KLASIFIKASI GEN ESENSIAL PADA SEKUENS DNA LALAT BUAH (*DROSOPHILA MELANOGASTER*) MENGGUNAKAN METODE *BIDIRECTIONAL LONG SHORT TERM MEMORY* (BiLSTM)

(Skripsi)

Oleh

PUTRI SANTIKA MAYANGSARI

NPM 2057051011

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM

UNIVERSITAS LAMPUNG

BANDAR LAMPUNG

2024

ABSTRAK

KLASIFIKASI GEN ESENSIAL PADA SEKUENS DNA LALAT BUAH (*DROSOPHILA MELANOGASTER*) MENGGUNAKAN METODE *BIDIRECTIONAL LONG SHORT TERM MEMORY (*BiLSTM*)*

Oleh

PUTRI SANTIKA MAYANGSARI

DNA berfungsi sebagai pembawa informasi genetik yang mengatur berbagai proses kehidupan, seperti pertumbuhan, perkembangan, dan kelangsungan hidup organisme, yang sangat penting untuk memahami fungsi biologis. Untuk mempelajari mekanisme tersebut, klasifikasi gen esensial, yang berperan penting dalam fungsi dasar seluler, sangat penting. Penelitian ini menggunakan metode *Bidirectional Long Short-Term* Memory (BiLSTM) untuk membuat model klasifikasi gen esensial pada sekuens DNA lalat buah. Pola sekuensial gen esensial dan non-esensial diidentifikasi melalui pengolahan dataset sekuens DNA dari *Drosophila melanogaster*. Untuk mencegah *overfitting*, metode regularisasi seperti *early stopping*, *dropout*, dan L2 *regularization* digunakan. Penggunaan *Random Undersampling* bertujuan untuk menyeimbangkan jumlah data tiap kelas dengan mengurangi sampel dari kelas yang lebih besar. Evaluasi model dilakukan menggunakan *Confusion Matrix* dengan metrik evaluasi yang digunakan yaitu *Sensitivity*, *Specificity*, ROC-AUC, dan PR-AUC. Hasil klasifikasi DNA *Drosophila melanogaster* pada data OEG mendapatkan hasil *Specificity* sebesar 73%, *Sensitivity* sebesar 80%, ROC-AUC sebesar 76%, dan PR-AUC sebesar 81%. Sedangkan, data CEG mendapatkan hasil Specificity sebesar 70%, Sensitivity sebesar 64%, ROC-AUC sebesar 67%, dan PR-AUC sebesar 46%. Pendekatan BiLSTM yang diusulkan membantu dalam pengembangan metode klasifikasi gen esensial.

Kata Kunci: Klasikasi, *Drosophila melanogaster*, Sekuens DNA, BiLSTM, Gen Esensial.

ABSTRACT

KLASIFIKASI GEN ESENSIAL PADA SEKUENS DNA LALAT BUAH (*DROSOPHILA MELANOGASTER*) MENGGUNAKAN METODE *BIDIRECTIONAL LONG SHORT TERM MEMORY (*BiLSTM*)*

By

PUTRI SANTIKA MAYANGSARI

DNA functions as the carrier of genetic information that regulates various life processes, such as growth, development, and the survival of organisms, which is crucial for understanding biological functions. To study these mechanisms, the classification of essential genes, which play a vital role in basic cellular functions, is important. This research employs the Bidirectional Long Short-Term Memory (BiLSTM) method to develop an essential genes classification model on the DNA sequences of fruit flies. The sequential patterns of essential and non-essential genes are identified through the processing of DNA sequence datasets from Drosophila melanogaster. To prevent overfitting, regularization methods such as early stopping, dropout, and L2 regularization are used. Random undersampling is employed to balance the number of samples in each class by reducing the samples from the larger class. Model evaluation is performed using a Confusion Matrix, with evaluation metrics including Sensitivity, Specificity, ROC-AUC, and PR-AUC. The classification results for Drosophila melanogaster DNA on the OEG dataset yielded a Specificity of 73%, Sensitivity of 80%, ROC-AUC of 76%, and PR-AUC of 81%. Meanwhile, the CEG dataset resulted in a Specificity of 70%, Sensitivity of 64%, ROC-AUC of 67%, and PR-AUC of 46%. The proposed BiLSTM approach aids in the development of essential gene classification methods.

Keyword: Classification, Drosophila melanogaster, DNA Sequences, BiLSTM, Essential Genes.

KLASIFIKASI GEN ESENSIAL PADA SEKUENS DNA LALAT BUAH (*DROSOPHILA MELANOGASTER*) MENGGUNAKAN METODE *BIDIRECTIONAL LONG SHORT TERM MEMORY* (BiLSTM)

Oleh

PUTRI SANTIKA MAYANGSARI

Skripsi

Sebagai Salah Satu Syarat untuk Mencapai Gelar
SARJANA ILMU KOMPUTER

Pada

Jurusan Ilmu Komputer
Fakultas Matematika dan Ilmu Pengetahuan Alam

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2024

Judul Skripsi	: **Klasifikasi Gen Esensial Pada Sekuens DNA Lalat Buah (*Drosophila Melanogaster*) menggunakan metode *Bidirectional Long Short Term Memory* (BiLSTM)**
Nama Mahasiswa	: Putri Santika Mayangsari
Nomor Pokok Mahasiswa	: 2057051011
Program Studi	: Ilmu Komputer
Fakultas	: Matematika dan Ilmu Pengetahuan Alam

MENYETUJUI

Ketua Jurusan Ilmu Komputer	Pembimbing Utama
Dwi Sakethi, M.Kom.	**Favorisen Rosyking Lumbanraja, S.Kom., M.Si., Ph.D**
NIP. 196806111998021001	NIP. 198301102008121002

MENGESAHKAN

1. Tim Penguji

   Ketua : **Favorisen Rosyking Lumbanraja, S.Kom., M.Si., Ph.D**

   Pembahas Pertama : **Fatma Indriani, S.T., MIT, Ph.D.**

   Pembahas Kedua : **Dr. rer. nat. Akmal Junaidi, M.Sc.**

2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam

Dr. Eng. Heri Satria, S.Si., M.Si.
NIP. 197110012005011002

Tanggal Lulus Ujian Skripsi: **1 Oktober 2024**

PERNYATAAN

Saya yang bertanda tangan di bawah ini, menyatakan bahwa skripsi saya dengan judul **"Klasifikasi Gen Esensial Pada Sekuens DNA Lalat Buah (*Drosophila Melanogaster*) menggunakan metode *Bidirectional Long Short Term Memory* (BiLSTM)"** merupakan karya saya sendiri dan bukan karya orang lain. Semua tulisan yang tertuang dalam skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Apabila di kemudian hari terbukti skripsi saya merupakan hasil penjiplakan atau dibuat orang lain, maka saya bersedia menerima sanksi berupa pencabutan gelar yang telah saya terima.

Bandar Lampung, 1 Oktober 2024

Putri Santika Mayangsari

NPM. 2057051011

RIWAYAT HIDUP

Lahir pada Hari Sabtu, 26 Januari 2002. Anak pertama dari Bapak Darso dan Ibu Siti Khodijah, telah menyelesaikan pendidikan dasar pada tahun 2014 di SDN 04 Karanggan. Kemudian menyelesaikan pendidikan menengah pada tahun 2017 di SMPN 02 Gunung Putri, dan lulus dari pendidikan menengah atas pada tahun 2020 di SMA Plus PGRI Cibinong. Pada tahun 2020, terdaftar sebagai mahasiswi Jurusan Ilmu Komputer Universitas Lampung melalui jalur SMMPTN. Terdapat beberapa kegiatan yang dilakukan selama menjadi mahasiswa yaitu sebagai berikut.

1. Menjadi asisten dosen mata kuliah Bioinformatika pada bulan Februari hingga bulan Juni Tahun 2024.
2. Mengikuti *Student Mobility* ke Universiti Malaya di Kuala Lumpur, Malaysia pada Bulan September Tahun 2023.
3. Melaksanakan Kerja Praktik pada Bulan Juni 2023 di Menara Mandiri, Jakarta Selatan.
4. Mengikuti Kuliah Kerja Nyata 2023 periode 1 di Desa Bandar Agung, Kecamatan Bandar Negeri Suoh, Kabupaten Lampung Barat.
5. Menjadi panitia dalam Rangka Pekan Raya Jurusan Ilmu Komputer ke10 dan Himakom *Tournament* pada Bulan Oktober Tahun 2022.
6. Mengikuti Gemastik (Pagelaran Mahasiswa Nasional Bidang Teknologi Informasi dan Komunikasi) Divisi UX Design Tahun 2021
7. Mengikuti Kursus Desain Interaksi Untuk UI/UX Designer Pemula – Program Kredensial Mikro Mahasiswa Indonesia (KMMI) pada Tahun 2021.

MOTTO

"Nothing in life is to be feared, it is only to be understood. Now is the time to understand more, so that we may fear less."

(Marie Curie)

"Success is not final, failure is not fatal: It is the courage to continue that counts."

(Winston Churchill)

PERSEMBAHAN

Alhamdulillahirabbil 'allamiin, segala puji dan syukur ke hadirat Allah SWT atas segala rahmat, ridho, serta karunia-Nya sehingga skripsi ini dapat diselesaikan dengan baik dan lancar dalam mendapatkan gelar sarjana komputer. Maka dari itu, dengan rasa syukur dan bahagia saya persembahkan skripsi ini kepada:

Nabi Muhammad SAW

Terima kasih atas cahaya petunjukmu yang menerangi hidupku. Engkau adalah teladan cinta dan kebijaksanaan. Semoga setiap langkahku sejalan dengan ajaranmu yang penuh rahmat.

Kedua Orang Tua ku Tersayang

Tulus Terima Kasih, Bapak dan Ibu. Kepada Bapak, pahlawan tanpa tanda jasa, dan Ibu, sinar dalam gelap. Persembahan ini adalah ungkapan kecil rasa syukur dan cinta tak terhingga. Kalian adalah fondasi kebahagiaan dan keberhasilan dalam hidupku. Terima kasih untuk kasih sayang, bimbingan, dan dukungan tak terbatas.

Seluruh Keluarga Besar Ilmu Komputer 2020

Jurusan Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Lampung

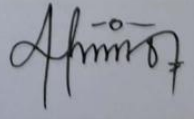
SANWACANA

Puji syukur ke hadirat Tuhan Yang Maha Esa atas rahmat dan berkat-Nya, penulis dapat menyelesaikan skripsi yang berjudul **"Klasifikasi Gen Esensial Pada Sekuens DNA Lalat Buah (*Drosophila Melanogaster*) menggunakan metode *Bidirectional Long Short Term Memory* (BiLSTM)"** Dalam melaksanakan penelitian dan pembuatan skripsi, penulis banyak mendapat bimbingan dan dukungan dari berbagai pihak, sehingga pada kesempatan ini penulis ingin menyampaikan ungkapan terima kasih kepada:

1. Allah *Subhaanahu wata 'aalaa* yang menjadi sumber kekuatan, sukacita, dan pengharapan yang selalu memberikan karunia dan rahmat-Nya selama penulis menyelesaikan skripsi.
2. Kedua orang tua, Bapak Darso dan Ibu Siti Khodijah, saudara kandung, Bagus Arya Dwi Pangga dan Raihana Yasmina Faiha yang telah memberikan dukungan dan motivasi kepada penulis untuk menyelesaikan skripsi dengan baik.
3. Bapak Dr. Eng. Heri Satria, S.Si., M.Si. selaku Dekan FMIPA Universitas Lampung.
4. Bapak Dwi Sakethi, M.Kom. selaku Ketua Jurusan Ilmu Komputer Universitas Lampung.
5. Ibu Anie Rose Irawati, ST., M.Cs. selaku Sekretaris Jurusan Ilmu Komputer Universitas Lampung
6. Ibu Yunda Heningtyas, M.Kom. selaku Dosen Pembimbing Akademik.
7. Bapak Favorisen Rosyking Lumbanraja S.Kom., M.Si., Ph.D. selaku Dosen Pembimbing Utama yang telah membimbing, memberi masukan, serta mendukung dalam proses pembuatan skripsi ini, sehingga penulis dapat menyelesaikan skripsi dengan baik.

8. Ibu Fatma Indriani, S.T., MIT, Ph.D. selaku Dosen Pembahas Pertama yang telah memberikan masukan dalam penelitian skripsi ini.
9. Bapak Dr. rer. nat. Akmal Junaidi, M.Sc. selaku Dosen Pembahas Kedua yang telah memberikan masukan dalam penelitian skripsi ini.
10. Bapak dan Ibu Dosen Jurusan Ilmu Komputer FMIPA Universitas Lampung yang telah memberikan ilmu serta pengalaman semasa perkuliahan.
11. Yulia, kak Zahra, dan kak Fajar, yang senantiasa memberikan arahan, dukungan, dan motivasi kepada saya untuk menyusun juga menyelesaikan skripsi hingga sampai saat ini.
12. Fitriah, Nabila, Yoan, Irma, Safira, Silvia, Nafasya, dan Nuk, yang senantiasa membersamai saya selama kegiatan belajar dikelas maupun kegiatan berprogres menyelesaikan skripsi.
13. Ayu, Hasna, Dila, Dewi, yang senantiasa mendukung secara emosional kepada saya disaat masa-masa sedih dan *overthinking*.
14. Kevin Lius Bong, salah satu peserta Clash Of Champions by Ruang Guru yang saya kagumi dan menjadi salah satu motivator saya untuk terus belajar dan pantang menyerah atas kegagalan.
15. Dan tidak lupa saya ucapkan terima kasih kepada diri saya sendiri yang telah berjuang, tidak berhenti berusaha, serta pantang menyerah hingga saat ini.

Bandar Lampung, 01 Oktober 2024



Putri Santika Mayangsari
NPM. 2057051011

DAFTAR ISI

Halaman

DAFTAR ISI ..... **xi**

DAFTAR GAMBAR ..... **xv**

DAFTAR KODE PROGRAM ..... **xvii**

DAFTAR TABEL ..... **xix**

I. PENDAHULUAN **1**

1.1 Latar Belakang 1

1.2 Rumusan Masalah 3

1.3 Batasan Masalah 3

1.4 Tujuan 4

1.5 Manfaat 4

II. TINJAUAN PUSTAKA **5**

2.1 Penelitian terdahulu 5

2.1.1 *Essential gene prediction in Drosophila melanogaster using* 8

machine learning approaches based on sequence and functional features..... 8

2.1.2 *Identifying essential genes across eukaryotes by machine* 8

Learning..... 8

2.1.3 *Performance evaluation of features for gene essentiality*..... 9

Prediction..... 9

2.2 Uraian tentang landasan teori 10
2.2.1 *Drosophila Melanogaster* (Lalat Buah) 10
2.2.2 *Essential Genes* 11
2.2.3 *Bidirectional Long Short Term Memory* (BiLSTM) 11
2.2.4 OEG *(Organismal Essential Genes)* 20
2.2.5 CEG *(Cellular Essential Genes)* 20
2.2.6 CRISPR 21
2.2.7 RNA interference (RNAi) 21
2.2.8 *Embedding Layer* 22
2.2.9 *Flatten Layer* 23
2.2.10 *Fully Connected Layer* 23
2.2.11 *Random Undersampling* 24
2.2.12 *K-mer Tokenization* 25
2.2.13 *Padding* 26
2.2.14 *Overfitting* 26
2.2.15 *Underfitting* 27
2.2.16 *Dropout* 27
2.2.17 *Performance Metrics* 28
III. METODOLOGI PENELITIAN **32**
3.1 Waktu dan Tempat 32
3.1.1 Tempat Penelitian 32
3.1.2 Waktu Penelitian 32

3.2 Data dan Alat 34
3.2.1 Data 34
3.2.2 Perangkat Keras (*Hardware*) 34
3.2.3 Perangkat Lunak (*Software*) 35
3.3 Metodologi Penelitian 38
3.3.1 Pengumpulan data 39
3.3.2 Prapemrosesan Data 39
3.3.3 Model Klasifikasi 40
3.3.4 Evaluasi Kinerja Metrik 41
IV. HASIL DAN PEMBAHASAN **42**
4.1 Pengumpulan Dataset 42
4.2 Prapemrosesan Data 44
4.2.1 Pembersihan data 44
4.2.2 Penggabungan Sekuens DNA dan Label Kelas DNA 46
4.2.3 Tokenisasi 50
4.2.4 *Padding* 52
4.3 Pembagian Data 54
4.3.1 Pembagian data OEG 55
4.3.2 Pembagian data CEG 57
4.4 Random Undersampling 58
4.5 Model *Bidirectional Long Short Term Memory* (BiLSTM) 61
4.6 Pelatihan model klasifikasi *Bidirectional Long Short Term Memory* (BiLSTM) 66

4.6.1 Pelatihan dataset OEG..... 66

4.6.2 Pelatihan dataset CEG..... 74

4.7 Pengujian model klasifikasi *Bidirectional Long Short Term Memory* (BiLSTM) 83

4.7.1 Pengujian dataset OEG 87

4.7.2 Pengujian dataset CEG..... 95

4.8 Pembahasan 103

4.8.1 Hasil Data CEG..... 104

4.8.2 Hasil Data OEG..... 108

4.8.3 Hasil Perbandingan Metode Terdahulu..... 112

4.8.4 Kesimpulan Hasil Metode BiLSTM 113

V. SIMPULAN DAN SARAN..... 118

5.1 SIMPULAN..... 118

5.2 SARAN 119

DAFTAR PUSTAKA..... 120

DAFTAR GAMBAR

Gambar Halaman

1. Arsitektur LSTM (Aldhyani & Alkahtani, 2021)..... 12
2. Struktur BiLSTM (Alharbi & Csala, 2021). 14
3. Fully Connected Layer (Cheng et al., 2023). 24
4. K-mer Tokenization (Gunasekaran et al., 2021). 25
5. Zero Padding (Lopez-del Rio et al., 2020)..... 26
6. Diagram Penelitian..... 39
7. Sequence DNA..... 44
8. Arsitektur BiLSTM Untuk Dataset OEG dan CEG 64
9. Grafik OEG Pre Padding k-mer 5 90:10 67
10. Grafik OEG Post Padding k-mer 5 90:10..... 68
11. Grafik OEG Pre Padding k-mer 7 90:10. 69
12. Grafik OEG Post Padding k-mer 7 90:10..... 70
13. Grafik OEG Pre Padding k-mer 5 80:20. 71
14. Grafik OEG Post Padding k-mer 5 80:20..... 72
15. Grafik OEG Pre Padding k-mer 7 80:20. 73
16. Grafik OEG Post Padding k-mer 7 80:20..... 74
17. Grafik CEG Pre Padding k-mer 5 90:10. 76
18. Grafik CEG Post Padding k-mer 5 90:10..... 76
19. Grafik CEG Pre Padding k-mer 7 90:10. 78
20. Grafik CEG Post Padding k-mer 7 90:10..... 78
21. Grafik CEG Pre Padding k-mer 5 80:20. 80
22. Grafik CEG Post Padding k-mer 5 80:20..... 80
23. Grafik CEG Pre Padding k-mer 7 80:20. 82

24. Grafik CEG Post Padding k-mer 7 80:20. 82
25. Confusion Matrix OEG Pre Padding k-mer 5 90:10. 88
26. Confusion Matrix OEG Post Padding k-mer 5 90:10. 89
27. Confusion Matrix OEG Pre Padding k-mer 7 90:10. 90
28. Confusion Matrix OEG Post Padding k-mer 7 90:10. 91
29. Confusion Matrix OEG Pre Padding k-mer 5 80:20. 92
30. Confusion Matrix OEG Post Padding k-mer 5 80:20. 93
31. Confusion Matrix OEG Pre Padding k-mer 7 80:20. 94
32. Confusion Matrix OEG Post Padding k-mer 7 80:20. 95
33. Confusion Matrix CEG Pre Padding k-mer 5 90:10. 96
34. Confusion Matrix CEG Post Padding k-mer 5 90:10. 97
35. Confusion Matrix CEG Pre Padding k-mer 7 90:10. 98
36. Confusion Matrix CEG Post Padding k-mer 7 90:10. 99
37. Confusion Matrix CEG Pre Padding k-mer 5 80:20. 100
38. Confusion Matrix CEG Post Padding k-mer 5 80:20. 101
39. Confusion Matrix CEG Pre Padding k-mer 7 80:20. 102
40. Confusion Matrix CEG Post Padding k-mer 7 80:20. 103
41. Grafik Perbandingan Pengujian 90:10 k-mer 5 CEG 104
42. Grafik Perbandingan Pengujian 90:10 k-mer 7 CEG 105
43. Grafik Perbandingan Pengujian 80:20 k-mer 5 CEG 106
44. Grafik Perbandingan Pengujian 80:20 k-mer 7 CEG 107
45. Grafik Perbandingan Pengujian 90:10 k-mer 5 OEG 108
46. Grafik Perbandingan Pengujian 90:10 k-mer 7 OEG 109
47. Grafik Perbandingan Pengujian 80:20 k-mer 5 OEG 110
48. Grafik Perbandingan Pengujian 80:20 k-mer 7 OEG 111
49. Hasil Data OEG Menggunakan BiLSTM 113
50. Hasil Data CEG Menggunakan BiLSTM 115

DAFTAR KODE PROGRAM

Kode Program Halaman

1. Membaca File 45
2. Menghapus Missing Value 45
3. Penggabungan Sekuens DNA dan Label Kelas DNA 46
4. Label Biner CEG 48
5. Label Biner OEG 49
6. k-mer Dictionary 51
7. k-mer Tokenization. 51
8. Pre Padding 52
9. Post Padding 53
10. Train Test Data OEG 55
11. Simpan data train dan data test OEG 55
12. Train 90% dan test 10% OEG 56
13. Train 80% dan Test 20% OEG 56
14. Train Test Data CEG 57
15. Simpan data train dan data test CEG 57
16. Training 90% dan Testing 10% CEG 57
17. Training 80% dan Testing 20% CEG 58
18. Distribusi Kelas 59
19. Random Undersampling 60
20. Seed Value 61
21. BiLSTM 62
22. Early Stopping 63
23. Learning Rate 63

24. Compile Model..... 65
25. Training and Validation Data..... 65
26. Sensitifity dan Specificity 83
27. ROC AUC. 84
28. PR AUC. 86

DAFTAR TABEL

Tabel Halaman

1. Penelitian Terdahulu Terkait Gen Esensial 5
2. Notasi LSTM (Smagulova & James, 2020). 19
3. Confusion Matrix (Monaghan et al., 2021). 28
4. Ringkasan Arti Setiap Metrik (Erickson & Kitamura, 2021). 28
5. Waktu Penelitian. 33
6. Label kelas esensial dan tidak esensial 44
7. Hasil Pembersihan Data 45
8. Penggabungan Sekuens CEG dan Label Kelas CEG 47
9. Mengubah Label Kelas CEG 48
10. Penggabungan Sekuens OEG dan Label Kelas OEG 49
11. Mengubah Label Kelas OEG 49
12. Pembagian Data OEG 56
13. Pembagian Data CEG 58
14. Random Undersampling CEG 60
15. Hasil pembagian data training 90% dan validation 10% k-mer 5 OEG 66
16. Hasil pembagian data *training* 90% dan *validation* 10% k-mer 7 OEG 68
17. Hasil pembagian data training 80% dan validation 20% k-mer 5 OEG 70
18. Hasil pembagian data training 80% dan validation 20% k-mer 7 OEG 72
19. Hasil pembagian data training 90% dan validation 10% k-mer 5 CEG 75
20. Hasil pembagian data training 90% dan validation 10% k-mer 7 CEG 77
21. Hasil pembagian data training 80% dan validation 20% k-mer 5 CEG 79
22. Hasil pelatihan data training 80% dan validation 20% k-mer 7 CEG 81
23. Hasil Pengujian 90% Training dan 10% Validation k-mer 5 OEG 87

24. Hasil Pengujian 90% training dan 10% validation k-mer 7 OEG..... 89
25. Hasil Pengujian 80% Training dan 20% Validation k-mer 5 OEG..... 91
26. Hasil pengujian 80% training dan 20% validation k-mer 7 OEG..... 93
27. Hasil Pengujian 90% Training dan 10% Validation k-mer 5 CEG..... 95
28. Hasil Pengujian 90% training dan 10% validation k-mer 7 CEG..... 97
29. Hasil Pengujian 80% Training dan 20% Validation k-mer 5 CEG..... 99
30. Hasil Pengujian 80% training dan 20% validation k-mer 7 CEG..... 101
31. Perbandingan Metode Terdahulu Dengan Metode BiLSTM..... 112

I. PENDAHULUAN

1.1 Latar Belakang

DNA adalah molekul yang mengandung informasi genetik yang penting bagi pertumbuhan, kelangsungan hidup, dan reproduksi suatu organisme. Teknik pengurutan DNA digunakan untuk menetapkan urutan nukleotida yang spesifik dalam suatu molekul DNA. Dasar urutan DNA adalah meneruskan informasi yang diperlukan oleh sel untuk menyusun komponen RNA dan protein (Hamed et al., 2023). Molekul-molekul biokimia seperti DNA (*Deoxyribonucleic acid*), RNA (*Ribonucleic acid*), dan protein memiliki peran krusial dalam organisasi seluler. Mereka memainkan peran yang sangat signifikan dalam sejumlah proses biologis pada organisme tertentu, mengatur aktivitas selama seluruh rentang hidup organisme tersebut. DNA dianggap sebagai mesin cetak biru semua organisme hidup yang membawa informasi genetik seluruh sel (Abd –Alhalem, et al., 2021).

Informasi genetik disimpan dalam bentuk kode kimia atau biasa disebut dengan empat basa nitrogen, yaitu *Adenine* (A), *Guanine* (G), *Cytosine* (C), dan *Thymine* (T). Sebagian besar DNA setiap manusia memiliki sekitar tiga miliar basa dalam genomnya. Urutan basa nitrogen ini memiliki peran penting dalam menentukan informasi yang diperlukan untuk membentuk suatu organisme. DNA yang membawa informasi genetik disebut dengan gen. Gen yang disusun sedemikian rupa menjadi struktur panjang yang disebut dengan kromosom dan genomnya terdiri dari 23 pasang kromosom, sehingga totalnya terdiri dari 46 kromosom. Jumlah gen untuk setiap manusia memiliki sekitar 20.000 hingga 30.000 (Nandhini & Tamilpavai, 2023). Setiap jenis nukleotida dapat membentuk ikatan dengan pasangan

komplementernya dalam untai ganda DNA yang berlawanan. *Adenine* (A) berpasangan dengan *Thymine* (T), sedangkan *Cytosine* (C) berpasangan dengan *Guanine* (G) (El-Tohamy et al., 2022).

Gen esensial memiliki peran yang sangat penting dalam mencapai kelangsungan hidup atau keberhasilan reproduksi suatu organisme. Oleh karena itu, pengetahuan tentang esensialitas gen menjadi fokus utama dalam berbagai penelitian ilmu hayati. Hal ini terutama bermanfaat untuk mengenali target obat, seperti dalam terapi kanker, atau menemukan target insektisida (Beder et al., 2021). *Drosophila Melanogaster* atau biasa disebut lalat buah dianggap sebagai organisme paling efektif untuk menganalisis akar penyebab penyakit manusia karena mirip dengan vertebrata, termasuk manusia dalam fungsi organ internal. Meskipun terdapat sedikit perbedaan seluler dan morfologi, sekitar 75% gen terkait penyakit manusia diyakini serupa dalam fungsinya pada *Drosophila Melanogaster*. *Drosophila Melanogaster* telah digunakan untuk penelitian penyakit pada manusia seperti gangguan saraf, penyakit kardiovaskular, kanker, dan metabolisme. Sistem saraf pada *Drosophila Melanogaster*, seperti pendengaran dan memori, telah membantu memahami disfungsi neurologis seperti epilepsi, degenerasi saraf, dan penyakit *Alzheimer*. Meskipun belum digunakan dalam penelitian asma, potensi *Drosophila Melanogaster* dalam menghubungkan proses genetik dengan fungsi biologis menjanjikan (Chola et al., 2022).

Penelitian tentang esensialitas gen melalui metode eksperimen laboratorium memerlukan waktu, tenaga, dan biaya yang besar. Oleh karena itu, dibuatlah pendekatan komputasi untuk mengklasifikasi sekuens DNA menggunakan metode *deep learning*, yaitu metode *Bidirectional Long Short Term Memory (*BiLSTM). Dengan menggunakan pendekatan ini dibangun model pembelajaran mendalam yang dievaluasi secara matematis untuk mengklasifikasikan sekuens DNA untuk penilaian esensialitas gen pada lalat buah (*Drosophila Melanogaster*) (Aromolaran et al., 2020).

1.2 Rumusan Masalah

Berdasarkan judul skripsi "Klasifikasi Gen Esensial pada Sekuens DNA Lalat Buah (*Drosophila Melanogaster*) menggunakan metode *Bidirectional Long Short Term Memory* (BiLSTM)" dan latar belakang yang telah diuraikan, maka terdapat beberapa permasalahan yang akan dibahas, yaitu sebagai berikut.

a) Apakah metode *Bidirectional Long Short Term Memory* (BiLSTM) dapat diimplementasikan untuk Klasifikasi Gen Esensial Lalat Buah.
b) Apakah parameter dan skenario yang terbaik untuk metode *Bidirectional Long Short Term Memory* (BiLSTM).
c) Seberapa baik perbandingan antara penelitian ini dengan penelitian terdahulu.

1.3 Batasan Masalah

Berdasarkan uraian pada latar belakang sebelumnya, terdapat batasan masalah dari penelitian ini, yaitu sebagai berikut.

a) Program Klasifikasi Gen Esensial pada Sekuens DNA Lalat Buah menggunakan metode *Bidirectional Long Short Term Memory* (BiLSTM).
b) Program mengatasi *imbalanced* data pada dataset menggunakan *NearMiss Undersampling*.
c) Program Klasifikasi Gen Esensial pada Sekuens DNA Lalat Buah menggunakan sumber data yang tersedia pada jurnal *Identifying essential genes across eukaryotes by machine learning* (Beder et al., 2021)

1.4 Tujuan

Tujuan dari penelitian Klasifikasi Gen Esensial pada Sekuens DNA Lalat Buah (*Drosophila Melanogaster*) menggunakan metode *Bidirectional Long Short Term Memory* (BiLSTM) dapat dilihat dengan detail sebagai berikut.

a) Menyajikan cara kerja metode *Bidirectional Long Short Term* Memory (BiLSTM) untuk Klasifikasi Gen Esensial pada Sekuens DNA Lalat Buah.

b) Membandingkan hasil evaluasi kinerja metode *Bidirectional Long Short Term Memory* dengan metode terdahulu.

1.5 Manfaat

Manfaat dari penelitian Klasifikasi Gen Esensial pada Sekuens DNA Lalat Buah (*Drosophila Melanogaster*) menggunakan metode *Bidirectional Long Short Term Memory* (BiLSTM), diuraikan sebagai berikut.

a) Terciptanya metode *Bidirectional Long Short Term Memory* untuk klasifikasi Gen Esensial pada Sekuens DNA Lalat Buah.

b) Melakukan perbandingan metode terdahulu dengan metode *Bidirectional Long Short Term Memory.*

c) Memberikan pemahaman tentang pentingnya esensialitas gen pada lalat buah yang memiliki dampak positif dalam organisme seluler.

II. TINJAUAN PUSTAKA

2.1 Penelitian terdahulu

Penelitian yang dilakukan sebelumnya disebut dengan penelitian terdahulu. Berikut ini adalah penelitian sebelumnya yang terkait dengan esensialitas gen sebagai bentuk kajian dasar untuk memperkuat rangka penyusunan terhadap penelitian yang dilakukan.

Tabel 1. Penelitian Terdahulu Terkait Gen Esensial.

NO	Penelitian	Data	Metode	Hasil
1	*Essential gene prediction in Drosophila melanogaster using machine learning approaches based on sequence and functional features* (Aromolaran et al., 2020)	OGEE & DEG *Drosophila melanogaster* Essential gen: 441 gen Non Essential gene: 11.788	*Generalised Linear Model (*GLM*), Support-Vector Machines(*SVM*), Artificial Neural Network (*NNET*), Random Forest (*RF*),* dan *Extreme Gradient Boosting(*XGB*)*	GLM ROC-AUC=89% PR-AUC=27% F1-*score*=28% SVM ROC-AUC=88% PR-AUC=27% F1-*score*=30% NNET ROC-AUC=85% PR-AUC=20% F1-*score*=24% RF ROC-AUC=90%

				PR-AUC=29% F1-Score=32% XGB ROC-UC=90% PR-AUC=30% F1-Score=34%
2	*Identifying essential genes across eukaryotes by machine learning* (Beder et al., 2021)	OGEE *Drosophila melanogaster* CEG: 11547 *Essential gene:* 1227 *Non Essential gene: 10320* OEG: 517 *Essential gene: 246* *Non Essential gene: 271*	*Random Forest (*RF*), Extreme Gradient Boosting (*XGB*), Oversampling* SMOTE	CEG RF: ROC-UC=84% PR-AUC=41% *Sensitivity*=54% *Specificity*=88% XGB: ROC-UC=83% PR-AUC=40% *Sensitivity*=55% *Specificity*=88% OEG RF: ROC-UC=92% PR-AUC=92% *Sensitivity*=82% *Specificity*=82% XGB: ROC-UC=91% PR-AUC=90% *Sensitivity*=81% *Specificity*=85%

3	*Performance evaluation of features for gene essentiality prediction* (Aromolaran et al., 2021).	OGEE & DEG *Saccharomyces Cerevisiae* dan *Schizosaccharo-myces pombe* *Essential gene:1037* *Non Essential gene: 4543*	*Random Forest* (RF), *Artificial Neural Network* (NNET), *Extreme Gradient Boosting* (XGB)	DNA *sequence S. cerevisiae* RF AUROC=67% AUPRC=32,3% *Precision*=54,5% *Sensitivity*=1,7% NNET AUROC=60,7% AUPRC=26,1% *Precision*=31,1% *Sensitivity*=21,1% XGB AUROC=66,3% AUPRC=31,7% *Precision*=47,3% *Sensitivity*=5,9%
				DNA *sequence S. pombe.* RF AUROC=60,8% AUPRC=35,1% *Precision*=41% *Sensitivity*=5,1%

Penjelasan mengenai masing-masing penelitian pada Tabel 1 dapat dijabarkan sebagai berikut.

2.1.1 Essential gene prediction in Drosophila melanogaster using machine learning approaches based on sequence and functional features

Penelitian yang dilakukan oleh (Aromolaran et al., 2020) menggabungkan daftar gen esensial yang komprehensif dari dataset *Online GEne Essentiality* (OGEE), dan *Database of Essential Genes* (DEG), juga menggunakan daftar gen nonesensial dari OGEE yang menghasilkan 411 gen esensial dan 11.788 gen nonesensial. Fitur yang digunakan berdasarkan delapan sumber berbeda, yaitu *protein sequence, gene sequence, domains, topology (transcription profiles), topology* (PPI), *evolution*, *localization*, dan *gene sets*.

Metode yang digunakan pada penelitian ini, yaitu *Generalised Linear Model (*GLM*), Support-Vector Machines(*SVM*), Artificial Neural Network (*NNET*), Random Forest (*RF*),* dan *Extreme Gradient Boosting(*XGB*). Extreme Gradient Boosting(*XGB*)* memiliki akurasi paling tinggi dibandingkan dengan metode yang lainnya, dengan nilai ROC-AUC sebesar 90%, PR-AUC sebesar 30%, dan F1-*Score* sebesar 34%.

2.1.2 Identifying essential genes across eukaryotes by machine Learning

Penelitian yang dilakukan oleh (Beder et al., 2021) menggunakan informasi gen esensial dari enam spesies, yaitu *C. elegans, D. melanogaster, H. sapiens, M. musculus, S. cerevisiae* dan *S. pombe* yang didapat dari database OGEE dan DEG. Untuk spesies *D. melanogaster, H. sapiens, M. musculus* menggunakan dua dataset dari *CEG (cellular essential gene)* dan OEG *(organismal essential gene).* Untuk *C. elegans* menggunakan dataset OEG. *S. cerevisiae* dan *S. pombe* menggunakan dataset *CEG.* Data gen esensial yang diperoleh sebanyak 11.038, dan data gen nonesensial sebanyak

67.035 dengan tujuh fitur kategori, yaitu *protein gene sequence, functional domains, topological features, evolution/conservation, subcellular localization,* dan *gene sets* dari *Gene Ontology.*

Klasifikasi gen esensial menggunakan metode *Random Forest (*RF*),* dan *Extreme Gradient Boosting(*XGB*),* dan untuk mengatasi ketidakseimbangan kelas digunakan metode resampling data, yaitu metode SMOTE (*Synthetic Minority Oversampling Technique*). Pada CEG, *Random Forest* menghasilkan ROC-AUC sebesar 84%, PR-AUC sebesar 40%, *sensitivity* sebesar 54%, dan specificity sebesar 88%, sedangkan *Extreme Gradient Boosting* menghasilkan ROC-AUC sebesar 83%, PR-AUC sebesar 40%, *sensitivity* sebesar 55%, dan specificity sebesar 86%. Pada OEG, *Random Forest* menghasilkan ROC-AUC sebesar 92%, PR-AUC sebesar 92%, *sensitivity* sebesar 82%, dan specificity sebesar 82%, sedangkan *Extreme Gradient Boosting* menghasilkan ROC-AUC sebesar 91%, PR-AUC sebesar 90%, *sensitivity* sebesar 81%, dan specificity sebesar 85%.

2.1.3 Performance evaluation of features for gene essentiality Prediction

Penelitian yang dilakukan (Aromolaran et al., 2021) menggunakan informasi gen esensial dari dua spesies, yaitu *S. cerevisiae* dan *Schizosaccharomyces pombe* yang dikumpulkan dari *Database of Essential Genes* (DEG) dan *Online GEne Essentiality* (OGEE). Diperoleh gen esensial sebanyak 1.037 dan gen nonesensial sebanyak 4.543 untuk spesies *S. cerevisiae*, sedangkan untuk *Schizosaccharomyces pombe* diperoleh gen esensial sebanyak 1.346 dan gen nonesensial sebanyak 3.689.

Fitur yang digunakan berdasarkan empat kategori berbeda, yaitu *protein sequence, gene sequence*, *topology (PPI)* dan *gene sets*

enrichment dari *gene ontology*. Metode yang digunakan pada penelitian ini adalah *Random Forest* (RF), *Artificial Neural Network* (NNET), *Extreme Gradient Boosting* (XGB).

2.2 Uraian tentang landasan teori

Berikut adalah beberapa uraian dari landasan teori yang berkaitan dengan penelitian yang dilakukan.

2.2.1 *Drosophila Melanogaster* (Lalat Buah)

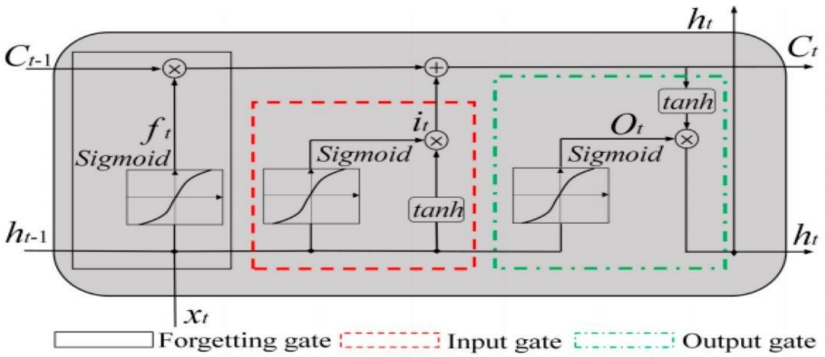
Drosophila Melanogaster atau lalat buah telah menjadi salah satu sistem model untuk penelitian genetika sejak awal tahun 1900-an. Meski sebagian besar penelitian berfokus pada genetika, para peneliti dengan cepat menyadari bahwa *Drosophila* dapat digunakan untuk penelitian dalam berbagai bidang lainnya, seperti perkembangan, fisiologi, perilaku, memori, dan pembelajaran. Meskipun terdapat sejarah evolusi yang signifikan yang memisahkan *Drosophila* dari manusia, banyak penemuan yang berasal dari penelitian pada lalat buah telah diterapkan pada mamalia, memberikan wawasan yang penting dalam biologi manusia, termasuk pemahaman tentang berbagai penyakit. Sebagai contoh, *Drosophila* telah berhasil digunakan dalam studi penuaan dan biologi otot. Mengingat konservasi jalur metabolisme dasar dan proses fisiologis, serta dengan keberagaman alat untuk manipulasi genetik dan komunitas riset yang dinamis dan kreatif, lalat buah menjadi sistem model yang sangat efektif untuk menyelidiki berbagai aspek biologi (Riddle, 2019).

2.2.2 Essential Genes

Gen esensial adalah bagian dari gen yang sangat diperlukan untuk kelangsungan hidup atau reproduksi organisme hidup. Prediksi esensialitas gen sangat penting untuk memahami kebutuhan minimal suatu organisme, mengidentifikasi gen penyakit, dan menemukan target obat baru (Zhang et al., 2020). Pengetahuan tentang gen esensial pada serangga khususnya lalat buah sangat relevan untuk layanan kesehatan dan pertanian, karena kelompok organisme ini merupakan vektor penting penyakit menular seperti malaria, demam berdarah, penyakit gangguan tidur, dan hama tanaman. Secara umum, prediksi gen esensial dapat didasarkan pada fitur intrinsik dari nukleotida atau urutan protein dan kombinasi keduanya. Fitur intrinsik dalam konteks ini menunjukkan ciri-ciri yang dapat diturunkan langsung dari rangkaian DNA maupun protein (Aromolaran et al., 2020).

2.2.3 *Bidirectional Long Short Term Memory* (BiLSTM)

Menurut (Alharbi & Csala, 2021) BiLSTM dikembangkan dengan mengadopsi pendekatan LSTM yang bertujuan untuk meningkatkan performa dalam proses klasifikasi. BiLSTM dirancang dengan mengintegrasikan RNN dengan pendekatan LSTM, yang memungkinkannya mengakses konteks jangka panjang. Dibandingkan dengan pendekatan satu arah seperti LSTM, BiLSTM unggul secara signifikan dalam kecepatan dan akurasi, melampaui kinerja RNN konvensional. Berikut ini merupakan arsitektur dari *Long Short Term Memory* (LSTM) yang dapat dilihat pada gambar 1.



Gambar 1. Arsitektur LSTM (Aldhyani & Alkahtani, 2021).

Model LSTM memiliki terdiri dari tiga gerbang yaitu forgetting gate, input gate, dan ouput gate. Fungsi dari masing-masing gerbang dijabarkan lebih rinci sebagai berikut (Tavakoli, 2019).

A. *Forgetting gate*:

Forgetting gate mengatur apakah informasi yang ada akan tetap disimpan didalam sel atau tidak. Gerbang ini menggunakan fungsi *sigmoid* untuk menentukan informasi mana yang perlu dihapus dari memori LSTM. Keputusan ini dipengaruhi oleh nilai $h_t - 1$ dan x_t. Dapat dilihat pada persamaan 1.

$$f_t = \sigma\left(W_{fh}[h_{t-1}] + W_{fx}[x_t] + b_f\right) \quad \text{.....(1)}$$

f_t adalah nilai dalam rentang antara 0 dan 1, dimana jika nilai mendekati 0 menunjukkan penghapusan penuh dari nilai yang telah dipelajari dan nilai menunjukkan mendekati 1 berarti mempertahankan nilai tersebut sepenuhnya. b_f menunjukkan nilai konstan yang dikenal dengan bias.

B. *Input gate*:

Input gate menentukan apakah informasi baru akan disimpan atau tidak kedalam memori LSTM. Gerbang ini terdiri dari dua lapisan, yaitu: 1. Lapisan *sigmoid*, dan 2. Lapisan *tanh*. Lapisan *sigmoid* digunakan untuk memutuskan nilai mana yang perlu diperbarui, sedangkan lapisan *tanh* digunakan untuk membuat vektor nilai kandidat baru yang dapat ditambahkan ke memori

LSTM. Berikut ini persamaan dari *input gate*. Dapat dilihat pada persamaan 2 dan 3.

$i_t = \sigma\,(\,W_{ih}[h_{t-1}] + W_{ix}[x_t] + b_i\,)$ ..... (2)

$\tilde{c}_t = tanh\,(\,W_{ch}[h_{t-1}] + W_{cx}[x_t] + b_c\,)$ ..... (3)

Dimana i_t mewakili keputusan nilai mana yang perlu diperbarui dan c menunjukkan vektor nilai kandidat baru yang dapat ditambahkan ke memori LSTM. Kedua lapisan digabung untuk membuat pembaruan memori LSTM, rumus pembaruan memori LSTM direpresentasikan sebagai berikut. Dapat dilihat pada persamaan 4.

$c_t = f_t \times c_{t-1} + i_t \times \tilde{c}_t$.....(4)

Dimana f_t adalah bilangan antara 0 dan 1 yang diperoleh dari gerbang forget. Bilangan 0 menyiratkan untuk membuang nilai dan 1 menyiratkan untuk mempertahankan nilai yang mewakili sejauh mana informasi lama tetap berada di memori LSTM.

C. *Output gate*:

Output gate mengambil keputusan apakah nilai yang ada di dalam sel akan digunakan untuk menghasilkan keluaran dari LSTM. Berikut rumus *output gate* direpresentasikan sebagai berikut. Dapat dilihat pada persamaan 5 dan 6.

$o_t = \sigma\,(\,W_{oh}[h_{t-1}] + W_{oh}[x_t] + b_o\,)$ ..... (5)

$h_t = O_t \times$

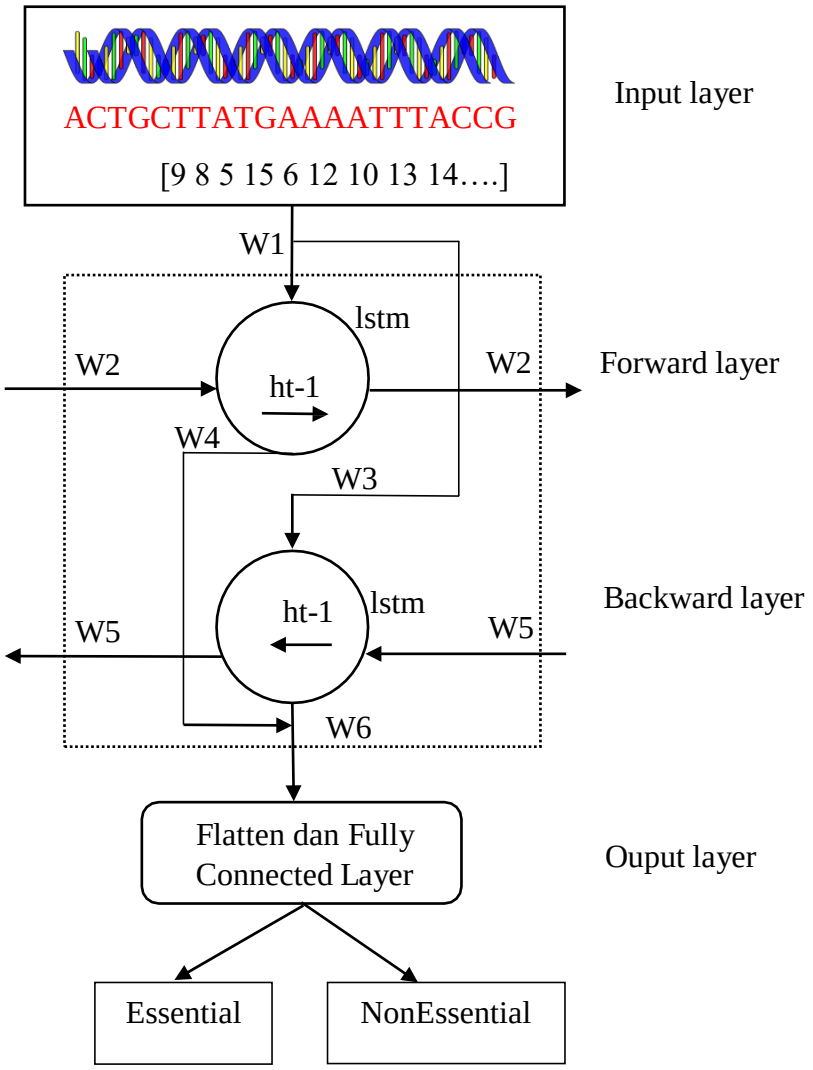
$tanh(C_t)$.....(6)

Semua gerbang bergantung pada ukuran data. W adalah matriks bobot antar sel. Nilai bias untuk setiap lapisan ditentukan oleh *b*, dan fungsi *sigmoid* σ dan fungsi tangen hiperbolik digunakan sebagai fungsi aktivasi. Rumus fungsi aktivasi direpresentasikan dalam Persamaan berikut. Dapat dilihat pada persamaan 7 dan 8.

sigmoid $= \frac{1}{1+e^{-z}}$ ……..(7)

tanh $= \frac{e^{z}-e^{-z}}{e^{z}-e^{-z}}$ ..... (8)

BiLSTM memberikan detail yang komprehensif terhadap informasi sekuensial untuk setiap langkah dalam urutan yang diberikan, mencakup tahap sebelum dan sesudahnya. Metodenya juga melibatkan penggunaan algoritma LSTM untuk perhitungan lapisan tersembunyi. Berbeda dengan LSTM, keunggulan BiLSTM terletak pada kemampuannya untuk memproses data dalam dua arah berbeda, memanfaatkan dua lapisan tersembunyi, dan mengalirkan hasilnya ke lapisan keluaran yang sama. Berikut ini adalah struktur dari *Bidirectional Long Short Term Memory* (BiLSTM) yang merujuk pada jurnal (Alharbi & Csala, 2021) yang dapat dilihat pada Gambar 2.



Gambar 2. Struktur BiLSTM (Alharbi & Csala, 2021).

Model BiLSTM memiliki kemampuan memproses data dalam dua arah yaitu *forward* dan *backward,* untuk rumus *backward propagation through time* (BPTT) dapat dijabarkan sebagai berikut (Smagulova & James, 2020). Dapat dilihat pada persamaan 9 sampai 17.

$$\text{Gate}S_t = \begin{matrix} \tilde{c}_t \\ i_t \\ f_t \\ o_t \end{matrix} \quad V = \begin{bmatrix} W^{(\tilde{c})} \\ W^{(i)} \\ W^{(f)} \\ W^{(o)} \end{bmatrix}, U = \begin{bmatrix} U^{(C)} \\ U^{(i)} \\ U^{(f)} \\ U^{(o)} \end{bmatrix}, b = \begin{bmatrix} b^{(C)} \\ b^{(i)} \\ b^{(f)} \\ b^{(o)} \end{bmatrix}$$

$\delta h_t = \Delta_t + \Delta h_t$ .....(9)

$\delta C_t = \delta h_t \odot o_t \odot (1 - tanh^2(C_t)) + \delta C_{t+1} \odot f_{t+1}$..... (10)

$\delta c = \delta C_t \odot i_t \odot (1 - \tilde{c}_t^2)$ .....(11)

$\delta i_t = \delta C_t \odot \tilde{c}_t \odot (1 - i_t)$ ..... (12)

$\delta f_t = \delta C_t \odot C_{t-1} \odot f_t \odot (1 - f_t)$ .....(13)

$\delta o_t = \delta h_t \odot \tanh(C_t) \odot o_t \odot (1 - f_t)$.....(14)

$\delta x_t = W^T . \delta gates_t$ ..... (15)

$\Delta h_{t-1} = U^T . \delta gates_1$.....(16)

Berdasarkan hal diatas didapat,

$W^{new} = W^{old} - \lambda . \delta W^{old}$..... (17)

Dimana λ menunjukkan koefisien *Stochastic Gradient Descent* (SGD) dan delta $\delta W = \sum_{t=1}^{T} \delta gates_t . x_t$, $\delta U = \sum_{t=1}^{T} \delta gates_{t+1} . h_t$, $\delta b = \sum_{t=1}^{T} \delta gates_{t+1}$

Contoh perhitungan manual *Bidirectional Long Short Term Memory* (BiLSTM) dijabarkan sebagai berikut:

$$X0 = \begin{bmatrix} 0.25 \\ 0.30 \end{bmatrix} \text{ dengan label } \begin{bmatrix} 0.5 \\ 0.3 \end{bmatrix}$$

$$X1 = \begin{bmatrix} 1 \\ 0.80 \end{bmatrix} \text{ dengan label } \begin{bmatrix} 1.25 \\ 1 \end{bmatrix}$$

Nilai matriks bobot yang sesuai disediakan di bawah ini:

$$W^{(f)} = \begin{bmatrix} 0.11 & 0.32 \\ 0.42 & 0.19 \end{bmatrix}, W^{(i)} = \begin{bmatrix} 0.60 & 0.17 \\ 0.16 & 0.17 \end{bmatrix},$$

$$W^{(g)} = \begin{bmatrix} 0.46 & 0.74 \\ 0.75 & 0.65 \end{bmatrix}, W^{(o)} = \begin{bmatrix} 0.98 & 0.08 \\ 0.15 & 0.54 \end{bmatrix}$$

Dan matriks bobot tersembunyi:

$$U^{(f)} = \begin{bmatrix} 0.87 & 0.50 \\ 0.23 & 0.67 \end{bmatrix}, U^{(i)} = \begin{bmatrix} 0.30 & 0.89 \\ 0.64 & 0.65 \end{bmatrix},$$

$$U^{(g)} = \begin{bmatrix} 0.60 & 0.12 \\ 1.00 & 0.01 \end{bmatrix}, U^{(o)} = \begin{bmatrix} 0.41 & 0.62 \\ 0.62 & 0.14 \end{bmatrix}$$

$$b^{(f)} = [0.30 \quad 0.1], b^{(i)} = [0.67 \quad 0.13],$$

$$b^{(g)} = [0.47 \quad 0.07], b^{(o)} = [0.75 \quad 0.09]$$

Forward propagation

1. Pertama, hitung gerbang t0:

$$g0 = tanh\left(\begin{bmatrix} 0.46 & 0.75 \\ 0.74 & 0.65 \end{bmatrix} \cdot \begin{bmatrix} 0.25 \\ 0.30 \end{bmatrix} + \begin{bmatrix} 0.60 & 1.00 \\ 0.12 & 0.01 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \end{bmatrix} + [0.47 \quad 0.07]\right) = \begin{bmatrix} 0.66959 \\ 0.42190 \end{bmatrix},$$

Dan demikian pula untuk $f0 = \begin{bmatrix}0.61147\\0.55897\end{bmatrix}, i0 = \begin{bmatrix}0.70432\\0.55564\end{bmatrix}, o0 = \begin{bmatrix}0.73885\\0.56758\end{bmatrix}$.

2. Karena tidak ada Ct−1, keadaan memori pada waktu t0 adalah $C0 = \begin{bmatrix}0.47161\\0.23442\end{bmatrix}$.
3. Akhirnya, keluaran sel pada t0 adalah $h0 = \begin{bmatrix}0.32472\\0.13067\end{bmatrix}$.
4. Setelah mengulangi langkah 1–4 untuk langkah waktu t1:

   $f1 = \begin{bmatrix}0.74248\\0.69481\end{bmatrix}, i1 = \begin{bmatrix}0.82911\\0.69219\end{bmatrix}, o1 = \begin{bmatrix}0.88740\\0.69463\end{bmatrix}$,

   $g1 = \begin{bmatrix}0.95231\\0.87876\end{bmatrix}$.

   keadaan memori pada waktu t1 adalah $C1 = \begin{bmatrix}1.13973\\0.77115\end{bmatrix}$.

   dan keluaran sel pada t1 adalah $h1 = \begin{bmatrix}0.72263\\0.44984\end{bmatrix}$.

Backward propagation

1. Delta pada propagasi mundur untuk jangka waktu t1:

$$\delta h_1 = \left(\begin{bmatrix}0.72263\\0.44984\end{bmatrix} - \begin{bmatrix}1.25\\1\end{bmatrix}\right) + \begin{bmatrix}0\\0\end{bmatrix} = \begin{bmatrix}-0.5274\\-0.5502\end{bmatrix},$$

$$\delta c_1 = \begin{bmatrix}-0.5274\\-0.5502\end{bmatrix} \odot \begin{bmatrix}0.88740\\0.69463\end{bmatrix} \odot \left(\begin{bmatrix}1 - 0.81432^2\\1 - 0.64760^2\end{bmatrix}\right) + 0 \odot 0$$

$$= \begin{bmatrix}-0.1577\\-0.2219\end{bmatrix},$$

$$\delta g_1 = \begin{bmatrix}-0.1577\\-0.2219\end{bmatrix} \odot \begin{bmatrix}0.82911\\0.69219\end{bmatrix} \odot \left(\begin{bmatrix}1 - 0.9068^2\\1 - 0.7722^2\end{bmatrix}\right)$$

$$= \begin{bmatrix}-0.01218\\-0.03498\end{bmatrix},$$

$$\delta i_1 = \begin{bmatrix}-0.1577\\-0.2219\end{bmatrix} \odot \begin{bmatrix}0.95231\\0.87876\end{bmatrix} \odot \left(\begin{bmatrix}1 - 0.82911\\1 - 0.69219\end{bmatrix}\right)$$

$$= \begin{bmatrix}-0.02567\\-0.06005\end{bmatrix},$$

$$\delta f_1 = \begin{bmatrix} -0.1577 \\ -0.2219 \end{bmatrix} \odot \begin{bmatrix} 0.47161 \\ 0.23442 \end{bmatrix} \odot \begin{bmatrix} 0.74248 \\ 0.69481 \end{bmatrix} \odot \left(\begin{bmatrix} 1 - 0.74248 \\ 1 - 0.69481 \end{bmatrix} \right) = \begin{bmatrix} -0.01422 \\ -0.01103 \end{bmatrix},$$

$$\delta o_1 = \begin{bmatrix} -0.5274 \\ -0.5502 \end{bmatrix} \odot \begin{bmatrix} 0.81432 \\ 0.64760 \end{bmatrix} \odot \left(\begin{bmatrix} 0.88740 \\ 0.69463 \end{bmatrix} \right) = \begin{bmatrix} 1 - 0.88740 \\ 1 - 0.69463 \end{bmatrix},$$

$$\delta x_1 = \begin{bmatrix} -0.04292 \\ -0.07558 \end{bmatrix},$$

$$\Delta h_0 = \begin{bmatrix} -0.16145 \\ -0.12819 \end{bmatrix}.$$

2. Langkah yang sama diambil untuk mengidentifikasi delta untuk t_t:

$$\delta h_0 = \left(\begin{bmatrix} 0.32472 \\ 0.13067 \end{bmatrix} - \begin{bmatrix} 0.5 \\ 0.3 \end{bmatrix} \right) + \begin{bmatrix} -0.21041 \\ -0.12374 \end{bmatrix} = \begin{bmatrix} -0.33673 \\ -0.29752 \end{bmatrix},$$

$$\delta C_0 = \begin{bmatrix} -0.31781 \\ -0.31409 \end{bmatrix}, \delta g_0 = \begin{bmatrix} -0.12348 \\ -0.14345 \end{bmatrix}, \delta i_0 = \begin{bmatrix} -0.08948 \\ -0.12264 \end{bmatrix},$$

$$\delta f_0 = 0,$$

$$\delta o_0 = \begin{bmatrix} -0.13132 \\ -0.04757 \end{bmatrix}, \delta x_0 = \begin{bmatrix} -0.43607 \\ -0.16384 \end{bmatrix}, \Delta h_{-1} = \begin{bmatrix} -0.63738 \\ -0.62015 \end{bmatrix}.$$

3. Delta matriks berat: $\delta W = \begin{matrix} \text{F} \\ \text{I} \\ \text{I} \\ \text{I} \\ \text{I} \\ \text{I} \\ \text{I} \\ \text{I} \\ \text{I} \\ [\end{matrix} \begin{matrix} -0.0916 & -0.1051 \\ -0.1135 & -0.1222 \\ -0.0565 & -0.0576 \\ -0.0959 & -0.0911 \\ -0.0142 & -0.0113 \\ -0.0110 & -0.0088 \\ -0.0758 & -0.0737 \\ -0.0875 & -0.0747 \end{matrix} \begin{matrix} 1 \\ \text{I} \\ \text{I} \\ \text{I} \\ \text{I} \\ \text{I} \\ \text{I} \\ ] \end{matrix},$

$$\delta U = \begin{bmatrix} -0.0040 & -0.0016 \\ -0.0114 & -0.0046 \\ -0.0083 & -0.0034 \\ -0.0195 & -0.0078 \\ -0.0046 & -0.0019 \\ -0.0036 & -0.0014 \\ -0.0139 & -0.0056 \\ [-0.0245 & -0.0099] \end{bmatrix}, \delta b = \begin{bmatrix} -0.3300 \\ -0.3491 \\ -0.1492 \\ -0.2035 \\ -0.0142 \\ -0.0110 \\ -0.1742 \\ [-0.1231] \end{bmatrix}.$$

4. Akhirnya, setelah menerapkan satu iterasi Stochastic Gradient Descent (SGD) dengan learning rate λ = 0,1, nilai bobot baru diperoleh:

$$W^{(g)} = \begin{bmatrix} 0.4614 & 0.7411 \\ 0.7011 & 0.6009 \end{bmatrix}, W^{(i)} = \begin{bmatrix} 0.6092 & 0.1814 \\ 0.1105 & 0.1122 \end{bmatrix},$$

$$W^{(f)} = \begin{bmatrix} 0.1157 & 0.3296 \\ 0.4058 & 0.1091 \end{bmatrix}, W^{(0)} = \begin{bmatrix} 0.9876 & 0.0887 \\ 0.1074 & 0.5075 \end{bmatrix},$$

$$U^{(g)} = \begin{bmatrix} 0.6005 & 0.1204 \\ 1.0002 & 0.0101 \end{bmatrix}, U^{(i)} = \begin{bmatrix} 0.3004 & 0.8911 \\ 0.6402 & 0.6505 \end{bmatrix},$$

$$U^{(f)} = \begin{bmatrix} 0.8708 & 0.5019 \\ 0.2303 & 0.6708 \end{bmatrix}, U^{(o)} = \begin{bmatrix} 0.4114 & 0.6225 \\ 0.6206 & 0.1409 \end{bmatrix},$$

$$b^{(g)} = [0.4714 \quad 0.0711], b^{(i)} = [0.7030 \quad 0.1649],$$

$$b^{(f)} = [0.3149 \quad 0.1203], b^{(o)} = [0.7674 \quad 0.1023].$$

Tabel 2. Notasi LSTM (Smagulova & James, 2020).

Sysmbols	*Name*
x_t	*Input vector*
h_{t-1}	*Output of a previous cell*
c_t	*Cell memory of current state*
c_{t-1}	*Cell memory of a previous cell*
$\tilde{c}_t = g_t$	*Candidate to a cell memory*
i_t	*Input gate*
f_t	*Forget gate*

o_t	*Output gate*
h_t	*Output of a current state*
σ	*Sigmoid function*
$tanh$	*Hyperbolic tangent function*
$V^{(*)}$	*Weight matrice*
$W^{(*)}= W_*[x_t]$	*Weight matrice*
$U^{(*)}= W_*[h_{t-1}]$	*Weight matrice*
b_*	*bias*

2.2.4 OEG *(Organismal Essential Genes)*

OEG adalah gen penting yang diperlukan untuk kelangsungan hidup organisme secara keseluruhan. Gen-gen ini penting untuk perkembangan embrio, morfogenesis saraf, dan fungsi organ dalam organisme multiseluler. Gen-gen ini mungkin tidak penting untuk sel individual atau organisme bersel tunggal seperti ragi, tetapi mereka mungkin sangat penting untuk organisme yang lebih kompleks, seperti gen yang terlibat dalam morfogenesis neural. Dalam penelitian perbandingan organisme, ditemukan bahwa korelasi gen OEG cukup kuat antar spesies (seperti manusia, tikus, dan lalat) meskipun ada korelasi antara OEG dan CEG dalam satu spesies. Ini menunjukkan bahwa gen-gen yang penting untuk kelangsungan hidup organisme mungkin berbeda dari gen-gen yang penting untuk kelangsungan hidup sel individu. OEG bertanggung jawab atas regulasi, perkembangan, dan proses sistem yang terkait fungsi fisiologis, seperti perkembangan saraf. (Beder et al., 2021).

2.2.5 CEG *(Cellular Essential Genes)*

CEG adalah gen penting untuk kelangsungan hidup sel individu, terutama diidentifikasi melalui eksperimen pada garis sel atau organisme bersel tunggal. Gen-gen ini penting untuk proses seperti

siklus sel, proliferasi, dan biogenesis makromolekul seluler. CEG sering ditemukan melalui skrining seluler yang menekankan pada fungsi dasar sel seperti pembentukan protein, replikasi DNA, dan pertumbuhan sel. Namun, dalam studi perbandingan antarspesies, CEG dan OEG memiliki korelasi yang lebih rendah antara manusia, lalat, tikus, dan ragi. Ini karena gen-gen ini memiliki fungsi yang lebih spesifik pada tingkat sel daripada organisme (Beder et al., 2021).

2.2.6 CRISPR

Teknik pengeditan genom yang disebut CRISPR *(Clustered Regularly Interspaced Short Palindromic Repeats)* memungkinkan modifikasi DNA di lokasi tertentu dalam genom. Sistem CRISPR-Cas terdiri dari kombinasi protein Cas dan RNA pemandu yang memiliki kemampuan untuk memotong DNA target. Metode ini dapat digunakan untuk memperkenalkan mutasi, memperbaiki gen yang rusak, inaktivasi gen target, atau penggantian gen dengan urutan baru. Keunggulan utama CRISPR adalah bahwa itu sederhana, murah, efisien, dan mudah digunakan. Akibatnya, ini telah menjadi alat penting di laboratorium biologi molekuler di seluruh dunia. Sebagai terapi gen, CRISPR digunakan dalam penelitian medis untuk mengobati berbagai penyakit, termasuk kanker dan penyakit genetik (Xu & Li, 2020).

2.2.7 RNA interference (RNAi)

Mekanisme biologis yang dikenal sebagai RNA *interference* (RNAi) dimulai dengan polimerase RNA tergantung RNA (RdRP), yang menghasilkan RNA untai ganda (dsRNA) panjang dari templat RNA untai tunggal. Kemudian, dsRNA ini diproses menjadi RNA interferensi kecil (siRNA) oleh enzim Dicer, yang kemudian dimuat

ke dalam kompleks RISC (RNA-induced sile yang disebabkan oleh RNA). Dalam penelitian biologi molekuler, RNAi digunakan untuk mengontrol ekspresi gen secara in vitro, yaitu di luar organisme hidup, biasanya pada kultur sel atau jaringan dalam lingkungan laboratorium terkontrol. Ini memungkinkan peneliti untuk mempelajari fungsi gen secara khusus dengan kontrol yang lebih baik atas variabel eksperimen. Selain itu, RNAi dapat digunakan in vivo, yaitu di dalam organisme hidup seperti hewan atau manusia, untuk meneliti bagaimana proses pembungkaman gen tersebut berfungsi dalam konteks biologis yang lebih kompleks dan realistis. Pendekatan in vivo memberikan pemahaman yang lebih baik tentang bagaimana RNAi mempengaruhi sistem biologis secara keseluruhan (Isenmann, et al., 2023).

2.2.8 Embedding Layer

Embedding layer diinisialisasi ke matriks acak. mengkonversi data berdimensi tinggi menjadi data berdimensi rendah dengan transformasi matriks (Pang et al., 2020). Lapisan ini mengubah kata-kata ke model ruang vektor berdasarkan seberapa sering sebuah kata muncul dekat dengan kata lain. *Embedding layer* menggunakan bobot acak untuk mempelajari penyematan semua istilah di dalamnya kumpulan data pelatihan (Gunasekaran et al., 2021). *Embedding layer* merupakan lapisan arsitektur yang di mana setiap indeks yang sesuai dengan kata unik dalam kumpulan data, diubah menjadi nilai nyata vektor fitur. Vektor-vektor bernilai riil ini ditumpuk menjadi satu untuk membentuk matriks, yang disebut matriks penyematan, seperti yang ditunjukkan pada persamaan 18. Intuisi di balik penyematan matriksnya adalah setiap baris menggambarkan indeks unik yang di dalamnya gilirannya sesuai dengan kata unik dalam kosa kata. matriks penyematan itu memiliki dimensi $v * d$, dimana v menggambarkan ukuran kosakata kumpulan

data dan d menggambarkan dimensi vektor padat (Aldhyani & Alkahtani, 2021). Dapat dilihat pada persamaan 18.

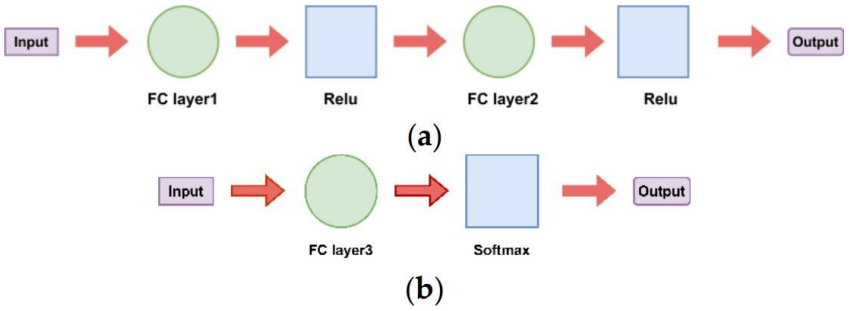
$$R = \begin{bmatrix} r1,1 & r1,2 & \cdots & r1,n \\ r2,1 & r2,2 & \cdots & r2,n \\ r3,1 & r3,2 & \cdots & r3,n \\ \vdots & \vdots & \ddots & \vdots \\ rv-1,1 & rv-1,2 & \cdots & rv-1,n \\ rv,1 & rv,2 & \cdots & rv,n \end{bmatrix}^{1} \quad \text{.....(18)}$$

2.2.9 Flatten Layer

Lapisan flatten memegang peran krusial dalam struktur arsitektur model jaringan saraf BiLSTM. Lapisan tersebut ditempatkan di antara lapisan BiLSTM dan lapisan dense berikutnya. Fungsi utama dari flatten layer adalah untuk melakukan operasi pengubahan bentuk output dari lapisan BiLSTM, sehingga output tersebut dapat disalurkan ke dalam lapisan dense. Output dari lapisan BiLSTM berupa tensor dengan dimensi yang bervariasi, sementara lapisan dense membutuhkan input dalam bentuk tensor datar. Oleh karena itu, flatten layer bekerja dengan mengubah tensor multidimensi menjadi tensor satu dimensi dengan melebarkan gulungan semua elemennya (Tan & Lee, 2023).

2.2.10 Fully Connected Layer

Fully connected layer atau dapat disebut juga *Dense Layer* menerima fitur keluaran BiLSTM sebagai masukan, dan kemudian mengeluarkannya ke lapisan keluaran akhir setelah pemrosesan yang komprehensif. Setiap neuron di *Fully connected layer* terhubung ke semua neuron di lapisan sebelumnya untuk mengintegrasikan fitur. lapisan keluaran memilih Dense sebagai pengklasifikasi, yang mengeluarkan hasil gen esensial atau gen nonesensial (Intelligence and Neuroscience, 2023).



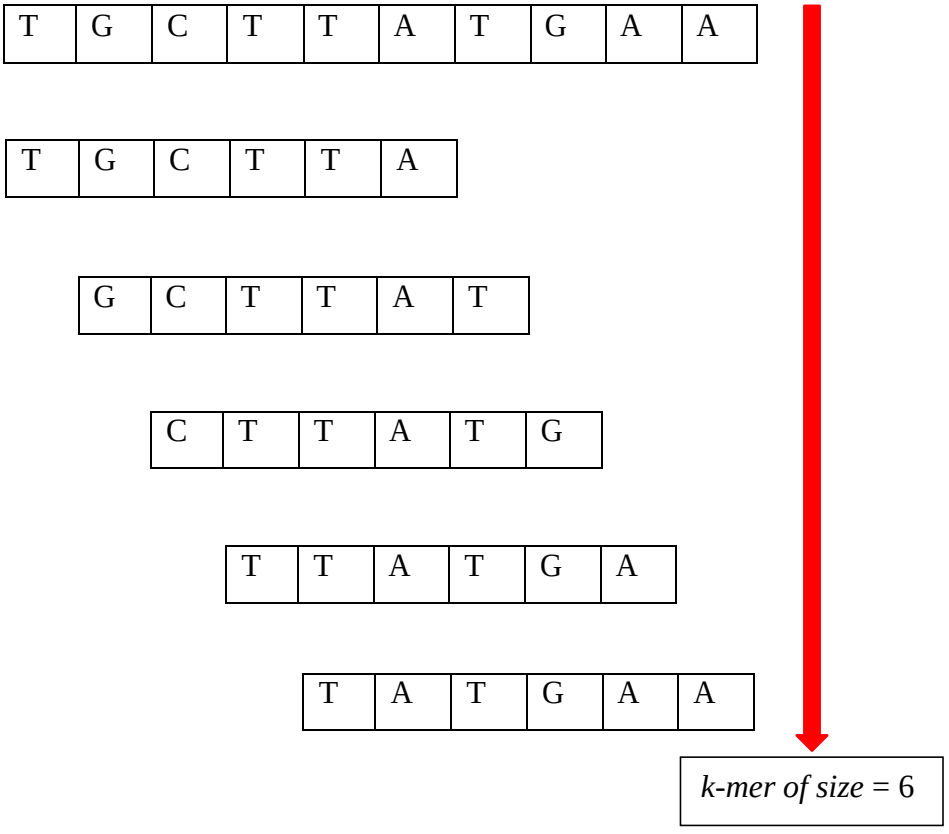
Gambar 3. *Fully Connected Layer* (Cheng et al., 2023).

2.2.11 Random Undersampling

Pengambilan sampel secara acak juga merupakan metode non-heuristic yang bertujuan untuk menyeimbangkan distribusi kelas melalui random penghapusan contoh kelas mayoritas (Batista et al., 2004). Random undersampling (RUS) adalah metode untuk mengatasi ketidakseimbangan kelas dengan menghapus beberapa contoh dari kelas mayoritas. Penghapusan ini dilakukan tanpa penggantian, artinya setelah contoh dihapus, mereka tidak akan dimasukkan kembali. Untuk menggambarkan cara kerja RUS, kita dapat melihat contoh dengan rasio 1:1. Misalnya, dalam satu dataset, terdapat 880 contoh gen esensial (kelas positif) dan 7433 contoh gen non esensial (kelas negatif). Tujuan RUS adalah mengurangi jumlah contoh dalam kelas mayoritas (kelas negatif) sehingga jumlahnya sama dengan kelas minoritas, yaitu 880. Ini dicapai dengan menghapus secara acak contoh-contoh dari kelas negatif hingga hanya tersisa 880 contoh, sehingga tercapai rasio 1:1. Menerapkan RUS dengan rasio 1:1 pada dataset ini berarti mengurangi jumlah contoh dalam kelas negatif secara drastis dari 7433 menjadi 880. Penting untuk dicatat bahwa RUS hanya diterapkan pada data pelatihan, bukan pada data pengujian. Contoh ini hanya untuk memberikan gambaran sederhana tentang cara kerja RUS. (Richard Zuech, 2021).

2.2.12 K-mer Tokenization

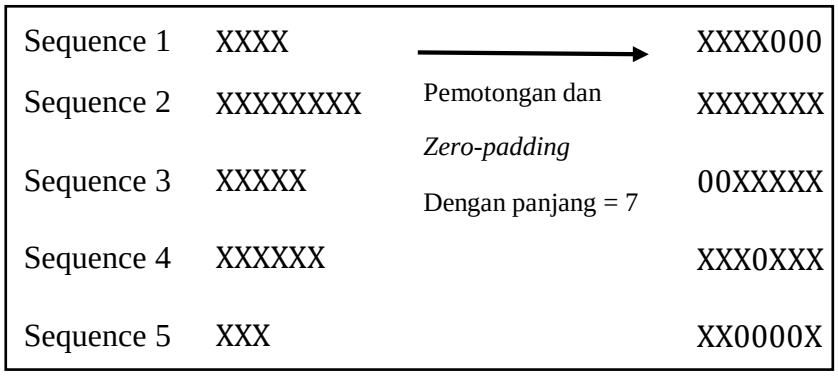
Proses tokenisasi *K-mer* melibatkan pembagian urutan DNA menjadi rangkaian berikutnya, atau yang dikenal sebagai 'k-mers,' dengan menggunakan mekanisme jendela geser. Di sini, 'k' mewakili ukuran jendela, yang menentukan panjang setiap jendela secara berurutan. Dalam kerangka ini, terdapat dua pendekatan umum, yaitu tumpang tindih yang diterapkan oleh *DNABERT*, dan tokenisasi tanpa tumpang tindih yang digunakan oleh *Nucleotide Transformer*. Untuk menggambarkan hal ini, mari kita ambil contoh urutan DNA 'ATGACG' dan lakukan tokenisasi menggunakan pendekatan 3-mer. Dengan menggunakan tumpang tindih, kita mendapatkan token ATG, TGA, GAC, dan ACG. Sebaliknya, dengan tidak menggunakan tumpang tindih, token yang dihasilkan hanya ATG dan ACG (Liang, et al., 2023)



Gambar 4. *k-mer Tokenization* (Gunasekaran et al., 2021).

2.2.13 Padding

Setiap rangkaian DNA pada dataset memiliki panjang yang berbeda, sedangkan semua vektor masukkan harus berukuran sama untuk dimasukkan kedalam model BiLSTM. Untuk mengatasi masalah ini, diterapkan pemotongan dan *padding*. Dengan menerapkan *padding* berarti menetapkan panjang yang sama untuk semua sekuens DNA pada dataset, kemudian memotong rangkaian DNA yang lebih panjang dari padding yang telah ditetapkan. Proses menyelesaikan suatu urutan ini disebut padding dan karakter yang digunakan untuk melengkapi rangkaian DNA bisa berupa apa saja yang tidak digunakan dalam urutan itu sendiri. Dalam hal ini, karakter nol (0) adalah yang paling umum digunakan. Padding nol dapat ditambahkan pada posisi mana pun pada rangkaian, tetapi pada umumnya padding nol dapat ditambahkan pada akhir rangkaian DNA. Contoh dari pemotongan dan padding nol dapat dilihat pada gambar 6 (Lopez-del Rio et al., 2020).



Gambar 5. *Zero Padding* (Lopez-del Rio et al., 2020).

2.2.14 Overfitting

Overfitting adalah kondisi ketika sebuah model terlalu cocok dengan data latih sehingga tidak mampu bekerja dengan baik pada data baru yang diambil dari distribusi yang sama. Ini terjadi karena pola yang dipelajari dari data latih tidak mewakili populasi secara umum.

Secara sederhana, model yang overfitting adalah model yang terlalu kompleks dibandingkan dengan model ideal yang sesuai untuk data dan masalah yang dihadapi. Beberapa penulis juga mendefinisikan overfitting sebagai kondisi ketika model mempelajari "noise" atau kebisingan dalam data, yaitu pola yang hanya ada dalam data latih dan tidak terdapat dalam populasi umum. Selain itu, overfitting juga bisa terjadi pada metode, protokol pemodelan, dan sistem dalam machine learning atau kecerdasan buatan, di mana metode atau sistem tersebut cenderung menghasilkan model yang overfitting terhadap data (Simon & Aliferis, 2024).

2.2.15 Underfitting

Underfitting adalah kondisi ketika sebuah model tidak mampu merepresentasikan data latih dengan baik dan juga gagal memberikan kinerja yang baik pada populasi umum. Secara lebih luas, model yang underfitting akan memiliki kesalahan generalisasi yang lebih besar dibandingkan dengan kesalahan generalisasi dari model terbaik yang mungkin bisa dibuat dengan data yang ad (Simon & Aliferis, 2024).

2.2.16 Dropout

Dalam konteks pembelajaran mesin, dropout mengacu pada teknik yang digunakan untuk mengatasi over-fitting. Over-fitting terjadi ketika model terlalu kompleks dan terlalu cocok dengan data pelatihan, yang dapat menghasilkan kinerja yang buruk pada data baru. Ini adalah masalah umum dalam machine learning di mana model tampaknya berkinerja baik pada data pelatihan, tetapi tidak pada data yang tidak terlihat. Untuk mengatasi over-fitting, dropout secara acak mengabaikan sebagian unit (neuron) dalam jaringan selama proses pelatihan, sehingga mendorong model untuk menjadi

lebih general dalam mempelajari pola dari data, bukan hanya "menghafal" data pelatihan tertentu (Xie, 2020).

2.2.17 Performance Metrics

Untuk mengukur *performance metrics*, dibutuhkan *confusion matrix* yang merupakan tabel pengukuran metrik kinerja algoritma klasifikasi. *Confusion matrix* Merupakan tabel yang menunjukkan sejumlah tindakan yang terdeteksi dengan benar dan salah (Al-shehari & Alsowail, 2021).

Tabel 3. *Confusion Matrix* (Monaghan et al., 2021).

	Actual		
Assignment	TP	FP	PPV = TP/TP+FP
	FN	TN	NPV = TN/TN+FN
	Sensitivity = TP/TP+FN	Specificity = TN/TN+FP	

Tabel 4 merupakan *Confusion Matrix* yang terdiri dari empat dasar kategori yaitu *True Positive* (TP), *False Positive* (FP), *False Negative* (FN), *True Negative* (TN). Pada penelitian ini metrik kinerja dari *Bidirectional Long Short Term Memory* (BiLSTM) dievaluasi menggunakan *sensitivity*, *specificity*, ROC-AUC, dan PR-AUC. Untuk memahami mengenai metrik kinerja algoritma klasifikasi, disajikan ringkasan arti setiap metrik pada Tabel 5.

Tabel 4. Ringkasan Arti Setiap Metrik (Erickson & Kitamura, 2021).

Metrik	Arti	Sinonim	Rumus
Sensitivity	Sebagian kecil kasus positif yang diprediksi positif	*Recall*, *true-positive rate*	*Sensitivity* = TP / TP+FN

Specificity	Sebagian kecil kasus negative yang diprediksi negatif	*Selectivity, true-negative rate*	*Specificity* = TN / TN+FP
False-positive rate (FPR)	Sebagian kecil kasus yang diprediksi positif, sebenarnya negatif	*Fall-out, probability of false alarm*	FPR = FP / TN+FP Atau = 1- *Specificity*
False-negative rate (FNR)	Sebagian kecil kasus yang diprediksi negatif, sebenarnya positif	*Miss rate*	FNR = FN / TP+FN
Positive-predictive value (PPV)	Sebagian kecil kasus yang benar-benar positif dari seluruh kasus diprediksi positif oleh model	*Precision*	PPV = TP / TP+FP
Negative-predictive value (NPV)	Sebagian kecil kasus yang benar-benar negatif dari seluruh kasus diprediksi negatif oleh model	*None*	NPV = TN / TN+FN
Accuracy	Sebagian kecil kasus diprediksi benar oleh model	*None*	Acc = TP+TN / TP+FN+TN+FP
F1 Score	Rata-rata harmonik dari *Positive-predictive value* dan *Sensitivity*	*F score, F measure, Dice similarity coefficient*	F1 = 2TP / 2TP+FP+FN

2.2.17.1 Sensitivity

Sensitivity mengukur keefektifan suatu tes dalam mengenali hasil positif yang sesungguhnya, yaitu seberapa baik tes tersebut dapat mengklasifikasikan subjek yang memang memiliki kondisi yang sedang diuji. Dengan kata lain, sensitivitas menilai proporsi subjek dengan *actual positive outcome* (TP + FN) yang secara akurat

diidentifikasi sebagai *positive assignment* (TP) (Monaghan et al., 2021). Dapat dilihat pada persamaan 19.

Sensitivity = TP / TP+FN ..... (19)

2.2.17.2 Specificity

Specificity mengukur seberapa baik suatu tes mengidentifikasi negatif sebenarnya, yaitu, seberapa baik suatu tes dapat mengklasifikasikan subjek yang benar-benar tidak mempunyai kondisi yang diminati. Sebagai alternatif, spesifisitas mengukur proporsi subjek dengan *actual negative outcome* (TN+FP) yang diidentifikasi sebagai *negative assignment* (TN) (Monaghan et al., 2021). Dapat dilihat pada persamaan 20.

Specificity = TN / TN+FP ..... (20)

2.2.17.3 ROC-AUC

ROC-AUC (*Receiver Operating Characteristic Area Under the Curve*) berfungsi sebagai representasi grafis yang secara efektif menggambarkan hal yang melekat *trade-off* antara *sensitivity* dan *false positive rate* (FPR) pada ambang klasifikasi yang berbeda. Jika kurva ROC menunjukkan kenaikan yang curam selama tahap awal, maka menandakan *true positive rate* tinggi dengan mempertahankan *false positive rate* rendah. Hal ini menunjukkan kemampuan yang luar biasa dalam mengklasifikasikan kasus-kasus positif secara akurat dengan meminimalkan terjadinya *False Positive* (Monaghan et al., 2021). Dapat dilihat pada persamaan 21 dan 22.

Sensitivity = TP / TP+FN ..... (21)

False-positive rate = FP / TN+FP ..... (22)

2.2.17.4 PR-AUC

PR-AUC (*Precision-Recall Area Under the Curve*) menawarkan perspektif yang berbeda dan mendetail tentang bagaimana pengklasifikasi dengan baik mengelola keseimbangan antara *Positive-predictive value* dan *sensitivity* di berbagai ambang klasifikasi. PR-AUC menggambarkan *trade-off* antara *precision* dan *recall*. Ambang batas (*Threshold*) menentukan titik tertentu pada kurva PR, namun tidak mengubah kurva itu sendiri. Area di bawah kurva PR-AUC dapat dihitung dengan skor presisi rata-rata, dan juga tidak dipengaruhi oleh ambang batas (Erickson & Kitamura, 2021). Dapat dilihat pada persamaan 23 dan 24.

Sensitivity = TP / TP+FN ..... (23)

Positive-predictive value = TP / TP+FP ..... (24)

III. METODOLOGI PENELITIAN

3.1 Waktu dan Tempat

Waktu dan tempat membentuk landasan peristiwa atau perjalanan yang berkelanjutan dalam proses penelitian. Berikut ini dijabarkan waktu dan tempat penelitian berlangsung.

3.1.1 Tempat Penelitian

Tempat penelitian dilaksanakan di Laboratorium Komputasi Dasar, Jurusan Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Lampung.

3.1.2 Waktu Penelitian

Penelitian yang dilakukan terhitung dari Bulan November 2023 pada semester ganjil sampai dengan Bulan Juli 2024 pada semester genap, dengan beberapa tahapan, yaitu pengumpulan data, prapemprosesan data, model klasifikasi, pengujian dan evaluasi. Berikut ini dijabarkan waktu penelitian yang dapat dilihat pada Tabel 5.

Tabel 5. Waktu Penelitian.

No	Kegiatan	2023												2024																							
		November				Desember				Januari				Februari				Maret				April				Mei				Juni				Juli			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	Studi literatur																																				
2	Pengumpulan data																																				
3	Penyusunan bab I-III																																				
4	*Cleaning* data																																				
5	*Kmer Tokenization*																																				
6	*Padding*																																				
7	*Data Splitting*																																				
8	*Random Undersampling*																																				
9	*K-fold cross validation*																																				
10	Pemodelan BiLSTM																																				
11	Evaluasi metrik kinerja																																				
12	Penyusunan bab IV-V																																				

3.2 Data dan Alat

3.2.1 Data

Data yang digunakan pada penelitian ini bersumber dari jurnal *Identifying essential genes across eukaryotes by machine learning* (Beder et al., 2021). Dataset yang digunakan adalah DNA lalat buah yang berjumlah 12.064 dibagi menjadi data CEG (*Cellular Essential Genes*) berjumlah 11547 dengan data *Essential gene* berjumlah 1227 dan data *Non Essential gene* berjumlah 10320. Data OEG (*Organismal Essential Genes*) berjumlah 517 dengan data *Essential gene* berjumlah 246 dan data *Non Essential gene* berjumlah 271 yang setiap sequence memiliki panjang yang berbeda-beda. Untuk panjang minimal data pada dataset yaitu 211 dan panjang maksimal data yaitu 394.150. Dataset terbagi menjadi dua yaitu data latih dan data uji. Data latih dibagi lagi menjadi dua bagian untuk traning dan validasi.

3.2.2 Perangkat Keras (*Hardware*)

Perangkat keras yang digunakan pada penelitian menggunakan laptop dengan spesifikasi sebagai berikut.

a) *Processor* : AMD® Ryzen 5 3400g with radeon vega graphics × 8
b) RAM : 16 GB
c) *System Type* : 64 bit
d) *Storage* : 500 GB
e) *Graphics* : NVIDIA Corporation GK210GL [Tesla K80]

3.2.3 Perangkat Lunak (*Software*)

Berikut adalah beberapa perangkat lunak yang digunakan untuk mendukung penelitian, diantara yaitu:

A. Operating System : *Linux (Ubuntu)*

B. Tools

a) Jupyter Notebook

Jupyter Notebook adalah aplikasi web yang memungkinkan pembuatan dan berbagi dokumen komputasi. Aplikasi ini menawarkan pengalaman yang sederhana, efisien, dan berpusat pada dokumen. *JupyterLab*, sebagai lingkungan pengembangan interaktif berbasis web terbaru, menyediakan fasilitas untuk bekerja dengan buku catatan, kode, dan data. Antarmukanya yang fleksibel memungkinkan pengguna untuk mengonfigurasi dan mengatur alur kerja dalam bidang ilmu data, komputasi ilmiah, jurnalisme komputasi, dan pembelajaran mesin. Dengan desain modular, *JupyterLab* memungkinkan ekstensi untuk memperluas dan memperkaya fungsionalitasnya. (jupyter.org, 2024)

b) *Python* versi 3.10.12

Python adalah bahasa pemrograman tingkat tinggi yang ditafsirkan, berorientasi objek, dengan semantik dinamis. Struktur data bawaan tingkat tinggi, dikombinasikan dengan pengetikan dinamis dan pengikatan dinamis, membuatnya sangat menarik untuk Pengembangan Aplikasi Cepat, serta untuk digunakan sebagai bahasa skrip atau perekat untuk menghubungkan komponen-komponen yang ada bersama-sama. Sintaks *Python* yang sederhana dan mudah dipelajari menekankan keterbacaan dan karenanya mengurangi biaya pemeliharaan program. *Python* mendukung modul dan paket, yang mendorong modularitas program dan penggunaan kembali kode (python.org, 2023).

C. Packages

a) *Scikit-learn* versi 1.3.2

Scikit-learn merupakan perpustakaan paling berguna dan tangguh untuk pembelajaran mesin dengan *Python*. Ini menyediakan pilihan alat yang efisien untuk pembelajaran mesin dan pemodelan statistik termasuk klasifikasi, regresi, pengelompokan, dan pengurangan dimensi melalui antarmuka konsistensi dengan *Python*. Perpustakaan ini, yang sebagian besar ditulis dengan *Python*, dibangun di atas *NumPy*, *SciPy* dan *Matplotlib*. Awalnya *scikit-learn* dikembangkan oleh David Cournapeau sebagai proyek kode musim panas Google pada tahun 2007. Kemudian, pada tahun 2010, Fabian Pedregosa, Gael Varoquaux, Alexandre Gramfort, dan Vincent Michel, dari FIRCA (Institut Penelitian Prancis di bidang Ilmu Komputer dan Otomasi), membawa proyek ini ke tingkat yang lebih tinggi dan membuat rilis publik pertama (v0.1 beta) pada tanggal 1 Februari 2010 (tutorialspoint, 2023).

b) *TensorFlow* versi 2.8.0

TensorFlow adalah perpustakaan sumber terbuka yang awalnya dikembangkan oleh Google, terutama untuk aplikasi pembelajaran mendalam, dan juga mendukung pembelajaran mesin tradisional. Awalnya *TensorFlow* dirancang untuk komputasi numerik besar tanpa mempertimbangkan pembelajaran mendalam, *TensorFlow* akhirnya terbukti sangat bermanfaat dalam pengembangan pembelajaran mendalam. Oleh karena itu, Google memutuskan untuk menjadikannya proyek sumber terbuka. *TensorFlow* menerima data dalam bentuk array multidimensi dengan dimensi lebih tinggi yang disebut tensor. Array multidimensi sangat berguna dalam menangani data dalam jumlah besar *TensorFlow* bekerja berdasarkan

grafik aliran data yang terdiri dari *node* dan *edge.* Dengan mekanisme eksekusi yang berbasis grafik, pelaksanaan kode *TensorFlow* secara terdistribusi di sekelompok komputer menjadi lebih efisien, terutama ketika menggunakan unit pemrosesan grafis (GPU) (Simplilern, 2023).

c) *Pandas* versi 1.5.3

Pandas adalah alat analisis dan manipulasi data sumber terbuka yang cepat, kuat, fleksibel dan mudah digunakan, dibangun di atas bahasa pemrograman *Python. Pandas* bertujuan untuk menjadi landasan dasar tingkat tinggi untuk melakukan analisis data dunia nyata yang praktis dengan *Python.* Selain itu, ia memiliki tujuan yang lebih luas untuk menjadi alat analisis/manipulasi data sumber terbuka paling kuat dan fleksibel yang tersedia dalam bahasa apa pun. Pandas sebagai proyek sumber terbuka kelas dunia (pypi.org, 2023).

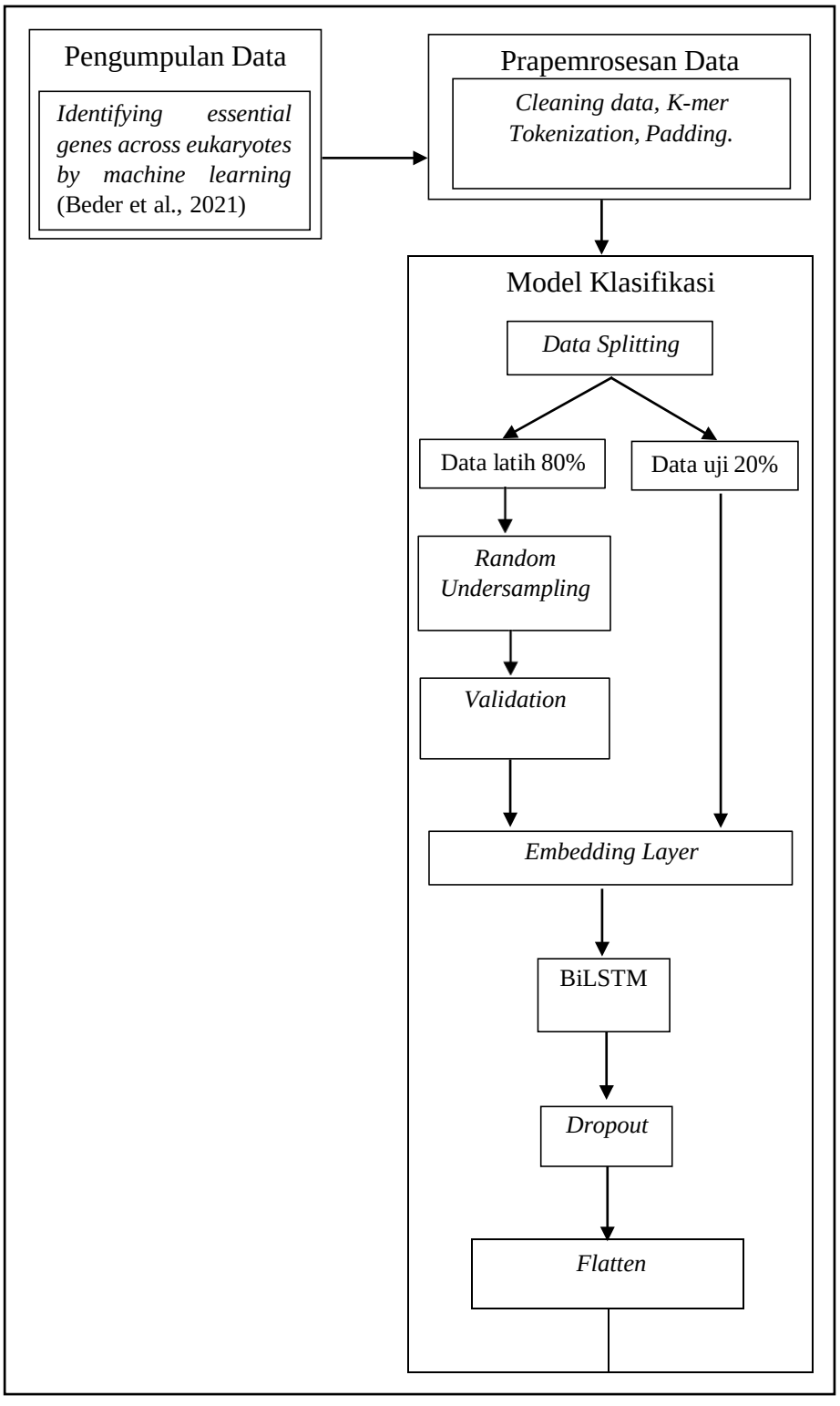
d) *NumPy* versi 1.23.5

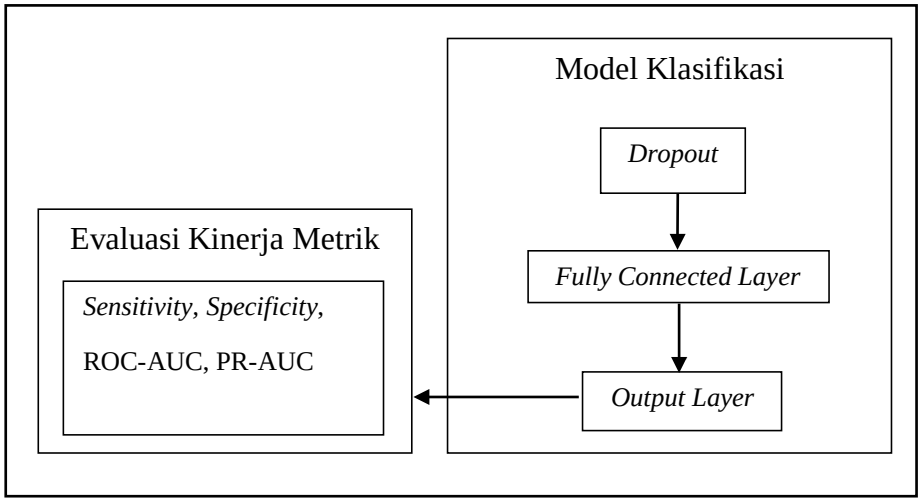
NumPy merupakan proyek sumber terbuka, yang memungkinkan eksekusi komputasi numerik menggunakan bahasa pemrograman *Python.* Diluncurkan pada tahun 2005 sebagai hasil pengembangan dari perpustakaan Numerik dan *Numarray*, *NumPy* berfungsi sebagai fondasi utama dalam komputasi ilmiah dengan *Python.* Sebagai sebuah pustaka *Python, NumPy* menawarkan kumpulan objek array multidimensi, berbagai turunan objek (seperti *masked arrays* dan matriks), serta beragam rutinitas untuk melakukan operasi cepat pada array. Fungsionalitas ini mencakup aspek matematika, logika, manipulasi bentuk, pengurutan, pemilihan, I/O, transformasi Fourier diskrit, aljabar linier dasar, operasi statistik dasar, simulasi acak, dan sejumlah fungsi lainnya. Inti dari *NumPy* terletak pada objek ndarray,

yang mewakili array n-dimensi dengan tipe data homogeny (numpy.org, About us, 2023).

3.3 Metodologi Penelitian

Metodologi penelitian berguna untuk mengetahui tahapan-tahapan yang akan dilakukan selama proses penelitian. Berikut metodologi penelitian yang dilakukan dapat dilihat sebagai berikut pada Gambar 6.





Gambar 6. Diagram Penelitian.

3.3.1 Pengumpulan data

Sumber data yang diperoleh selama penelitian hanya terdiri dari data sekunder. Data sekunder didapat dari studi literatur untuk mencari sumber informasi yang berhubungan dengan penelitian Klasifikasi Sekuens DNA untuk penilaian Esensialitas Gen pada Lalat Buah (*Drosophila Melanogaster)* menggunakan metode *Bidirectional Long Short Term Memory* (BiLSTM). Dataset yang digunakan pada penelitian ini bersumber dari penelitian terdahulu yaitu penelitian oleh (Beder et al., 2021), dengan menggunakan dataset fasta dan dataset csv dari sekuens DNA *Drosophila Melanogaster*.

3.3.2 Prapemrosesan Data

Pemrosesan awal data melibatkan beberapa tahapan untuk memastikan keakuratan analisis. Langkah pertama adalah melakukan *cleaning data* untuk menangani nilai yang hilang dalam kumpulan data, untuk menghindari potensi distorsi hasil. Setelah itu, dilakukan proses *Tokenization* menggunakan k-mer tokenization dengan panjang k = 5 dan 7. Selanjutnya dilakukan Pemotongan dan *Padding* untuk menyamakan jumlah atau panjang rangkaian DNA pada dataset.

3.3.3 Model Klasifikasi

Data splitting merupakan proses di mana dataset dibagi menjadi subset pelatihan dan pengujian. Pembagian ini digunakan untuk mengukur kemampuan model dalam generalisasi pada data yang tidak digunakan selama pelatihan. *Validation* digunakan untuk menguji dan mengevaluasi kinerja model dengan membagi dataset menjadi data pelatihan untuk melatih model dan data validasi untuk menguji model. Hal ini dilakukan untuk memastikan hasil evaluasi lebih representatif dan mengurangi risiko bias yang terjadi dalam proses evaluasi model. Untuk menangani ketidakseimbangan kelas dalam dataset Ini melibatkan strategi menggunakan *Random Undersampling* untuk memastikan setiap kelas memiliki representasi yang cukup, sehingga model dapat belajar tanpa terpengaruh oleh dominasi kelas mayoritas atau minoritas. Setelah itu, beralih ke proses *embedding layer* yaitu mengkonversi data berdimensi tinggi menjadi data berdimensi rendah dengan mengubah kata unik dalam kumpulan data menjadi nilai nyata vektor fitur yang ditumpuk menjadi satu membentuk matriks. Model arsitektur yang digunakan yaitu *Bidirectional Long Short Term Memory* (BiLSTM) karena memiliki kemampuan lebih baik untuk meningkatkan performa dalam proses klasifikasi. Dalam konteks pembelajaran mesin, *Dropout* mengacu pada teknik yang digunakan untuk mengatasi over-fitting. Over-fitting terjadi ketika model terlalu kompleks dan terlalu cocok dengan data pelatihan, yang dapat menghasilkan kinerja yang buruk pada data baru. *Flatten layer* bekerja dengan mengubah tensor multidimensi menjadi tensor satu dimensi dengan melebarkan gulungan semua elemennya agar dapat dapat disalurkan ke dalam lapisan dense. Proses terakhir yaitu *fully connected layer* atau *dense layer* yang digunakan untuk proses output dari klasifikasi BiLSTM.

3.3.4 Evaluasi Kinerja Metrik

Setelah proses pemodelan selesai, langkah selanjutnya adalah melakukan evaluasi kinerja metrik menggunakan berbagai metrik kinerja klasifikasi. Beberapa metrik yang diambil sebagai tolak ukur adalah *sensitivity, specificity*, ROC-AUC, dan PR-AUC. Evaluasi ini penting untuk memahami sejauh mana model dapat memprediksi dengan akurat dan memberikan wawasan yang mendalam tentang performa klasifikasi pada data yang digunakan.

V. SIMPULAN DAN SARAN

5.1 SIMPULAN

Penelitian ini bertujuan untuk mengklasifikasikan gen esensial pada sekuens DNA lalat buah (*Drosophila melanogaster*) menggunakan metode *Bidirectional Long Short Term Memory* (BiLSTM). Berdasarkan hasil yang diperoleh, beberapa kesimpulan utama dapat diambil sebagai berikut.

1. Penelitian ini menggunakan skema pembagian dataset yang mencakup data pelatihan 80% dan data pengujian 20%, dengan data pelatihan 80% dibagi lagi menjadi data pelatihan dan data validasi. Skema pembagian optimal pada data pelatihan 90% dan data validasi 10%.
2. Hasil klasifikasi menunjukkan bahwa untuk dataset OEG, skema pembagian data pelatihan 90% dan data validasi 10% memberikan hasil terbaik dengan nilai pengujian yang diperoleh adalah *specificity* 73%, *sensitifity* 80%, *ROC-AUC* 76%, *PR AUC* 81%. Sedangkan untuk dataset CEG, skema yang sama menghasilkan nilai pengujian yang diperolah adalah *specificity* 70%, *sensitifity* 64%, *ROC-AUC* 67%, *PR AUC* 46%.
3. Hasil perbandingan dengan penelitian pada jurnal (Beder et al., 2021), menunjukkan beberapa nilai yang lebih tinggi dan beberapa nilai yang lebih rendah. Hal ini menunjukkan bahwa Implementasi BiLSTM pada sekuens DNA lalat buah menunjukkan potensi yang baik, tetapi masih memerlukan pengoptimalan lebih lanjut untuk mencapai kinerja yang lebih baik, dengan melakukan eksperimen

lebih lanjut terkait hyperparameter maupun parameter yang digunakan, sehingga diharapkan akan menghasilkan hasil terbaik.

5.2 SARAN

Berdasarkan pembahasan mengenai implementasi BiLSTM yang telah dilakukan, terdapat beberapa hal yang dapat ditingkatkan, yaitu sebagai berikut.

1. Untuk meningkatkan hasil klasifikasi, dapat dilakukan perbandingan dengan metode klasifikasi lainnya seperti DNABERT, BioBERT, atau *Genome-scale Transformer*. Penelitian lebih lanjut dengan model-model ini dapat memberikan wawasan yang lebih mendalam tentang kinerja dan efektivitas berbagai pendekatan dalam klasifikasi gen esensial.
2. Penelitian ini dapat diperluas dengan mengaplikasikan teknik-teknik penanganan ketidakseimbangan data lainnya. Metode seperti oversampling dan undersampling dapat diterapkan pada dataset CEG untuk mengevaluasi pengaruhnya terhadap kinerja model. Pendekatan ini diharapkan dapat memperbaiki distribusi data dan meningkatkan hasil klasifikasi.

DAFTAR PUSTAKA

Abd –Alhalem, S. M., El-Rabaie, E.-S. M., Soliman, N. F., Abdulrahman, S. S., Ismail, N. A., & Abd El-samie, F. E. (2021). *DNA Sequences Classification with Deep Learning: A Survey*, 41–51.

Al-shehari, T., & Alsowail, R. A. (2021). An insider data leakage detection using one-hot encoding, synthetic minority oversampling and machine learning techniques. *Entropy*, 1–24.

Aldhyani, T. H. H., & Alkahtani, H. (2021). A bidirectional long short-term memory model algorithm for predicting covid-19 in gulf countries. *Life*, *11*(11), 1–26.

Alharbi, F. R., & Csala, D. (2021). Wind speed and solar irradiance prediction using a bidirectional long short-term memory model based on neural networks. *Energies*, 1–22.

Aromolaran, O., Beder, T., Oswald, M., Oyelade, J., Adebiyi, E., & Koenig, R. (2020). Essential gene prediction in Drosophila melanogaster using machine learning approaches based on sequence and functional features. *Computational and Structural Biotechnology Journal*, *18*, 612–621.

Aromolaran, O., Oyelade, J., & Adebiyi, E. (2021). Performance evaluation of features for gene essentiality prediction. *IOP Conference Series: Earth and Environmental Science*, 1–14.

Batista, G. E. A. P. A., Prati, R. C., & Monard, M. C. (2004). A study of the behavior of several methods for balancing machine learning training data. *ACM SIGKDD Explorations Newsletter*, *6*(1), 20–29.

Beder, T., Aromolaran, O., Dönitz, J., Tapanelli, S., Adedeji, E. O., Adebiyi, E., Bucher, G., & Koenig, R. (2021). Identifying essential genes across eukaryotes by machine learning. *NAR Genomics and Bioinformatics*, *3*(4), 1–13.

Cheng, X., Tang, H., Wu, Z., & Liang, D. (2023). *applied sciences BILSTM-Based Deep Neural Network for Rock-Mass Classification Prediction Using Depth-Sequence MWD Data : A Case Study of a Tunnel in Yunnan , China*, 1– 20.

Chola, C., Benifa, J. V. B., Guru, D. S., Muaad, A. Y., Hanumanthappa, J., Al-Antari, M. A., Alsalman, H., & Gumaei, A. H. (2022). Gender Identification and Classification of Drosophila melanogaster Flies Using Machine Learning Techniques. *Computational and Mathematical Methods in Medicine*, *2022*, 1–9.

El-Tohamy, A., Maghwary, H. A., & Badr, N. (2022). A Deep Learning Approach for Viral DNA Sequence Classification using Genetic Algorithm. *International Journal of Advanced Computer Science and Applications*, *13*(8), 530–538.

Erickson, B. J., & Kitamura, F. (2021). Magician's corner: 9. performance metrics for machine learning models. In *Radiology: Artificial Intelligence* (Vol. 3, Issue 3). Radiological Society of North America Inc, 1–7.

Gunasekaran, H., Ramalakshmi, K., Rex Macedo Arokiaraj, A., Kanmani, S. D., Venkatesan, C., & Dhas, C. S. G. (2021). Analysis of DNA Sequence Classification Using CNN and Hybrid Models. *Computational and Mathematical Methods in Medicine*, *2021*(C), 1–12.

Hamed, B. A., Ibrahim, O. A. S., & Abd El-Hafeez, T. (2023). Optimizing classification efficiency with machine learning techniques for pattern matching. *Journal of Big Data*, *10*(1), 1–18.

Intelligence and Neuroscience, C. (2023). Retracted: Emotion Analysis Model of Microblog Comment Text Based on CNN-BiLSTM. *Computational Intelligence and Neuroscience*, *2023*, 1–10.

Isenmann, M., Stoddart, M. J., Schmelzeisen, R., Gross, C., Bella , E. D., & Rothweiler, R. M. (2023). Basic Principles of RNA Interference: Nucleic Acid Types and In Vitro Intracellular Delivery Methods. *micromachines*, 1–22.

Liang, C., Bai, W., Qiao, L., Ren, Y., Sun, J., Ye, P., Yan, H., MA, X., Zhuo, W., & Ouyang, W. (2023). RETHINKING THE BERT-LIKE PRETRAINING FOR DNA SEQUENCES. 1-13.

Luo, H., Lin, Y., Liu, T., Lai, F.-L., Zhang, C.-T., Gao, F., & Zhang, R. (2020). DEG15, an update of the Database of Essential Genes that includes built-in analysis tools. *Nucleic Acids Research*, 677-686.

Lopez-del Rio, A., Martin, M., Perera-Lluna, A., & Saidi, R. (2020). Effect of sequence padding on the performance of deep learning models in archaeal protein functional prediction. *Scientific Reports*, *10*(1), 1–14.

Monaghan, T. F., Rahman, S. N., Agudelo, C. W., Wein, A. J., Lazar, J. M., Everaert, K., & Dmochowski, R. R. (2021). Foundational statistical principles in medical research: Sensitivity, specificity, positive predictive value, and negative predictive value. *Medicina (Lithuania)*, *57*(5), 1–6.

Nandhini, K., & Tamilpavai, G. (2023). An Optimal Stacked ResNet-BiLSTM-Based Accurate Detection and Classification of Genetic Disorders. *Neural Processing Letters*, *55*(7), 9117–9138.

Pang, X., Zhou, Y., Wang, P., Lin, W., & Chang, V. (2020). An innovative neural network approach for stock market prediction. *Journal of Supercomputing*, *76*(3), 2098–2118.

Richard Zuech, J. H. (2021). Detecting web attacks using random undersampling and ensemble learners. *journal of big data*, 11, 1–20.

Riddle, N. C. (2019). Drosophila melanogaster, a new model for exercise research. In *Acta Physiologica* (Vol. 227, Issue 3). Blackwell Publishing Ltd, 1–3.

Simon, G., & Aliferis, C. (2024). *From "Human versus Machine" to "Human with Machine" BT - Artificial Intelligence and Machine Learning in Health Care and Medical Sciences: Best Practices and Pitfalls*, 480-484.

Smagulova, K., & James, A. P. (2020). Overview of long short-term memory neural networks. In *Modeling and Optimization in Science and Technologies* (Vol.

14). Springer International Publishing, 139–153 .

Tan, K. L., & Lee, C. P. (2023). *applied sciences RoBERTa-GRU : A Hybrid Deep Learning Model for Enhanced Sentiment Analysis*, 1–16.

Tavakoli, N. (2019). Modeling genome data using bidirectional LSTM. *Proceedings - International Computer Software and Applications Conference*, *2*(October), 183–188.

Xie, J. (2020). A Novel Method of Music Generation Based on Three Different Recurrent Neural Networks. *Journal of Physics: Conference Series*, *1549*(4), 1–8.

Xu, Y., & Li, Z. (2020). CRISPR-Cassystems : Overview, innovations and applications in human disease research and gene therapy. *Computational and Structural Biotechnology Journal*, 2402–2415.

Zhang, X., Xiao, W., & Xiao, W. (2020). DeepHE: Accurately predicting human essential genes based on deep learning. *PLoS Computational Biology*, *16*(9), 1–17.