

BAB II

TINJAUAN PUSTAKA

Bab ini membahas tentang pencapaian terdahulu sebagai referensi penelitian yang telah dilakukan dan tinjauan teoritis yang akan memperkuat teori dalam penelitian.

2.1 Pencapaian Terdahulu

Hasil riset yang telah dilakukan oleh peneliti lain akan dijadikan bahan referensi dalam penelitian ini. Riset yang dijadikan referensi adalah seputar DL dan kesehatan mental. Tabel 2.1 adalah tabel daftar penelitian terdahulu yang digunakan sebagai referensi atau acuan penelitian.

Tabel 2.1 Hasil Pencapaian Terdahulu

No.	Publikasi Jurnal	Judul Penelitian	Hasil Penelitian
1.	Ivan Dwi Nugraha & Yufis Azhar (2022)	Deteksi Depresi Pengguna Twitter Indonesia Menggunakan LSTM-RNN	Penelitian ini menghasilkan sebuah sistem yang mampu mengklasifikasikan kelasnya pada sistem klasifikasi depresi menggunakan LSTM-RNN dari tweet Twitter.
2.	Nila Zaimatus Septiana (2021)	Dampak Penggunaan Media Sosial Terhadap Kesehatan Mental dan Kesejahteraan Sosial Remaja Dimasa COVID-19	Penelitian ini menghasilkan kesimpulan bahwa media sosial yang digunakan secara bijak dapat meminimalkan dampak positif bagi seorang individu.
3.	Kuncahyo Setyo Nugroho, Ismail Akbar, Affi Nizar Suksmawati., & Istiadi (2021)	Deteksi Depresi dan Kecemasan Pengguna Twitter Menggunakan Bidirectional LSTM	Penelitian ini menghasilkan model yang memiliki akurasi mencapai 94.12% dengan menggunakan BiLSTM. Kelemahan dari BiLSTM adalah memerlukan sebuah dataset yang besar untuk membuatnya tidak <i>over-fitting</i> dan waktu komputasi yang lama.

4.	Fajarudin Zakariya, Junta Zeniarja & Sri Winarno (2024)	Pengembangan Chatbot Kesehatan Mental Menggunakan Algoritma Long Short-Term Memory	Penelitian ini menghasilkan sebuah sistem chat bot yang mampu memberikan informasi seputar kesehatan mental menggunakan arsitektur algoritma <i>Long Short-Term Memory</i> (LSTM). Model tersebut memiliki akurasi sebesar 93%, val_akurasi sebesar 83%, loss 0.3%. Penelitian ini terjadi <i>over-fitting</i> pada validasi loss yang diakibatkan oleh keterbatasan data.
5.	Alfan Syahada Hutagalung, Arif Bijaksana. Putra Negara, & Enda Esyudha Pratama (2021)	Aplikasi Pendeteksi <i>Cyberbullying</i> Terhadap Komentar Postingan Media Sosial Instagram dengan Metode <i>Naïve Bayes Classifier</i> Berbasis Website	Penelitian ini menghasilkan sebuah <i>website</i> yang dapat mendeteksi komentar <i>bullying</i> terhadap sebuah postingan di media sosial Instagram dengan penggunaan metode <i>naïve bayes classifier</i> .

Penelitian yang dilakukan (Nugraha & Azhar, 2022) adalah membuat model untuk deteksi depresi pengguna Twitter dengan kombinasi arsitektur LSTM dan RNN. Penelitian ini memiliki hasil akurasi model sebesar 86%. Kelebihan dari penelitian ini adalah hasilnya yang konsisten dengan akurasi, presisi, *recall*, dan *f1-score* sebesar 86%.

Penelitian yang dilakukan (Nugroho, Akbar, Suksmawati, & Istiada, 2021) adalah membuat model untuk deteksi depresi dan kecemasan dengan *Bidirectional LSTM* (BiLSTM). Hasil dari penelitian ini memperoleh akurasi model sebesar 94%. Penelitian tersebut juga menggunakan LSTM tradisional sebagai komparasi dan mendapatkan akurasi sebesar 84%. Kesimpulan dari penelitian tersebut adalah BiLSTM lebih unggul 10% dari LSTM tradisional.

Penelitian lainya dalam penggunaan LSTM adalah miliki (Zakariya, Zeniarja, & Winarno, 2024). Penelitian ini membahas pembuatan chatbot yang dapat menangani masalah kesehatan mental. Model yang dibuat dalam penelitian tersebut memperoleh akurasi sebesar 93% dengan epochs sebanyak 200. Kelebihan dari penelitian ini adalah sistem juga di-*deploy* untuk digunakan secara *real-time*.

Penelitian terakhir adalah milik (Hutagalung, Negara, & Pratama, 2021). Penelitian ini tidak menggunakan LSTM melainkan dengan metode *naïve bayes classifier* untuk deteksi *cyberbullying* terhadap komentar *postingan* komentar Instagram. Keunggulan dari penelitian ini adalah sistem dibuat dengan berbasis website, tidak hanya pembuatan klasifikasi saja. Akurasi dihitung dengan menggunakan *confusion matrix* dengan akurasi 98,5%.

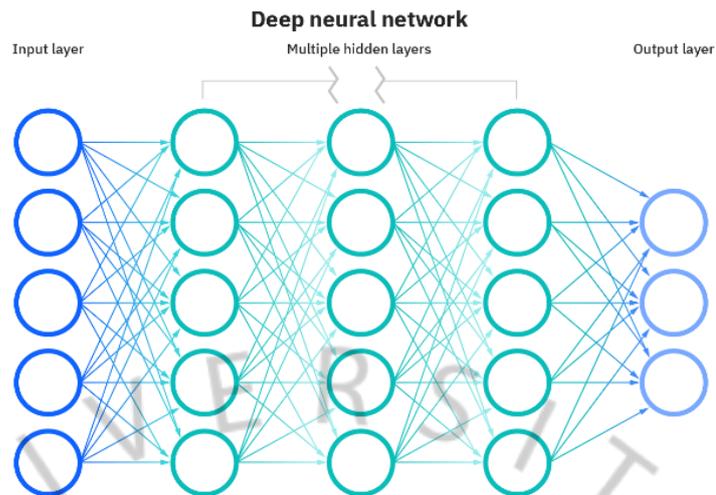
Jika dibandingkan dengan penelitian yang dilakukan oleh peneliti, peneliti memiliki keunggulan akurasi sebesar 99% hanya dengan menggunakan arsitektur LSTM, selain itu dataset yang dimiliki peneliti lebih banyak dibandingkan penelitian yang dilakukan peneliti lain, yaitu 15.478 dataset. Jumlah dataset yang banyak memiliki potensi belajar untuk model lebih baik dan memberikan akurasi lebih baik. Selain itu, peneliti juga melakukan *deployment* aplikasi agar dapat digunakan ke publik, sama seperti yang dilakukan oleh peneliti Zakariya, et al. dan Hutagalung, et al.

2.2 Tinjauan Teoritis

Tinjauan teoritis digunakan oleh peneliti sebagai dasar ilmu pengetahuan penelitian. Teori-teori yang digunakan peneliti dalam penelitian akan dijelaskan pada tinjauan teoritis.

2.2.1 Deep Learning (DL)

DL merupakan teknik kelas *machine learning* yang mengeksplorasi banyak lapisan pemrosesan informasi non-linier untuk mengekstrak dan transformasi fitur yang diawasi atau tidak diawasi, serta untuk analisis dan klasifikasi pola (Deng & Yu, 2014). DL mampu mempelajari suatu data yang kompleks dan besar secara hierarkis dan otomatis yang terinspirasi dari cara kerja otak manusia dengan membuat sebuah jaringan syaraf tiruan atau *neuron*.

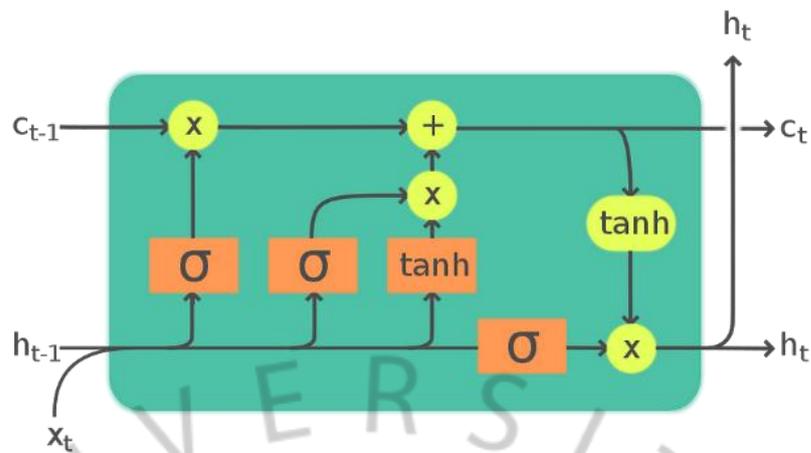


Gambar 2.1 Arsitektur *Deep Learning* (Sumber: www.dqlab.id)

Gambar 2.1 merupakan ilustrasi arsitektur DL yang saling terhubung. Lapisan yang saling terhubung ini disebut juga dengan *hidden layer*. *Hidden layer* mengubah input menjadi sesuatu yang dapat digunakan oleh lapisan luar. Proses ini memungkinkan jaringan mempelajari hubungan non-linear antara data input dan output.

2.2.2 Arsitektur *Long Short-Term Memory* (LSTM)

LSTM merupakan sebuah arsitektur jaringan syaraf tiruan yang terkategoriikan sebagai *Recurrent Neural Network* (RNN) (Zakariya, Zeniarja, & Winarno, 2024). Arsitektur LSTM menyempurnakan kekurangan RNN dalam memprediksi kata berdasarkan informasi lampau yang disimpan jangka waktu panjang. Arsitektur LSTM terbentuk oleh sekelompok subnet yang terkoneksi secara berulang, dikenal sebagai blok memori. Blok-blok ini dapat diibaratkan sebagai chip memori yang dapat dibedakan dalam komputer digital. Setiap blok terdiri dari satu atau lebih sel memori yang terkoneksi secara independen dan tiga unit perkalian input (Graves, 2012).



Gambar 2.2 Arsitektur LSTM (Sumber: www.medium.com)

Gambar 2.2 merupakan arsitektur dari LSTM. LSTM terdiri dari 3 gerbang untuk pemrosesan informasi, gerbang tersebut adalah gerbang lupa, gerbang masukan, dan gerbang luaran. Gerbang lupa merupakan gerbang yang menentukan jumlah informasi yang dihapus dari sel memori. Gerbang masukan merupakan gerbang jumlah informasi baru yang perlu disimpan didalam sel memori. Dan yang terakhir adalah gerbang luaran, gerbang ini berfungsi untuk menentukan jumlah informasi yang akan dikeluarkan sebagai output sel memori (Azrul, Purnamasari, & Ali, 2024). Secara rinci, LSTM merupakan kemajuan dalam algoritma RNN, yang mampu menyimpan informasi untuk jangka waktu yang lebih lama (Lombu, Papatungan, & Dewa, 2024).

2.2.3 Case Folding

Case Folding merupakan proses mengubah teks yang didapatkan dari dataset menjadi huruf kecil/*lowercase* dan pembersihan terhadap teks berupa nomor, url, html tag, dan tanda baca. Tahap ini dilakukan sehingga dapat memudahkan proses analisis dan pengolahan data teks (Bramantyo, Putra, & Hendrastuty, 2023). Penelitian ini menggunakan fungsi *replace* pada *Python* dan *library Regular Expressions* (ReGex). Contoh proses *case folding* dapat dilihat pada Tabel 2.2.

Tabel 2.2 Proses *Case Folding* Pada Dataset

Sebelum <i>Case Folding</i>	Sesudah <i>Case Folding</i>
More about eating disorders please! Recently I have been addicted to looking at my weight and	more about eating disorders please! recently i have been addicted to looking at my weight and

trying to not eat for days on end just to not eat but then eating too much all at once and then feeling guilty about it. Is that an eating disorder??	trying to not eat for days on end just to not eat but then eating too much all at once and then feeling guilty about it. is that an eating disorder??
---	---

2.2.4 Tokenisasi

Tokenisasi adalah proses penguraian kalimat menjadi sebuah potongan kata-kata yang dilakukan dengan memotong kata pada white space atau spasi (Pipin & Kurniawan, 2022). Proses ini penting untuk mempersiapkan teks agar lebih mudah diproses oleh mesin. Penelitian ini menggunakan *library* ReGex untuk proses tokenisasi. Contoh proses tokenisasi dapat dilihat pada Tabel 2.3.

Tabel 2.3 Proses Tokenisasi Pada Dataset

Sebelum Tokenisasi	Sesudah Tokenisasi
more about eating disorders please! recently i have been addicted to looking at my weight and trying to not eat for days on end just to not eat but then eating too much all at once and then feeling guilty about it. is that an eating disorder??	['more', 'about', 'eating', 'disorders', 'please', 'recently', 'i', 'have', 'been', 'addicted', 'to', 'looking', 'at', 'my', 'weight', 'and', 'trying', 'to', 'not', 'eat', 'for', 'days', 'on', 'end', 'just', 'to', 'not', 'eat', 'but', 'then', 'eating', 'too', 'much', 'all', 'at', 'once', 'and', 'then', 'feeling', 'guilty', 'about', 'it', 'is', 'that', 'an', 'eating', 'disorder']

2.2.5 Stop Words Removal

Stop Words Removal dilakukan dengan tujuan untuk menghilangkan kata yang umum digunakan dan memiliki sedikit bobot dalam maknanya (Sarica & Luo, 2021). Kata-kata tersebut sering dihilangkan dalam proses analisis teks. Penelitian ini menggunakan *library natural language toolkit* (NLTK) pada *Python* untuk proses *stop words removal*. Contoh proses *stop words removal* dapat dilihat pada Tabel 2.4.

Tabel 2.4 Proses *Stop Words Removal* Pada Dataset

Sebelum <i>Stop Words Removal</i>	Sesudah <i>Stop Words Removal</i>
['more', 'about', 'eating', 'disorders', 'please', 'recently', 'i', 'have', 'been', 'addicted', 'to', 'looking', 'at', 'my', 'weight', 'and', 'trying', 'to', 'not', 'eat', 'for', 'days', 'on', 'end', 'just', 'to', 'not', 'eat', 'but', 'then', 'eating', 'too', 'much', 'all', 'at',	['eating', 'disorders', 'please', 'recently', 'addicted', 'looking', 'weight', 'trying', 'eat', 'days', 'end', 'eat', 'eating', 'much', 'feeling', 'guilty', 'eating', 'disorder']

'once', 'and', 'then', 'feeling', 'guilty', 'about', 'it', 'is', 'that', 'an', 'eating', 'disorder']	
--	--

2.2.6 Lematisasi

Lematisasi adalah proses mengekstrakan kata dasar dari sebuah kata dengan pendekatan menggunakan kosakata/kamus data dan analisis morfologi kata. (Arimbawa & ER, 2020). Lematisasi bertujuan untuk mendapatkan kata-kata yang valid dari kata dasar. Penelitian ini menggunakan *library SpaCy* pada *Python* untuk proses lematisasi pada sebuah kata. Contoh proses lematisasi dapat dilihat pada Tabel 2.5.

Tabel 2.5 Proses Lematisasi Pada Dataset

Sebelum Lematisasi	Sesudah Lematisasi
['eating', 'disorders', 'please', 'recently', 'addicted', 'looking', 'weight', 'trying', 'eat', 'days', 'end', 'eat', 'eating', 'much', 'feeling', 'guilty', 'eating', 'disorder']	['eat', 'disorder', 'please', 'recently', 'addicted', 'look', 'weight', 'try', 'eat', 'day', 'end', 'eat', 'eat', 'much', 'feel', 'guilty', 'eat', 'disorder']

2.2.7 Website

Website merupakan platform yang umum digunakan dalam pembuatan aplikasi. *Website* umumnya berisikan informasi seperti teks, gambar, video, atau sebuah *link* yang disesuaikan kontennya sesuai dengan kebutuhan informasi.

2.2.8 Python

Python merupakan bahasa pemrograman yang sering digunakan untuk pengembangan *machine learning* dan keperluan *data scientist* untuk analisis data. *Python* memiliki sintaks yang mudah untuk dipahami dan mudah digunakan. Selain itu, *Python* juga memiliki banyak *library* yang memudahkan dalam proses pembuatan model *machine learning* maupun visualisasi data.

2.2.9 Streamlit Framework

Streamlit adalah sebuah kerangka kerja *open-source* gratis untuk membangun sebuah aplikasi website *machine learning*. *Streamlit* merupakan *library* bahasa pemrograman *Python* yang khusus untuk *machine learning engineer*. Penggunaan *streamlit* cukup mudah sehingga sering digunakan *data*

scientist dan *machine learning engineer* untuk menampilkan data atau mengumpulkan parameter yang diperlukan untuk pemodelan.

2.2.10 *Unified Modeling Language (UML)*

UML merupakan bahasa yang sering digunakan untuk analisis bisnis dan pengembangan perangkat lunak. Penggunaan UML seperti menggambarkan, mendeskripsikan, merancang, menentukan, dan mendokumentasikan proses alur bisnis, struktur dan perilaku artefak dari sistem perangkat lunak yang akan dibuat (Unified Modeling Language, 2009). UML dapat memudahkan proses analisa sebuah sistem sebelum masuk ke tahap pembuatan. Diagram yang sudah dibuat akan digunakan dalam tahap pembuatan sistem dan memberikan gambaran alur dari sebuah sistem berjalan.

2.2.11 *Confusion Matrix*

● Proses pengujian model dengan menggunakan *confusion matrix* dilakukan dengan tujuan untuk mengukur kinerja prediksi model. *Confusion matrix* memberikan gambaran kombinasi hasil nilai sebenarnya (*actual*) dan nilai yang diprediksi (*prediction*). *Confusion matrix* sering digunakan pada masalah klasifikasi yang terdiri dari dua atau lebih kelas (Putra, Suprpto, & Bukhori, 2022). Terdapat empat atribut yang digunakan untuk dasar perhitungan, yaitu:

1. *True Positive (TP)*: Jumlah data yang bernilai positif baik pada kategori yang diprediksi maupun kategori yang sebenarnya.
2. *False Positive (FP)*: Jumlah data yang bernilai positif pada kategori yang diprediksi namun bernilai negatif pada kategori yang sebenarnya.
3. *True Negative (TN)*: Jumlah data yang bernilai negatif baik pada kategori yang diprediksi maupun kategori yang sebenarnya.
4. *False Negative (FN)*: Jumlah data yang bernilai negatif pada kategori yang diprediksi namun bernilai positif pada kategori yang sebenarnya.

Perhitungan *accuracy*, *precision*, *recall*, dan *F1-score* dari empat atribut *confusion matrix* dijelaskan sebagai berikut:

1. *Accuracy*

Accuracy merupakan jumlah titik data yang diprediksi benar dari semua titik data yang ada (Nugraha & Azhar, 2022). Prediksi benar (positif dan negatif) dihitung benar dalam pengklasifikasiannya akan digunakan untuk mengestimasi akurasi dari prediksi. Nilai *accuracy* dapat diperoleh dengan persamaan berikut:

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

2. *Precision*

Precision adalah seberapa sering model memprediksi kelas positif dengan benar (Nugraha & Azhar, 2022). Rasio antara TP dengan keseluruhan data yang diprediksi positif yang bertujuan untuk meminimalisir terjadinya FP. Nilai *precision* dapat diperoleh dengan persamaan berikut:

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

3. *Recall*

Recall merupakan perhitungan probabilitas label positif berlabel positif (Nugraha & Azhar, 2022). Rasio antara TP dengan keseluruhan data yang sebenarnya bernilai positif bertujuan meminimalisir terjadi FN. Nilai *recall* dapat diperoleh dengan persamaan berikut:

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

4. *F1-score*

F1-score / *F-Measure* memberikan skor tunggal yang menyeimbangkan perhatian terhadap presisi dan daya ingat dalam satu angka (Nugraha & Azhar, 2022). Nilai *F1-score* dapat diperoleh dengan persamaan berikut:

$$F1 \text{ score} = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \quad (4)$$

