

$$J = \sum_{i,j=1}^{V} f(X_{ij}) \left(w_i^T \tilde{w}_j + b_i + \tilde{b}_j - \log X_{ij}\right)^2$$

Dengan:

X_{ij} = jumlah co-occurrence antara kata i dan konteks j

w_i, $\tilde{w}_j$ = vektor kata dan vektor konteks

b_i , $\tilde{b}_j$= bias kata dan konteks

$f(X_{ij})$ = fungsi pembobotan untuk menghindari dominasi kata yang terlalu sering muncul

$$f(x) = \begin{cases} (\frac{x}{x_{\max}})^{\alpha} & jika\ x < x_{max} \\ 1 & jika\ x < x_{max} \end{cases}$$

Dengan α=0.75 sebagai nilai umum yang digunakan.

GloVe menggabungkan kekuatan statistik global, seperti LSA (*Latent Semantic Analysis*), dengan kemampuan pembelajaran representasi kata dari model neural seperti Word2Vec. Hal ini membuat GloVe lebih stabil dan mampu menangkap relasi semantik yang lebih kompleks, seperti analogi kata (misalnya, king – man + woman ≈ queen)[12].

2.2.6 *Recurrent Neural Network*

Recurrent Neural Networks (RNN) merupakan jenis jaringan saraf tiruan yang dirancang untuk mengolah data berurutan. RNN memiliki kemampuan untuk mempertahankan memori terhadap input sebelumnya melalui status internal (*hidden state*) yang terus diperbarui selama proses propagasi. Hal ini membuat RNN cocok untuk tugas seperti NLP, pengenalan suara (*speech recognition*), dan *time series forecasting*[13].

Namun, RNN memiliki kelemahan dalam menangani dependensi jangka panjang karena masalah *vanishing gradient*, sehingga sulit mempertahankan informasi pada urutan yang panjang.

2.2.7 *Long Short Term Memory*

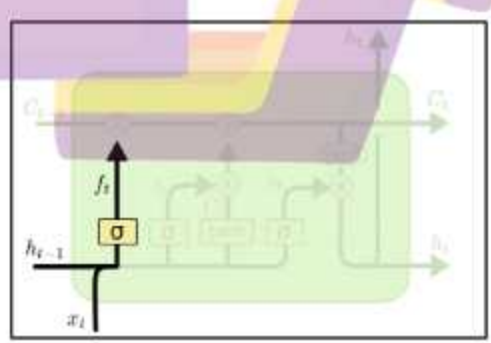
LSTM (*Long Short-Term Memory*) dikembangkan sebagai solusi untuk mengatasi masalah vanishing gradient pada RNN dengan menggunakan mekanisme memori khusus yang terdiri dari *forget gate, input gate*, dan *output gate*. Dengan arsitektur ini, LSTM dapat mempertahankan informasi penting dari data berurutan dalam jangka panjang[14].

LSTM terdiri dari unit-unit yang disebut *memory cell* yang bertanggung jawab untuk menyimpan informasi dari waktu ke waktu. Setiap *memory cell* memiliki tiga gerbang utama, yaitu *forget gate, input gate*, dan *output gate*, yang masing-masing berfungsi untuk mengatur aliran informasi di dalam jaringan[14].

2.2.7.1 Forget Gate

Forget gate berfungsi mengatur informasi mana dari cell state sebelumnya yang perlu dilupakan atau disimpan. Gerbang ini menghasilkan nilai antara 0 dan 1 melalui fungsi aktivasi sigmoid, yang bertindak sebagai bobot pengaruh untuk menentukan seberapa besar informasi tersebut disimpan (nilai mendekati 1) atau dilupakan (nilai mendekati 0)[14]. Persamaan matematisnya dituliskan sebagai:

$$f_t = \sigma\left(W_f . [h_{t-1}, x_t] + b_f\right.$$



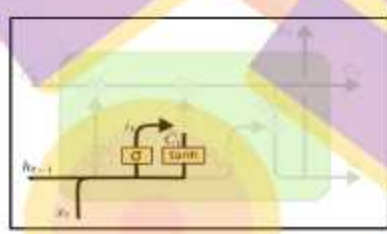
Gambar 2.1. Ilustrasi *Forget Gate*

2.2.7.2. *Input Gate*

Input gate mengontrol informasi baru yang akan diperbarui ke dalam cell state. Gerbang ini terdiri dari dua komponen: lapisan sigmoid yang menentukan bagian mana dari informasi yang akan diperbarui, dan lapisan tanh yang menghasilkan vektor kandidat nilai baru yang dapat ditambahkan ke *cell state*[14]. Persamaan dari *input gate* adalah:

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

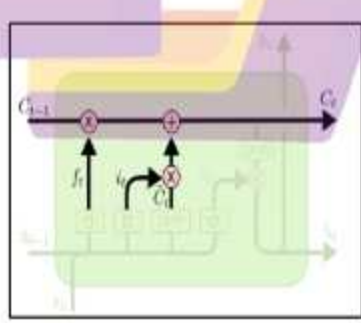
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t]$$



Gambar 2.2. Ilustrasi *Input Gate*

Cell state diperbarui dengan menggabungkan hasil dari forget gate dan input gate:

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$



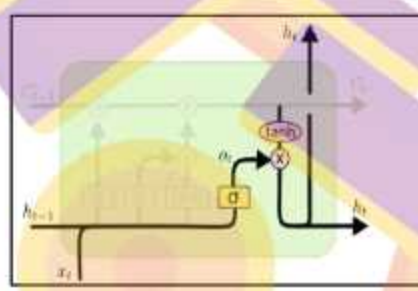
Gambar 2.3. Ilustrasi *Cell State Update*

2.2.7.3 *Output Gate*

Output gate menentukan bagian dari cell state yang akan dijadikan output pada waktu tertentu. Nilai output diperoleh dengan mengalikan hasil aktivasi sigmoid dari output gate dan hasil fungsi tanh dari cell state, untuk menghasilkan output yang terkontrol dan berskala[14].

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

$$h_t = o_t * \tanh\left(C_t\right)$$



Gambar 2.4. Ilustrasi *Output Gate*

Melalui struktur gerbang ini, LSTM memiliki kemampuan untuk mempertahankan informasi relevan dalam jangka waktu panjang serta menyaring informasi yang tidak diperlukan. Hal ini membuat LSTM sangat efektif untuk digunakan dalam berbagai tugas pemrosesan data sekuensial, seperti *speech recognition, language modeling, machine translation*, dan *time series forecasting*.

Dengan demikian, LSTM secara efektif mengatasi keterbatasan RNN tradisional dalam mempertahankan informasi jangka panjang, serta memberikan fleksibilitas dalam mengatur aliran informasi melalui mekanisme gate yang terstruktur, sehingga sangat cocok untuk berbagai tugas pemrosesan data sekuensia