

**PREDIKSI KEBERLANJUTAN RAWAT INAP PASIEN RUMAH SAKIT  
MENGUNAKAN XLNET-BIGRU-ATT PADA CATATAN KHUSUS  
KLINIS PASIEN**



**ANNISA DARWIS  
D121201032**



**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS TEKNIK  
UNIVERSITAS HASANUDDIN  
MAKASSAR  
2024**

**PREDIKSI KEBERLANJUTAN RAWAT INAP PASIEN RUMAH SAKIT  
MENGUNAKAN XLNET-BIGRU-ATT PADA CATATAN KHUSUS  
KLINIS PASIEN**

**ANNISA DARWIS  
D121 20 1032**



**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS TEKNIK  
UNIVERSITAS HASANUDDIN  
MAKASSAR  
2024**

**PREDIKSI KEBERLANJUTAN RAWAT INAP PASIEN RUMAH SAKIT  
MENGUNAKAN XLNET-BIGRU-ATT PADA CATATAN KHUSUS  
KLINIS PASIEN**

ANNISA DARWIS

D121 20 1032

Skripsi

sebagai salah satu syarat untuk mencapai gelar sarjana

Program Studi Teknik Informatika

pada

**PROGRAM STUDI TEKNIK INFORMATIKA  
DEPARTEMEN TEKNIK INFORMATIKA  
FAKULTAS TEKNIK  
UNIVERSITAS HASANUDDIN  
MAKASSAR  
2024**

**SKRIPSI**

**PREDIKSI KEBERLANJUTAN RAWAT INAP PASIEN RUMAH SAKIT  
MENGUNAKAN XLNET-BIGRU-ATT PADA CATATAN KHUSUS KLINIS  
PASIE**

**ANNISA DARWIS**

**D121201032**

Skripsi,

telah dipertahankan di depan Panitia Ujian Sarjana Teknik Informatika pada tanggal  
bulan tahun dan dinyatakan telah memenuhi syarat kelulusan  
pada

Program Studi Teknik Informatika  
Departemen Teknik Informatika  
Fakultas Teknik  
Universitas Hasanuddin  
Makassar

Mengesahkan:  
Pembimbing tugas akhir,



Anugrayani Bustamin, S.T., M.T.

NIP 19901201 201807 4 001

Mengetahui:  
Ketua Program Studi,



Prof. Dr. Ir. Indrabayu, S. T., M. T., M.  
Bus. Sys., IPM, ASEAN. Eng.

NIP 19750716 200212 1 004

## PERNYATAAN KEASLIAN SKRIPSI DAN PELIMPAHAN HAK CIPTA

Dengan ini saya menyatakan bahwa, skripsi berjudul "Prediksi Keberlanjutan Rawat Inap Pasien Rumah Sakit Menggunakan XLNet-BiGRU-ATT pada Catatan Khusus Klinis Pasien" adalah benar karya saya dengan arahan dari pembimbing ibu Anugrayani Bustamin, S.T., M.T. Karya ilmiah ini belum diajukan dan tidak sedang diajukan dalam bentuk apa pun kepada perguruan tinggi mana pun. Sumber informasi yang berasal atau dikutip dari karya yang diterbitkan maupun tidak diterbitkan dari penulis lain telah disebutkan dalam teks dan dicantumkan dalam Daftar Pustaka skripsi ini. Apabila di kemudian hari terbukti atau dapat dibuktikan bahwa sebagian atau keseluruhan skripsi ini adalah karya orang lain, maka saya bersedia menerima sanksi atas perbuatan tersebut berdasarkan aturan yang berlaku.

Dengan ini saya melimpahkan hak cipta (hak ekonomis) dari karya tulis saya berupa skripsi ini kepada Universitas Hasanuddin.

Makassar, 11 September 2024



## Ucapan Terima Kasih

Assalamualaikum Warahmatullahi Wabarakatuh.

Alhamdulillahirabbil'aalamiin, puji dan syukur penulis panjatkan atas kehadiran Allah Subhanahu wa ta'ala yang telah mencurahkan rahmat serta karunia-Nya, sehingga penulis dapat menyelesaikan tugas akhir ini yang berjudul " Prediksi Keberlanjutan Rawat Inap Pasien Rumah Sakit Menggunakan XLNet-BiGRU-ATT pada Catatan Khusus Klinis Pasien" sebagai salah satu persyaratan yang harus dipenuhi dalam menyelesaikan jenjang Strata-1 pada Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin.

Penulis menyadari bahwa dalam penyusunan dan penulisan laporan skripsi ini tidak lepas dari bantuan, bimbingan serta dukungan dari berbagai pihak, dari masa perkuliahan sampai dengan masa penyusunan skripsi. Oleh karena itu, penulis dengan senang hati menyampaikan terima kasih kepada :

1. Kedua Orang tua penulis, Bapak Darwis dan Ibu Nurlan, yang selalu memberikan dukungan dan doa yang tiada henti, memberi semangat kepada penulis, serta selalu sabar dalam mendidik penulis sejak kecil.
2. Seluruh Keluarga dan Saudara penulis, yang selalu memberikan semangat dan mendoakan penulis selama ini.
3. Ibu Anugrayani Bustamin, S.T., M.T. selaku pembimbing penulis yang selalu membimbing, menyediakan waktu, tenaga, pikiran, dan perhatian yang luar biasa untuk mengarahkan penulis dalam penyusunan tugas akhir ini.
4. Segenap Dosen dan Staf Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin yang telah banyak membantu selama perkuliahan hingga penyelesaian tugas akhir ini.
5. Ica, Echa dan Nabila yang telah memberikan semangat kepada penulis dalam menyelesaikan tugas akhir ini.
6. Teman - Teman Rezolver 20 atas dukungan, bantuan, dan semangat yang telah diberikan selama ini.
7. Serta seluruh pihak yang tak sempat penulis sebutkan satu per satu, tanpa sadar telah menjadi motivasi penulis dalam menyelesaikan tugas akhir.

Akhirnya dengan segala kerendahan hati, penulis menyadari bahwa tugas akhir ini masih terdapat kekurangan dan jauh dari kata sempurna baik dari isi maupun cara penyajian. Oleh karena itu penulis mengharapkan adanya saran maupun kritik yang bersifat membangun demi kesempurnaan tugas akhir ini. Semoga tugas akhir ini dapat memberikan manfaat bagi pembaca terlebih khusus bagi penulis.

Wassalamu'alaikum Warahmatullahi Wabarakatuh.

Penulis,



Annisa Darwis

## ABSTRAK

ANNISA DARWIS. **Prediksi Keberlanjutan Rawat Inap Pasien Rumah Sakit Menggunakan XLNet-BiGRU-ATT pada Catatan Khusus Klinis Pasien** (Dibimbing oleh Anugrayani Bustamin, S.T., M.T.)

**Latar Belakang.** Penerimaan kembali pasien rumah sakit dalam jangka waktu singkat setelah keluar merupakan salah satu indikator penting dalam menilai kualitas perawatan kesehatan. Prediksi penerimaan kembali pasien berdasarkan catatan medis yang dihasilkan selama perawatan sebelumnya dapat membantu penyedia layanan kesehatan dalam mencegah kejadian ini. **Metode.** Penelitian ini mengembangkan model prediksi menggunakan arsitektur kombinasi XLNet, BiGRU (*Bidirectional Gated Recurrent Unit*), dan *attention mechanism*. Arsitektur ini memanfaatkan representasi konteks dari XLNet, kemampuan BiGRU dalam menangkap informasi sekuensial dua arah, serta *attention mechanism* untuk fokus pada bagian catatan medis yang relevan. *Hyperparameter* yang digunakan termasuk *learning rate*  $2e-5$ , *batch size* 32, dan *max sequence length* 512. **Hasil.** Hasil terbaik diperoleh pada *epoch* ke-4, dengan nilai ROC-AUC, PR-AUC, dan PR80 yang mencapai 0.742, 0.723 dan 0.237. **Kesimpulan.** Arsitektur yang dikembangkan mampu memprediksi penerimaan kembali pasien berdasarkan data teks klinis yang kompleks. Model ini berpotensi digunakan sebagai alat bantu bagi penyedia layanan kesehatan dalam mengidentifikasi pasien dengan risiko tinggi untuk kembali dirawat, sehingga langkah pencegahan yang tepat dapat diambil.

**Kata kunci:** Penerimaan kembali pasien, XLNet, BiGRU, Mekanisme *Attention*, Prediksi, Catatan klinis.

## ABSTRACT

ANNISA DARWIS. *Predicting Hospital Readmissions Using XLNet-BiGRU-ATT on Patient Clinical Notes* (supervised by Anugrayani Bustamin, S.T., M.T.)

**Background.** Hospital readmission within a short period after discharge is a key indicator of healthcare quality. Predicting patient readmission based on medical records generated during previous care can assist healthcare providers in preventing this event. **Methods.** This study developed a predictive model using a combination of XLNet, BiGRU (Bidirectional Gated Recurrent Unit), and attention mechanism architecture. This architecture leverages contextual representations from XLNet, the ability of BiGRU to capture bidirectional sequential information, and the attention mechanism to focus on relevant parts of the medical records. The hyperparameters used include a learning rate of  $2e-5$ , a batch size of 32, and a max sequence length of 512. **Results.** The best results were obtained in the 4th epoch, with ROC-AUC, PR-AUC, and PR80 scores of 0.742, 0.723, and 0.237, respectively. **Conclusion.** The developed architecture successfully predicts hospital readmissions based on complex clinical text data. This model has the potential to be used as a tool to help healthcare providers identify patients at high risk of readmission, enabling appropriate preventive measures to be taken.

**Keywords:** Patient readmission, XLNet, BiGRU, Attention mechanism, Prediction, Clinical records.

## DAFTAR ISI

	Halaman
HALAMAN JUDUL .....	i
PERNYATAAN PENGAJUAN .....	ii
LEMBAR PENGESAHAN SKRIPSI .....	iii
PERNYATAAN KEASLIAN SKRIPSI .....	iv
UCAPAN TERIMA KASIH .....	v
ABSTRAK .....	ii
ABSTRACT .....	iii
DAFTAR ISI .....	iv
DAFTAR TABEL .....	vi
DAFTAR GAMBAR .....	i
DAFTAR LAMPIRAN .....	i
DAFTAR SINGKATAN, ISTILAH, DAN LAMBANG .....	ii
BAB I PENDAHULUAN .....	4
1.1 Latar Belakang .....	4
1.2 Landasan Teori .....	6
1.2.1 Catatan Klinis .....	6
1.2.2 <i>Electronic Medical Record</i> .....	7
1.2.3 ICU ( <i>Intensive Care Unit</i> ) .....	9
1.2.4 Prediksi Penerimaan Kembali Pasien .....	9
1.2.5 <i>Text Classification</i> .....	10
1.2.6 <i>Natural Language Processing</i> .....	11
1.2.7 XLNet ( <i>eXtreme Language Understanding NETwork</i> ) .....	14
1.2.8 BiGRU ( <i>Bidirectional Gated Recurrent Unit</i> ) .....	17
1.2.9 <i>Attention Layer</i> .....	20
1.2.10 <i>PhysioNet</i> .....	21
1.3 Rumusan Masalah .....	22
1.4 Tujuan dan Manfaat .....	22
1.5 Ruang Lingkup .....	23
BAB II METODE PENELITIAN .....	24
2.1 Tahapan Penelitian .....	24
2.2 Waktu dan Lokasi Penelitian .....	25
2.3 Instrumen Penelitian .....	25
2.4 Teknik Pengambilan Data .....	26
2.5 Pelaksanaan Penelitian .....	30
2.5.1 <i>Preprocessing Data</i> .....	31
2.5.2 Training dan Implementasi Model .....	37
2.5.3 Evaluasi Model .....	49
BAB III HASIL DAN PEMBAHASAN .....	56
3.1 Hasil .....	56
3.1.1 Pemrosesan <i>Text</i> Medis .....	56
3.1.2 Hasil Pelatihan Model .....	60
3.2 Pembahasan .....	62
3.2.1 Pemilihan Model XLNet-BiGRU-ATT .....	62

3.2.2 Perbandingan dengan Studi Sebelumnya .....	62
3.2.3 Interpretasi Data Hasil Pengujian Model .....	63
3.2.4 <i>Sample</i> Alur Sistem dan Visualisasi Model.....	68
3.2.5 Interpretabilitas Model.....	77
BAB IV KESIMPULAN DAN SARAN.....	78
4. 1 Kesimpulan .....	78
4. 2 Saran.....	79
DAFTAR PUSTAKA .....	80
LAMPIRAN.....	82

## DAFTAR TABEL

Nomor urut	Halaman
Tabel 1. Atribut <i>admissions</i> .....	27
Tabel 2. Atribut <i>noteevents</i> .....	28
Tabel 3. Hasil pembagian <i>dataset</i> .....	36
Tabel 4. Inisialisasi model XLNet-BiGRU-ATT .....	42
Tabel 5. Optimasi model.....	46
Tabel 6. Hasil tokenisasi teks medis .....	58
Tabel 7. Performa model pada setiap <i>epoch</i> .....	60
Tabel 8. <i>Curva</i> performa model pada setiap <i>epoch</i> .....	60
Tabel 9. Perbandingan performa model dengan studi sebelumnya.....	63
Tabel 10. Data hasil pengujian model .....	63
Tabel 11. Pemetaan nilai <i>Query</i> (Q) dan <i>Key</i> (K) .....	70
Tabel 12. Sampel data dengan skor probabilitas yang dihasilkan model .....	73

## DAFTAR GAMBAR

Nomor urut	Halaman
Gambar 1. <i>Two-stream self-attention mechanism structure</i> .....	15
Gambar 2. (a) <i>Content stream</i> & (b) <i>Query stream</i> .....	16
Gambar 3. <i>GRU structure</i> .....	18
Gambar 4. <i>BiGRU structure</i> .....	20
Gambar 5. Tahapan penelitian.....	24
Gambar 6. Lokasi penelitian .....	25
Gambar 7. Contoh data pasien .....	28
Gambar 8. Catatan klinis pasien .....	29
Gambar 9. Pelaksanaan penelitian .....	30
Gambar 10. <i>Preprocessing data</i> .....	33
Gambar 11. Data tabel <i>admissions</i> .....	34
Gambar 12. Data tabel <i>noteevents</i> .....	34
Gambar 13. Urutan kronologis penerimaan pasien .....	34
Gambar 14. Menambahkan <i>next admission type</i> .....	34
Gambar 15. Memfilter penerimaan <i>elective</i> .....	35
Gambar 16. <i>Backfill</i> nilai .....	35
Gambar 17. Plot histogram hari antara penerimaan kembali pasien .....	35
Gambar 18. Aritektur <i>XLNet-BiGRU-ATT</i> .....	37
Gambar 19. Output <i>XLNet</i> .....	44
Gambar 20. Output <i>BiGRU</i> .....	45
Gambar 21. Output <i>Attention Mechanism</i> .....	45
Gambar 22. <i>Computational graph</i> .....	68
Gambar 23. <i>Attention recap</i> .....	69
Gambar 24. Visualisasi <i>attention</i> .....	76

## DAFTAR LAMPIRAN

Nomor urut	Halaman
Lampiran 1. <i>Dataset</i> Penelitian: Kelas Positif .....	82
Lampiran 2. <i>Dataset</i> Penelitian: Kelas Negatif .....	83
Lampiran 3. <i>Source Code Program: Preprocessing</i> .....	84
Lampiran 4. <i>Source Code Program: Modeling</i> XLNet-BiGRU-ATT .....	87
Lampiran 5. <i>Source Code Program: Training Model</i> XLNet-BiGRU-ATT .....	88
Lampiran 6. <i>Source Code Program: Evaluasi Model</i> XLNet-BiGRU-ATT .....	90
Lampiran 7. Permohonan akses data pada <i>PhysioNet</i> .....	92
Lampiran 8. <i>PhysioNet Credentialed Health: Signed Data Use Agreement</i> .....	93

## DAFTAR SINGKATAN, ISTILAH, DAN LAMBANG

Lambang/singkatan	Arti dan Penjelasan
NLP	<i>Natural Language Processing</i>
XLNet	<i>eXtreme Language Understanding NETWORK</i>
BiGRU	<i>Bidirectional Gated Recurrent Unit</i>
ATT	<i>Attention Mechanism</i>
SOAP	<i>Subjective, Objective, Assessment, Plan</i>
MIMIC-III	<i>Medical Information Mart for Intensive Care III</i>
EHR	<i>Electronic Medical Record</i>
ICU	<i>Intensive Care Unit</i>
$P(X)$	<i>Permutation Language Model</i>
$g_{zt}^{(m)}$	<i>Content Stream</i>
$h_{zt}^{(m)}$	<i>Query Stream</i>
$h_t$	<i>Recurrence mechanism</i>
$\odot$	<i>Hadamard Product</i>
$x^t$	<i>Current Node Input</i>
$y^t$	<i>Current Node Output</i>
$h^t$	<i>Hidden State</i>
$\sigma$	<i>Sigmoid</i>
$r$	<i>control reset gate</i>
$z$	<i>control update gate</i>
$h_i$	<i>Representasi Kontekstual Self-attention</i>
$x$	<i>Urutan Kata dalam Bentuk Vektor</i>
$y$	<i>Label Sebenarnya</i>
$\hat{y}$	<i>Logist yang Dihasilkan Model</i>
$\overrightarrow{h}_t$	<i>Forward Representation</i>
$\overleftarrow{h}_t$	<i>Backward Representation</i>
$a_t$	<i>Attention Score</i>
$t$	<i>Sequence Length</i>
$T$	<i>Jumlah keseluruhan token dalam catatan medis</i>
$c$	<i>Context Vector</i>
$w$	<i>Fully Connected Layer Weight</i>
$\tanh$	<i>Hyperbolic tangen</i>
ROC	<i>Receiver Operating Characteristic</i>
AUC	<i>Area Under the Curve</i>
PR	<i>Precision-Recall</i>
TPR	<i>True Positive Rate</i>
FPR	<i>False Positive Rate</i>
TP	<i>True Positives</i>
FN	<i>False Negatives</i>
FP	<i>False Positives</i>

TN  
PR80

*True Negatives*  
*Recall at Precision 80*

---

# BAB I PENDAHULUAN

## 1.1 Latar Belakang

*Intensive Care Unit* (ICU) adalah ruang rawat di rumah sakit yang dilengkapi dengan staf dan peralatan khusus untuk merawat dan mengobati pasien dengan perubahan fisiologi yang cepat memburuk dan mempunyai intensitas defek fisiologi yang merupakan keadaan kritis dan dapat menyebabkan kematian. Perawatan intensif yang diberikan kepada setiap pasien kritis tersebut berkaitan erat dengan tindakan-tindakan yang memerlukan pencatatan medis yang berkesinambungan dan monitoring untuk memantau secara cepat perubahan fisiologis yang terjadi atau akibat dari penurunan fungsi organ-organ tubuh lainnya (Murwani, A., 2008).

Penerimaan kembali ke *Unit Perawatan Intensif* (ICU) di rumah sakit dapat menyebabkan beban yang tinggi pada sistem perawatan kesehatan, dengan efek sosial ekonomi pada pasien, keluarga, dan praktisi kesehatan. Penerimaan kembali ICU yang dini dan tidak direncanakan, dikaitkan dengan peningkatan risiko mortalitas, morbiditas, tinggal lebih lama di rumah sakit dan ICU, dan peningkatan biaya. Penelitian saat ini telah menunjukkan bahwa tingkat penerimaan kembali ICU dipengaruhi oleh faktor-faktor selain kualitas perawatan, seperti karakteristik pasien dan lama tinggal, dan secara umum semua sumber data yang memungkinkan (González-Nóvoa et al., 2023).

Peningkatan penggunaan teknologi dalam bidang kesehatan telah memberikan landasan yang kuat bagi pengembangan sistem yang mampu memproses dan menganalisis data medis dengan cepat dan efisien. *Natural Language Processing* (NLP) adalah bidang ilmu yang berfokus pada pengolahan bahasa alami oleh komputer. Dalam konteks pemulangan pasien, NLP dapat digunakan untuk menganalisis catatan medis dan dokumen terkait untuk mengidentifikasi faktor-faktor yang mungkin berkontribusi terhadap penerimaan kembali pasien.

Dibandingkan dengan gambaran terstruktur, catatan klinis memberikan gambaran pasien terkait gejala, alasan diagnosis, hasil radiologi, aktivitas sehari-hari, dan riwayat pasien. Dalam membuat prediksi klinis yang akurat diperlukan pembacaan data catatan klinis dalam jumlah besar, sementara dokter yang bekerja di unit perawatan intensif, perlu mengambil keputusan dalam waktu terbatas. Hal ini bisa saja menambah beban kerja dokter, sehingga sistem prediksi berdasarkan catatan klinis dapat berguna dalam praktiknya (Huang et al., 2020).

Di Indonesia rekam medis pasien mulai beralih menjadi berbasis elektronik dengan diterbitkannya Peraturan Menteri Kesehatan (PMK) nomor 24 tahun 2022 tentang rekam medis. Dimana fasilitas pelayanan kesehatan (fasyankes) diwajibkan menjalankan sistem pencatatan riwayat medis pasien secara elektronik. Berdasarkan aturan ini dituliskan bahwa Isi rekam medis paling sedikit terdiri atas identitas pasien, hasil pemeriksaan fisik dan penunjang, diagnosis, pengobatan, dan

rencana tindak lanjut pelayanan kesehatan, dan nama serta tanda tangan tenaga kesehatan pemberi pelayanan kesehatan.

Dikutip dari artikel infokes.co.id (Infokes Indonesia) yang terintegrasi dengan standar dan regulasi Kementerian Kesehatan Republik Indonesia (Kemenkes), format pencatatan rekam medis yang sering digunakan adalah format SOAP (*Subjective, Objective, Assessment, Plan*). Dalam SOAP, *Subjective* mencakup informasi yang dilaporkan oleh pasien atau keluarga mengenai keluhan atau gejala yang dialami serta riwayat penyakit pasien. *Objective* mencakup data yang diukur atau ditemukan secara objektif oleh tenaga medis. Contoh data yang dicatat di bagian ini termasuk hasil pemeriksaan fisik, laboratorium, atau hasil pemeriksaan penunjang lainnya. *Assessment* mencakup diagnosis, prognosis, dan rencana perawatan. *Plan* mencakup rencana tindakan yang akan dilakukan untuk mengatasi masalah kesehatan pasien seperti pemberian obat, terapi, tindakan medis, atau edukasi kepada pasien (*Pencatatan SOAP Rekam Medis Dan Contoh Penggunaannya - eClinic, 2023*).

Catatan SOAP adalah cara bagi petugas kesehatan untuk mendokumentasikan secara terstruktur dan terorganisir (Gogineni et al., 2019). Catatan SOAP membantu memandu petugas layanan kesehatan menggunakan data klinis untuk menilai, mendiagnosis, dan merawat pasien berdasarkan informasi yang mereka berikan. Catatan SOAP adalah informasi penting tentang status kesehatan pasien serta dokumen komunikasi antar profesional kesehatan (Podder et al., 2024).

Oleh karena itu, dataset yang akan digunakan dalam penelitian ini adalah MIMIC-III (*Medical Information Mart for Intensive Care III*) yang merupakan proyek database dari *Institutional Review Boards* dari Beth Israel Deaconess Medical Center (Boston, MA) dan *Massachusetts Institute of Technology* (Cambridge, MA). MIMIC III adalah database besar berisi catatan kesehatan elektronik (EHR) dari pasien ICU, yang mencakup informasi seperti diagnosis utama, prosedur yang dilakukan, dan rencana tindak lanjut, catatan terkait dengan pengelolaan obat-obatan, laporan konsultasi dari spesialis atau dokter, catatan mengenai arahan perawatan lanjutan, hasil laboratorium, dan lain-lain (Johnson et al., 2015), yang relevan dengan data rekam medis di Indonesia.

Asumsi *bag-of-words* telah digunakan untuk memodelkan teks klinis, selain model penyematan kata *log-bilinear* seperti *Word2Vec*. Namun catatan klinis yang panjang dan kata-kata yang saling bergantung, membuat metode ini tidak dapat menangkap ketergantungan jangka panjang yang diperlukan untuk menangkap makna klinis (Huang et al., 2020).

XLNet (*eXtreme Language Understanding NETwork*) – BiGRU (*Bidirectional Gated Recurrent Unit*) – ATT (*Attention Mechanism*) adalah salah satu jenis model dalam bidang *Natural Language Processing* atau NLP yang digunakan untuk berbagai tugas, seperti pemahaman bahasa alami, terjemahan mesin, dan lain-lain. Model ini menggabungkan beberapa teknik dan arsitektur yang efektif dalam penanganan teks. XLNet adalah model untuk *semantic understanding pre-training*. Model ini dibangun berdasarkan model sebelumnya seperti *mask language models* dan model bahasa *autoregresif* (AR), dan mengatasi tantangan spesifik dalam tahap

*pre-training* BERT. Salah satu keunggulan utama XLNet adalah kemampuannya menangani ketidakkonsistenan antara *mask flag* dan proses penyesuaian (Han et al., 2023).

Di dalam XLNet-BiGRU-ATT, ada tambahan lapisan-lapisan lain yang meningkatkan kemampuan model. Pertama, ada lapisan BiGRU, yang merupakan jenis arsitektur *recurrent neural network* (RNN) yang memungkinkan model untuk memproses data sekuensial dalam dua arah sekaligus: maju dan mundur. Ini membantu model dalam menangkap konteks yang lebih kaya dan bergantung pada sejarah lebih luas dalam teks. Dibandingkan dengan LSTM, GRU memiliki struktur yang lebih sederhana dengan performa serupa, dan *bidirectional* GRU (BiGRU) dapat menangkap informasi konteks lebih baik daripada RNN *unidirectional* (Han et al., 2023).

Kemudian, ada juga lapisan *attention* (perhatian) yang memberikan bobot pada bagian-bagian penting dari input. *Attention mechanism* memungkinkan model untuk fokus pada bagian-bagian penting yang paling relevan dari input saat membuat prediksi, yang sangat berguna dalam menangani data teks klinis yang panjang. Gabungan dari XLNet sebagai *pre-trained language model*, BiGRU untuk *bidirectional sequential processing*, dan *attention layer* untuk memberikan penekanan pada informasi penting dalam teks, menjadikan XLNet-BiGRU-ATT sebagai model yang sangat kuat dalam penanganan berbagai tugas pemrosesan bahasa alami.

Dengan melihat permasalahan diatas, penulis mengusulkan judul “Prediksi Keberlanjutan Rawat Inap Pasien Rumah Sakit Menggunakan XLNet-BiGRU-ATT pada Catatan Khusus Klinis Pasien” untuk membangun model prediksi penerimaan kembali pasien rumah sakit dengan memanfaatkan rekam medis untuk membantu pertimbangan dokter atau petugas medis yang bekerja di unit perawatan intensif, yang perlu mengambil keputusan berdasarkan catatan klinis dalam jumlah besar, sehingga dapat menghemat waktu dan tenaga dokter.

## 1.2 Landasan Teori

### 1.2.1 Catatan Klinis

Catatan klinis atau rekam medis adalah berkas berisi catatan dan dokumen tentang pasien yang meliputi identitas, pemeriksaan, pengobatan, tindakan medis lain pada sarana pelayanan kesehatan untuk rawat jalan dan rawat inap baik dikelola pemerintah maupun swasta (permenkes nomor 209 / MENKES / PER / III / 2008). Rekam medis juga didefinisikan sebagai segala bentuk identifikasi dan dokumentasi pasien, baik dalam bentuk tertulis dan grafis. Sejarah rekam medis telah berlangsung selama ribuan tahun, dengan akar sejarah yang lebih awal dalam peradaban kuno. Hingga abad ke-19, rekam medis terutama digunakan untuk tujuan pendidikan, kemudian digunakan untuk tujuan lain seperti dalam prosedur asuransi atau hukum (Lorkowski & Pokorski, 2022).

Rekam medis tidak memiliki formalisasi selama ribuan tahun, dan pendekatan sistematis modern merupakan pencapaian 100–150 tahun terakhir.

Dalam aspek kualitas catatan, sering kali naskah medis yang membutuhkan banyak waktu untuk dibuat kemudian dianggap tidak berguna karena kurangnya keterbacaan. Banyak penulis mengatakan bahwa kualitas rekam medis tidak dapat dinilai ketika kasus dan pasien yang dijelaskan bersifat anonim. Oleh karena itu, kemampuan memilih dan menganalisis data dari catatan historis sangat penting. Riwayat penyakit yang tercatat dianggap sebagai alat pendidikan yang berharga, asalkan data dan deskripsinya seimbang dan saling melengkapi (Lorkowski & Pokorski, 2022).

### 1.2.2 *Electronic Medical Record*

Rekam medis mengalami perubahan revolusioner dari format berbasis kertas ke format elektronik, yang mencerminkan perkembangan sistem *eHealth*. Proses migrasi ke catatan kesehatan elektronik melibatkan penggunaan algoritma kecerdasan buatan atau *artificial intelligence* (AI) yang menyederhanakan layanan medis dengan menggunakan metode kerja yang lebih cepat dan lebih sederhana. AI menguntungkan pasien dan penyedia layanan karena meningkatkan manajemen dan komunikasi pasien di antara pusat-pusat medis, menghemat sumber daya, mengidentifikasi kontaminasi atau infeksi, dan membatasi biaya kesehatan (Lorkowski & Pokorski, 2022).

Torkman et al., (2024) dalam studi eksplorasinya mengenai rekam medis elektronik menyatakan bahwa *electronic medical record* (EMR) adalah salah satu aplikasi teknologi informasi kesehatan terpenting dalam perawatan kesehatan. EMR adalah sistem komputerisasi yang menyediakan pendekatan efisien untuk mengumpulkan, menyimpan, dan menampilkan informasi terkait kesehatan. EMR dikembangkan untuk mendukung tugas-tugas profesional perawatan kesehatan dalam praktik medis. Fungsi utama sistem ini meliputi tetapi tidak terbatas pada penyediaan catatan konsultasi medis, informasi tentang vaksinasi, rincian masalah kesehatan, resep, dan pembaruan, pengingat otomatis untuk berbagai tugas, dan peringatan janji temu medis. Meskipun manfaatnya sudah diakui, penerapan rekam medis elektronik di kalangan profesional kesehatan masih belum konsisten. Berbagai faktor memengaruhi keberhasilan adopsi dan pemanfaatan. Faktor organisasi seperti dukungan kepemimpinan, pelatihan yang memadai, dan infrastruktur teknis sangatlah penting. Misalnya, penelitian telah menunjukkan bahwa dukungan manajemen puncak yang kuat secara signifikan meningkatkan kemungkinan keberhasilan implementasi EMR, karena hal ini mendorong lingkungan yang mendukung bagi para profesional kesehatan untuk mengadopsi teknologi baru. Selain itu, keberadaan sistem pendukung TI di dalam rumah sakit sangat penting dalam mengatasi tantangan teknis dan mengurangi resistensi terhadap adopsi EMR. Kompleksitas dan kegunaan EMR yang dirasakan juga memainkan peran penting dalam penerimaan mereka. Profesional perawatan kesehatan lebih cenderung mengadopsi EMR jika sistemnya ramah pengguna dan selaras dengan alur kerja klinis mereka. Faktor-faktor seperti kemudahan penggunaan, waktu respons, dan kegunaan keseluruhan sistem EMR adalah yang terpenting. Sistem yang dianggap

rumit atau mengganggu pola kerja cenderung tidak diterima oleh penyedia layanan kesehatan. Oleh karena itu, memastikan bahwa EMR dirancang dengan masukan pengguna akhir dan dapat beradaptasi dengan lingkungan klinis dapat secara signifikan memengaruhi penerimaan dan pemanfaatannya.

Selain itu, atribut individu seperti literasi komputer, sikap terhadap teknologi, dan pengalaman sebelumnya dengan memengaruhi tingkat adopsi. Profesional perawatan kesehatan yang lebih akrab dengan sistem komputer dan yang memiliki sikap positif terhadap teknologi lebih cenderung menggunakan EMR secara efektif. Program pelatihan yang meningkatkan literasi komputer dan membiasakan staf dengan fungsi EMR dapat mengurangi resistensi dan mempromosikan sikap positif terhadap EMR. Studi telah menyoroti pentingnya pengembangan dan pelatihan profesional berkelanjutan dalam memastikan keberhasilan jangka panjang sistem EMR dalam pengaturan perawatan kesehatan.

Ada berbagai faktor yang dapat memengaruhi penggunaan EMR di sektor perawatan kesehatan. Misalnya, faktor-faktor berikut ditemukan menjadi pendorong niat perilaku mengenai penggunaan EMR: harapan kinerja, pengaruh sosial, dan kebiasaan menemukan bahwa fungsionalitas EMR dapat diperkuat dengan menambahkan fitur tambahan seperti kamus fasilitas penelitian dan entitas medis. Selain itu, ditemukan bahwa faktor individu seperti literasi komputer, norma pribadi, dan pengalaman individu dapat memainkan peran penting dalam penggunaan EMR oleh profesional perawatan kesehatan. Seperti halnya sistem informasi lainnya, keberhasilan implementasi EMR bergantung pada penggunaan pengguna.

Torkman et al. (2024) juga menjelaskan bahwa kesulitan yang dirasakan dalam penerapan EMR, seperti kompleksitas sistem dan dokumentasi yang memakan waktu, merupakan hambatan yang signifikan. Masalah kompatibilitas perangkat keras/perangkat lunak juga menjadi perhatian, karena beberapa sistem tidak terintegrasi dengan baik dengan alur kerja yang ada. Ketidakpastian kinerja pekerjaan, termasuk ketakutan kehilangan otonomi dan penghapusan data, semakin memengaruhi penerimaan. Kemudahan pengoperasian dan risiko yang dirasakan merupakan faktor penting, dengan pengguna menunjukkan bahwa sistem yang ramah pengguna lebih mungkin diadopsi. Dukungan sosial, seperti dukungan kolega dan *supervisor*, sangat penting untuk mengatasi tantangan adopsi. Kepercayaan pengguna dan dukungan organisasi, termasuk program pengembangan profesional, sangat penting untuk implementasi yang sukses. Akhirnya, dukungan teknologi, seperti bantuan dan pelatihan TI yang tepat waktu, diperlukan untuk mengatasi tantangan teknis.

Isu utama yang harus di atasi dalam implementasi EMR, yaitu: (1) Kebutuhan terhadap standar data di bidang terminologi klinik, (2) Aspek *privacy*, kerahasiaan dan keamanan data, (3) Pelaksanaan entri data oleh dokter dan tenaga medis lainnya, (4) Kesulitan integrasi sistem rekam medis dengan sumber informasi lain dalam pelayanan kesehatan (Berg, 2004).

Manfaat teknologi informasi dalam rekam medis elektronik, selain untuk efisiensi pencatatan dan pengolahan data, serta menyediakan informasi yang lebih akurat dan terpercaya, yaitu memiliki tujuan untuk mengurangi *medical error* dan

meningkatkan keamanan pasien (*patient safety*). Dengan adanya sistem aplikasi manajemen rekam medis, maka *medical error* dalam pengambilan keputusan oleh tenaga kesehatan dapat dikurangi, karena setiap pengambilan keputusan akan berdasarkan data rekam medis pasien yang telah ada dan sudah terintegrasi dengan unit pelayanan lainnya (Moody, L.E, et.al. 2004).

### 1.2.3 ICU (*Intensive Care Unit*)

*Intensive care unit* (ICU) adalah ruang rawat di rumah sakit yang dilengkapi dengan staf dan peralatan khusus untuk merawat dan mengobati pasien dengan perubahan fisiologi yang cepat memburuk yang mempunyai intensitas defek fisiologi satu organ ataupun mempengaruhi organ lainnya sehingga merupakan keadaan kritis yang dapat menyebabkan kematian. Perawatan intensif yang diberikan kepada setiap pasien kritis tersebut berkaitan erat dengan tindakan-tindakan yang memerlukan pencatatan medis yang berkesinambungan dan monitoring untuk memantau secara cepat perubahan fisiologis yang terjadi atau akibat dari penurunan fungsi organ-organ tubuh lainnya (Murwani, A., 2008).

ICU memainkan peran penting dalam sistem perawatan kesehatan dengan menyediakan perawatan khusus bagi pasien dengan kondisi yang parah atau mengancam jiwa. Salah satu aspek penting dalam mengelola sumber daya ICU dan memastikan perawatan pasien yang optimal adalah memperkirakan lamanya waktu perawatan di ICU (Rhazzafe et al., 2024).

ICU merupakan suatu bagian rumah sakit yang dilengkapi dengan staf khusus dan perlengkapan yang khusus yang ditujukan untuk observasi, perawatan dan terapi pasien-pasien yang menderita penyakit atau cedera yang mengancam jiwa atau potensial mengancam jiwa yang diharapkan masih dapat *reversible*. Umumnya pasien yang dirawat di ICU berada dalam keadaan tertentu, misalnya pasien dengan penyakit kritis yang menderita kegagalan satu atau lebih dari sistem organnya ( Tabrani, 2007).

### 1.2.4 Prediksi Penerimaan Kembali Pasien

Prediksi penerimaan kembali pasien di rumah sakit dan lamanya rawat inap memberikan informasi tentang cara mengelola kapasitas tempat tidur rumah sakit dan jumlah staf yang dibutuhkan. Penerimaan kembali rumah sakit memainkan peran utama dalam pengeluaran rumah sakit. Baru-baru ini, fokus utama sistem perawatan kesehatan adalah pada pasien yang diterima kembali ke rumah sakit dalam jangka waktu yang singkat (sebagian besar pada penerimaan kembali yang terjadi dalam 30 hari) setelah keluar dari rumah sakit. Menurut laporan terbaru, beban sistem perawatan kesehatan Amerika Serikat adalah 41 miliar dolar, karena penerimaan kembali pasien diabetes di rumah sakit dalam waktu 30 hari. Sebuah studi di Spanyol mengungkapkan bahwa, sementara total biaya tahunan pasien diabetes adalah EUR 1803,6 per orang, biaya rawat inap untuk pasien ini adalah EUR 801,6. Studi lain yang dilakukan di Amerika Serikat menunjukkan bahwa biaya tahunan langsung

diabetes adalah sekitar USD 9595 per orang. Ada biaya langsung dan tidak langsung dari sistem perawatan kesehatan yang terkait dengan rawat inap pasien rawat inap. Biaya medis langsung mencakup biaya yang terkait dengan layanan yang diberikan di rumah sakit, seperti rawat inap, rawat inap ICU, tes laboratorium, dan jenis kunjungan rumah sakit lainnya (Tavakolian et al., 2023)

Penerimaan kembali ke ICU selama penerimaan di rumah sakit yang sama merupakan kejadian buruk yang jarang terjadi dan dapat menyebabkan beban yang tinggi pada sistem perawatan kesehatan, dengan efek sosial ekonomi yang sangat penting pada pasien, keluarga, dan praktisi kesehatan. Penerimaan kembali ICU yang dini dan tidak direncanakan, dengan tingkat penerimaan kembali berkisar antara 1,3% hingga 13,7%, dikaitkan dengan peningkatan risiko mortalitas, morbiditas, tinggal lebih lama di rumah sakit dan ICU, dan peningkatan biaya. Akibatnya, ada minat yang tinggi pada tingkat penerimaan kembali ICU sebagai indikator kualitas perawatan kritis. Namun demikian, penelitian saat ini telah menunjukkan bahwa tingkat penerimaan kembali ICU dipengaruhi oleh faktor-faktor selain kualitas perawatan, seperti karakteristik pasien dan lama tinggal, dan secara umum semua sumber data yang memungkinkan. Ini membuka masalah untuk penggunaan teknik kecerdasan buatan baru untuk mengeksplorasi semua informasi yang tersedia (González-Nóvoa et al., 2023)

Dalam beberapa tahun terakhir, penggunaan teknik pembelajaran mesin di bidang kesehatan telah meningkat untuk meningkatkan kualitas perawatan pasien dan untuk memfasilitasi pekerjaan tenaga kesehatan. Karena besarnya jumlah data yang dihasilkan di rumah sakit saat ini, sangat menarik untuk mengembangkan teknik untuk menganalisis data ini secara otomatis dan efisien, memfasilitasi pengambilan keputusan yang benar oleh tenaga kesehatan. Analisis manual dari semua data ini akan memerlukan waktu yang tidak tersedia dalam kerangka kerja sehari-hari rumah sakit, yang menyebabkan hanya sebagian kecil darinya yang dianalisis, kehilangan kesempatan untuk menganalisis data yang tersedia secara global. Pemantauan pasien yang berkelanjutan dan menyeluruh selama tinggal di ICU menghasilkan berbagai macam data biomedis dengan potensi besar untuk aplikasi (González-Nóvoa et al., 2023).

### 1.2.5 **Text Classification**

*Text classification* telah menjadi salah satu tugas fundamental dalam bidang *natural language processing* (NLP) dengan berbagai aplikasi penting dalam dunia nyata. Menurut Minaee et al. (2021) klasifikasi teks adalah proses mengategorikan dokumen teks ke dalam satu atau lebih kelas atau kategori yang telah ditentukan sebelumnya. Tugas ini memiliki berbagai aplikasi praktis, termasuk analisis sentimen, deteksi spam, kategorisasi berita, dan pengelompokan topik. Dalam beberapa tahun terakhir, kemajuan signifikan telah dicapai dalam bidang ini, terutama dengan munculnya teknik *deep learning*.

Metode *deep learning* telah mengungguli pendekatan *machine learning* tradisional dalam banyak *benchmark text classification*. Arsitektur *neural network*

seperti *convolutional neural networks* (CNN), *recurrent neural networks* (RNN), dan lebih baru lagi, model berbasis transformer seperti BERT dan XLNet, telah menunjukkan performa yang luar biasa dalam berbagai tugas klasifikasi teks. Keunggulan utama dari metode *deep learning* adalah kemampuannya untuk secara otomatis mempelajari representasi fitur yang relevan dari data teks mentah, menghilangkan kebutuhan akan ekstraksi fitur manual yang rumit.

Namun, tantangan masih tetap ada dalam domain ini. Salah satu isu utama adalah kebutuhan akan dataset berlabel yang besar untuk melatih model *deep learning* yang kompleks. Selain itu, interpretabilitas model *deep learning* seringkali lebih sulit dibandingkan dengan metode klasik seperti *naive bayes* atau *decision trees*. Terlepas dari tantangan ini, *text classification* terus menjadi area penelitian yang aktif dengan potensi besar untuk inovasi dan aplikasi praktis di masa depan (Minaee et al., 2021)

### 1.2.6 *Natural Language Processing*

Model *natural language processing* (NLP) adalah disiplin ilmu komputer yang bertujuan untuk memahami konsep dan maksud dari bahasa manusia. Sementara manusia cukup mahir memahami sintaks linguistik dan tata bahasa serta hubungan spasial tersirat, komputer memiliki kesulitan besar pengolahan *query* bahasa alami (Allen, 1995).

Teknologi *machine learning* dalam *natural language processing* (NLP) memberi komputer kemampuan untuk menginterpretasikan, memanipulasi, dan memahami bahasa manusia. Integrasi NLP dalam *software requirements engineering* (SRE) menawarkan solusi yang lebih efisien, meskipun semi-otomatis, untuk mendeteksi ambiguitas. Integrasi NLP dan AI ke dalam rekayasa perangkat lunak mewakili pergeseran paradigma yang signifikan, yang menjanjikan tidak hanya untuk mengotomatisasi tetapi juga untuk meningkatkan akurasi dan efektivitas proses ini. NLP, komponen penting AI, menjembatani kesenjangan antara bahasa manusia dan mesin, memungkinkan mesin untuk memproses, menganalisis, dan memahami data bahasa manusia, dan untuk menghasilkan respons seperti manusia. Integrasi ini sangat berdampak dimana NLP yang dikombinasikan dengan AI dapat meningkatkan kualitas persyaratan perangkat lunak dengan menganalisis atribut leksikal dan sintaksisnya, mengotomatiskan pemetaan ke representasi formal, dan menyempurnakan spesifikasi dengan mengekstraksi pengetahuan domain yang relevan. (Necula et al., 2024)

Necula et al. (2024), memetakan tren dalam NLP untuk bidang SRE dalam periode 1991–2023. Analisis ini mengungkapkan beberapa wawasan yang dikelompokkan berdasarkan periode waktu:

1. Periode 1: Hingga tahun 2000. Pada periode menjelang tahun 2000, "model matematika" menjadi titik fokus, yang menunjukkan tahap dasar pemodelan perangkat lunak, di mana model abstrak dan landasan teoritis sangat penting. Selama masa ini, landasan untuk kemajuan masa depan dalam rekayasa

perangkat lunak dan NLP sedang diinisialisasi, meskipun data menunjukkan tahap yang lebih baru untuk topik terkait NLP.

2. Periode 2: 2001–2010. Dari tahun 2001 hingga 2010, bidang ini mulai berkembang, dengan "linguistik komputasional" dan "pemrograman berorientasi objek" yang mendapat perhatian penting. Periode ini menandai pergeseran dari teori dasar ke metodologi yang lebih terapan dalam pengembangan perangkat lunak. Pertumbuhan "linguistik komputasional" selama dekade ini mengarah pada peningkatan minat dalam mengintegrasikan teknik pemrosesan bahasa ke dalam pengembangan perangkat lunak, khususnya di bidang seperti "*requirements analysis*" dan "pemilihan dan evaluasi perangkat lunak komputer".
3. Periode 3: 2011–2021. Dekade ini menyaksikan pertumbuhan signifikan dalam aplikasi NLP dalam SRE. Sistem pemrosesan bahasa alami, *software requirements*, dan *engineering requirements* mengalami peningkatan fokus yang substansial, yang menggambarkan dorongan industri untuk memanfaatkan NLP guna pengumpulan, analisis, dan pengelolaan yang lebih efektif. Munculnya *deep learning* menjelang akhir periode ini menandakan pergeseran ke arah teknik AI yang lebih canggih untuk mengatasi masalah kompleks dalam rekayasa perangkat lunak.
4. Periode 4: 2022–2023. Dalam beberapa tahun terakhir, terdapat fokus yang jelas pada teknologi AI mutakhir, dengan *deep learning*, *machine learning*, dan *learning systems* menjadi pusat perhatian. Tren ini mencerminkan kemajuan pesat yang terjadi dalam AI dan peningkatan penerapannya dalam pemodelan perangkat lunak dan *engineering requirements*. Penyebutan *language model* dan *requirement classification* selama periode ini menyoroti upaya berkelanjutan untuk menyempurnakan aplikasi NLP dalam pengembangan perangkat lunak, menjadikannya lebih akurat dan efisien.

Lebih jauh, Necula et al., (2024) mendalami evolusi tematik dalam ranah NLP dengan memberi gambaran tentang pergeseran signifikan dan kontinuitas fokus dalam bidang tersebut selama periode waktu yang berbeda. Dengan merangkum transisi dalam konsentrasi tematik dari satu era ke era lain, yang disorot oleh berbagai metrik seperti indeks inklusi tertimbang, indeks inklusi, jumlah kejadian, dan indeks stabilitas. Metrik ini menawarkan wawasan kuantitatif tentang tren yang berkembang dan pentingnya tema tertentu dari waktu ke waktu:

1. *Natural language* ke sistem NLP (1991–2000 ke 2001–2010): Transisi dari *natural language* ke *natural language processing system* dengan skor sempurna (1,00) pada indeks inklusi tertimbang dan indeks inklusi menunjukkan pergeseran signifikan ke arah operasionalisasi pemahaman bahasa alami dalam sistem komputasi, yang mencerminkan gerakan penting dalam penelitian NLP dengan hanya dua kejadian tetapi stabilitas tinggi.
2. Semantik dalam sistem NLP (2001–2010 hingga 2011–2020): Fokus pada "semantik" selama periode dengan skor maksimum ini menunjukkan minat yang mendalam dalam memahami dan memproses makna dalam bahasa, yang penting untuk aplikasi NLP yang efektif.

3. Sintaksis ke spesifikasi formal (2001–2010 ke 2011–2020): Pergeseran dari sintaksis ke spesifikasi formal dengan skor tertinggi menunjukkan minat yang semakin matang dalam mengintegrasikan analisis sintaksis ke dalam konteks yang lebih terstruktur dan formal, yang berpotensi memengaruhi area seperti spesifikasi dan desain perangkat lunak. Pergeseran ini dan yang sebelumnya menggarisbawahi pentingnya pemahaman makna dan struktur *natural language requirements* dalam literatur.
4. Dari otomatisasi ke ekstraksi (2001–2010 ke 2011–2020): Peralihan dari otomatisasi ke ekstraksi menunjukkan pergeseran fokus ke arah ekstraksi konsep berharga dari *software requirements*, yang penting untuk mengotomatisasi proses pengembangan *software*, meskipun indeks menunjukkan penekanan dan stabilitas sedang.
5. Kontinuitas ekstraksi (2011–2021 hingga 2022–2023): Fokus berkelanjutan pada ekstraksi dengan indeks inklusi tertimbang yang layak tetapi indeks inklusi yang lebih rendah menunjukkan minat yang berkelanjutan tetapi sedikit berkurang di area ini, yang mungkin mencerminkan kemajuan atau kejenuhan dalam metodologi ekstraksi.
6. Sistem NLP ke *user story* (2011–2021 ke 2021–2023): Transisi ke *user story* mungkin menunjukkan meningkatnya minat dalam menerapkan NLP untuk memahami dan memproses *requirements* dan narasi pengguna dalam proses pengembangan perangkat lunak yang tangkas.
7. Siklus hidup ke penilaian kualitas dan penambangan data (2011–2021 ke 2021–2023): Transisi ini mencerminkan meningkatnya minat dalam menilai kualitas dan mengekstraksi wawasan dari data di seluruh siklus hidup sistem atau proyek, yang menunjukkan penerapan konsep ini yang lebih luas dalam rekayasa perangkat lunak dan analisis data.
8. *Requirements engineering* ke sistem NLP (1991–2000 ke 2001–2010): Pergeseran ini, dengan indeks inklusi berbobot sedang tetapi indeks inklusi lebih rendah, menunjukkan meningkatnya kepentingan yang meskipun tidak terlalu menonjol seperti topik lainnya, mencerminkan integrasi prinsip-prinsip *engineering requirements* dengan sistem NLP.
9. Rekayasa perangkat lunak berbantuan komputer ke sistem NLP (1991–2000 ke 2001–2010): Transisi signifikan dengan indeks inklusi berbobot tinggi, mengarah pada integrasi alat rekayasa berbantuan komputer dengan NLP untuk meningkatkan proses pengembangan perangkat lunak.

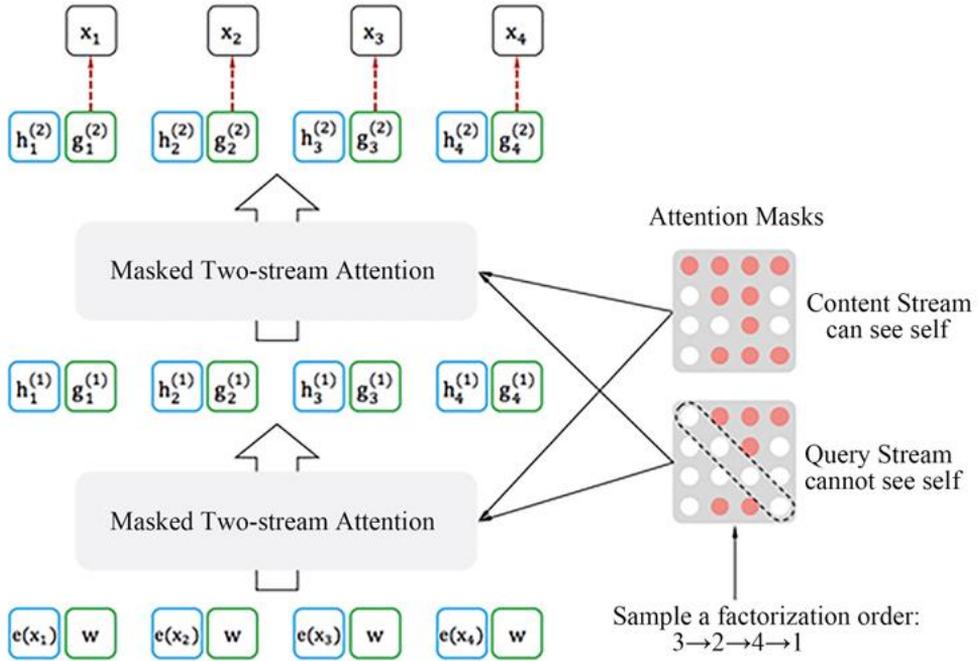
Meskipun penerapan NLP menunjukkan hasil yang menjanjikan, ada beberapa tantangan, termasuk keahlian teknis yang dibutuhkan untuk menggunakan model tersebut dan kebutuhan untuk mempertimbangkan isu etika dan mengatasi potensi bias. Diperlukan pengukuran yang ditetapkan untuk mengevaluasi etika penggunaan ML dan NLP. Oleh karena itu, komunitas akademis harus mengeksplorasi bagaimana teknologi ini dapat mendukung upaya dalam mengatasi tantangan ini untuk menghindari hasil negatif (Sarzaeim et al., 2023)

### 1.2.7 XLNet (*eXtreme Language Understanding NETWORK*)

XLNet merupakan model bahasa berbasis *transformation*, XLNet memperkenalkan pendekatan pelatihan berbasis permutasi yang memeriksa kata-kata dalam urutan yang berbeda dan dilatih dengan kumpulan data yang lebih besar. Model kasus dasar XLNet terdiri dari 12 *transformer layers* dan 12 *attention heads per-layer* dengan 768 *hidden size* yang menghasilkan 110 juta parameter. Pendekatan ini memungkinkan XLNet untuk memahami teks dari kedua arah dengan lebih efisien, dan menangkap pola kompleks dalam data teks. Fitur ini berpotensi meningkatkan hasil analisis sentimen dalam skenario kompleks seperti percakapan klinis. XLNet menggunakan metode tokenisasi berbasis permutasi unik yang menyusun ulang kata-kata dalam kalimat dan menggunakan kata-kata terakhir untuk memprediksi kata berikutnya dalam urutan tersebut. (Chatzimina et al., 2024)

XLNet dapat memproses kalimat pendek dan memperoleh informasi semantik dengan menggunakan informasi konteks untuk mencapai representasi teks yang efektif. XLNet merupakan model mutakhir untuk *pre-training* dalam pemahaman semantik. Model ini dibangun berdasarkan model-model sebelumnya seperti *mask language models* dan *autoregressive (AR) language models*, model XLNet juga mengatasi tantangan-tantangan khusus dalam tahap *pre-training* BERT. Salah satu keuntungan utama XLNet adalah kemampuannya untuk menangani ketidakkonsistenan antara *masked flag* (penanda bagian yang di-mask dalam data) dan *refinement process* (proses penyempurnaan atau penyelarasan model selama pelatihan). Hal ini dicapai melalui pendekatan unik yang merekonstruksi *teks input* dengan cara permutasi dan komposisi. Hal ini memungkinkan integrasi fitur kontekstual ke dalam proses pelatihan model (Han et al., 2023)

XLnet menggunakan ide *random sorting* untuk memecahkan masalah model AR yang tidak dapat memperkenalkan informasi teks dua arah. Proses *random sorting* hanya mengurutkan nomor urut setiap posisi secara acak. Setelah pengurutan, kata-kata sebelum setiap kata dapat digunakan untuk memprediksi probabilitasnya. Di sini, kata-kata sebelumnya dapat diambil dari kata-kata sebelum kata saat ini dalam kalimat asli atau dari kata-kata setelahnya, yang setara dengan menggunakan informasi urutan dua arah (Han et al., 2023)



Gambar 1. *Two-stream self-attention mechanism structure*  
(Sumber: (Han et al., 2023))

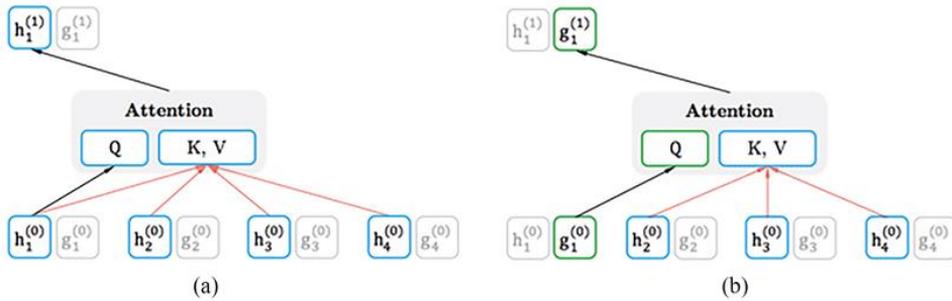
XLNet merupakan model bahasa *autoregresif* yang dapat memperoleh informasi kontekstual dua arah. Untuk memperoleh informasi kontekstual dua arah, XLNet terutama mengadopsi tiga metode, yaitu model bahasa permutasi, *two-stream self-attention*, dan mekanisme sirkulasi. Inti dari XLNet adalah model bahasa permutasi. Untuk mengekstrak informasi kontekstual dua arah, algoritma menggunakan urutan acak dari urutan input awal sambil mempertahankan model satu arah dari model *autoregresif*. XLNet mengintegrasikan model bahasa *autoregresif* dan memperkenalkan dua teknik utama *Transformer-XL* ke dalam XLNet, yaitu skema pengodean posisi relatif dan mekanisme pengulangan segmen (Shi et al., 2022)

**Permutation Language Model.** XLNet menggunakan model bahasa permutasi untuk mengatasi keterbatasan dari *masked language model* (MLM) yang digunakan di BERT. Dalam *masked language modeling*, beberapa token dalam urutan teks di-*mask* dan kemudian diprediksi oleh model. Namun, pendekatan ini mengganggu aliran alami teks dan memisahkan token yang di-*mask* dari konteks aslinya.

XLNet, sebaliknya, menggunakan permutasi urutan token yang memungkinkan model mempelajari probabilitas dari semua urutan token yang mungkin. Persamaan untuk memprediksi token berdasarkan urutan permutasi (1) adalah sebagai berikut:

$$P(x) = \prod_{t=1}^T P(x_{z_t} | x_{z_1}, \dots, x_{z_{t-1}}) \quad (1)$$

Dimana,  $x_{z_t}$  adalah token yang berada di posisi ke- $t$  dalam urutan permutasi  $z$ , dan  $T$  adalah jumlah total token dalam urutan permutasi tersebut. Dengan mempelajari semua urutan yang mungkin, XLNet dapat menangkap hubungan antar-token baik secara searah (kiri ke kanan) maupun terbalik (kanan ke kiri), yang menghasilkan pemahaman yang lebih kaya tentang konteks token. Studi yang dilakukan oleh (Yang et al., 2020) menyatakan bahwa pendekatan ini secara efektif meningkatkan performa pemahaman teks.



Gambar 2. (a) *Content stream* & (b) *Query stream*  
(Sumber: (Han et al., 2023))

**Two-Stream Self-Attention.** Permutasi urutan token dapat membuat posisi asli token menjadi sulit dilacak. Untuk mengatasi masalah ini, XLNet memperkenalkan *two-stream self-attention*. Aliran ini terbagi menjadi dua jenis: *content stream* dan *query stream*, yang keduanya memiliki peran penting dalam memahami konten dan urutan token.

*Content stream* mengelola informasi tentang konten token dalam urutan, memperbarui representasi konten dengan memperhatikan semua token dalam sekuens. *Content stream* menggunakan informasi dari *query stream* sebelumnya untuk *key* dan *value*. Persamaan untuk representasi konten token dihitung melalui mekanisme *self-attention* (2) yaitu:

$$g_{z_t}^{(m)} \leftarrow \text{Attention} (Q = g_{z_t}^{(m-1)}, KV = h_{z_{<t}}^{(m-1)}; \theta) \quad (2)$$

Dimana:

- $g_{z_t}^{(m)}$  : *output content stream* untuk token pada *layer* ke- $m$
- $Q = g_{z_t}^{(m-1)}$ : *query* menggunakan representasi *content stream* dari *layer* sebelumnya
- $KV = h_{z_{<t}}^{(m-1)}$  : *key* dan *value* menggunakan representasi *query stream* dari *layer* sebelumnya
- $\theta$  : mewakili parameter yang dapat dilatih dalam operasi *attention*

*Content stream* menggunakan informasi dari kedua *stream* dengan tujuan mengumpulkan informasi kontekstual dari seluruh sekuens.

*Query stream* digunakan untuk memprediksi token berikutnya dalam urutan permutasi, memperbarui representasi *query* dengan hanya memperhatikan token-token sebelumnya (*autoregressive*). *Query stream* menggunakan informasi dari

*query stream* sendiri untuk *query*, *key*, dan *value*. *Stream* ini mempertahankan informasi tentang posisi permutasi dan diwakili oleh persamaan (3):

$$h_{z_t}^{(m)} \leftarrow \text{Attention}(Q = g_{z_t}^{(m-1)}, KV = h_{z_{\leq t}}^{(m-1)}; \theta) \quad (3)$$

Dimana:

- a)  $h_{z_t}^{(m)}$  : *output query stream* untuk token pada *layer* ke- $m$
- b)  $Q = g_{z_t}^{(m-1)}$  : *query* menggunakan representasi *query stream* dari *layer* sebelumnya
- c)  $KV = h_{z_{\leq t}}^{(m-1)}$  : *key* dan *value* menggunakan representasi *query stream* dari *layer* sebelumnya
- d)  $\theta$  : mewakili parameter yang dapat dilatih dalam operasi *attention*

*Query stream* hanya menggunakan informasi dari dirinya sendiri dengan tujuan mempertahankan sifat *autoregressive* untuk prediksi token berikutnya.

Dengan memisahkan konten dan urutan posisi, XLNet memastikan bahwa model memahami dengan baik arti token maupun posisi token tersebut dalam konteks keseluruhan. Studi dari (Raffel et al., 2023) mendukung pentingnya model berbasis atensi untuk menangkap hubungan dependensi jarak jauh dalam teks.

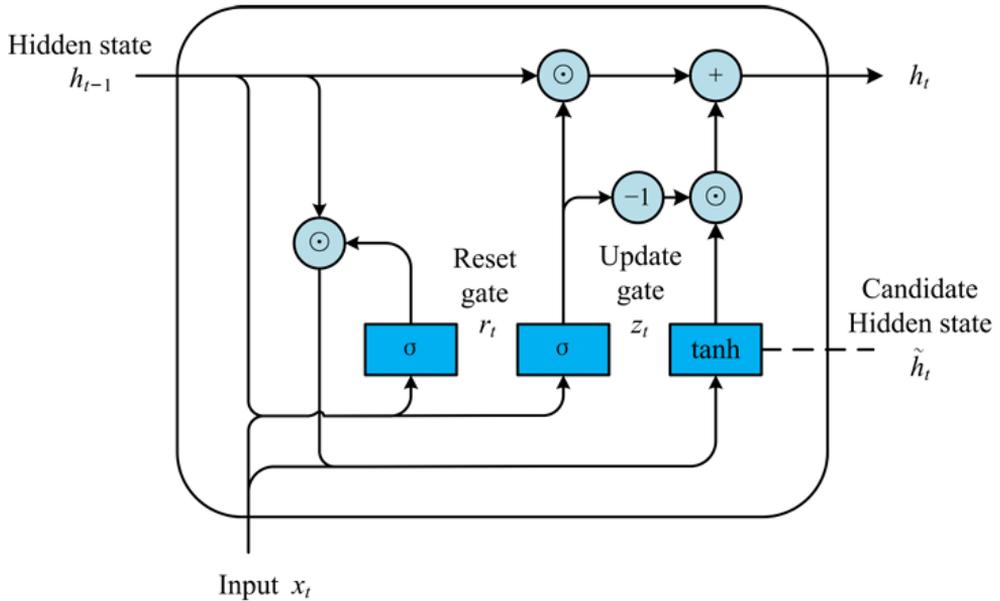
**Mekanisme Sirkulasi (*Recurrence Mechanism*)**. Selain *self-attention*, XLNet menggabungkan prinsip dari *recurrent neural networks* (RNN) untuk memastikan bahwa informasi dari token-token sebelumnya disirkulasikan ke token-token berikutnya dalam urutan. Ini penting untuk memproses teks dalam urutan panjang tanpa kehilangan konteks. Persamaan dasar untuk mekanisme sirkulasi dalam XLNet (4) serupa dengan yang digunakan dalam RNN:

$$h_t = \sigma(W_h h_{t-1} + W_x x_t + b) \quad (4)$$

Dimana,  $h_t$  adalah *hidden state* yang mengandung informasi tentang token pada langkah  $t$ , sedangkan  $h_{t-1}$  membawa informasi dari token sebelumnya. Mekanisme ini membantu model mempertahankan memori jangka panjang dari urutan token, meningkatkan performa dalam tugas yang melibatkan konteks teks yang panjang.

### 1.2.8 BiGRU (*Bidirectional Gated Recurrent Unit*)

*Gated recurrent unit* (GRU) adalah jenis jaringan saraf rekursif yang berbeda dari jaringan saraf lainnya karena struktur gerbang internalnya. Struktur unik ini memungkinkan jaringan untuk menentukan data mana yang relevan dan data mana yang dapat dibuang berdasarkan hubungan mereka. Struktur ini memfasilitasi transmisi data yang efektif dalam jaringan sambil secara efektif mengendalikan informasi yang berlebihan. Sehingga, GRU dapat mengatasi masalah ketergantungan jangka panjang dalam jaringan saraf (Han et al., 2023)



Gambar 3. GRU structure  
(Sumber: (Han et al., 2023))

*Gated recurrent unit* (GRU) menunjukkan efisiensi yang lebih tinggi dan kinerja yang lebih baik daripada RNN dan LSTM saat menangani dependensi jangka panjang. Dengan memperkenalkan dua *gating mechanisms* yaitu *reset gate* dan *update gate*. *Reset gate* menentukan seberapa banyak informasi masa lalu yang akan dipertahankan pada langkah waktu saat ini, sementara *update gate* memutuskan bagaimana informasi baru diintegrasikan ke dalam status saat ini. Desain struktural ini tidak hanya meningkatkan pemahaman model terhadap data sekuensial tetapi juga meningkatkan kemampuannya dalam generalisasinya dalam pembelajaran sampel kecil dengan mengurangi jumlah parameter. Atribut ini membuat GRU dapat digunakan dalam skenario dengan sumber daya terbatas atau di mana pelatihan cepat diperlukan (Zhao et al., 2024)

GRU menggabungkan *input node* saat ini atau *current node input*  $x^t$  dan *state*  $h^{t-1}$  ditransmisikan dari *node* sebelumnya untuk menghasilkan *output node* saat ini atau *current node's output*  $y^t$  dan *hidden state*  $h^t$  diteruskan ke-*node* berikutnya. Persamaan transmisi dan pembaruan parameter internal dalam jaringan (5) – (8) yaitu:

$$r^t = \sigma(w^{rx}x^t + w^{rh}h^{t-1} + b^r) \quad (5)$$

$$z^t = \sigma(w^{zx}x^t + w^{zh}h^{t-1} + b^z) \quad (6)$$

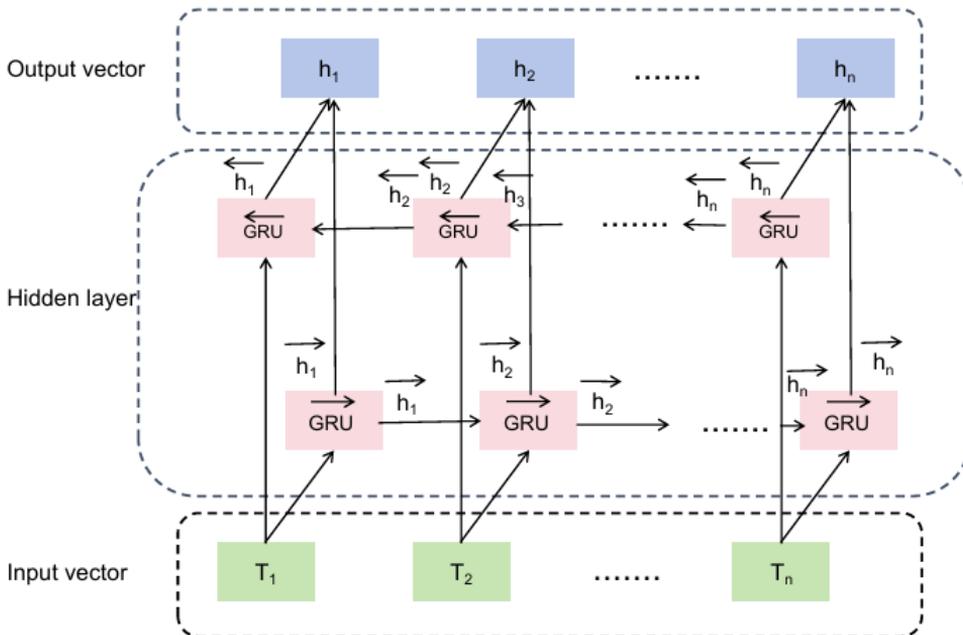
$$h' = \tanh(w^{xh}x^t + r^t \odot w^{hh}h^{t-1}) \quad (7)$$

$$h^t = (1 - z^t) \odot h^{t-1} + h' \odot z^t \quad (8)$$

Dalam persamaan ini,  $\sigma$  menunjukkan fungsi *sigmoid*, yang berfungsi sebagai sinyal *gate control* untuk menjaga nilai dalam rentang  $[0, 1]$ . Semakin dekat sinyal *gate* ke-1, semakin banyak data yang diingat. Jika sebaliknya, semakin banyak data yang dilupakan.  $r$  adalah kontrol *reset gate*, dan  $z$  adalah kontrol *update gate*.  $h'$  menunjukkan keadaan tersembunyi dari kandidat *hidden state*.  $w^{rx}$ ,  $w^{rh}$ , dan lainnya adalah *matriks weight*, sedangkan  $b^r$ ,  $b^z$ , dan lainnya adalah *bias terms*.  $\odot$  mewakili *Hadamard product*, yang berarti perkalian *element-wise* dari matriks (Zhao et al., 2024)

*Reset gate* menghitung data setelah *reset*  $r^t \odot h^{t-1}$  yang kemudian digabungkan dengan  $x^t$  dan melewati fungsi aktivasi *tanh* untuk menjaga nilai *output* dalam  $[-1,1]$ , menghasilkan *hidden state*  $h'$ . Secara bersamaan, *update gate* melakukan fungsi *forgetting* dan *selective memory*, di mana  $(1 - z^t) \odot h^{t-1}$  secara selektif melupakan keadaan *node* sebelumnya, dan  $h' \odot z^t$  secara selektif mengingat *hidden state*. Dalam *tasks named entity recognition*, informasi kontekstual sangat penting untuk memprediksi label entitas. Untuk mendapatkan informasi kontekstual dari masa lalu dan masa depan dalam teks, jaringan unit berulang dua arah (*bidirectional gated recurrent unit networks*, BiGRUs) biasanya digunakan. *Hidden state*  $h'$  dari GRU hanya dapat mengakses informasi dari konteks masa lalu, bukan masa depan (Zhao et al., 2024)

Untuk GRU, output hanya dipengaruhi oleh input saat ini dan *hidden state* sebelumnya, dan tidak terkait dengan *next state*, menjadikan GRU sebagai model searah. Namun, untuk mengekstraksi fitur yang efektif, perlu memperhatikan informasi saat ini, informasi sebelumnya, dan informasi berikutnya. Dapat dengan mudah dipahami bahwa menggabungkan konteks-konteks ini memudahkan pemahaman informasi tekstual. Berdasarkan ide ini, BiGRU diusulkan untuk menggabungkan *forward* GRU dan *backward* GRU, menjadikan hasil output sebagai kombinasi dari dua GRU berdasarkan *weight matrix* (Han et al., 2023)



Gambar 4. BiGRU structure  
(Sumber: (Zhao et al., 2024))

BiGRU (*Bidirectional Gated Recurrent Unit*) memanfaatkan GRU *forward* dan *backward* untuk ekstraksi fitur dan bobot informasi kontekstual, menjumlahkan output, dan kemudian melewatinya melalui lapisan linier untuk memetakan vektor  $d$ -dimensional ke vektor  $m$ -dimensional, sehingga menghasilkan daftar vektor label *output* akhir dari jaringan BiGRU,  $H = \{h_1, h_2, \dots, h_n\}$ , dengan  $H \in \mathbb{R}^{n \times m}$ , dimana  $n$  adalah panjang urutan teks dan  $m$  adalah jumlah *entity type labels*. (Zhao et al., 2024)

### 1.2.9 Attention Layer

Regularisasi ketika berhadapan dengan data tekstual para peneliti sering menghadapi tantangan berupa kalimat panjang yang memuat banyak sekali informasi. Meskipun Bi-GRU dapat mempertimbangkan informasi kontekstual teks secara komprehensif, jaringan ini memiliki keterbatasan dalam menekankan informasi spesifik yang secara langsung relevan dengan tugas target. Oleh karena itu, untuk meningkatkan fokus model pada bagian konteks yang relevan dengan tugas saat ini dan mengekstraksi fitur semantik yang lebih efektif, diperkenalkan *attention layer* di atas jaringan Bi-GRU. Secara khusus, *attention mechanism* yang ditambahkan ke jaringan Bi-GRU dapat menetapkan bobot yang berbeda untuk berbagai bagian informasi kontekstual. Untuk informasi yang terkait erat atau relevan dengan tugas target, model akan mengalokasikan lebih banyak perhatian, sedangkan untuk informasi yang kurang relevan, model akan memberikan lebih sedikit perhatian. Mekanisme ini memungkinkan model untuk memproses informasi yang terkait dengan tugas-tugas tertentu secara lebih efektif, sehingga mencapai kinerja yang lebih tinggi. Dengan menggabungkan Bi-GRU dengan *attention*

*mechanism*, model tidak hanya dapat menangkap ketergantungan jarak jauh tetapi juga menekankan pentingnya fitur yang relevan dengan memberikan bobot yang lebih tinggi (Zhao et al., 2024)

### 1.2.10 *PhysioNet*

*PhysioNet* adalah bagian dari proyek *Research Resource for Complex Physiologic Signals* yang dikelola oleh *MIT Lab for Computational Physiology*, didukung oleh *National Institutes of Health (NIH)*. Didirikan pada tahun 1999, *PhysioNet* bertujuan untuk menyediakan data fisiologis kompleks dan alat untuk analisis sinyal dan sistem biologis. Ini adalah inisiatif terbuka yang memungkinkan peneliti di seluruh dunia untuk berbagi, mengakses, dan menganalisis data medis dan fisiologis.

#### A. Fitur Utama *PhysioNet*

##### 1. Akses ke *Dataset* Medis dan Fisiologis Besar

*PhysioNet* menyediakan berbagai dataset medis yang kaya dan kompleks, yang digunakan untuk berbagai tujuan penelitian di bidang biomedis, kecerdasan buatan, dan ilmu data kesehatan. Beberapa dataset paling terkenal di platform ini termasuk:

- a) *MIMIC-III (Medical Information Mart for Intensive Care)*: Kumpulan data besar dari catatan pasien ICU, termasuk data demografis, vitals, obat-obatan, dan catatan klinis. Digunakan dalam penelitian model prediksi, diagnosis, dan pengelolaan penyakit.
- b) *Cinc Challenge Datasets*: Digunakan dalam kompetisi tahunan yang mengajak peneliti untuk memecahkan masalah dalam analisis sinyal fisiologis, seperti deteksi aritmia jantung dari sinyal EKG.

##### 2. Data Beranotasi Lengkap dan Terperinci

*Dataset* di *PhysioNet* sering kali dilengkapi dengan anotasi terperinci, termasuk label diagnostik, waktu kejadian medis, atau interpretasi klinis, yang sangat berguna untuk pengembangan model prediksi berbasis *machine learning* dan *deep learning*.

##### 3. Alat dan Perangkat Lunak untuk Analisis Sinyal Fisiologis

*PhysioNet* juga menyediakan alat dan perangkat lunak untuk analisis sinyal fisiologis, seperti sinyal EKG, EEG, atau tekanan darah. Perangkat lunak ini dirancang untuk membantu peneliti mengolah, memvisualisasikan, dan mengekstrak fitur dari sinyal biologis.

##### 4. *Open Access*

Semua data dan alat yang tersedia di *PhysioNet* dapat diakses secara terbuka, meskipun beberapa dataset tertentu memerlukan prosedur persetujuan karena adanya sensitivitas data pasien. Kebijakan akses terbuka ini memungkinkan kolaborasi lintas institusi dan negara, serta mempercepat kemajuan penelitian dalam berbagai bidang kesehatan.

##### 5. Komunikasi Ilmiah Aktif

*PhysioNet* telah menjadi pusat bagi komunitas global peneliti dan ilmuwan data yang bekerja di bidang pemrosesan sinyal fisiologis, kesehatan digital,

prediksi medis, dan AI dalam kesehatan. Platform ini secara rutin mengadakan kompetisi seperti *PhysioNet Challenges*, yang menantang komunitas untuk mengembangkan solusi inovatif dalam analisis sinyal fisiologis.

#### B. Penggunaan dalam Riset

*PhysioNet* mendukung berbagai penelitian, seperti:

1. Prediksi Readmisi Rumah Sakit: Banyak studi menggunakan *dataset* MIMIC-III untuk mengembangkan model prediksi berbasis *deep learning* atau *machine learning* untuk memprediksi *readmisi* pasien.
2. Analisis Sinyal Jantung: *Dataset* seperti EKG telah digunakan untuk mengembangkan algoritma yang mendeteksi aritmia, fibrilasi atrium, dan kondisi jantung lainnya.
3. Analisis Tren Klinis: *Dataset* ini sering digunakan untuk mempelajari pola klinis dalam pengobatan, seperti respon pasien terhadap obat-obatan atau prosedur ICU.

#### C. Penggunaan Dataset MIMIC-III

Salah satu *dataset* terpenting di *PhysioNet* adalah MIMIC-III, yang terdiri dari lebih dari 58.000 penerimaan pasien ICU dari 2001 hingga 2012 di Rumah Sakit Beth Israel Deaconess *Medical Center*. MIMIC-III mencakup berbagai data:

1. Catatan klinis: Ringkasan medis, pemeriksaan laboratorium, riwayat obat.
2. Data Fisiologis: Tekanan darah, EKG, dan sinyal vital lainnya.
3. Data demografis: Usia, jenis kelamin, dan diagnosis awal pasien. *Dataset* ini banyak digunakan dalam pengembangan model prediktif untuk hasil pasien, termasuk prediksi kematian, lama tinggal di ICU, serta potensi *readmisi*.

### 1.3 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan, penelitian ini merumuskan masalah bagaimana cara mengimplementasikan dan tingkat performa model XLNet-BiGRU-ATT dalam memprediksi kemungkinan keberlanjutan rawat inap pasien setelah pemulangan dengan memanfaatkan catatan klinis dan data medis?

### 1.4 Tujuan dan Manfaat

#### 1.4.1 Tujuan

Berdasarkan rumusan masalah yang ada, maka tujuan dari penelitian ini adalah untuk mengetahui cara mengimplementasikan dan tingkat performa model XLNet-BiGRU-ATT dalam memprediksi kemungkinan keberlanjutan rawat inap pasien setelah pemulangan dengan memanfaatkan catatan klinis dan data medis.

### 1.4.2 Manfaat

Penelitian ini diharapkan dapat memberikan manfaat sebagai berikut:

1. Memberikan informasi sebagai dasar pengambilan keputusan atau pertimbangan dalam proses pemulangan pasien.
2. Peningkatan efisiensi waktu pembacaan (analisa) catatan klinis pasien, penelitian ini diharapkan dapat membantu pertimbangan dokter atau petugas medis yang bekerja di unit perawatan intensif, yang perlu mengambil keputusan berdasarkan catatan klinis dalam jumlah besar, sehingga dapat menghemat waktu dan tenaga dokter.

### 1.5 Ruang Lingkup

Adapun runang lingkup dalam penelitian ini adalah sebagai berikut:

1. Model klasifikasi untuk memprediksi pasien mana yang berisiko untuk masuk kembali ke rumah sakit dalam waktu 30 hari yang tidak direncanakan dengan menggunakan catatan klinis pasien dalam bentuk teks bebas.
2. Data yang digunakan berupa *database* MIMIC-III (*Medical Information Mart for Intensive Care III*). Basis data rumah sakit yang berisi data yang tidak teridentifikasi dari lebih dari 40.000 pasien yang dirawat di unit perawatan kritis di Beth Israel Deaconess *Medical Center* di Boston, Massachusetts dari tahun 2001 hingga 2012. *Database* ini mencakup informasi seperti demografi, pengukuran tanda vital yang dilakukan di samping tempat tidur (~1 titik data per jam), hasil tes laboratorium, prosedur, pengobatan, catatan perawat, laporan pencitraan, dan kematian (termasuk keluar dari rumah sakit).

Sebelum data dimasukkan ke dalam *database* MIMIC-III, data tersebut terlebih dahulu diidentifikasi sesuai dengan standar Undang-Undang Portabilitas dan Akuntabilitas Asuransi Kesehatan atau *Health Insurance Portability and Accountability Act* (HIPAA) menggunakan pembersihan data terstruktur dan pergeseran tanggal.

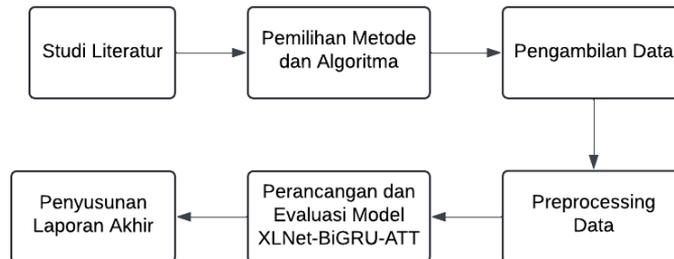
Proyek *database* ini disetujui oleh *Institutional Review Boards* dari Beth Israel Deaconess *Medical Center* (Boston, MA) dan Massachusetts *Institute of Technology* (Cambridge, MA).

3. Keluaran sistem berupa prediksi keberlanjutan rawat inap pasien rumah sakit dengan catatan klinis.

## BAB II METODE PENELITIAN

### 2.1 Tahapan Penelitian

Penelitian ini akan berlangsung dalam beberapa alur tahapan yang terdapat pada Gambar 5.



Gambar 5. Tahapan penelitian

#### 1. Studi Literatur

Pada tahap ini, dilakukan tinjauan literatur untuk mengumpulkan informasi dan penelitian yang relevan tentang prediksi penerimaan kembali pasien rumah sakit, teknik *Natural language processing* (NLP), model *deep learning* seperti XLNet, BiGRU (*Bidirectional Gated Recurrent Unit*), dan *attention mechanism*. Tujuannya adalah memahami dasar teori, metode yang telah ada, dan menemukan gap penelitian yang bisa dijadikan dasar skripsi.

#### 2. Pemilihan Metode dan Algoritma

Setelah melakukan studi literatur, langkah berikutnya adalah memilih metode dan algoritma yang paling sesuai untuk menyelesaikan masalah prediksi penerimaan kembali pasien. Pada kasus ini, dipilih XLNet sebagai model *language representation*, BiGRU untuk menangani urutan data dengan memperhitungkan konteks sebelumnya dan sesudahnya, dan *attention mechanism* untuk memberikan bobot pada informasi penting dalam catatan klinis.

#### 3. Pengambilan Data

Tahap ini melibatkan pengumpulan data catatan klinis pasien. Data yang dikumpulkan bisa berupa data tekstual dari rekam medis elektronik atau *electronic health records* (EHR) yang berisi informasi diagnosis, perawatan, riwayat medis, dan lainnya. Data harus mencakup informasi tentang pasien yang diterima kembali di rumah sakit setelah perawatan.

#### 4. Preprocessing Data

Data yang telah dikumpulkan kemudian diproses untuk meningkatkan kualitas dan mempersiapkannya untuk analisis lebih lanjut. Tahap preprocessing meliputi pembersihan data (menghapus data yang tidak lengkap atau salah), normalisasi (menyamakan format data), tokenisasi (memecah teks menjadi kata-kata atau token)

dan encoding (mengubah teks menjadi representasi numerik yang bisa diproses oleh model deep learning).

### 5. Perancangan dan Evaluasi Model XLNet-BiGRU-ATT

Tahap ini mencakup desain arsitektur sistem dan evaluasi model prediksi. Model XLNet digunakan untuk mengubah catatan klinis menjadi representasi vektor, kemudian BiGRU digunakan untuk menangani urutan data dengan konteks dari kedua arah (*forward* dan *backward*). *Attention mechanism* diterapkan untuk memfokuskan perhatian pada bagian penting dari catatan klinis. Sistem ini diimplementasikan menggunakan *framework deep learning* yaitu *PyTorch*, dan dilatih menggunakan data yang telah di-*preprocessing*.

### 6. Penyusunan Laporan Akhir

Langkah terakhir adalah menyusun laporan akhir yang mendokumentasikan seluruh proses penelitian, mulai dari studi literatur, pemilihan metode, pengambilan dan preprocessing data, hingga perancangan dan evaluasi sistem. Laporan akhir juga mencakup hasil eksperimen, analisis kinerja model, dan kesimpulan dari penelitian.

## 2.2 Waktu dan Lokasi Penelitian

Waktu penelitian dimulai sejak disetujuinya proposal penelitian ini, dimulai pada bulan Juni 2024. Penelitian ini dilaksanakan di Laboratorium *Artificial Intelligence* (AI), Departemen Teknik Informatika, Fakultas Teknik, Universitas Hasanuddin, Kelurahan Romang Lompoa, Kecamatan Bontomarannu, Kabupaten Gowa.



Gambar 6. Lokasi penelitian

## 2.3 Instrumen Penelitian

Instrument penelitian yang digunakan dalam penelitian ini yaitu:

### A. Software:

1. *Windows 10x64*
2. *Jupyter Notebook*
3. *Google Colab*
4. *Microsoft (Word, Excel dan Power Point)*
5. *Python 3.11.2*
7. *PyTorch*

## B. Hardware:

1. ASUS X540UBR Intel® Core™ i5-6200U CPU @2.30GHz 2.4GHz
2. GPU NVIDIA A100 (colab hardware):
  - a) Memori GPU (VRAM): 80 GB HBM2e (*High Bandwidth Memory*)
  - b) RAM Sistem (CPU Memory): 52 GB

## 2.4 Teknik Pengambilan Data

Data yang digunakan dalam penelitian ini berupa *database* MIMIC-III (*Medical Information Mart for Intensive Care III*). Basis data rumah sakit yang berisi data yang tidak teridentifikasi dari lebih dari 40.000 pasien yang dirawat di unit perawatan kritis di Beth Israel Deaconess *Medical Center* di Boston, Massachusetts dari tahun 2001 hingga 2012. *Database* ini mencakup informasi seperti demografi, pengukuran tanda vital yang dilakukan di samping tempat tidur (~1 titik data per jam), hasil tes laboratorium, prosedur, pengobatan, catatan perawat, laporan pencitraan, dan kematian (termasuk keluar dari rumah sakit). Sebelum data dimasukkan ke dalam *database* MIMIC-III, data tersebut terlebih dahulu diidentifikasi sesuai dengan standar Undang-Undang Portabilitas dan Akuntabilitas Asuransi Kesehatan atau *Health Insurance Portability and Accountability Act* (HIPAA) menggunakan pembersihan data terstruktur dan pergeseran tanggal. Proyek *database* ini disetujui oleh *Institutional Review Boards* dari Beth Israel Deaconess *Medical Center* (Boston, MA) dan Massachusetts *Institute of Technology* (Cambridge, MA).

Berikut adalah beberapa komponen utama dari data yang terdapat dalam MIMIC-III:

1. Demografi Pasien:  
Informasi dasar seperti usia, jenis kelamin, ras, tanggal lahir, dan tanggal kematian.
2. Data ICU:  
Informasi tentang rawat inap di ICU termasuk tanggal masuk dan keluar, jenis ICU, serta lama tinggal.
3. Catatan Klinik:  
Catatan teks bebas yang berisi catatan harian dokter, perawat, dan laporan radiologi.
4. Data Laboratorium:  
Hasil tes laboratorium seperti tes darah, tes urin, dan parameter biokimia lainnya.
5. Pengukuran Vital:  
Data vital *signs* seperti tekanan darah, denyut jantung, suhu tubuh, dan laju pernapasan yang dicatat secara berkala.
6. Pemeriksaan Pencitraan:  
Informasi tentang prosedur pencitraan seperti X-ray, CT scan, MRI, termasuk laporan interpretasi.
7. Obat-obatan:

Informasi tentang obat yang diberikan termasuk nama obat, dosis, waktu pemberian, dan *route* administrasi.

8. **Prosedur:**  
Data tentang prosedur medis dan bedah yang dilakukan pada pasien selama di ICU.
9. **Diagnosis:**  
Kode diagnosis (misalnya ICD-9) yang diberikan kepada pasien selama rawat inap di ICU.
10. **Input/Output Cairan:**  
Catatan tentang cairan yang masuk (misalnya, infus) dan keluar (misalnya, urin) dari tubuh pasien.
11. **Scoring dan Skala:**  
Skor klinis dan skala penilaian seperti APACHE, SAPS, SOFA yang digunakan untuk menilai kondisi pasien.
12. **Intervensi Mekanis:**  
Informasi tentang penggunaan alat bantu mekanis seperti ventilator, dialisis, dan perangkat lain yang digunakan selama perawatan di ICU.

Data yang digunakan dalam penelitian ini adalah data dalam tabel *ADMISSIONS* dan *NOTEVENTS* yang terdapat dalam *dataset* MIMIC III yang relevan dengan penelitian terkait prediksi penerimaan kembali pasien.

#### A. Tabel *ADMISSIONS*:

Tabel *admissions* mencatat informasi terkait setiap kali pasien masuk rumah sakit. Kolom-kolom penting dalam tabel ini meliputi:

Tabel 1. Atribut *admissions*

<i>Attribute</i>	<i>Information</i>
<i>HADM_ID</i>	ID unik untuk setiap rawat inap ( <i>hospital admission</i> )
<i>SUBJECT_ID</i>	ID unik untuk setiap pasien
<i>ADMITTIME</i>	Waktu dan tanggal ketika pasien diterima di rumah sakit
<i>DISCHTIME</i>	Waktu dan tanggal ketika pasien dipulangkan dari rumah sakit
<i>DEATHTIME</i>	Waktu kematian, jika pasien meninggal selama rawat inap
<i>ADMISSION_TYPE</i>	Jenis penerimaan (misalnya, elective, emergency)
<i>DIAGNOSIS</i>	Diagnosis utama terkait penerimaan pasien

Tabel ini berfungsi untuk melacak waktu rawat inap, penyebab rawat inap, dan apakah pasien kembali dirawat dalam jangka waktu tertentu setelah dipulangkan. Berikut contoh data pasien yang terdapat pada tabel *admissions*:

ROW_ID	SUBJECT_ID	HADM_ID	ADMITTIME	DISCHTIME	DEATHTIME	ADMISSION_TYPE	ADMISSION_LOCATION	DISCHARGE_LOCATION	INSURANCE	LANGUAGE	RELIGION
21	22	165315	2196-04-09 12:26:00	2196-04-10 15:54:00		EMERGENCY	EMERGENCY ROOM ADMIT	DISC-TRAN CANCER/CHLDRN H	Private		UNOBTAINABLE
22	23	152223	2153-09-03 07:15:00	2153-09-08 19:10:00		ELECTIVE	PHYS REFERRAL/NORMAL DELI	HOME HEALTH CARE	Medicare		CATHOLIC
23	23	124321	2157-10-18 19:34:00	2157-10-25 14:00:00		EMERGENCY	TRANSFER FROM HOSP/EXTRAM	HOME HEALTH CARE	Medicare	ENGL	CATHOLIC
24	24	161859	2139-06-06 16:14:00	2139-06-09 12:48:00		EMERGENCY	TRANSFER FROM HOSP/EXTRAM	HOME	Private		PROTESTANT QUAKER
25	25	129635	2160-11-02 02:06:00	2160-11-05 14:55:00		EMERGENCY	EMERGENCY ROOM ADMIT	HOME	Private		UNOBTAINABLE
26	26	197661	2126-05-06 15:16:00	2126-05-13 15:00:00		EMERGENCY	TRANSFER FROM HOSP/EXTRAM	HOME	Medicare		CATHOLIC
27	27	134931	2191-11-30 22:16:00	2191-12-03 14:45:00		NEWBORN	PHYS REFERRAL/NORMAL DELI	HOME	Private		CATHOLIC
28	28	162569	2177-09-01 07:15:00	2177-09-06 16:00:00		ELECTIVE	PHYS REFERRAL/NORMAL DELI	HOME HEALTH CARE	Medicare		CATHOLIC

Gambar 7. Contoh data pasien

## B. Tabel NOTEEVENTS

Tabel *noteevents* berisi catatan klinis (*clinical notes*) yang ditulis oleh tenaga kesehatan selama pasien menjalani perawatan. Kolom-kolom penting dalam tabel ini meliputi:

Tabel 2. Atribut *noteevents*

Attribute	Information
<i>SUBJECT_ID</i>	ID unik untuk setiap pasien
<i>HADM_ID</i>	ID unik untuk rawat inap yang terkait dengan catatan tersebut
<i>CHARTTIME</i>	Waktu dan tanggal catatan dibuat
<i>CATEGORY</i>	Kategori catatan (misalnya, <i>discharge summary</i> , <i>radiology report</i> , <i>nursing notes</i> )
<i>TEXT</i>	Isi catatan klinis dalam format teks yang mencakup informasi tentang status kesehatan pasien, tindakan medis, diagnosis, dan perawatan yang diberikan

Tabel ini berisi data yang kaya informasi berupa teks tidak terstruktur yang mencerminkan kondisi pasien selama dirawat. Berikut contoh data pasien yang terdapat pada tabel *noteevents*:

STORETIME	CATEGORY	DESCRIPTION	CGID	ISERROR	TEXT
	Discharge summary	Report			Admission Date: [**2151-7-16**] Discharge Date: [**2151-8-4**]Service:ADDENDUM:RADIOLOGIC STUDIES: Radiologic studies also included a chestCT, which confirmed cavitary lesions in the left lung apexconsistent with infectious process/tuberculosis. This alsomoderate-sized left pleural effusion.HEAD CT: Head CT showed no intracranial hemorrhage or masseffect, but old infarction consistent with past medicalhistory.ABDOMINAL CT: Abdominal CT showed lesions ofT10 and sacrum most likely secondary to osteoporosis. These canbe followed by repeat imaging as an outpatient. [**First Name8 (NamePattern2) **] [**First Name4 (NamePattern1) 1775**] [**Last Name (NamePattern1) **], M.D. [**MD Number(1) 1776**]Dictated By:[**Hospital 1807**]MEDQUIST36D: [**2151-8-5**] 12:11T: [**2151-8-5**] 12:21JOB#: [**Job Number 1808**]

Gambar 8. Catatan klinis pasien

Untuk memprediksi penerimaan kembali pasien, data dari tabel *ADMISSIONS* dan *NOTEEVENTS* dapat digunakan dalam kombinasi:

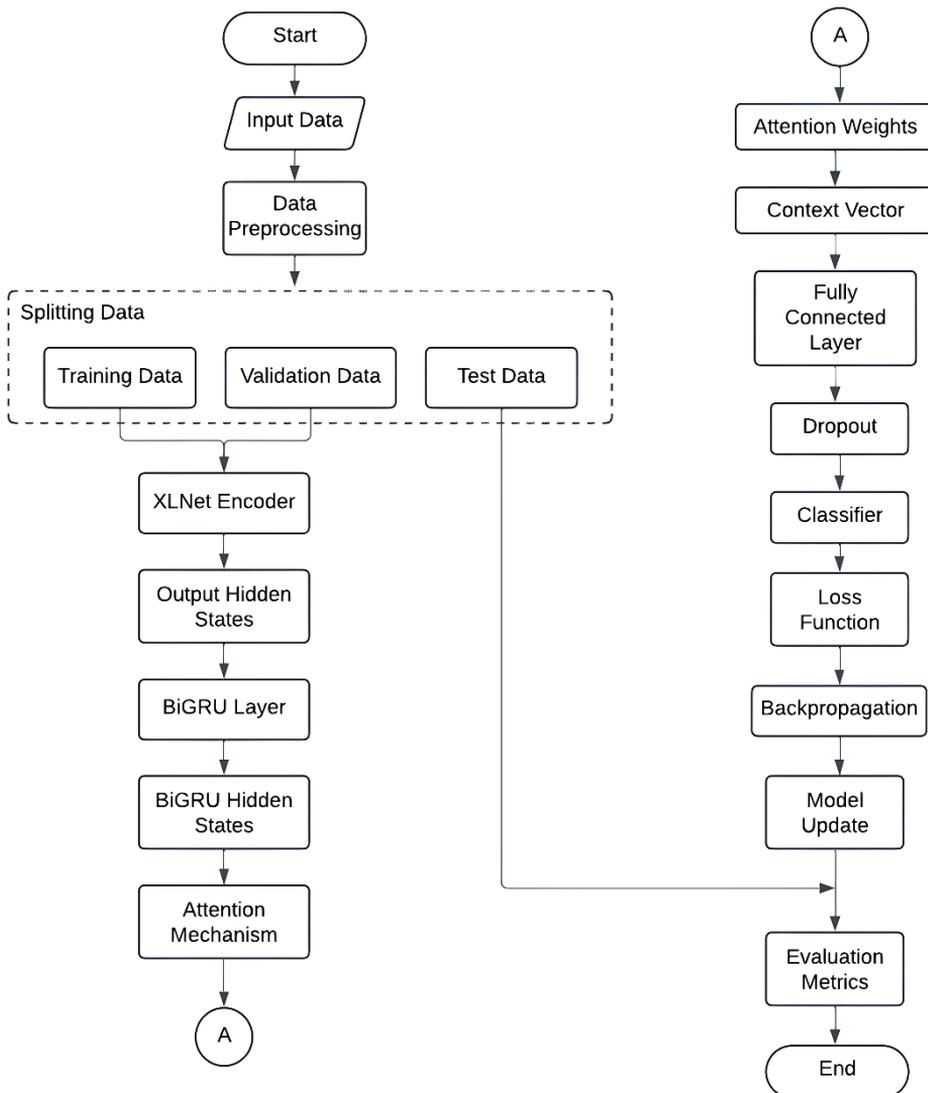
1. Mengidentifikasi Periode Waktu antara Rawat Inap:  
Menggunakan kolom *ADMITTIME* dan *DISCHTIME* dari tabel *ADMISSIONS*, sehingga dapat menghitung waktu sejak pasien dipulangkan hingga mereka kembali dirawat. Penerimaan kembali sering didefinisikan sebagai rawat inap yang terjadi dalam 30 hari setelah pasien dipulangkan.
2. Menggabungkan Catatan Klinis untuk Membuat Fitur:
  - a) Tabel *NOTEEVENTS*, terutama catatan yang berada dalam kategori *discharge summary*, digunakan untuk mengekstraksi informasi teks penting mengenai status pasien pada saat dipulangkan.
  - b) Model prediktif memanfaatkan catatan medis ini untuk mendeteksi pola atau kata-kata kunci yang terkait dengan risiko penerimaan kembali.
3. Membangun Model Prediksi:
  - a) Catatan klinis dari *NOTEEVENTS* diolah menggunakan teknik pemrosesan bahasa alami atau *natural language processing* (NLP). Arsitektur model XLNet-BiGRU-ATT digunakan untuk memahami konteks teks klinis secara lebih baik.
  - b) Selain itu, fitur numerik dari *ADMISSIONS* seperti diagnosis, jenis penerimaan (*emergency/elective*), dan informasi demografis juga dimasukkan sebagai input tambahan dalam model untuk meningkatkan akurasi prediksi.

Dengan menggabungkan data dari tabel *ADMISSIONS* yang memberikan informasi numerik dan waktu serta *NOTEEVENTS* yang berisi data teks tidak terstruktur, model prediksi penerimaan kembali pasien dapat dibuat dengan lebih

akurat. Model ini membantu memprediksi pasien mana yang memiliki risiko tinggi untuk kembali dirawat dalam waktu dekat setelah keluar dari rumah sakit, memungkinkan tindakan pencegahan lebih dini dilakukan.

## 2.5 Pelaksanaan Penelitian

Pelaksanaan penelitian terbagi ke dalam beberapa bagian, seperti *preprocessing data*, proses pelatihan model, proses evaluasi model & pengujian pada *test data*. Gambar 9 memperlihatkan seluruh alur pelaksanaan penelitian yang akan dilakukan.



Gambar 9. Pelaksanaan penelitian

### 2.5.1 *Preprocessing Data*

*Preprocessing data* adalah tahap awal dalam proses pengolahan data yang bertujuan untuk membersihkan, mengorganisir, dan mempersiapkan data agar siap untuk digunakan dalam analisis lebih lanjut. Dalam konteks prediksi penerimaan kembali pasien rumah sakit dengan catatan klinis menggunakan teknik NLP, beberapa langkah yang perlu dilakukan dalam pra-pemrosesan data antara lain:

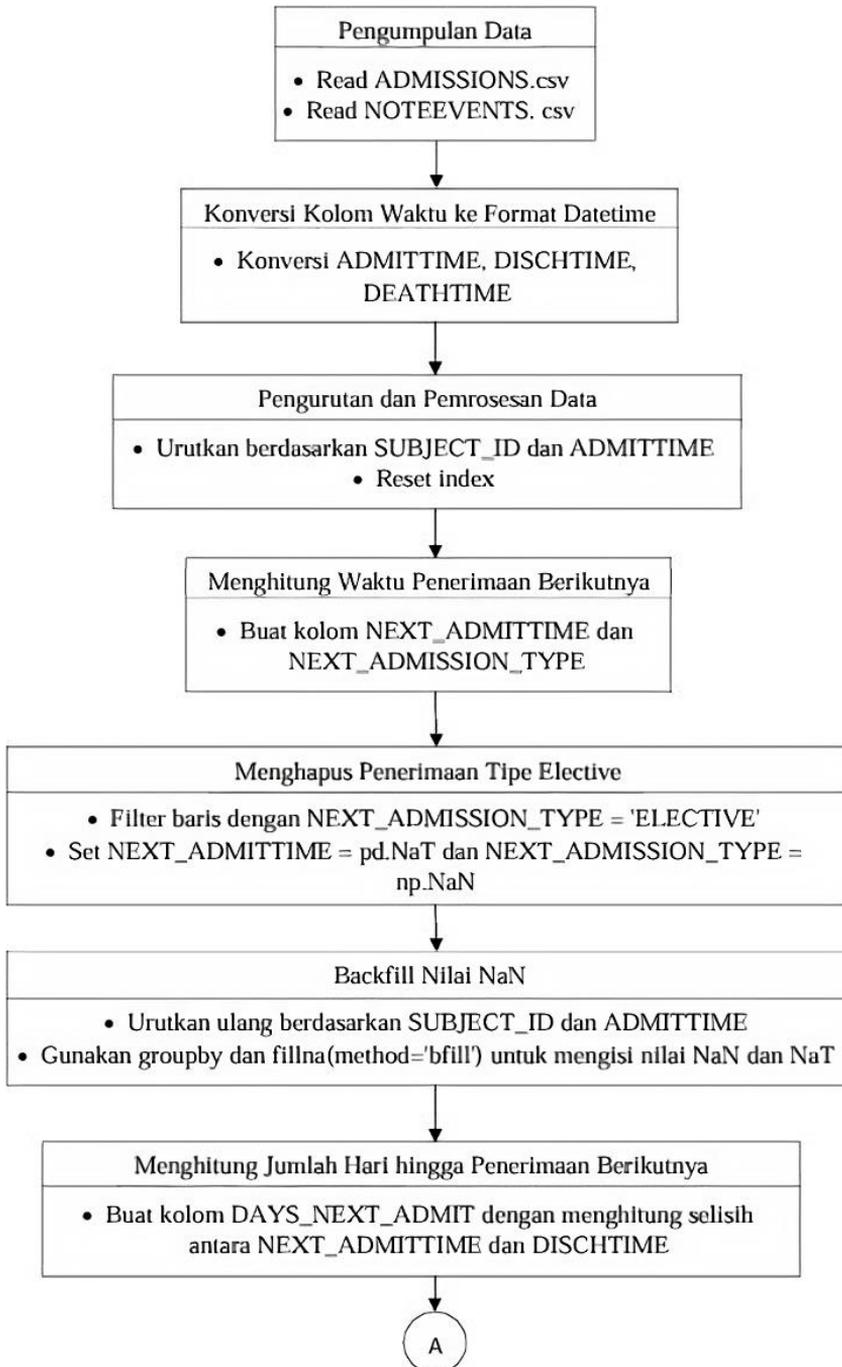
#### 1. Normalisasi Data

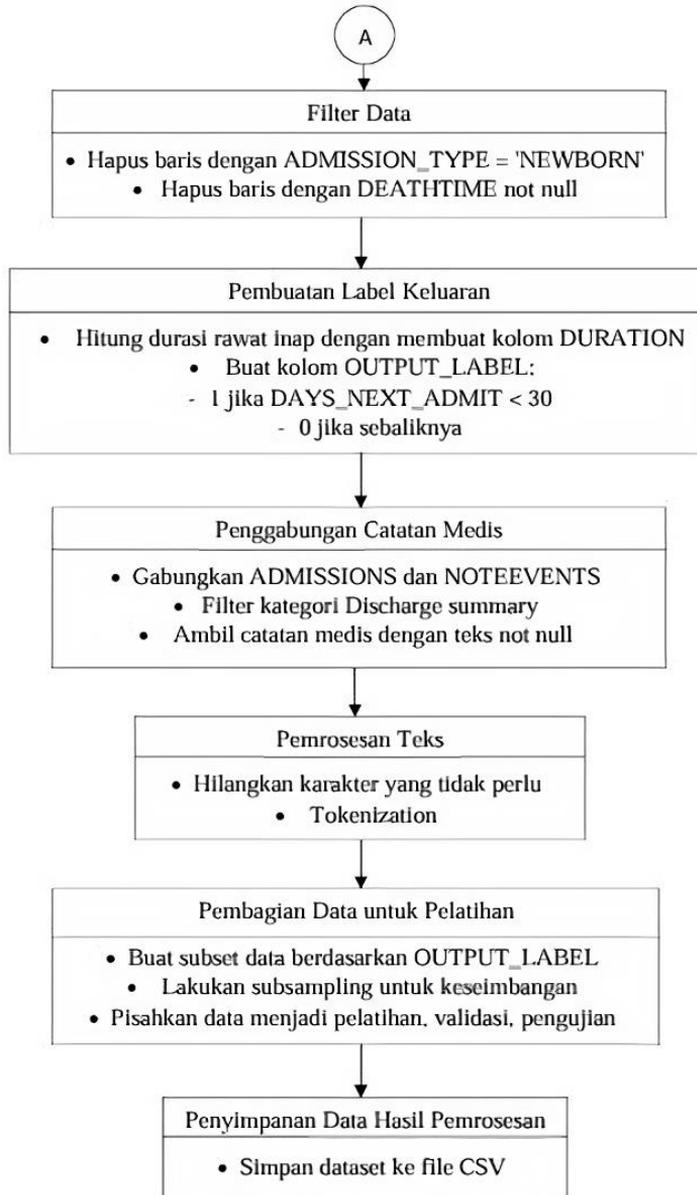
Normalisasi Data adalah proses mengubah data ke dalam format yang seragam atau standar. Dalam konteks klinis, catatan medis memiliki berbagai format dan representasi yang berbeda-beda. Oleh karena itu, normalisasi data bisa meliputi langkah-langkah seperti standarisasi format tanggal, penyesuaian skala, atau penanganan nilai yang hilang.

#### 2. Pemilihan Atribut yang Paling Relevan untuk Analisis

Dalam analisis prediksi penerimaan kembali pasien, tidak semua informasi dalam catatan medis relevan. Oleh karena itu, penting untuk memilih atribut-atribut yang paling relevan atau informatif untuk memprediksi kemungkinan penerimaan kembali pasien. Misalnya, diagnosis awal, riwayat medis, prosedur medis yang dilakukan, dan instruksi tindak lanjut setelah pemulangan dapat menjadi atribut yang relevan untuk dipertimbangkan.

Gambar 10 menunjukkan proses pengumpulan dan pemrosesan data hingga penyimpanan dataset yang telah diproses untuk digunakan dalam pelatihan model prediksi penerimaan kembali pasien rumah sakit.





Gambar 10. *Preprocessing data*

#### 1. Pengumpulan Data:

Mengumpulkan data dari file `ADMISSIONS.csv` dan `NOTEVENTS.csv` untuk mendapatkan informasi tentang pasien dan catatan medis.

ROW_ID	SUBJECT_ID	HADM_ID	ADMITTIME	DISCHTIME	DEATHTIME	ADMISSION_TYPE	ADMISSION_LOCATION	DISCHARGE_LOCATION	INSURANCE	LF
0	21	22	165315	2196-04-09 12:26:00	2196-04-10 15:54:00	NaN	EMERGENCY	EMERGENCY ROOM ADMIT	DISC-TRAN CANCER/CHLDRN H	Private
1	22	23	152223	2153-09-03 07:15:00	2153-09-08 19:10:00	NaN	ELECTIVE	PHYS REFERRAL/NORMAL DELI	HOME HEALTH CARE	Medicare
2	23	23	124321	2157-10-18 19:34:00	2157-10-25 14:00:00	NaN	EMERGENCY	TRANSFER FROM HOSP/EXTRAM	HOME HEALTH CARE	Medicare
3	24	24	161859	2139-06-06 16:14:00	2139-06-09 12:48:00	NaN	EMERGENCY	TRANSFER FROM HOSP/EXTRAM	HOME	Private
4	25	25	129635	2160-11-02 02:06:00	2160-11-05 14:55:00	NaN	EMERGENCY	EMERGENCY ROOM ADMIT	HOME	Private

Gambar 11. Data tabel *admissions*

```
df_notes.head()
```

ROW_ID	SUBJECT_ID	HADM_ID	CHARTDATE	CHARTTIME	STORETIME	CATEGORY	DESCRIPTION	CGID	ISERROR	TEXT	
0	174	22532	167853.0	2151-08-04	NaN	NaN	Discharge summary	Report	NaN	NaN	Admission Date: [**2151-7-16**] Dischar...
1	175	13702	107527.0	2118-06-14	NaN	NaN	Discharge summary	Report	NaN	NaN	Admission Date: [**2118-6-2**] Discharg...
2	176	13702	167118.0	2119-05-25	NaN	NaN	Discharge summary	Report	NaN	NaN	Admission Date: [**2119-5-4**] D...
3	177	13702	196489.0	2124-08-18	NaN	NaN	Discharge summary	Report	NaN	NaN	Admission Date: [**2124-7-21**] ...
4	178	26880	135453.0	2162-03-25	NaN	NaN	Discharge summary	Report	NaN	NaN	Admission Date: [**2162-3-3**] D...

Gambar 12. Data tabel *noteevents*

2. Konversi Kolom Waktu:

Mengubah kolom yang berkaitan dengan waktu ke format *datetime* untuk memudahkan analisis waktu dan durasi.

3. Pengurutan dan Pemrosesan Data:

Mengurutkan data berdasarkan ``SUBJECT_ID`` dan ``ADMITTIME`` untuk memastikan urutan kronologis dari penerimaan pasien.

	SUBJECT_ID	ADMITTIME	ADMISSION_TYPE
165	124	2160-06-24 21:25:00	EMERGENCY
166	124	2161-12-17 03:39:00	EMERGENCY
167	124	2165-05-21 21:02:00	ELECTIVE
168	124	2165-12-31 18:55:00	EMERGENCY

Gambar 13. Urutan kronologis penerimaan pasien

4. Menghitung Waktu Penerimaan Berikutnya:

Menambahkan kolom yang mencatat waktu dan jenis penerimaan berikutnya untuk setiap pasien.

	SUBJECT_ID	ADMITTIME	ADMISSION_TYPE	NEXT_ADMITTIME	NEXT_ADMISSION_TYPE
165	124	2160-06-24 21:25:00	EMERGENCY	2161-12-17 03:39:00	EMERGENCY
166	124	2161-12-17 03:39:00	EMERGENCY	2165-05-21 21:02:00	ELECTIVE
167	124	2165-05-21 21:02:00	ELECTIVE	2165-12-31 18:55:00	EMERGENCY
168	124	2165-12-31 18:55:00	EMERGENCY	NaN	NaN

Gambar 14. Menambahkan *next admission type*

### 5. Menghapus Penerimaan Tipe *Elective*:

Memfilter penerimaan yang bersifat *elective* (direncanakan) untuk fokus pada penerimaan yang tidak direncanakan.

	SUBJECT_ID	ADMITTIME	ADMISSION_TYPE	NEXT_ADMITTIME	NEXT_ADMISSION_TYPE
165	124	2160-06-24 21:25:00	EMERGENCY	2161-12-17 03:39:00	EMERGENCY
166	124	2161-12-17 03:39:00	EMERGENCY	NaT	NaN
167	124	2165-05-21 21:02:00	ELECTIVE	2165-12-31 18:55:00	EMERGENCY
168	124	2165-12-31 18:55:00	EMERGENCY	NaT	NaN

Gambar 15. Memfilter penerimaan *elective*

### 6. *Backfill* Nilai:

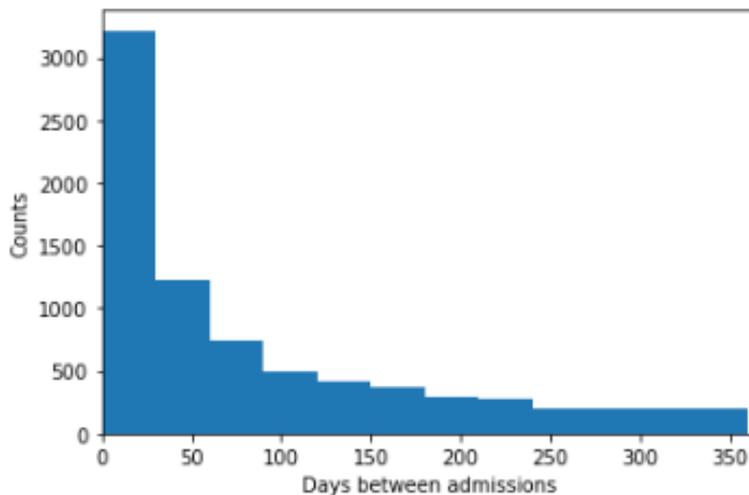
Mengisi nilai yang hilang dengan informasi yang relevan untuk memastikan kelengkapan data.

	SUBJECT_ID	ADMITTIME	ADMISSION_TYPE	NEXT_ADMITTIME	NEXT_ADMISSION_TYPE
165	124	2160-06-24 21:25:00	EMERGENCY	2161-12-17 03:39:00	EMERGENCY
166	124	2161-12-17 03:39:00	EMERGENCY	2165-12-31 18:55:00	EMERGENCY
167	124	2165-05-21 21:02:00	ELECTIVE	2165-12-31 18:55:00	EMERGENCY
168	124	2165-12-31 18:55:00	EMERGENCY	NaT	NaN

Gambar 16. *Backfill* nilai

### 7. Menghitung Jumlah Hari hingga Penerimaan Berikutnya:

Menentukan berapa hari hingga penerimaan berikutnya untuk memahami pola penerimaan ulang pasien.



Gambar 17. Plot histogram hari antara penerimaan kembali pasien

### 8. Filter Data:

Menghapus data yang tidak relevan atau yang bisa mengganggu analisis, seperti data bayi baru lahir dan pasien yang meninggal.

### 9. Pembuatan Label Keluaran:

Membuat label keluaran (*output label*) untuk membedakan pasien yang akan diterima kembali dalam waktu kurang dari 30 hari dari yang tidak, guna keperluan model prediktif.

10. Penggabungan Catatan Medis:

Menggabungkan informasi penerimaan pasien dengan catatan medis untuk analisis yang lebih komprehensif.

11. Pemrosesan Teks:

Membersihkan dan menyiapkan teks catatan medis untuk analisis tekstual, seperti tokenisasi.

12. Pembagian Data:

Membagi data menjadi set pelatihan, validasi, dan pengujian untuk membangun dan menguji model prediktif.

- a) Data Pelatihan (*Training*): Data pelatihan digunakan untuk melatih model prediktif. Model XLNet-BiGRU-ATT akan mempelajari pola dari data pelatihan ini untuk membuat prediksi di masa depan.
- b) Data Validasi: Setelah melatih model dengan data training, perlu mengukur kinerja model di luar sampel data tersebut. Data validasi digunakan untuk mengevaluasi kinerja model selama pelatihan.
- c) Data Pengujian (*Testing*): Setelah model dilatih menggunakan data pelatihan, data pengujian digunakan untuk menguji kinerja model. Data ini tidak digunakan dalam proses pelatihan agar dapat memberikan evaluasi yang obyektif terhadap kemampuan prediksi model. Evaluasi tersebut membantu memastikan bahwa model tidak hanya belajar dari data yang telah dilihat sebelumnya, tetapi juga mampu menggeneralisasi dan membuat prediksi yang akurat terhadap data baru.

Hasil dari pembagian dataset terdapat di tabel 3.

Tabel 3. Hasil pembagian *dataset*

Data	Hasil Pembagian <i>Dataset</i>	Persentase
<i>Training</i>	26245	80%
<i>Validation</i>	3037	10%
<i>Test</i>	3063	10%

13. Penyimpanan Data Hasil Pemrosesan:

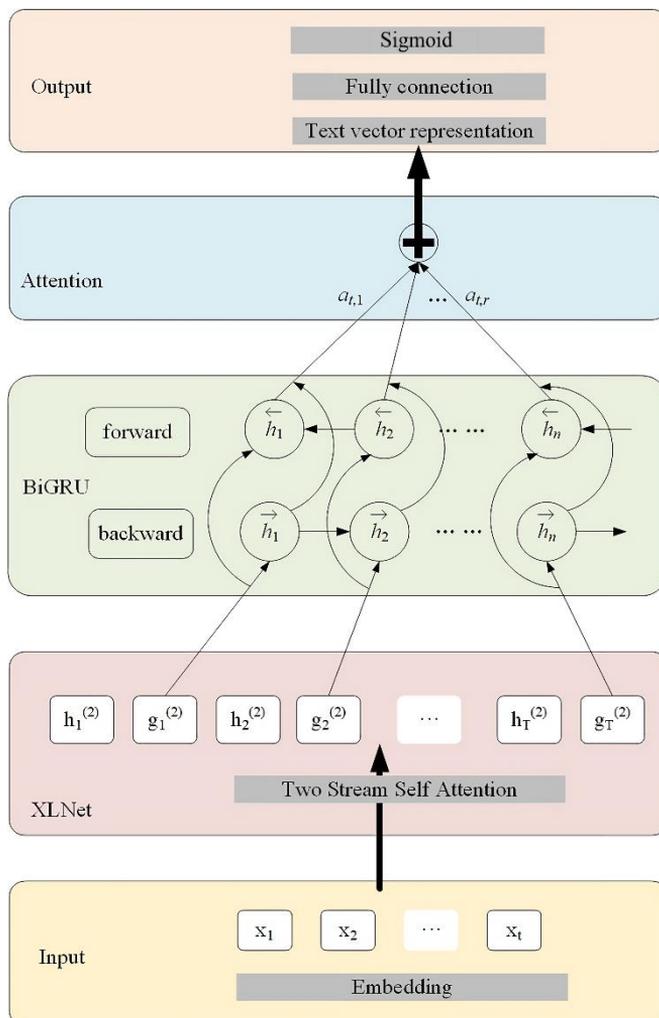
Menyimpan dataset yang telah diproses untuk digunakan dalam analisis lebih lanjut atau pembangunan model.

Secara keseluruhan, tujuan dari proses ini adalah untuk mempersiapkan data yang bersih dan terstruktur agar dapat digunakan dalam membangun model prediktif untuk mengidentifikasi pasien yang berisiko tinggi kembali dirawat dalam waktu 30 hari setelah dipulangkan.

## 2.5.2 Training dan Implementasi Model

### 2.5.2.1 Model XLNet

Gambar 18. Menunjukkan arsitektur model XLNet-BiGRU-ATT.



Gambar 18. Aritektur XLNet-BiGRU-ATT

Arsitektur model XLNet-BiGRU-ATT dikembangkan berdasarkan penelitian Han et al., (2023) dengan modifikasi yang bertujuan untuk meningkatkan kinerja dalam tugas klasifikasi biner. Perubahan kunci yang diterapkan adalah penggantian fungsi aktivasi softmax pada lapisan output dengan sigmoid. Fungsi sigmoid lebih cocok untuk klasifikasi biner karena memprediksi probabilitas setiap kelas secara independen, menghasilkan output antara 0 hingga 1. Berbeda dengan softmax, yang membatasi total probabilitas antar kelas menjadi 1, sigmoid memberikan fleksibilitas lebih besar dalam memprediksi probabilitas setiap kelas, yang diharapkan dapat meningkatkan efektivitas model.

XLNet dikembangkan untuk menangkap relasi antara kata dalam urutan teks secara lebih kompleks dibandingkan metode *masked language model* konvensional seperti BERT. XLNet dilatih dengan teknik *permuted language modeling*, di mana model mempelajari urutan kata dari berbagai kombinasi permutasi, sehingga mampu menangkap konteks yang lebih kaya dari teks yang bersifat urutan panjang, seperti catatan medis.

Model XLNet menerima input berupa token yang dihasilkan dari proses *tokenization* menggunakan tokenizer yang sesuai dengan XLNet. Setiap catatan medis pasien diubah menjadi sekumpulan token  $(x_1, x_2, \dots, x_t)$  yang mewakili kata atau frasa dalam catatan tersebut. Token ini kemudian diubah menjadi embedding vektor melalui lapisan *embedding* dari XLNet.

Arsitektur XLNet dilengkapi dengan beberapa lapisan *self-attention*, di mana setiap kata dalam teks dapat berinteraksi secara langsung dengan semua kata lainnya. Proses ini dilakukan untuk menghasilkan representasi kata yang memperhitungkan konteks sekitarnya, yang disebut sebagai *contextualized word embeddings*.

Alur kerja arsitektur xlnet yang menggunakan "Two-Stream Self-Attention":

1. Dua Aliran Informasi (*Two-Stream Information*): Terdapat dua jenis aliran (*stream*) informasi, yaitu *content stream* dan *query stream*. Kedua aliran ini dieksplorasi secara bersamaan untuk melakukan prediksi.
2. *Masked Two-Stream Attention*: Pada bagian utama, terdapat *attention layer* yang dirancang khusus untuk bekerja dengan dua aliran ini. Setiap token yang diinputkan, seperti  $x_1, x_2, \dots$ , memiliki representasi yang diproses dalam *query stream* dan *content stream*. Pada aliran *content*, token bisa melihat informasi dari masa lalu dan dirinya sendiri, sedangkan pada *query stream*, token tidak bisa melihat dirinya sendiri (ini disebut *masking*).
3. *Positional Encoding and Factorization*: Ada juga pemilihan urutan faktorisasi, di mana contoh yang ditunjukkan dalam gambar 1 adalah " $3 \rightarrow 2 \rightarrow 4 \rightarrow 1$ ". Ini menggambarkan urutan di mana informasi token akan diolah, yang memungkinkan model menangani *permutation-based language modeling*.
4. *Masked Self-Attention*: Model ini memanfaatkan mekanisme *masked self-attention*, di mana perhatian pada token tertentu dipengaruhi oleh posisi sebelumnya, sehingga bisa memprediksi dengan lebih fleksibel.

Persamaan dasar yang digunakan dalam XLNet (9) adalah sebagai berikut:

$$h_i = \text{Attention}(x_1, x_2, \dots, x_i) \quad (9)$$

Dimana:

- a)  $x$  adalah urutan kata dalam bentuk vektor yang dihasilkan dari tokenisasi teks.
- b)  $h_i$  adalah representasi kontekstual dari kata  $i$  setelah melalui proses *self-attention*.

XLNet memiliki fungsi *feed-forward* untuk memproses *embedding* dari kata-kata secara bersamaan dengan kompleksitas waktu yang jauh lebih rendah

dibandingkan model sekuensial lainnya, seperti RNN atau GRU. Hasil dari XLNet kemudian diteruskan ke lapisan selanjutnya untuk diproses lebih lanjut.

Fungsi *loss* yang digunakan dalam arsitektur XLNet untuk klasifikasi biner ini adalah *binary cross-entropy with logits (BCEWithLogitsLoss)*, yang cocok untuk klasifikasi dengan dua kemungkinan hasil, yaitu pasien diterima kembali atau tidak. Persamaan dari fungsi *loss* (10) adalah:

$$L(y, \hat{y}) = -(y \cdot \log(\sigma(\hat{y})) + (1 - y) \cdot \log(1 - \sigma(\hat{y}))) \quad (10)$$

Dimana:

- a)  $y$  adalah label sebenarnya (0 atau 1).
- b)  $\hat{y}$  adalah logits yang dihasilkan oleh model.
- c)  $\sigma(\hat{y})$  adalah fungsi aktivasi sigmoid yang mengubah logits menjadi probabilitas.

Persamaan untuk fungsi *sigmoid*  $\sigma(z)$  (11) didefinisikan sebagai :

$$\sigma(z) = \frac{1}{1 + e^{-z}} \quad (11)$$

### 2.5.2.2 Penambahan Lapisan BiGRU (*Bidirectional Gated Recurrent Unit*)

Meskipun XLNet sangat baik dalam memahami konteks global dari teks, penggunaan lapisan BiGRU bertujuan untuk menangkap informasi sekuensial lebih lanjut dari catatan medis pasien. BiGRU (*Bidirectional GRU*) adalah modifikasi dari arsitektur *recurrent neural network* (RNN) yang memungkinkan model untuk mengingat informasi masa lalu dengan lebih efektif sambil tetap menjaga efisiensi komputasi.

Keunggulan *BiGRU* dibandingkan RNN standar adalah kemampuannya untuk memproses sekuens dari dua arah sekaligus: maju (dari awal hingga akhir teks) dan mundur (dari akhir ke awal teks). Dengan demikian, informasi kontekstual dari kata di posisi sebelumnya dan sesudahnya dapat diambil secara lebih akurat. Pada lapisan BiGRU, keluaran dari arah maju dan mundur digabungkan untuk setiap waktu  $t$ , sehingga menghasilkan representasi tersembunyi  $h_t$  yang dirumuskan pada persamaan (12) sebagai berikut:

$$h_t = [\vec{h}_t, \overleftarrow{h}_t] \quad (12)$$

Dimana:

- a)  $\vec{h}_t$  adalah representasi dari arah maju.
- b)  $\overleftarrow{h}_t$  adalah representasi dari arah mundur.
- c)  $h_t$  adalah gabungan dari keduanya.

Dengan menggunakan BiGRU, model mampu menangkap urutan kata yang lebih panjang secara lebih efektif, yang merupakan ciri khas dari catatan medis yang kompleks dan terkadang ambigu.

### 2.5.2.3 Attention Mechanism

*Attention mechanism* bertujuan untuk memfokuskan perhatian model pada kata atau frasa yang lebih penting dalam catatan medis. Dalam kasus prediksi penerimaan kembali pasien, tidak semua informasi dalam catatan medis sama pentingnya, beberapa kata atau bagian tertentu mungkin lebih relevan dalam menentukan hasil.

*Attention mechanism* memungkinkan model untuk "memperhatikan" hubungan antara setiap pasangan token dalam teks *input*, dan menentukan seberapa pentingnya setiap token dalam konteks keseluruhan teks. Dalam konteks prediksi penerimaan kembali pasien rumah sakit, *attention mechanism* akan membantu model untuk memahami hubungan antara berbagai bagian informasi pasien, seperti nama, usia, riwayat medis, gejala, dan diagnosis terakhir. Peran *attention mechanism*:

- Pemahaman Konteks: *Attention mechanism* memungkinkan model untuk memahami konteks dari setiap token dalam teks *input*. Ini memungkinkan model untuk mengetahui bagaimana informasi tentang nama pasien, usia, riwayat medis, gejala, dan diagnosis terakhir berinteraksi dan saling mempengaruhi.
- Pentingnya Informasi: *Attention mechanism* menentukan seberapa pentingnya setiap token dalam konteks teks keseluruhan. Ini memungkinkan model untuk mengetahui apakah informasi seperti riwayat medis yang mencakup diabetes tipe 2 dan hipertensi lebih relevan daripada gejala tertentu seperti sakit kepala dalam konteks prediksi penerimaan kembali pasien.
- Menggabungkan Informasi: *Attention mechanism* memungkinkan model untuk menggabungkan informasi dari berbagai bagian teks *input*, untuk membuat representasi yang lebih kaya dan informatif dari teks keseluruhan. Ini membantu model untuk membuat prediksi yang lebih akurat tentang apakah pasien akan kembali ke rumah sakit atau tidak.

Dengan menggunakan *attention mechanism*, model dapat secara efektif memproses informasi dari teks input, dan menggunakan pemahaman yang diperolehnya untuk membuat prediksi yang lebih baik dalam kasus prediksi penerimaan kembali pasien rumah sakit.

*Attention mechanism* berfungsi dengan menghitung bobot untuk setiap representasi kata dalam teks. Bobot ini menunjukkan seberapa penting kata tersebut untuk keputusan akhir. Bobot *attention*  $a_t$  untuk waktu  $t$  dihitung dalam persamaan (13) sebagai:

$$a_t = \frac{\exp(e_t)}{\sum_{i=1}^T \exp(e_i)} \quad (13)$$

Dimana:

- $a_t$  adalah skor attention yang dihitung dari representasi tersembunyi di waktu  $t$ .
- $T$  adalah panjang sekuens (jumlah kata dalam catatan medis).

Setelah mendapatkan bobot *attention*, kemudian menghitung vektor konteks  $c$  dalam persamaan (14) sebagai:

$$c = \sum_{t=1}^T a_t h_t \quad (14)$$

Dimana:

- a)  $c$  adalah vektor konteks yang dihasilkan dari penjumlahan tertimbang representasi tersembunyi oleh bobot *attention*.
- b)  $h_t$  adalah representasi tersembunyi di waktu  $t$ .

Dengan demikian, vektor konteks  $c$  mengandung informasi yang sudah dipilih secara selektif berdasarkan relevansinya terhadap tugas prediksi.

#### 2.5.2.4 Lapisan Klasifikasi dan Sigmoid

Setelah mekanisme *attention* menghasilkan vektor konteks, vektor ini diteruskan ke lapisan *fully connected* untuk menghasilkan keluaran berupa *logits*. Proses ini diwakili oleh persamaan (15) berikut:

$$z = W \cdot c + b \quad (15)$$

Dimana:

- a)  $z$  adalah *logits*.
- b)  $W$  adalah bobot lapisan *fully connected*.
- c)  $c$  adalah vektor konteks dari mekanisme *attention*.
- d)  $b$  adalah bias.

Nilai *logits* ini kemudian dilewatkan melalui fungsi sigmoid untuk mengubahnya menjadi probabilitas:

$$\hat{y} = \sigma(z) \quad (16)$$

Dimana  $\sigma(z)$  adalah fungsi aktivasi sigmoid yang sudah dijelaskan sebelumnya. Nilai  $\hat{y}$  ini kemudian digunakan untuk menentukan apakah pasien akan diterima kembali atau tidak, berdasarkan ambang batas tertentu, yang diinisialisasi ke 0.5. Secara keseluruhan inisialisasi model XLNet-BiGRU-ATT ditunjukkan pada tabel 4.

Tabel 4. Inisialisasi model XLNet-BiGRU-ATT

Layer/Block	Details
<i>transformer</i> ( <i>XLNetModel</i> )	a) <i>Embedding</i> : (32000, 768) b) <i>Layer</i> : 12 x <i>XLNetLayer</i>
<i>rel_attn</i> ( <i>XLNetRelativeAttention</i> )	c) <i>LayerNorm</i> : <i>LayerNorm</i> (768) d) <i>Dropout</i> : $p=0.1$
<i>ff</i> ( <i>XLNetFeedForward</i> )	e) <i>LayerNorm</i> : <i>LayerNorm</i> (768) f) <i>Linear</i> 1: <i>in_features</i> =768, <i>out_features</i> =3072 g) <i>Linear</i> 2: <i>in_features</i> =3072, <i>out_features</i> =768 h) <i>Dropout</i> : $p=0.1$ i) <i>Activation</i> : GELU
<i>dropout</i>	<i>Dropout</i> : $p=0.1$ , <i>inplace</i> =False
<i>sequence_summary</i>	j) <i>Linear</i> : <i>in_features</i> =768, <i>out_features</i> =768 k) <i>Activation</i> : Tanh l) <i>Dropout</i> : $p=0.1$
<i>logits_proj</i>	<i>Linear</i> : <i>in_features</i> =768, <i>out_features</i> =1
<i>bigru</i>	<i>GRU</i> : <i>in_features</i> =768, <i>hidden_size</i> =384, <i>batch_first</i> =True, <i>bidirectional</i> =True
<i>attention</i>	<i>Linear</i> : <i>in_features</i> =768, <i>out_features</i> =1
<i>fc</i>	<i>Linear</i> : <i>in_features</i> =768, <i>out_features</i> =768
<i>classifier</i>	<i>Linear</i> : <i>in_features</i> =768, <i>out_features</i> =1
<i>loss_fct</i>	<i>BCEWithLogitsLoss</i>

Alur proses yang terjadi pada tabel diatas yaitu:

#### 1. *Embedding Layer*

- a) *Layer*: *XLNetModel* -> *Embedding*(32000, 768)
- b) Fungsi: *Embedding layer* berfungsi untuk mengonversi *input* token (dalam bentuk *indeks integer*) menjadi vektor berdimensi 768. *Embedding* ini merupakan representasi distribusi dari token yang mengandung informasi semantik, yang nantinya digunakan sebagai *input* untuk lapisan-lapisan selanjutnya dalam model.

#### 2. *XLNet Layers (Self-Attention dan Feedforward)*

- a) *Layer*: *XLNetLayer*(12 layers) yang terdiri dari:
  - 1) *XLNetRelativeAttention*: Menggunakan *two-stream self-attention*, yaitu teknik *autoregressive* yang memungkinkan pemrosesan token secara kontekstual. *Attention* yang digunakan juga memperhitungkan posisi relatif antar token.
  - 2) *LayerNorm*: Normalisasi dilakukan dengan parameter *LayerNorm*(768) untuk menjaga stabilitas distribusi nilai dalam jaringan.
  - 3) *Dropout*: Digunakan untuk regularisasi dengan nilai  $p=0.1$  guna mencegah *overfitting*.

b) *XLNetFeedForward*:

- 1) *LayerNorm*: Sama seperti pada lapisan *attention*, normalisasi dilakukan untuk stabilitas.
- 2) *Linear Layer 1*: Mengubah dimensi dari 768 menjadi 3072, memperbesar representasi internal dari fitur *input*.
- 3) *Linear Layer 2*: Mengembalikan dimensi dari 3072 kembali ke 768.
- 4) *Activation (GELU)*: Fungsi aktivasi yang digunakan adalah GELU (*Gaussian Error Linear Unit*). GELU secara efektif memberikan aktivasi *non-linear* yang lebih halus dibandingkan ReLU, khususnya berguna dalam pemrosesan model berbasis NLP.
- 5) *Dropout*: *Dropout* sebesar  $p=0.1$  kembali diterapkan sebagai mekanisme regularisasi untuk mengurangi *overfitting*.

3. *Sequence Summary Layer*

*Layer: SequenceSummary*

- a) *Linear Layer*: Mengambil *output* dari lapisan XLNet dengan dimensi 768, kemudian diproyeksikan kembali ke dimensi yang sama menggunakan *Linear(in\_features=768, out\_features=768)*.
- b) *Activation (Tanh)*: Aktivasi Tanh digunakan untuk memampatkan *output* ke dalam rentang (-1, 1), membantu dalam normalisasi *output*.
- c) *Dropout*: Setelah aktivasi, *dropout* sebesar  $p=0.1$  diterapkan.

4. *BiGRU Layer*

*Layer: GRU(768, 384, batch\_first=True, bidirectional=True)*

- a) Fungsi: Lapisan ini memiliki *hidden size* 384, yang berarti setelah digabungkan (*forward + backward*), ukuran *output* akan menjadi 768.
- b) BiGRU berguna untuk memproses data sekuensial seperti teks medis yang memiliki informasi penting di awal atau akhir catatan.

5. *Attention Mechanism*

*Layer: Attention -> Linear(in\_features=768, out\_features=1)*

- a) Fungsi: Lapisan *attention* bertujuan untuk memprioritaskan informasi penting dalam sekuens input. Setiap vektor output BiGRU akan diberi bobot relevansi melalui fungsi linear dengan proyeksi ke output bernilai 1. Ini memungkinkan model untuk memfokuskan perhatiannya pada fitur-fitur tertentu dalam catatan medis pasien.
- b) Setelah bobot *attention* dihitung, *context vector* diperoleh melalui proses penggandaan (*dot-product*) antara bobot *attention* dan *output* dari BiGRU.

6. *Fully Connected Layer (FC)*

*Layer: Linear(in\_features=768, out\_features=768)*

- a) Fungsi: Lapisan *fully connected (FC)* ini bertujuan untuk memampatkan representasi *context vector* yang diperoleh dari

mekanisme *attention*. Proyeksi linear ini digunakan untuk mengurangi dimensi representasi fitur sebelum menuju ke lapisan *output*.

- b) *Dropout*: *Dropout* sebesar  $p=0.1$  diterapkan kembali untuk regularisasi sebelum klasifikasi akhir dilakukan.

### 7. Classification Layer

Layer: *Linear*(*in\_features*=768, *out\_features*=1)

- a) Fungsi: Lapisan ini merupakan *linear classifier* yang bertugas untuk menghasilkan *logits*. *Output* dari lapisan ini adalah skor prediksi, yang dalam kasus ini merupakan nilai real yang menunjukkan kemungkinan apakah pasien akan kembali dirawat atau tidak.
- b) Output berupa satu nilai (logit), yang akan diproses lebih lanjut oleh fungsi *loss*.

### 8. Loss Function

Layer: *BCEWithLogitsLoss*

- a) Fungsi: Fungsi *loss* yang digunakan adalah *Binary Cross-Entropy with Logits* (*BCEWithLogitsLoss*). Fungsi ini secara otomatis menggabungkan fungsi sigmoid ke dalam perhitungan *loss*, yang berarti *logits* dari *classifier* akan diubah menjadi probabilitas antara 0 dan 1 sebelum dihitung selisihnya dengan label yang sebenarnya (0 atau 1).
- b) *Loss* ini menghitung perbedaan antara prediksi model dan label sebenarnya, serta digunakan untuk mengoptimalkan parameter selama proses pelatihan.

Kesimpulan proses:

1. *Input* token diubah menjadi *embedding vector*. *Embedding* diproses oleh 12 lapisan XLNet, yang terdiri dari *self-attention* dan *feedforward* dengan *LayerNorm* dan GELU sebagai aktivasi.
2. Setelah input token di-embed, vektor *embedding* ini diproses oleh XLNet yang dilengkapi dengan *two-stream self-attention*. XLNet berbeda dari model-model transformer lain karena memproses data secara *autoregressive* dan *autoregressive permuted*, yang memungkinkan model ini untuk menangkap informasi kontekstual secara lebih efektif.
3. Output dari XLNet berupa representasi vektor sekuensial  $h_1^{(2)}, h_2^{(2)}, \dots, h_r^{(2)}$  yang digunakan sebagai input bagi lapisan selanjutnya.

```
Output XLNet (hidden states dari token):
tensor([[[[-0.8984, -0.5233, -0.9776, ..., -0.4563,  0.3893,  0.2800],
          [-0.8479, -0.4829, -0.9435, ..., -0.4333,  0.3550,  0.2761],
          [-0.7998, -0.4382, -0.9690, ..., -0.4226,  0.3180,  0.2627],
          ...,
          [-1.0260, -1.2186, -2.4152, ..., -0.7135,  0.3664, -0.7094],
          [-0.4430, -1.0411, -1.2441, ..., -0.6757,  0.1345,  0.0515],
          [-0.4250, -1.0077, -1.1801, ..., -0.7421,  0.1171,  0.0188]]]])
```

Gambar 19. Output XLNet

4. Vektor representasi sekuensial dari XLNet kemudian diproses oleh lapisan BiGRU. BiGRU (*Bidirectional GRU*) merupakan varian dari GRU (*Gated Recurrent Unit*) yang memproses data sekuensial dalam dua arah (*forward* dan *backward*). Ini memungkinkan model untuk menangkap informasi dari masa lalu dan masa depan dalam catatan medis pasien.
5. Pada proses ini, keluaran dari BiGRU adalah representasi dua arah yang digabungkan dalam satu vektor, yaitu:  $\overrightarrow{h_1}, \overleftarrow{h_1}, \overrightarrow{h_2}, \overleftarrow{h_2}, \dots, \overrightarrow{h_n}, \overleftarrow{h_n}$

```
Output BiGRU (representasi sekuensial dari token):
tensor([[[[-0.5338,  0.3906, -0.5204, ..., -0.7470, -0.6333, -0.4927],
          [-0.6989,  0.4273, -0.5134, ..., -0.7590, -0.6528, -0.5038],
          [-0.7525,  0.4180, -0.4643, ..., -0.7715, -0.6707, -0.5226],
          ...,
          [-0.6273,  0.8097, -0.6945, ..., -0.9253, -0.6011, -0.4749],
          [-0.5657,  0.7313, -0.8762, ..., -0.9644, -0.9237, -0.3388],
          [-0.4928,  0.6710, -0.9175, ..., -0.8861, -0.8439, -0.2094]]]])
grad_fn=<TransposeBackward1>
```

Gambar 20. Output BiGRU

6. Setelah melalui BiGRU, mekanisme *attention* diterapkan untuk memberi bobot pada informasi penting dalam catatan medis.
7. Nilai *attention* ini kemudian digunakan untuk menghitung *context vector* dengan mengalikan *attention* dengan *output* BiGRU.

```
Output Attention (vektor konteks berdasarkan bobot perhatian):
tensor([[[-0.8103,  0.6798, -0.4622,  0.8542, -0.3122,  0.8693,  0.2371, -0.4482,
          -0.3588,  0.8155,  0.6928,  0.7847,  0.5619,  0.3382,  0.4381,  0.6227,
          -0.5449,  0.8357, -0.0524,  0.7968,  0.9858,  0.1926, -0.3837,  0.4631,
          0.5473,  0.5408,  0.0501,  0.3142,  0.8622,  0.4217,  0.2322, -0.2657,
          0.8239, -0.6904, -0.8183,  0.5167,  0.6485,  0.9631,  0.5325, -0.5526,
          0.5068,  0.1828, -0.4083,  0.9776,  0.4431,  0.7389,  0.8741,  0.9439,
          0.8448,  0.1249, -0.8908,  0.0506,  0.4034, -0.8902, -0.8177, -0.3561,
          -0.8673,  0.7097, -0.2838, -0.1083,  0.1515,  0.8525,  0.1434, -0.2255,
          0.5652,  0.4329,  0.2162,  0.7963, -0.3109, -0.9163, -0.6978, -0.2045,
```

Gambar 21. Output Attention Mechanism

8. Setelah *context vector* diperoleh, representasi teks ini kemudian diproses melalui lapisan *fully connected* yang mengubahnya menjadi vektor yang lebih ringkas dan ditambah dengan lapisan *dropout* untuk mencegah *overfitting*.
9. Vektor representasi teks akhir ini kemudian diumpankan ke lapisan *output (linear classifier)* untuk menghasilkan *logits*, yang merupakan skor prediksi apakah pasien akan diterima kembali atau tidak.
10. Untuk pelatihan, model menggunakan fungsi *loss binary cross-entropy* dengan *logits (BCEWithLogitsLoss)*, yang menghitung perbedaan antara nilai prediksi dengan label sebenarnya.

### 2.5.2.5 Optimasi Model

Pelatihan model dilakukan menggunakan optimasi AdamW yang dikombinasikan dengan penjadwalan *learning rate* berbasis *warmup*. AdamW adalah variasi dari algoritma optimisasi Adam yang menggabungkan regulasi *weight decay* untuk mencegah *overfitting*. Secara keseluruhan optimasi model XLNet-BiGRU-ATT ditunjukkan pada tabel 5.

Tabel 5. Optimasi model

<i>Step</i>	<i>Description</i>
Pengelompokan Parameter	Mendefinisikan dua grup parameter untuk optimizer: <ol style="list-style-type: none"> <li><i>no_decay</i>: Bias &amp; bobot <i>LayerNorm</i> dikecualikan dari <i>weight decay</i>.</li> <li><i>decay</i>: Parameter model lainnya dengan <i>weight decay</i> sebesar 0.01.</li> </ol>
Pengaturan <i>Optimizer</i>	Inisialisasi optimizer AdamW menggunakan parameter yang dikelompokkan dan <i>learning rate</i>
Perhitungan Langkah <i>Warmup</i>	Menghitung jumlah langkah <i>warmup</i> ( <i>warmup_steps</i> = <i>warmup_proportion</i> x <i>num_train_steps</i> ).
Pengaturan <i>Scheduler</i>	Mengatur <i>scheduler learning rate linear</i> dengan <i>warmup</i> Parameter scheduler: <ul style="list-style-type: none"> <li>- <i>num_warmup_steps</i>: Jumlah langkah <i>warmup</i> yang telah dihitung.</li> <li>- <i>num_training_steps</i>: Total jumlah langkah pelatihan (<i>num_train_steps</i>).</li> </ul>

Alur proses yang terjadi pada tabel 5 yaitu:

#### A. Inisialisasi Variabel *no\_decay*

Variabel *no\_decay* adalah daftar yang berisi nama-nama parameter yang tidak akan dikenakan *weight decay*. Parameter ini biasanya mencakup bias dari neuron dan parameter dari *layer normalization* seperti *LayerNorm.bias* dan *LayerNorm.weight*. *Weight decay* merupakan salah satu bentuk regularisasi L2 yang digunakan untuk mencegah model *overfitting* dengan menambahkan penalti terhadap nilai besar dari bobot model. Namun, dalam arsitektur *neural network*, bias dan parameter *normalization layer* dikecualikan dari penalti ini, karena *weight decay* tidak relevan untuk arsitektur tersebut. *Weight decay* membantu mengurangi kompleksitas model dengan mendorong model untuk memiliki bobot yang lebih kecil, sehingga membuat model lebih general dan mengurangi risiko *overfitting* pada data latih.

#### B. Membuat *optimizer\_grouped\_parameters*

Parameter model dibagi menjadi dua kelompok:

- Kelompok pertama mencakup semua parameter yang tidak ada dalam daftar *no\_decay*. Kelompok ini akan dikenakan *weight decay* dengan nilai 0.01. Penambahan *weight decay* dilakukan melalui optimasi yang bertujuan untuk mengurangi magnitudo dari bobot model.

Persamaan Regularisasi L2 atau *Weight Decay* (17) yaitu:

$$L_{total} = L_{original} + \lambda \sum_i w_i^2 \quad (17)$$

Dimana:

- a)  $L_{original}$  adalah fungsi *loss* asli.
  - b)  $\lambda$  adalah *hyperparameter* yang mengontrol tingkat regularisasi (*weight decay*).
  - c)  $w_i$  adalah bobot dari parameter model.
2. Kelompok kedua mencakup parameter yang ada di dalam daftar *no\_decay*, seperti bias dan *LayerNorm*. Untuk parameter ini, *weight decay* diatur ke 0 karena tidak memerlukan penalti regularisasi.

### C. Membuat *Optimizer* AdamW

Mendefinisikan *optimizer* yang akan digunakan untuk memperbarui bobot model selama proses pelatihan. *Optimizer* yang digunakan adalah **AdamW** (*Adam with Weight Decay*), yang merupakan variasi dari algoritma optimasi Adam yang telah dioptimalkan untuk menerapkan *weight decay* secara lebih efisien.

AdamW menggabungkan keuntungan dari Adam dan penambahan *weight decay* secara eksplisit, tanpa mempengaruhi perhitungan gradien seperti yang terjadi pada regularisasi L2 dalam Adam standar.

AdamW mengombinasikan perhitungan gradien dan momentum dalam persamaan (18) – (21) sebagai berikut:

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \quad (18)$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \quad (19)$$

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t}, \quad \hat{v}_t = \frac{v_t}{1 - \beta_2^t} \quad (20)$$

$$w_t = w_{t-1} - \eta \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon} - \eta \lambda w_{t-1} \quad (21)$$

Dimana:

- a)  $m_t$  adalah estimasi gradien momentum.
- b)  $v_t$  adalah estimasi gradien kuadrat (momentum kedua).
- c)  $\hat{m}_t$  dan  $\hat{v}_t$  adalah *bias-corrected* momentum.
- d)  $g_t$  adalah gradien.
- e)  $\eta$  adalah *learning rate*.
- f)  $\lambda$  adalah *weight decay*.
- g)  $w_t$  adalah bobot pada waktu  $t$ .

Berbeda dengan Adam standar, AdamW memisahkan *weight decay* dari perhitungan gradien, sehingga *weight decay* hanya diterapkan langsung ke parameter model setelah gradien diperhitungkan.

#### D. Menghitung *Warmup\_steps*

*Warmup\_steps* adalah jumlah langkah (*steps*) selama fase *warmup* pada saat pelatihan. *Warmup* adalah strategi di mana *learning rate* secara bertahap ditingkatkan selama beberapa langkah awal pelatihan, kemudian secara bertahap diturunkan pada langkah-langkah berikutnya.

*Warmup\_steps* dihitung berdasarkan proporsi dari total langkah pelatihan (*num\_train\_steps*). Jika *warmup\_proportion* adalah 0.1, berarti 10% dari seluruh langkah pelatihan digunakan untuk fase *warmup*.

Persamaan untuk menghitung *warmup\_steps* (22) yaitu:

$$warmup\_steps = warmup\_proportion \times num\_train\_steps \quad (22)$$

Dimana:

- a) *warmup\_proportion* adalah persentase langkah pelatihan yang dialokasikan untuk *warmup*.
- b) *num\_train\_steps* adalah jumlah total langkah pelatihan.

#### E. Membuat *Scheduler*

Membuat penjadwal (*scheduler*) yang mengatur nilai *learning rate* selama pelatihan menggunakan pendekatan *linear schedule with warmup*. Fungsi ini melakukan dua hal:

1. Pada fase *warmup*, *learning rate* akan meningkat secara *linear* dari 0 hingga mencapai nilai puncak (*learning rate* awal yang diatur dalam *optimizer*).
2. Setelah fase *warmup*, *learning rate* akan menurun secara *linear* hingga akhir pelatihan.

Persamaan untuk penjadwalan *linear* dengan *warmup* (23) – (24) yaitu:

Untuk  $t \leq warmup\_steps$ :

$$\eta_t = \frac{t}{warmup\_steps} \cdot \eta_{max} \quad (23)$$

Untuk  $t > warmup\_steps$ :

$$\eta_t = \eta_{max} \cdot \left(1 - \frac{t - warmup\_steps}{num\_train\_steps - warmup\_steps}\right) \quad (24)$$

Dimana:

- a)  $\eta_t$  adalah *learning rate* pada langkah  $t$ .
- b)  $\eta_{max}$  adalah *learning rate* maksimum (yang diatur dalam *optimizer*).
- c)  $t$  adalah langkah pelatihan saat ini.
- d) *warmup\_steps* adalah jumlah langkah dalam fase *warmup*.
- e) *num\_train\_steps* adalah total langkah pelatihan.

Pada fase awal, *learning rate* dimulai dari 0 dan meningkat hingga mencapai nilai puncak selama langkah-langkah *warmup*. Setelah fase *warmup* selesai, *learning rate* akan menurun secara *linear* hingga mendekati nol pada akhir pelatihan.

Sehingga secara keseluruhan proses yang terjadi yaitu:

1. Inisialisasi: Model diinisialisasi dengan bobot *pre-trained* XLNet, kemudian menambahkan lapisan BiGRU dan *attention mechanism*.
2. Proses *Forward Pass*: Data catatan medis pasien yang telah di-tokenisasi melewati arsitektur XLNet, BiGRU, dan *attention mechanism*. Hasil akhirnya adalah nilai *logits*.
3. Perhitungan *Loss*: *Logits* yang dihasilkan dibandingkan dengan label sebenarnya menggunakan fungsi *BCEWithLogitsLoss*.
4. *Backward Pass*: *Gradien loss* dihitung dan digunakan untuk memperbarui bobot model.
5. Optimasi: Bobot model diperbarui menggunakan AdamW, dengan *weight decay* diterapkan untuk mencegah *overfitting*.
6. Penjadwalan *Learning Rate*: *Learning rate* diatur agar menurun secara bertahap selama pelatihan, dengan *warmup* awal untuk stabilisasi.

### 2.5.3 Evaluasi Model

#### 1. Uji Model

Menguji Model pada Data Pengujian untuk Mengevaluasi Kinerjanya:

- a. Setelah melatih model dengan data pelatihan, langkah selanjutnya adalah menguji kinerja model pada data pengujian yang belum pernah dilihat sebelumnya.
  - b. Data pengujian memberikan gambaran objektif tentang seberapa baik model dapat menggeneralisasi pola yang telah dipelajari dari data pelatihan untuk membuat prediksi yang akurat terhadap data baru.
  - c. Uji model pada data pengujian memastikan bahwa model tidak hanya "menghafal" data pelatihan tetapi juga mampu mengidentifikasi pola umum yang dapat diterapkan pada situasi baru.
2. Hitung Metrik Evaluasi

Evaluasi model dilakukan menggunakan metrik berikut:

#### A. ROC-AUC (*Area Under the ROC Curve*):

##### 1. ROC (*Receiver Operating Characteristic*):

ROC adalah kurva yang menggambarkan performa sebuah model klasifikasi biner dengan memplot *True Positive Rate* (TPR) di sumbu y melawan *False Positive Rate* (FPR) di sumbu x pada berbagai *threshold* klasifikasi. TPR dikenal juga sebagai *recall*, dan FPR adalah persentase negatif yang salah diklasifikasikan sebagai positif. ROC digunakan untuk mengevaluasi keseimbangan antara sensitivitas (*recall*) dan 1-spesifisitas (FPR).

##### 2. AUC (*Area Under the Curve*):

Mengukur performa sebuah model klasifikasi dengan menghitung luas di bawah kurva evaluasi. AUC saja tidak bisa digunakan untuk mengukur performa model tanpa adanya kurva seperti ROC atau *Precision-Recall* (PR). AUC adalah metrik yang menghitung luas di

bawah kurva, sehingga ia memerlukan kurva seperti ROC atau PR untuk diukur. Artinya, AUC adalah ringkasan dari performa model yang dinilai berdasarkan kurva tersebut. AUC memberikan gambaran keseluruhan dari performa model pada berbagai *threshold*, tetapi tanpa kurva (ROC atau PR), tidak ada yang diukur. ROC atau PR adalah alat visual yang memetakan kinerja model pada berbagai ambang batas keputusan, dan AUC hanya menghitung area yang berada di bawah kurva tersebut.

### 3. ROC-AUC (Area Under the ROC Curve):

AUC adalah ukuran luas di bawah kurva ROC, yang memberikan nilai tunggal untuk menilai performa model klasifikasi. Nilai AUC berkisar antara 0 dan 1. Semakin besar nilai AUC (mendekati 1), semakin baik model dalam membedakan antara kelas positif dan negatif.

*ROC Curve* sendiri tidak memberikan satu nilai tunggal untuk menilai kinerja model, tapi memberikan gambaran tentang bagaimana model berperilaku pada berbagai *threshold*. Namun, *ROC Curve* digunakan untuk menghasilkan AUC, yang merupakan nilai tunggal yang dapat digunakan untuk menilai kinerja model secara keseluruhan.

AUC memetakan performa model ke dalam satu angka yang menunjukkan keseimbangan antara *True Positive Rate* (TPR) dan *False Positive Rate* (FPR) atau antara *Precision* dan *Recall*, tergantung pada jenis kurva yang digunakan (ROC atau *Precision-Recall*).

Fungsi ini menghitung skor prediksi berdasarkan beberapa metrik, menggabungkan prediksi individual dari satu pasien, dan kemudian mengevaluasinya menggunakan kurva ROC.

Alur proses perhitungan:

#### 1. Mengurutkan Data Berdasarkan ID:

Data diurutkan berdasarkan ID. Setiap ID mewakili satu pasien, sehingga jika ada beberapa contoh data untuk satu pasien, mereka akan dikelompokkan bersama.

#### 2. Menghitung Skor Agregat:

Skor agregat untuk setiap pasien dihitung sebagai gabungan dari skor maksimal dan rata-rata skor untuk semua contoh data dari pasien tersebut. Bagian ini mengagregasi skor prediksi per ID dengan menghitung rata-rata yang melibatkan maksimum, jumlah, dan jumlah nilai per ID. Persamaan untuk menghitung Skor Agregat (25) yaitu:

$$\text{Agregat Skor} = \frac{\max(s_i) + \frac{1}{2} \sum_{i=1}^n s_i}{1 + \frac{1}{2}n} \quad (25)$$

Di mana  $s_i$  adalah skor prediksi individual dan  $n$  adalah jumlah contoh data untuk satu pasien.

Penjelasan:

- a)  $\max(s_i)$ : Mengambil nilai maksimum dari semua prediksi untuk satu pasien. Ini menganggap bahwa skor prediksi tertinggi adalah yang paling relevan.
- b)  $\sum_{i=1}^n s_i$ : Mengambil jumlah dari semua prediksi untuk pasien tersebut. Bagian ini memberi kontribusi dari semua prediksi, tetapi dibagi dua untuk menyeimbangkan kontribusinya dengan nilai maksimum.
- c)  $1 + \frac{1}{2}n$ : Merupakan faktor pembagi yang menormalkan skor akhir, sehingga hasilnya dipengaruhi oleh jumlah prediksi yang ada (semakin banyak prediksi, semakin besar kontribusi dari rata-rata prediksi).

Skor agregat ini memberikan keseimbangan antara menggunakan prediksi maksimum dan rata-rata prediksi pasien untuk menghasilkan satu nilai representatif yang akan digunakan untuk evaluasi model.

### 3. Mengambil Nilai Label Terendah:

Nilai label terendah untuk setiap pasien diambil menggunakan fungsi agregat  $\min$ . Ini memastikan bahwa jika ada label yang berbeda untuk pasien yang sama, maka label negatif (0) lebih diutamakan. Label "0" berarti pasien tidak dirawat kembali, sementara "1" berarti dirawat kembali.

### 4. Plot kurva ROC dan AUC:

- a) *False Positive Rate* (FPR) dan *True Positive Rate* (TPR) dihitung menggunakan fungsi *roc\_curve* dengan persamaan (26) dan (27) berikut:

$$TPR = \frac{TP}{TP + FN} \quad (26)$$

$$FPR = \frac{FP}{FP + TN} \quad (27)$$

Dimana:

*TPR* : Merupakan *recall* atau sensitivitas, yang menunjukkan seberapa baik model mengenali kelas positif.

*TP* : *True Positives* (prediksi positif yang benar).

*FN* : *False Negatives* (prediksi negatif yang salah).

*FPR* : Menunjukkan seberapa sering model salah memprediksi negatif sebagai positif.

*FP* : *False Positives* (prediksi positif yang salah).

*TN* : *True Negatives* (prediksi negatif yang benar).

ROC (*Receiver Operating Characteristic*) mengukur performa model klasifikasi dengan memplot *True Positive Rate* (TPR) melawan *False Positive Rate* (FPR) di berbagai nilai *threshold*.

Fungsi `roc_curve()` akan menghasilkan nilai FPR, TPR, dan `threshold` berdasarkan:

1. *True Labels*: Ini digunakan untuk membandingkan prediksi model dengan label sebenarnya (0 atau 1). Dalam rumus TPR dan FPR:

- a) TP (*True Positives*): Diukur dengan berapa banyak prediksi positif yang benar-benar positif sesuai dengan  $x$ .
  - b) FP (*False Positives*): Diukur dengan berapa banyak prediksi positif yang sebenarnya negatif sesuai dengan  $x$ .
- $x$  digunakan untuk menghitung nilai-nilai TP, FN, FP, dan TN yang dibutuhkan untuk menghitung TPR dan FPR.

2. *Temp values (Predicted Scores)*: Ini adalah skor probabilitas atau *logits* yang dihasilkan oleh model. Dalam setiap iterasi *threshold*, *temp values* digunakan untuk menentukan apakah suatu sampel harus diklasifikasikan sebagai positif atau negatif. ROC curve dihitung dengan menggunakan beberapa *threshold* berbeda pada *temp values*, untuk menghitung berapa banyak TP, FP, FN, dan TN di berbagai ambang batas tersebut.

- a) Misalnya, jika *threshold* adalah 0.5, maka:  
Jika *temp values* > 0.5, prediksi dianggap positif.  
Jika *temp values* ≤ 0.5, prediksi dianggap negatif.
- b) Dengan *threshold* yang berbeda-beda, dapat menghasilkan berbagai nilai TPR dan FPR, yang kemudian diplot sebagai kurva ROC.

*True Labels*: Menyediakan informasi *actual labels* yang digunakan untuk menghitung TP, FP, FN, dan TN, yang diperlukan untuk mendapatkan TPR dan FPR pada setiap *threshold*.

*temp values (Predicted Scores)*: Memberikan skor probabilitas yang dihasilkan model, yang dibandingkan dengan berbagai *threshold* untuk memutuskan klasifikasi (positif/negatif). Ini digunakan untuk menentukan prediksi pada setiap *threshold*, yang kemudian digunakan untuk menghitung TPR dan FPR.

Dengan demikian, kedua variabel (*true labels* dan *temp values*) bersama-sama digunakan untuk menghitung ROC melalui perhitungan TPR dan FPR di berbagai *threshold*.

b) AUC (*Area Under Curve*) dihitung dari FPR dan TPR menggunakan fungsi `auc`.

Persamaan AUC (28) yaitu:

$$AUC = \int_0^1 TPR(FPR)d(FPR) \quad (28)$$

AUC mengukur performa model dalam membedakan antara kelas positif dan negatif. Nilai AUC berkisar antara 0 dan 1, di mana 1 menunjukkan model yang sempurna dan 0.5 menunjukkan model acak.

B. PR-AUC (*Area Under the Precision-Recall Curve*):

a) PR (*Precision-Recall Curve*):

*Precision-Recall (PR) curve* adalah kurva yang memplot *Precision* (presisi) di sumbu y melawan *Recall* (atau *True Positive Rate*) di sumbu x. *Precision* adalah proporsi prediksi positif yang benar-benar benar, sedangkan *Recall* adalah proporsi positif aktual yang terprediksi dengan benar. PR Curve lebih fokus pada performa pada kelas minoritas (positif) dan sangat berguna ketika dataset tidak seimbang.

b) PR-AUC (*Area Under the Precision-Recall Curve*):

PR-AUC mengukur luas di bawah kurva *Precision-Recall*, memberikan gambaran performa keseluruhan model dalam konteks presisi dan recall. PR-AUC berguna terutama dalam kasus di mana kelas positif lebih jarang. Nilai yang lebih besar menunjukkan performa yang lebih baik, dengan PR-AUC maksimal adalah 1.

Fungsi ini menghasilkan kurva *Precision-Recall* dari hasil prediksi.

Alur proses perhitungan:

1. Menghitung *Precision* dan *Recall*:

Persamaan untuk menghitung Precision (29) yaitu:

$$Precision = \frac{TP}{TP + FP} \quad (29)$$

Persamaan untuk menghitung Recall (30) yaitu:

$$Recall = \frac{TP}{TP + FN} \quad (30)$$

2. Menghitung AUC untuk Precision-Recall:

AUC untuk kurva Precision-Recall dihitung menggunakan fungsi auc.

C. PR80 (*Recall at Precision 80*):

RP80 adalah metrik yang menunjukkan *Recall* pada titik di mana *Precision* mencapai nilai 80%. Ini menilai seberapa baik model dapat menangkap kelas positif (*recall*) dengan tingkat presisi yang tinggi (minimal 80%). Metrik ini berguna jika tujuan evaluasi model adalah untuk mempertahankan tingkat presisi yang cukup tinggi.

Fungsi ini menghitung *precision-recall* untuk prediksi yang dihasilkan dan kemudian mengevaluasi *recall* pada *precision* yang lebih dari 80%.

Menghitung *Recall* pada *Precision* > 0.8:

- a) *Threshold* untuk *Precision* ditetapkan pada 80%. Jika *precision* lebih dari 0.8, maka *recall* pada titik tersebut dihitung.

- b) Jika tidak ada sampel yang memenuhi syarat ini, maka RP80 (*Recall at Precision 80*) dianggap nol.

Interpretasi:

- a) *Recall at Precision 80* (PR80) berarti seberapa banyak contoh positif yang berhasil diidentifikasi model (*recall*) dengan presisi yang setidaknya 80%.
- b) PR80 adalah metrik yang sangat berguna dalam situasi di mana ingin memastikan bahwa ketika model memprediksi positif, model ini 80% yakin bahwa prediksi tersebut benar, sekaligus memaksimalkan kemampuan model untuk mendeteksi sebanyak mungkin kasus positif.

PR80 dihitung dengan mencari nilai *recall* dari *precision* yang lebih dari 0.80.

Jika *precision* lebih besar dari 80%, maka *recall*-nya disebut sebagai PR80.

Sehingga secara keseluruhan proses evaluasi yang dilakukan yaitu:

- a) ROC-AUC (*Area Under the ROC Curve*):  
Menghitung skor agregat prediksi dan mengevaluasi performa model menggunakan ROC dan AUC.
- b) PR-AUC (*Area Under the Precision-Recall Curve*):  
Menghasilkan kurva *Precision-Recall* untuk memvisualisasikan *trade-off* antara *Precision* dan *Recall*.
- c) PR80 (*Recall at Precision 80*):  
Menggabungkan perhitungan *Precision-Recall* dan mengevaluasi *recall* pada *precision* di atas 80%, yang dikenal sebagai RP80.

Hubungan Antar Metrik:

- a. **ROC-AUC** dan **PR-AUC**: Kedua metrik ini menggambarkan performa model, namun fokusnya berbeda. ROC-AUC lebih cocok digunakan ketika terdapat keseimbangan antara kelas positif dan negatif, sementara PR-AUC lebih fokus pada kinerja untuk mendeteksi kelas positif, khususnya ketika data imbalanced (jumlah kelas negatif lebih banyak).
- 1) Jika menggunakan ROC, maka AUC-ROC akan menunjukkan keseimbangan antara:
- True Positive Rate* (TPR): Proporsi positif yang benar-benar terdeteksi oleh model (juga dikenal sebagai *recall*).
  - False Positive Rate* (FPR): Proporsi negatif yang salah terdeteksi sebagai positif oleh model.
- AUC-ROC menunjukkan seberapa baik model dapat memisahkan antara kelas positif dan negatif dengan mempertimbangkan bagaimana model mengelola TPR dan FPR di berbagai ambang batas (*thresholds*).
- 2) Jika menggunakan *Precision-Recall* (PR) *curve*, maka AUC-PR akan menunjukkan keseimbangan antara:
- Precision*: Proporsi prediksi positif yang benar-benar positif.
  - Recall* (TPR): Proporsi positif yang benar-benar terdeteksi oleh model.

AUC-PR lebih fokus pada evaluasi bagaimana model menjaga keseimbangan antara *precision* dan *recall*, terutama saat menangani dataset yang tidak seimbang (di mana kelas positif lebih sedikit dari kelas negatif).

- b. **RP80** adalah metrik yang spesifik digunakan untuk mengevaluasi recall ketika presisi diatur pada tingkat tertentu (80%), yang sangat penting dalam skenario di mana kesalahan prediksi positif lebih kritis.

Kombinasi metrik ini memberi gambaran yang lebih komprehensif tentang performa model dalam berbagai skenario evaluasi.