

USULAN PENELITIAN DISERTASI

**EVALUASI MODEL PADA PENGEMBANGAN LINGKUNGAN
PEMBELAJARAN BERBASIS LEARNING ANALYTICS**

*(LEARNING ANALYTICS BASED MODEL EVALUATION FOR
DEVELOPMENT OF LEARNING ENVIRONMENT)*

disusun dan diajukan oleh

Billy Eden William Asrul
D053192005



SEKOLAH PASCASARJANA
PROGRAM STUDI DOKTOR (S3) TEKNIK ELEKTRO
UNIVERSITAS HASANUDDIN
MAKASSAR
2022

PENGESAHAN PROPOSAL SEMINAR USULAN PENELITIAN

**EVALUASI MODEL PADA PENGEMBANGAN LINGKUNGAN
PEMBELAJARAN BERBASIS LEARNING ANALYTICS**

Diajukan Oleh

Billy Eden William Asrul

D053192005

Telah diperiksa dan dinyatakan memenuhi syarat untuk melaksanakan
Seminar Usul Penelitian

Menyetujui

Tim Promotor

Prof. Dr.Eng. Ir. Syafaruddin, ST., M.Eng., IPU
Promotor

Tanggal: 9 Mei 2022

Dr. Amil Ahmad Ilham, ST., M.IT.
Co-Promotor

Dr.Eng. Wardi, ST., M.Eng.
Co-Promotor

PRAKATA

Assalamu Alaikum Wr. Wb.

Salam Sejahtera bagi kita.

Puji syukur kita panjatkan kehadirat Allah SWT. atas berkah dan rahmat-Nya sehingga pengajuan proposal disertasi dengan judul “Evaluasi Model Pada Pengembangan Lingkungan Pembelajaran Berbasis *Learning Analytics*” dapat terselesaikan.

Pada penyusunan kali ini kami menyajikan beberapa hal yang menyangkut judul yang telah kami angkat dan telah melalui proses pencarian dari berbagai sumber baik jurnal penelitian, buku maupun dari situs-situs di internet, selain dari penelitian sebelumnya. Untuk itu melalui kesempatan ini kami ingin mengucapkan terima kasih kepada seluruh pihak yang telah membantu dalam penyusunan ini, termasuk secara khusus Penulis berterima kasih kepada bapak Prof. Dr.Eng. Ir. Syafaruddin, ST., M.Eng.,IPU. selaku promotor pada Fakultas Teknik, Universitas Hasanuddin atas arahan dan bimbingan pada saat penyusunan pra-proposal dan rasionalisasi roadmap riset sehingga menjadi proposal disertasi, kemudian kepada bapak Dr. Amil Ahmad Ilham, ST., M.IT. Dan bapak Dr.Eng. Wardi, ST., M.Eng. selaku Co-promotor yang selalu mengarahkan, memberikan masukan serta motivasi sehingga penyusunan proposal penelitian ini dapat terselesaikan, juga kepada para dosen, dan spesial kepada orang tua, dan keluarga besar.

Walaupun dalam penyusunannya, penulis telah berusaha secara maksimal, namun jika masih ada kekurangan baik dari segi akademik keilmuan, maupun dari segi teknis penulisan penulis memohon maaf dan mohon petunjuk berupa kritik dan saran, demi penyusunan selanjutnya agar lebih baik lagi.

Akhir kata penulis mengucapkan terima kasih dan mohon maaf atas segala kekurangan. Semoga kesehatan dan kesuksesan senantiasa menyertai kita semua.

Wassalam.

Gowa, Mei 2022

PENULIS

DAFTAR ISI

PRAKATA	ii
DAFTAR GAMBAR	v
DAFTAR TABEL	vi
BAB I PENDAHULUAN	1
A. LATAR BELAKANG	1
B. RUMUSAN MASALAH	7
C. TUJUAN PENELITIAN	7
D. MANFAAT PENELITIAN	8
E. BATASAN MASALAH	9
BAB II TINJAUAN PUSTAKA	10
A. LANDASAN TEORI	10
1. Learning Analytics	10
2. Mekanisme Pelaksanaan LA dalam pembelajaran	10
3. Machine Learning	11
4. Deep Learning	13
5. Convolutional Neural Network	15
6. Recurrent Neural Network	21
7. Long Short Term Memory	22
8. CNN-LSTM	27
9. Confusion Matrix	32
B. PENELITIAN TERKAIT	35
C. STATE OF THE ART	39
BAB III METODE PENELITIAN	48
A. TAHAPAN PENELITIAN	48
B. ROADMAP PENELITIAN	49
C. WAKTU DAN LOKASI PENELITIAN	50
D. JENIS PENELITIAN	50
E. PERANCANGAN MEKANISME PREDIKSI	51
F. SUMBER DATA	55
G. INSTRUMEN PENELITIAN	56
H. JADWAL PENELITIAN	57
DAFTAR PUSTAKA	58

DAFTAR GAMBAR

Gambar 2.1. Perbandingan Neural Network sederhana dengan Deep Learning Neural Network [22].	14
Gambar 2. 2. Arsitektur MLP sederhana	15
Gambar 2. 3. Arsitektur model CNN dan LSTM	16
Gambar 2. 4. Jaringan dengan 3 convolution layer	17
Gambar 2. 5. Arsitektur RNN [23]	22
Gambar 2. 6. Looping pada arsitektur RNN [23]	22
Gambar 2. 7. Modul pengulang RNN yang berisi satu layer [23]	23
Gambar 2. 8. Modul pengulang dalam LSTM berisi empat layer [23]	23
Gambar 2. 9. Ilustrasi cell state [23]	24
Gambar 2. 10. Ilustrasi langkah pertama LSTM [23]	25
Gambar 2. 11. Ilustrasi langkah kedua LSTM [23]	25
Gambar 2. 12. Ilustrasi langkah ketiga LSTM [23]	26
Gambar 2. 13. Ilustrasi langkah keempat LSTM [23]	26
Gambar 2. 14. Kerangka Pikir Penelitian	46
Gambar 3. 1. Tahapan Penelitian	48
Gambar 3. 2. Roadmap Penelitian	49
Gambar 3. 3. Flowchart Sistem Yang Diusulkan	51
Gambar 3. 4. Rencana Mekanisme Prediksi	53
Gambar 3. 5. Skema Tahapan Eksperimen	54

DAFTAR TABEL

Tabel 2. 1. Confusion Matrix	33
Tabel 2. 2. State Of The Art	40
Tabel 3. 1. Tabel Fitur Data	55
Tabel 3. 2. Tabel Jadwal Penelitian	57

BAB I

PENDAHULUAN

A. LATAR BELAKANG

Learning Analytics (LA) merupakan mekanisme pengukuran, pengumpulan, analisis dan pelaporan data tentang peserta didik, yang bertujuan untuk memahami dan mengoptimalkan pembelajaran serta lingkungan dimana proses pembelajaran tersebut dilaksanakan [1]. Seiring dengan perkembangan teknologi informasi dan penerapannya yang mulai meningkat dalam dunia pendidikan, menjadikan data-data pembelajaran lebih mudah diperoleh, dan peran *Learning Analytics* menjadi lebih signifikan [2].

Dalam penerapan metode pembelajaran luar jaringan (luring) atau dalam jaringan (daring), Internet dan penggunaan *Learning Management System* (LMS) merupakan hal yang mendukung suksesnya proses belajar mengajar. Salah satu mekanisme untuk memantau proses pembelajaran tersebut dapat dilihat dari riwayat penggunaan sistem yang terekam pada LMS [2].

Dengan adanya ketersediaan data berskala besar, dan dengan dilakukannya penerapan sistem diseluruh bagian proses belajar mengajar, menjadi terobosan penting untuk studi efektivitas dalam dunia pendidikan, yang memungkinkan untuk menerapkan studi tentang analisis faktor yang terkait dengan efektivitas dan tingkat kompleksitas yang mendalam dan

dapat mengarah pada peningkatan proses dan kebijakan pendidikan. Selain itu, berkat prevalensi peralatan komputer dengan kemampuan pemrosesan yang tinggi, sekarang dimungkinkan untuk menerapkan pendekatan yang berbeda dari analisis inferensial dan multivariat klasik, yang mengoptimalkan perolehan informasi berharga dari data berskala besar [3],[4].

Berdasarkan ISO/IEC 40180:2017 [5], untuk pembelajaran yang optimal dan agar sistem pendidikan dapat dioptimalkan, sangat penting untuk memiliki sarana untuk mengumpulkan data umpan balik terkait kinerja pembelajaran, dan menggunakan penilaian sistemik untuk mengidentifikasi kelemahan dalam mekanisme pembelajaran bagi para pelajar secara adaptif [6],[7]. Salah satunya yaitu dengan melakukan prediksi kinerja akademik. Dengan berkembangnya mekanisme pembelajaran saat ini, penggabungan pembelajaran luring dan pembelajaran daring (Blended Learning) mendapatkan kendala bagi pelajar dalam mengakses materi pembelajaran, pengumpulan tugas serta interaksi pembelajaran yang membuat kinerja pelajar sangat tergantung dengan pemanfaatan *virtual learning environment (VLE)* yang menjadi salah satu faktor kelulusan dalam pembelajaran saat ini [8]. Kurangnya interaksi dengan teman sekelas dan pengajar menjadi salah satu kendala dalam transisi pembelajaran saat ini. Keterlibatan dalam lingkungan pembelajaran yang berkurang, karena transisi cepat ke pembelajaran online juga merupakan tantangan lain yang membuat peningkatan pada tingkat kegagalan pelajar dalam perkuliahan.

Transisi pembelajaran saat ini memungkinkan untuk performa pelajar dapat diprediksi dengan memperkenalkan mekanisme evaluasi pembelajaran yang relevan dengan mekanisme pembelajaran *hybrid learning* [9],[10].

Prediksi kinerja akademik bermanfaat bagi pengajar dan pelajar untuk memahami status pembelajaran saat ini, yang akan berhubungan dengan kinerja akhir pelajar. *Log* pembelajaran skala besar dapat digunakan untuk membuat mekanisme prediksi. Penggunaan LMS, menghasilkan file *log* yang berisi data tentang interaksi pengguna (misalnya, tampilan materi kuliah, pengiriman dan evaluasi tugas, serta interaksi kuis dan forum). Informasi tersebut dapat digunakan untuk membuat mekanisme prediktif untuk tujuan seperti meramalkan kinerja pelajar [1],[11]. Data dalam file *log* menggambarkan pola bagaimana pelajar berinteraksi dengan LMS, dan pola tersebut menghasilkan beberapa korelasi dengan kinerja mereka.

Prediksi awal kinerja pelajar merupakan sumber yang baik digunakan untuk meningkatkan proses pembelajaran. Jika dapat mendeteksi pelajar yang berisiko “lulus” maupun “tidak lulus/gagal” pada tahap awal perkuliahan, maka akan memiliki lebih banyak waktu untuk meningkatkan kinerja mereka [6],[12]. Dengan memanfaatkan Kemajuan teknologi *machine learning* (ML), mendapat perhatian luas untuk memperkirakan kinerja pelajar di minggu-minggu awal proses belajar mengajar. Dalam hal ini, VLE dapat merekam data interaksi siswa. Interaksi data pelajar pada VLE ini digunakan untuk mengembangkan mekanisme

prediksi karena berbagai alasan, termasuk memprediksi kinerja siswa. Misalnya, arsitektur *multi-layer perceptron* (MLP) diusulkan untuk memprediksi kinerja pelajar menggunakan database VLE [8],[13]. Skema prediksi awal ini memungkinkan pengajar untuk campur tangan terhadap pelajar yang berisiko, dengan tingkat kegagalan yang tinggi dalam proses pembelajaran. Selain itu, model analitik *machine learning* dapat memberikan wawasan statistik kepada pengajar untuk membantu proses belajar mengajar. Demikian juga, pelajar yang sangat baik kinerjanya dapat dimotivasi dengan kegiatan tambahan yang disesuaikan, dan yang kurang dapat diberikan mekanisme tambahan.

Beberapa masalah yang terjadi pada bidang *learning analytics* terkhusus pada *academic performance prediction* diantaranya telah dibahas pada beberapa penelitian yang membandingkan metode klasifikasi dengan model jaringan syaraf tiruan, dan mendapatkan hasil bahwa performa jaringan syaraf tiruan mampu memprediksi lebih baik [14], juga dalam beberapa kasus penelitian masih menggunakan data Indeks Prestasi kumulatif dalam memprediksi kinerja akhir pada masa kelulusan [15], hal tersebut dianggap membutuhkan waktu yang cukup lama untuk mengumpulkan data nilai akhir semua perkuliahan, yang kemudian proses memprediksi dan peningkatan kinerja pelajar cenderung membutuhkan waktu yang lama, sehingga dalam penelitian ini akan digunakan data keterlibatan pelajar dalam mengikuti satu tahun akademik perkuliahan[16],[17],[18], data demografi pelajar, dan data history nilai

diawal perkuliahan yang dikumpulkan dari beberapa sistem, diantaranya LMS dan sistem informasi akademik (SIKA). dengan harapan semakin cepat data diolah diawal proses pembelajaran, maka akan semakin banyak waktu yang dapat digunakan untuk memperbaiki performa pembelajaran pelajar, serta para pengajar juga mampu mengevaluasi mekanisme pembelajaran yang akan diterapkan.

Saat ini ada beberapa penelitian tentang *academic performance prediction* masih sangat banyak diteliti yang kemudian dipublikasikan pada jurnal bereputasi, penelitian membahas tentang penggunaan model prediksi untuk mengidentifikasi pelajar yang beresiko pada tahap awal perkuliahan dan memberi peringatan dini kepada pengajar dan pelajar. Menggunakan beberapa jenis metode prediksi *logistic regression, support vector machine, decision tree, multi layer perceptron, naive bayes classifier, dan k-nearest neighbor* untuk menguji model prediksi terbaik. Dengan data input berupa nilai kuis, partisipasi kelompok, tugas rumah, hasil praktikum, dan tugas tertulis hasil menunjukkan bahwa model yang terbentuk sangat tergantung pada data kinerja pelajar yang dikumpulkan selama semester berjalan [6]. Nilai akhir sangat mempengaruhi kualitas data dan daya prediksi variabel, yang mengakibatkan pengaruh signifikan pada akurasi dari hasil prediksi [13]. Dengan langkah mengeksplorasi faktor yang mempengaruhi peningkatan kinerja prediksi nilai perkuliahan pelajar, dari data *log* perkuliahan serta dari sistem pembelajaran, dapat kemudian digunakan untuk memprediksi keberhasilan pelajar, dengan memanfaatkan

algoritma *machine learning* yang memungkinkan dilakukan eksperimen dengan memperbarui nilai *hyperparameter* dan mempelajari parameter optimal [4], [10], [11], [19]–[22].

Pada penelitian berikutnya, membahas tentang bagaimana prediksi kinerja kelompok pelajar, menggunakan kerangka *deep neural network* (DNN) untuk memprediksi kinerja awal kelompok pelajar. Kerangka DNN kemudian di evaluasi dengan fungsi aktivasi yang berbeda dan dilakukan tahapan untuk mengoptimalkan kerangka DNN tersebut. Hasil dari penelitian ini menunjukkan bahwa akurasi prediksi dari kerangka kerja algoritma DNN ketika dioptimalkan, mampu memprediksi kinerja tim pelajar secara memadai dan akurat [8],[10],[13].

Dari beberapa penelitian terdahulu maka pada penelitian ini, Kemudian akan membangun model prediksi dengan menggunakan lapisan *convolutional neural network* (CNN) dan *long short-term memory* (LSTM) untuk prediksi kinerja siswa. Dengan mengintegrasikan mekanisme convolutional dan LSTM, model *deep learning* (DL) yang diusulkan dapat secara akurat mengekstrak fitur spatiotemporal dari data aktivitas pelajar. sehingga fokus pada penelitian ini diantaranya; pertama, membuat mekanisme evaluasi pembelajaran yang relevan dengan model pembelajaran hybrid; kedua, membuat model *academic performance prediction* serta memperbaiki kinerja algoritma dengan teknik optimasi, sehingga hasil prediksi model diharapkan semakin akurat dan mampu menjadi mekanisme prediksi *early warning system* untuk mengoptimalkan

proses belajar mengajar. Selanjutnya penelitian ini memiliki potensi perluasan yang terletak pada kemungkinan dilakukan eksperimen pada strategi pembelajaran *hybrid* dengan menggunakan algoritma prediksi konvolusi, juga dapat memeriksa kinerja pelajar jika *log* intervensi pengajar dijadikan fitur selama kegiatan pembelajaran.

B. RUMUSAN MASALAH

Berdasarkan latar belakang sebelumnya, maka rumusan masalah yang diusulkan dalam penelitian sebagai berikut:

1. Bagaimana menemukan algoritma *machine learning* terbaik yang dapat digunakan untuk memprediksi kinerja pelajar diawal perkuliahan pada mekanisme evaluasi pembelajaran yang dirancang?
2. Bagaimana mengukur performa prediksi yang diusulkan dalam hal proses klasifikasi dan akurasi prediksi?
3. Bagaimana mengoptimalkan performa prediksi yang diusulkan pada *machine learning* untuk memenuhi kinerja akurasi sehingga akan meningkatkan kualitas proses pembelajaran?

C. TUJUAN PENELITIAN

Berdasarkan rumusan masalah, maka tujuan dari penelitian yang diusulkan sebagai berikut:

1. Untuk menemukan algoritma *machine learning* terbaik yang dapat digunakan untuk memprediksi kinerja pelajar diawal perkuliahan pada mekanisme evaluasi pembelajaran yang dirancang.

2. Untuk mengukur performa prediksi yang diusulkan dalam hal proses klasifikasi dan akurasi prediksi.
3. Untuk mengoptimalkan performa prediksi yang diusulkan pada *machine learning* untuk memenuhi kinerja akurasi sehingga akan meningkatkan kualitas proses pembelajaran.

D. MANFAAT PENELITIAN

1. Menghasilkan mekanisme evaluasi pembelajaran serta algoritma *machine learning* yang mampu memprediksi tingkat resiko keberhasilan pelajar diawal semester, sehingga mempunyai waktu yang banyak untuk mengoptimalkan proses pembelajaran hingga akhir masa studi.
2. Mengetahui kualitas performa prediksi prestasi akademik mahasiswa di perguruan tinggi secara keseluruhan dibandingkan dengan metodologi prediktif lainnya.
3. Memberikan informasi tentang fitur yang mempengaruhi dalam pencapaian akademik, yang didapatkan dari prestasi pelajar sebelumnya, keterlibatan aktivitas pembelajaran, dan kondisi demografi pelajar sebagai prediktor kinerja akademik.
4. Menjadi rujukan bagi pengembangan penelitian dibidang *Learning Analytics*.

E. BATASAN MASALAH

Untuk mempertajam sasaran penelitian dan membatasi ruang lingkup penelitian maka peneliti membatasi masalah dari penelitian yang diusulkan, yaitu:

1. Penelitian ini menerapkan teknik klasifikasi dan prediksi menggunakan algoritma *CNN-LSTM*.
2. Data Inputan menggunakan kondisi demografi, data *history* pelajar dan data *log* aktivitas pembelajaran.
3. Data penelitian diambil dari *log* pembelajaran melalui LMS, data demografi didapatkan dari sistem informasi akademik, dan data tambahan bersasal dari *dashboard* yang digunakan untuk mengumpulkan data awal.
4. Data yang dianalisis berasal dari data mata kuliah dalam satu semester.

BAB II
TINJAUAN PUSTAKA
A. LANDASAN TEORI

1. Learning Analytics

Secara terminologi *learning analytics (LA)* terdiri dari *learning* yang berarti pembelajaran dimana peserta didik melakukan interaksi dengan pendidik maupun sumber belajar dan *analytics* yang kerap didefinisikan sebagai suatu teknik terapan di bidang ilmu komputer, matematika, dan statistik untuk membantu manusia dalam mengartikan berbagai informasi berukuran besar. *Learning analytics* [23], didefinisikan sebagai pengukuran, pengumpulan, analisis dan penyajian data tentang pembelajar dengan tujuan memahami dan memaksimalkan pembelajaran dan lingkungan pembelajaran. Lebih lanjut LA didefinisikan sebagai aktivitas interpretasi data mengenai peserta didik, yang diperoleh semasa yang bersangkutan melakukan aktivitas akademik, untuk kebutuhan memprediksi kinerjanya di masa mendatang dan mendeteksi potensi masalah yang dapat timbul di kemudian hari.

2. Mekanisme Pelaksanaan LA dalam pembelajaran

Data secara eksplisit diambil dari beragam sumber seperti: hasil ujian, rekaman diskusi dan interaksi dengan instruktur atau sesama pelajar, aktivitas ekstrakurikuler, dan lain sebagainya. Secara sederhana

LA fokus pada peningkatan kinerja peserta didik, sehingga agregat data yang dikumpulkan dapat dimanfaatkan oleh guru, dosen, instruktur, dan pimpinan satuan pendidikan dalam memperbaiki atau merevisi ekosistem atau komponen lingkungan pembelajaran yang ada di sekitarnya, seperti: kurikulum, program pendidikan, desain pembelajaran, dan lain sebagainya.

3. Machine Learning

Machine learning merupakan salah satu pendekatan yang digunakan pada data mining. Pendekatan *machine learning* dapat menangani masalah dengan menemukan model algoritma yang tepat dan lebih baik dalam menghasilkan nilai prediksi dari sebuah variabel input. Hal ini bisa dilakukan karena metode *machine learning* menginterpretasikan bagaimana komputer dapat belajar, bekerja atau memperbaiki performansi berdasarkan data. Metode ini disebut sebagai metode yang dapat membuat keputusan yang tepat dan cerdas dari kompleksitas pola data. *Machine learning* merupakan pendekatan yang cepat berkembang dan beradaptasi.

Pendekatan *machine learning* memiliki 4 kategori yang umumnya diaplikasikan pada konsep data mining :

1. Supervised learning sering disebut sebagai metode klasifikasi.

Metode ini berawal dari data label pada training dataset. Hal pertama pada proses klasifikasi yaitu tahapan mempelajari mapping atau fungsi dimana dapat memprediksi kelas yang terkait dengan label

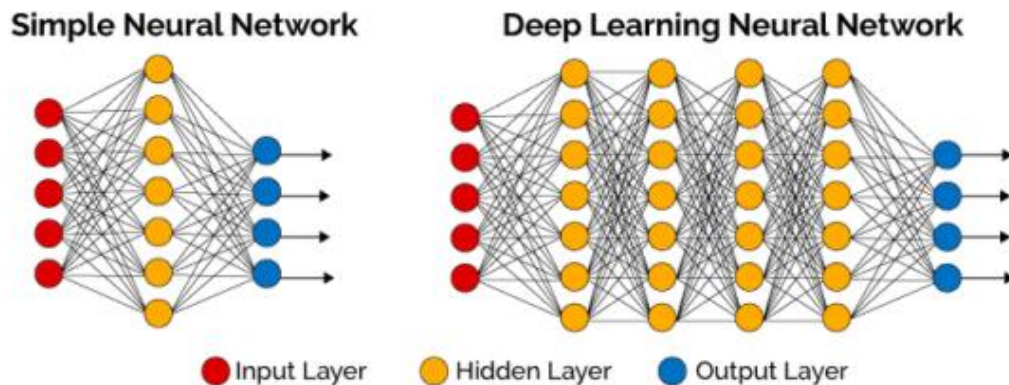
dari sebuah input. Metode ini dapat menggunakan dataset dari berbagai sumber baik itu data yang mudah dipahami atau data dengan tingkat kerumitan yang tinggi.

2. **Unsupervised learning** merupakan metode pengelompokan (clustering). Proses clustering sendiri adalah bertujuan untuk mengelompokkan obyek untuk menemukan kelas-kelas data yang tidak diketahui pada dataset.
3. **Semi-supervised learning** merupakan teknik *machine learning* dimana untuk mempelajari model algoritma menggunakan dua jenis data yaitu labeled data dan unlabeled data. Dalam arti, labeled data digunakan untuk mengetahui dan mempelajari model yang digunakan sementara unlabeled data digunakan sebagai batasan antara masing-masing kelas pada model. Unlabeled data yang dimaksud dapat dikelompokkan menjadi kategori positif atau negatif dimana masing-masing berperan menjadi batasan dalam menentukan keputusan.
4. **Active learning**, salah satu pendekatan *machine learning* dimana user berperan aktif dalam proses pendekatan. Pengguna dapat melakukan pelabelan pada contoh data dimana kemungkinan data merupakan unlabeled data atau dataset yang dihasilkan oleh perpaduan dari hasil pembelajaran program. Tujuan metode ini adalah untuk mengoptimalkan kualitas model dengan cara

memberikan peran atau pemahaman user untuk membuat batasan jumlah data yang akan diuji sesuai dengan kebutuhan user.

4. Deep Learning

Deep learning merupakan sebuah algoritma neural network yang menggunakan metadata sebagai input dan mengolah input tersebut menggunakan sekumpulan fungsi transformasi non-linier yang ditata berlapis-lapis dan mendalam [24]. Algoritma *deep learning* dirancang untuk menganalisis data dengan struktur logika yang mirip dengan bagaimana manusia mengambil keputusan. Algoritma ini memiliki kemampuan menangkap fitur atau ciri yang relevan dari suatu data untuk keperluan pemecahan suatu masalah. Pada *deep learning* terdapat *hidden layer* (lapisan tersembunyi) yang bertugas untuk melatih serangkaian fitur unik berdasarkan output dari jaringan sebelumnya. Algoritma ini menjadi semakin kompleks dan bersifat abstrak ketika jumlah *hidden layer* semakin bertambah banyak. Neural network yang dimiliki *deep learning* terbentuk dari hirarki sederhana dengan beberapa lapisan tingkat tinggi atau banyak lapisan (*multi layer*). Berikut pada Gambar 2.1 adalah ilustrasi perbedaan antara neural network sederhana yang hanya menggunakan satu atau dua *hidden layer* dengan *deep learning* yang menggunakan banyak *hidden layer*.



Gambar 2.1. Perbandingan *Neural Network* sederhana dengan *Deep Learning Neural Network* [22].

Seperti yang tampak pada Gambar 2.1, neural network sederhana yang menggunakan jaringan feed forward hanya mempunyai satu atau dua *hidden layer*. Sedangkan pada *deep learning neural network* terdapat *hidden layer* yang jumlahnya diatas dua *layer* keatas yang bahkan bisa sampai ratusan *layer*. Penggunaan *hidden layer* yang banyak inilah yang memungkinkan *deep learning* mampu menganalisis data dari berbagai dimensi, sehingga dapat menarik kesimpulan dari suatu permasalahan dengan lebih akurat dan detail.

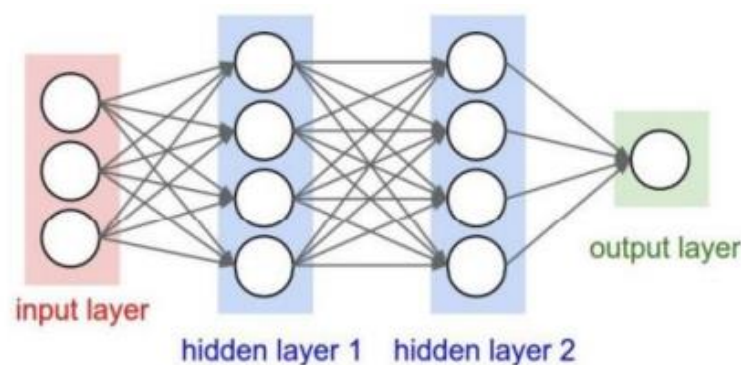
Sama seperti neural network pada umumnya, *layer* pada *deep learning neural network* terbagi atas tiga *layer* utama yakni *input layer*, *hidden layer*, dan *output layer*. Ketiga *layer* tersebut memiliki perannya masing-masing. *Input layer* berfungsi untuk menampung data mentah seperti suara, teks, atau gambar. *Hidden layer* bertugas untuk meneliti dan mengklasifikasikan data tersebut berdasarkan referensi yang sudah ada. *Output layer* berfungsi untuk menyajikan hasil penelitian dan klasifikasi dari suatu data. Algoritma *deep learning* terbagi lagi

kedalam beberapa jenis, namun algoritma yang paling sering digunakan yakni *convolutional neural network (CNN)*, *Recurrent Neural Network (RNN)*, Dan *Long Short-Term Memory Network (LSTM)*.

5. Convolutional Neural Network

Pengembangan dari *Multilayer Perceptron (MLP)* yang digunakan untuk mengolah data dua dimensi adalah ide awal dari *Convolutional Neural Network (CNN)*. Karena kedalaman jaringan dan memiliki *layer* atau lapisan yang banyak maka CNN termasuk ke dalam jenis *deep neural network*.

CNN memiliki cara kerja yang sama dengan MLP, yang membedakannya adalah setiap neuron pada CNN dilewati dalam bentuk dua dimensi dan pada MLP ukuran setiap neuronnya hanya satu dimensi. Pada Gambar 2.2. ditunjukkan arsitektur MLP.

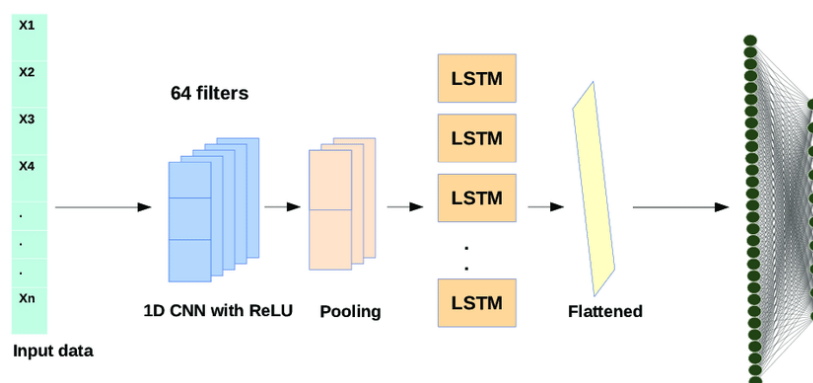


Gambar 2. 2. Arsitektur MLP sederhana

MLP memiliki i layer yang pada Gambar 2.2, digambarkan dengan kotak merah dan biru dengan setiap *layer* berisi j_i neuron yang diilustrasikan dengan lingkaran putih. Input data berbentuk satu dimensi yang diterima oleh MLP dan dipropagasikan pada jaringan sebelum

MLP menghasilkan output. Setiap hubungan antar neuron pada dua *layer* yang bersebelahan memiliki parameter bobot satu dimensi yang menentukan kualitas mode. Setiap data input pada *layer* dilakukan operasi linear dengan nilai bobot yang ada, kemudian hasil komputasi akan ditransformasi menggunakan operasi non linear yang disebut sebagai fungsi aktivasi [24].

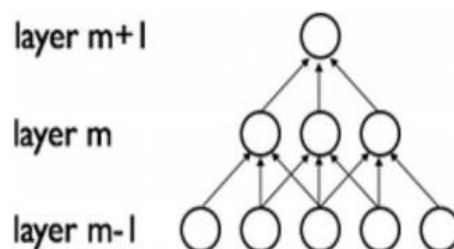
Convolutional Neural Network (CNN) merupakan jaringan saraf tiruan yang memberi umpan balik dimana jaringan saraf mempertahankan struktur hierarki dengan mempelajari representasi fitur internal dan menggeneralisasi fitur-fitur dalam masalah gambar secara umum seperti pengenalan objek dan masalah computer vision. Penggunaanya tidak terbatas pada gambar, bisa juga digunakan untuk mencapai hasil dalam masalah pemrosesan data dengan vektor *feature* skala besar dalam hal ini inputan data log pembelajaran [8].



Gambar 2. 3. Arsitektur model CNN dan LSTM

a. Convolution *layer*

Filter konvolusi menggunakan korelasi lokal dengan memaksakan pola konektivitas lokal di antara *layer* yang berdekatan pada setiap convolution *layer*. *Layer* yang lebih atas, m didapat dari sebuah subhimpunan satuan dari *layer* yang lebih rendah, $m - 1$. Keuntungan dari *convolutional layer* dibandingkan dengan MLP adalah jumlah parameter yang berkurang secara signifikan disebabkan oleh parameter bagi. Konektivitas tersebut digambarkan dalam Gambar 2.4.



Gambar 2. 4. Jaringan dengan 3 convolution *layer*

Proses ini adalah *layer* pertama pada arsitektur jaringan CNN-LSTM. Sesuai dengan namanya, pada *layer* ini dilakukan proses konvolusi untuk semua vektor data yang ada pada fitur inputan, dimana untuk vektor setelah kata terakhir yang masih berada pada window bernilai nol.

Namun pada penelitian ini, konvolusi berakhir hingga vektor fitur ke- $n - h + 1$. Pada dasarnya proses konvolusi adalah proses mengoperasikan setiap input dengan setiap bobot menggunakan operasi dot produk lalu menjumlahkan hasil operasi tersebut dengan nilai bias pada masing-masing bobot. Hasil dari operasi dot produk

dengan nilai bobot dan penjumlahan dengan nilai bias akan diproses kedalam activation function.

Untuk menggunakan matriks nilai bobot berukuran 2x3 dan stride sebanyak 1. Ukuran matriks nilai bobot ditunjukkan dengan bersatunya 3 vektor input yang masing-masing vektornya memiliki 2 elemen. bahwa pergeseran operasi konvolusi sebanyak 1 vektor, hal tersebut menunjukkan jumlah stride.

Setelah beberapa *vektor fitur* digabung, ulasan direpresentasikan seperti pada persamaan (2.1)

$$x_{1:n} = x_1 \oplus x_2 \oplus x_3 \oplus \dots \oplus x_n \quad (2.1)$$

Keterangan:

n : banyaknya vektor fitur

x_i : vektor fitur yang berada pada $\mathbb{R}^{50}$ dengan indeks ke- i

Dimana $\oplus$ merupakan operator penggabungan. Secara umum $x_{i:i+j}$ berarti hasil gabungan antara beberapa *vektor fitur* dari indeks ke- i sampai ke- $i + j$ seperti berikut:

$$x_i, x_{i+1}, x_{i+2}, \dots, x_{i+j} \quad (2.2)$$

Filter (bobot) pada operasi konvilusi dibutuhkan untuk mendapatkan nilai feature map, berikut adalah hubungan antara feature map dan filter:

$$net = w \cdot x_{i:i+h-1} + b \quad (2.3)$$

$$c_i = f(net) \quad (2.4)$$

Keterangan:

c_i : nilai feature map pada indeks ke- i

h : ukuran window fitur

w : filter yang berada pada $\mathbb{R}^{50h}$

b : parameter bias

Activation function atau fungsi non linear yang digambarkan sebagai $f(net)$ pada persamaan (2.7) pada penelitian ini adalah fungsi Rectified Linear Unit (ReLU). Hasil dari fungsi tersebut memiliki batasan output bernilai positif. Berikut adalah rumus dari Rectified Linear Unit (ReLU) function:

$$ReLU(x) = \max(0, x) \quad (2.5)$$

Jadi, persamaan feature map menjadi seperti berikut:

$$net = w \cdot x_{i:i+h-1} + b \quad (2.6)$$

$$c_i = ReLu(net) \quad (2.7)$$

Untuk setiap window fitur yang ada dalam kalimat ulasan diterapkan filter w . Seluruh fitur pada data digambarkan sebagai $\{x_{1:h}, x_{2:h}, x_{3:h}, \dots, x_{n-h+1:n}\}$ sehingga didapatkan feature map sebagai berikut:

$$c = [c_1, c_2, c_3, \dots, c_{n-h+1}] \quad (2.8)$$

c adalah feature map.

Pada penelitian ini akan digunakan lebih dari satu filter dan window fitur untuk mendapatkan hasil yang maksimal.

b. Pooling *layer*

Pooling *layer* adalah komponen penting kedua pada CNN setelah convolution *layer*. Pooling *layer* memiliki fungsi sebagai down-sampling yang bersifat non-linear. Pooling *layer* bertujuan untuk menyeleksi informasi yang penting dari feature map. Seleksi tersebut dilakukan dengan mengeliminasi nilai yang kurang maksimal untuk mengurangi proses komputasi. Selain itu pooling juga dilakukan untuk menyediakan bentuk translasi invarian. Pooling juga mengatasi masalah yang ditimbulkan dari perbedaan panjang feature map sehingga didapatkan dimensi yang sama.

Operasi pooling dilakukan setelah didapatkan feature map. Feature map yang telah didapat dari convolution *layer* akan diproses di pooling *layer*. Inti dari lapisan ini adalah pengambilan nilai-nilai penting dari setiap feature map dengan mengambil nilai yang maksimum. Terdapat beberapa metode untuk proses pooling yaitu max pooling, average pooling, dan lain-lain. Pada penelitian ini metode yang digunakan adalah metode pooling yang sering digunakan untuk task NLP yaitu max pooling. Operasi pooling dirumuskan sebagai berikut:

$$\hat{c} = \max\{c\} \quad (2.9)$$

Hasil dari pooling *layer* merupakan suatu vektor yang terdiri atas nilai maksimum tiap feature map dan vektor tersebut memiliki

jumlah elemen sebanyak m . Hal ini dikarenakan banyaknya filter berjumlah m pula. Vektor tersebut digambarkan seperti persamaan (2.10).

$$z = [\hat{c}_1, \hat{c}_2, \hat{c}_3, \dots \hat{c}_m] \quad (2.11)$$

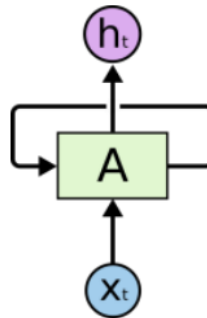
c. Output *layer*

Peran output *layer* adalah klasifikasi ulasan yang telah diekstrak fiturnya pada *layer-layer* sebelumnya menggunakan jaringan syaraf tiruan dengan kelas-kelas tujuan yang ditentukan antara kelas positif dan negatif. Vektor feature yang merupakan output dari pooling *layer* akan melewati hubungan penuh jaringan Long Short Term Memory (LSTM) untuk mendapatkan skor prediksi. Label untuk ulasan positif dan ulasan negatif pada penelitian ini adalah +1 dan -1.

6. Recurrent Neural Network

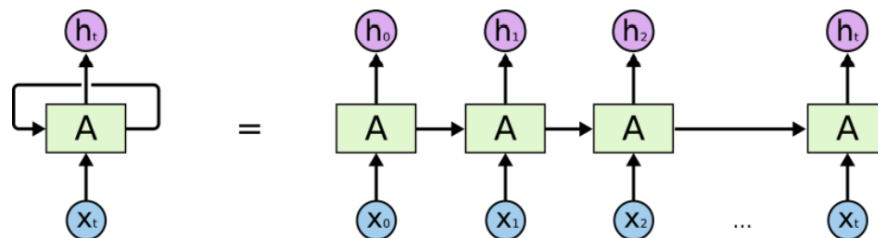
Dasar dari pengembangan *Recurrent Neural Network* (RNN) adalah cara berfikir manusia pada umumnya yang tidak setiap saat membuat keputusan secara tunggal. Manusia cenderung menimbang atau memperhitungkan masa lalu dalam pengambilan keputusan. Seperti analogi tersebut, RNN tidak membuang begitu saja informasi dari masa lalu. Hal itulah yang membuat RNN berbeda dari *Artificial Neural Network* biasa. RNN adalah salah satu *neural network* yang diperuntukkan memproses data bersambung (*sequential data*). RNN

dapat menyimpan memori terdahulu karena proses *looping* pada arsitekturnya.



Gambar 2. 5. Arsitektur RNN [23]

Gambar 2.5 menunjukkan arsitektur dari algoritma RNN yang berulang. Pada Gambar 2.6 akan ditunjukkan arsitektur RNN yang mengilustrasikan *looping* pada arsitektur RNN.



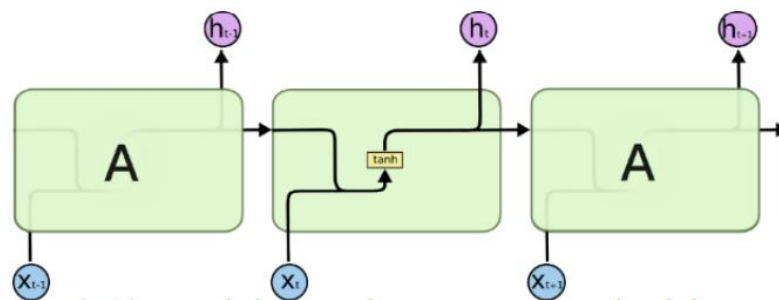
Gambar 2. 6. Looping pada arsitektur RNN [23]

7. Long Short Term Memory

Long Short Term Memory (LSTM) adalah sebuah jenis khusus dari algoritma *Recurrent Neural Network* (RNN) yang mampu mempelajari ketergantungan jangka panjang. LSTM cocok digunakan untuk masalah yang memiliki ketergantungan jangka panjang. Mengingat informasi jangka panjang adalah perilaku bawaan LSTM. LSTM juga memiliki

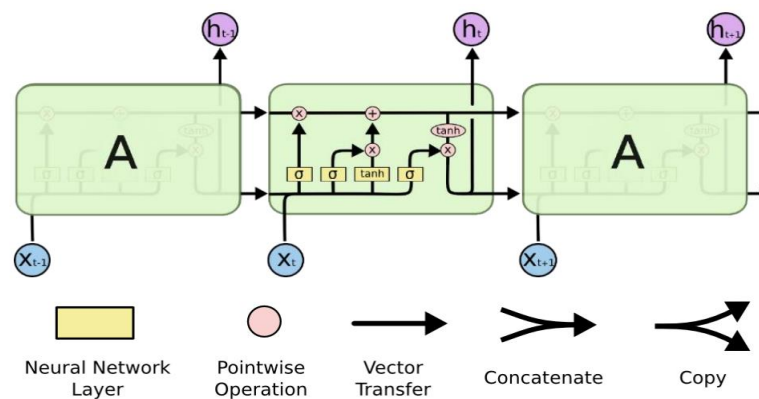
struktur rantai seperti struktur RNN, perbedaannya terletak pada struktur modul pengulangannya.

Seluruh *Recurrent Neural Network* memiliki bentuk rantai modul pengulangan jaringan syaraf tiruan. Dalam RNN standar, Modul pengulangan ini akan memiliki struktur yang sangat sederhana, seperti lapisan *tanh* tunggal [25].



Gambar 2. 7. Modul pengulang RNN yang berisi satu layer [23]

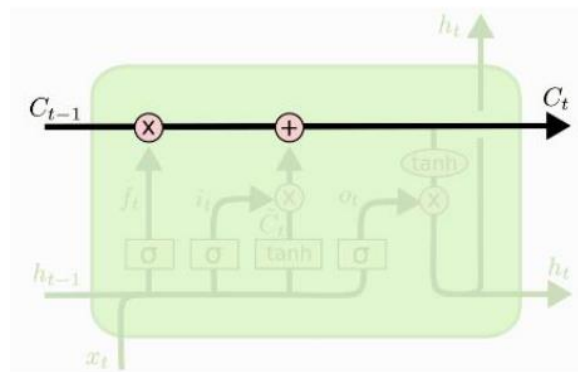
LSTM juga memiliki struktur seperti rantai, namun modul pengulangan memiliki struktur yang berbeda. Sebagai gantinya, LSTM memiliki empat lapisan jaringan syaraf tunggal yang berinteraksi dengan cara yang berbeda dengan RNN.



Gambar 2. 8. Modul pengulang dalam LSTM berisi empat layer [23]

Pada Gambar 2.8. setiap baris membawa keseluruhan vektor, dari *output* satu simpul ke *input* lain. Lingkaran merah muda mewakili operasi *pointwise*, seperti penambahan vektor, sedangkan kotak kuning adalah lapisan jaringan syaraf tiruan yang dipelajari. Penggabungan garis menunjukkan rangkaian (*concatenation*), sementara garis bercabang menunjukkan kontennya disalin dan salinannya masuk ke lokasi yang berbeda [8].

Kunci pada algoritma LSTM adalah *cell state* atau garis horizontal yang melewati bagian atas diagram. *Cell state* bekerja seperti *conveyor belt*. *Cell state* berjalan ke seluruh rantai hanya dengan sedikit interaksi linear. Sangat mudah bagi informasi untuk mengalir begitu saja tanpa perubahan.

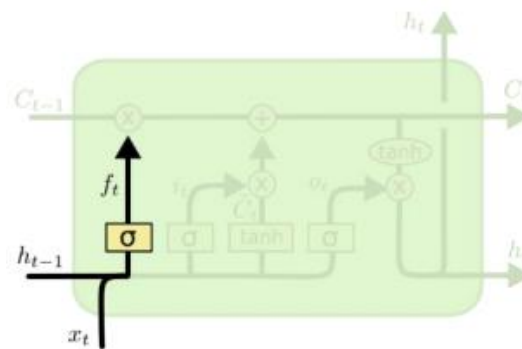


Gambar 2. 9. Ilustrasi *cell state* [23]

LSTM memiliki kemampuan untuk menambah dan menghapus informasi ke *cell state*, diatur secara teliti oleh struktur yang disebut *gates*. *Gates* adalah cara untuk melepas informasi yang lewat. Mereka terdiri dari lapisan jaringan saraf sigmoid dan operasi perkalian *pointwise*.

Berikut adalah langkah-langkah pada LSTM:

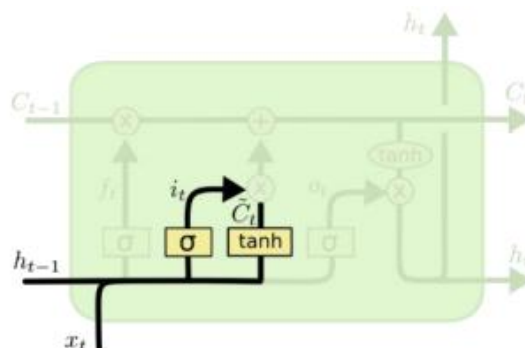
1. Menentukan informasi yang akan dibuang dari *cell state* ini dari tahap ini terletak pada *layer* sigmoid atau yang juga disebut *forget gate*. Sesuai dengan namanya, *layer* ini berperan untuk menyeleksi informasi yang akan dilupakan. *Forget gate* menghasilkan angka 0 dan 1 untuk *cell state* C_{t-1} . Angka 1 merepresentasikan untuk menjaga memori dan 0 representasi untuk melupakan memori.



Gambar 2. 10. Ilustrasi langkah pertama LSTM [23]

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (2.12)$$

2. Menentukan informasi yang akan dimasukkan ke cell state



Gambar 2. 11. Ilustrasi langkah kedua LSTM [23]

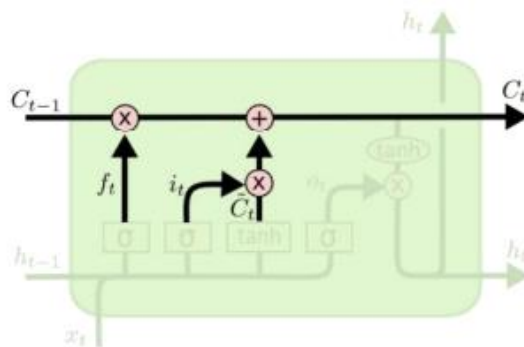
Pada tahap ini terdapat dua *layer* yaitu *layer* sigmoid dan *layer* *tanh*. *Layer* sigmoid dinamakan *input layer* yang memutuskan nilai

mana yang akan diperbarui. Selanjutnya *layer tanh* berperan untuk membentuk vektor dari nilai kandidat baru $\tilde{C}_t$ yang dapat ditambahkan ke *cell state*. Lalu *cell state* yang lama, C_{t-1} di-*update* menjadi *cell state* baru, C_t .

$$i_t = \sigma(W_i.[h_{t-1}, x_t] + b_i) \quad (2.13)$$

$$\tilde{C}_t = \tanh(W_c.[h_{t-1}, x_t] + b_c) \quad (2.14)$$

3. Menambahkan informasi baru

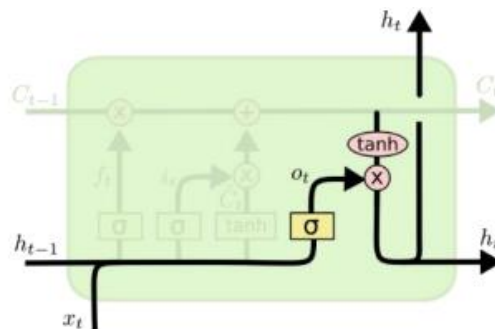


Gambar 2. 12. Ilustrasi langkah ketiga LSTM [23]

Selanjutnya akan kita eksekusi apa yang sudah diputuskan pada tahap sebelumnya. Persamaan (2.14) adalah cara meng-*update* informasi atau memori baru:

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \quad (2.15)$$

4. Menemukan output



Gambar 2. 13. Ilustrasi langkah keempat LSTM [23]

Akan dijalankan *layer* sigmoid yang menentukan bagian dari *cell state* mana yang akan dijadikan output. Lalu masukkan *cell state* melewati *tanh* (untuk memaksa nilainya menjadi antara -1 dan 1) dan mengalikannya dengan hasil dari *layer* sigmoid agar kita hanya menghasilkan bagian yang telah ditentukan.

$$o_t = \sigma(W_o [h_{t-1}, x_t] + b_o) \quad (2.16)$$

$$h_t = o_t * \tanh(C_t) \quad (2.17)$$

8. CNN-LSTM

CNN-LSTM adalah kombinasi antara algoritma *Convolutional Neural Network* (CNN) dan algoritma *Long Short Term Memory* (LSTM). Sesuai dengan urutan namanya, konstruksi model CNN-LSTM meletakkan algoritma LSTM pada *layer* terakhir di model jaringan CNN sehingga sebelum memasuki jaringan LSTM vektor input harus melalui proses algoritma CNN setelah itu jaringan LSTM. Tahapan pada model CNN-LSTM terdiri atas 3 tahapan:

1. Tahap Umpan Maju

Tahap umpan maju adalah proses masuknya input data yang melewati jaringan dari awal hingga didapatkan skor dari prediksi performa pelajar.

a. *Convolution Layer*

Lapisan pertama yang harus dilewati pada tahapan umpan maju merupakan *convolution layer*. Pada *layer* ini data masukan yang sudah berupa vektor-vektor *fitur* akan diolah dengan proses konvolusi sehingga didapatkan nilai *fitur* dari kumpulan vektor masukan.

Persamaan (2.1) digunakan untuk melakukan proses penggabungan dari vektor fitur dalam satu *window* dan bergerak sepanjang banyaknya kata. Setelah itu persamaan (2.6) dan (2.7) digunakan untuk mendapatkan nilai fitur dari data masukan. Hasil dari *layer* ini merupakan sebuah vektor *feature map*.

b. *Pooling Layer*

Lapisan kedua pada algoritma CNN-LSTM yaitu *pooling layer*. Pada lapisan ini akan diambil nilai maksimum dari *feature map* yang telah didapatkan dari *convolution layer*. Untuk mendapatkan nilai dari *pooling layer* digunakan persamaan (2.9).

c. *Output Layer*

Untuk mendapatkan skor dari prediksi performa akademik, maka digunakan persamaan (2.17).

2. Tahap Umpan Mundur

Tahap umpan mundur adalah proses pembelajaran atau pelatihan bagi model. Sesuai dengan namanya, proses ini bergerak mundur dari *layer output* ke *layer input*. Hal ini bertujuan untuk mengevaluasi *error* yang dihasilkan.

Tujuan akhir pelatihan pada jaringan adalah untuk mencari *gradien* pada setiap filter dan parameter yang berkenaan dengan keluaran. Sehingga filter dapat diperbarui secara bertahap menggunakan metode *mini-batch gradient descent*. Prinsip dari metode ini yaitu pembaharuan dilakukan setiap *mini-batch* yang ditentukan. *Mini-batch* merupakan jumlah data

yang dibutuhkan untuk tiap pembelajaran. *Mini batch* yang digunakan pada penelitian ini yaitu 64. Misalkan θ merupakan vektor semua parameter yang digunakan pada algoritma, maka pembaharuan dirumuskan sebagai berikut:

$$\theta_{t+1} = \theta_t - \eta \cdot \nabla_{\theta} L \quad (2.18)$$

η adalah parameter pembelajaran. Untuk menentukan nilai η dilakukan berdasarkan aturan Adam. Berdasarkan aturan Adam persamaan untuk pembaharuan parameter pembelajaran berubah menjadi berikut:

$$\theta_{t+1} = \theta_t + \Delta\theta_t \quad (2.19)$$

Dimana

$$\Delta\theta_t = - \frac{RMS[\Delta\theta]_{t-1}}{RMS[\nabla_{\theta} L]_t} \nabla_{\theta} L$$

RMS adalah *root mean square*.

Perlu dilakukan pembelajaran dengan terus melakukan pembaharuan terhadap semua parameter yang ada pada Penelitian ini untuk mendapatkan hasil yang optimal.

a. *Output Layer*

Terdapat parameter W dan B_0 pada *output layer* untuk menentukan skor data nilai fitur. Maka dari itu dibentuk nilai optimal dengan cara mengurangi parameter sebelumnya dengan gradiennya. Persamaan (2.20) merupakan penurunan rumus untuk tiap data ke- i dan bergerak sampai banyaknya data dan konstanta $C=1$ untuk

mempermudah perhitungan. Untuk mendapatkan gradien dari parameter W adalah sebagai berikut:

$$\begin{aligned}
 \frac{\partial L}{\partial W} &= \frac{\partial}{\partial W} \left[\frac{1}{2} \|W\|^2 + \max(0, 1 - y_i f(z_i))^2 \right] \\
 &= \frac{\partial}{\partial W} \left[\frac{1}{2} \|W\|^2 \right] + \frac{\partial}{\partial W} [\max(0, 1 - y_i f(z_i))^2] \\
 &= W - 2y_i \max(0, 1 - y_i f(z_i)) \frac{\partial f}{\partial w} \\
 &= W - 2y_i z_i \max(0, 1 - y_i f(z_i)) \quad (2.20)
 \end{aligned}$$

Gradien bias didapatkan dengan menurunkan fungsi loss terhadap bias b_0 . Persamaan (2.21) merupakan penurunan rumus untuk tiap data ke- i dan bergerak sampai banyaknya data dan konstanta $C=1$ untuk mempermudah perhitungan.

$$\begin{aligned}
 \frac{\partial L}{\partial b_0} &= \frac{\partial}{\partial b_0} \left[\frac{1}{2} \|W\|^2 + \max(0, 1 - y_i f(z_i))^2 \right] \\
 &= \frac{\partial}{\partial b_0} \left[\frac{1}{2} \|W\|^2 \right] + \frac{\partial}{\partial b_0} [\max(0, 1 - y_i f(z_i))^2] \\
 &= -2y_i \max(0, 1 - y_i f(z_i)) \frac{\partial f}{\partial b_0} \\
 &= -2y_i z_i \max(0, 1 - y_i f(z_i)) \quad (2.21)
 \end{aligned}$$

$\frac{\partial L}{\partial W}$ dan $\frac{\partial L}{\partial b_0}$ selanjutnya digunakan untuk memperbaharui parameter W dan b_0 .

b. *Convolution Layer*

Terdapat banyak parameter w dan b pada *convolution layer* karena setiap *window* dioperasikan parameter yang berbeda. Misal w_k dan b_k merupakan parameter yang digunakan untuk mendapatkan *feature map* ke- k dan konstanta $C = 1$ untuk mempermudah perhitungan, maka dapat diperoleh turunan gradien dari parameter tersebut sebagai berikut:

$$\begin{aligned}
 \frac{\partial L}{\partial w_k} &= \frac{\partial}{\partial w_k} \left[\frac{1}{2} \|W\|^2 + \max(0, 1 - y_i f(z_i))^2 \right] \\
 &= \frac{\partial}{\partial w_k} \left[\frac{1}{2} \|W\|^2 \right] + \frac{\partial}{\partial w_k} [\max(0, 1 - y_i f(z_i))^2] \\
 &= -2y_i \max(0, 1 - y_i f(z_i)) \frac{\partial f}{\partial w_k} \\
 &= -2y_i W_k \max(0, 1 - y_i f(z_i)) \frac{\partial z_i}{\partial w_k} \\
 &= -2y_i W_k \max(0, 1 - y_i f(z_i)) (1 - (z_i^k)^2) x_{j:j+h-1} \quad (2.22)
 \end{aligned}$$

Dimana,

$$j = \operatorname{argmax} c_j^k$$

Parameter bias diperoleh dari persamaan berikut:

$$\begin{aligned}
 \frac{\partial L}{\partial b_k} &= \frac{\partial}{\partial b_k} \left[\frac{1}{2} \|W\|^2 + \max(0, 1 - y_i f(z_i))^2 \right] \\
 &= \frac{\partial}{\partial b_k} \left[\frac{1}{2} \|W\|^2 \right] + \frac{\partial}{\partial b_k} [\max(0, 1 - y_i f(z_i))^2] \\
 &= -2y_i \max(0, 1 - y_i f(z_i)) \frac{\partial f}{\partial b_k} \\
 &= -2y_i W_k \max(0, 1 - y_i f(z_i)) \frac{\partial z_i}{\partial b_k} \\
 &= -2y_i W_k \max(0, 1 - y_i f(z_i)) (1 - (z_i^k)^2) \quad (2.23)
 \end{aligned}$$

Indeks k bergerak sampai dengan dimensi dari vektor z yang merepresentasikan banyaknya filter yang digunakan sedangkan indeks $k-i$ bergerak sampai banyaknya data pembelajaran.

3. Perhitungan Nilai *Loss* (*Error*)

Terdapat beberapa parameter yang digunakan untuk mendapatkan skor ulasan pada penelitian ini. Parameter tersebut menentukan akurasi pada klasifikasi. Oleh karena itu, dilakukan evaluasi terhadap hasil skor yang didapatkan.

Perhitungan nilai *loss* atau *error* dapat dihasilkan dari persamaan (2.17) karena penelitian ini menggunakan LSTM untuk klasifikasi. Dapat dilihat seberapa jauh parameter atau bobot yang digunakan dapat mengklasifikasi data masukkan kedalam kelas yang diinginkan berdasarkan nilai error yang diperoleh. Pembaharuan nilai bobot atau parameter yang digunakan ditentukan oleh nilai error tersebut.

9. Confusion Matrix

Confusion Matrix merupakan metode evaluasi yang digunakan untuk mengukur performansi pemodelan berdasarkan tabel matrix (2x2) dengan dua nilai kelas yaitu nilai Y atau nilai N. Pada tabel 2.1 dijelaskan bahwa nilai sel vertikal (kolom) berisi data hasil observasi sementara nilai sel horizontal (baris) berisi data prediksi . Penentuan nilai akurasi dikelompokkan menjadi 4 cara yang berbeda yaitu :

- a. Jika nilai observasi positif sesuai dengan nilai ekspektasi yang ditentukan, maka hasilnya adalah *True Positive* (TP).
- b. Jika nilai observasi negatif namun masih dikategorikan sesuai dengan nilai ekspektasi, maka hasilnya adalah *False Positive* (FP).
- c. Jika nilai observasi positif tetapi tidak sesuai dengan nilai ekspektasi, maka hasilnya adalah *False Negative* (FN).
- d. Nilai observasi yang negatif dan dinyatakan tidak sesuai dengan nilai ekspektasi, maka hasilnya adalah *True Negative* (TN).

Tabel 2. 1. Confusion Matrix

Total Population = P + N	Actual Value		
		True (P)	False (N)
Prediction Value	True (P)	True Positive (TP)	False Positive (FN)
	False (N)	False Negative (FN)	True Negative (TN)

Umumnya, ada 4 tipe pengukuran evaluasi yang dapat dilakukan menggunakan metode *Confusion Matrix*, diantaranya adalah :

- a. Mengukur kemampuan model dalam menemukan peluang nilai observasi yang tepat dengan nilai ekspektasi (nilai positif) disebut dengan *sensitivity test*. Konsep pengukuran ini sama dengan pengukuran *recall rate* dimana dilakukan untuk menemukan proporsi dari jumlah nilai yang relevan dari keseluruhan data. Sebagai contoh, untuk menentukan drop out, ekspektasi nilai positif (Y) adalah $IPK < 2.5$

sedangkan nilai negatif (N) adalah $IPK > 2.5$. Jika angka *sensitivity test* adalah 90% maka mahasiswa dengan $IPK < 2.5$ memiliki peluang 95% terindikasi *drop out*. Rumus menghitung *sensitivitas (Recall)* adalah sebagai berikut :

$$Recall = \frac{TP}{(TP+FN)} \quad (2.24)$$

- b. Mengukur besar kecilnya peluang nilai ekspektasi negatif yang terjadi disebut *specificity test*. Jika angka *specificity test* sebesar 85% pada mahasiswa dengan $IPK > 2.5$ maka dapat dinyatakan bahwa kelompok mahasiswa tersebut tidak terindikasi dropout. Rumus yang digunakan untuk menghitung spesifitas adalah :

$$Specificity = \frac{TN}{(TN+FP)} \quad (2.25)$$

- c. Untuk mengetahui proporsi dari nilai observasi yang dinyatakan sangat relevan dengan nilai ekspektasi dapat dilakukan menggunakan rumus *precision test* di bawah ini:

$$Precision = \frac{TP}{(TP+FP)} \quad (2.26)$$

- d. Untuk mengetahui apakah metode yang digunakan dapat mengeksekusi data nilai observasi dengan memilih nilai yang tepat, dalam arti kemungkinan data bernilai negatif nihil. Pengukuran ini dapat dilakukan dengan menghitung nilai akurasi menggunakan rumus:

$$Accuracy = \frac{(TP+TN)}{(TP+FP+TN+FN)} \quad (2.27)$$

- e. F-Measure (F-1 Score) menggambarkan perbandingan rata-rata *precision* dan *recall* yang dibobotkan. *Accuracy* tepat kita gunakan sebagai acuan performansi algoritma jika dataset kita memiliki jumlah

data *False Negative* dan *False Positive* yang sangat mendekati (*symmetric*). Namun jika jumlahnya tidak mendekati, maka sebaiknya kita menggunakan *F1 Score* sebagai acuan.

$$F1\ Score = \frac{2*(Recall*Precision)}{(Recall+Precision)} \quad (2.28)$$

B. PENELITIAN TERKAIT

Saat ini ada beberapa penelitian tentang *academic performance prediction* yang masih sangat banyak diteliti dan kemudian dipublikasikan pada jurnal bereputasi, salah satunya dengan judul *Models for early prediction of at-risk students in a course using standards-based grading* [21], pada penelitian ini membahas tentang penggunaan Model Prediksi untuk mengidentifikasi pelajar yang beresiko lebih awal dan memberi peringatan dini kepada pengajar dan pelajar. Menggunakan 6 jenis metode prediksi *Logistic Regression*, *Support Vector Machine*, *Decision Tree*, *Multi layer Preceptron*, *Naive Bayes Classifier*, dan *K-Nearest Neighbor* untuk menguji model prediksi terbaik. Dengan data input berupa Nilai Kuis, Partisipasi Kelompok, Tugas Rumah, Hasil Praktikum, Kegiatan Pemodelan Matematika, dan Tugas Tertulis hasil menunjukkan bahwa model yang terbentuk sangat tergantung pada data kinerja pelajar yang dikumpulkan selama semester berjalan. Nilai akhir sangat mempengaruhi kualitas data dan daya prediksi variabel yang mengakibatkan pengaruh signifikan pada akurasi dari hasil prediksi [21].

Penelitian berikut dengan judul *Evaluating the effectiveness of educational data mining techniques for early prediction of students' academic failure in introductory programming courses* [26], pada penelitian ini membanding tentang efektivitas teknik data mining untuk memprediksi lebih awal, pelajar yang cenderung gagal dalam kuliah pengantar pemrograman. Mencari pengaruh pra processing data untuk meningkatkan efektivitas teknik mining, algoritma yang digunakan diantaranya SVM, Decision Tree J48, NN, dan Naive Bayes Pada eksperimennya untuk mencari dan Mengevaluasi teknik Mining pada data yang telah diproses *fine-tuning* sebelumnya dan data tanpa *fine-tuning* Menggunakan metode *grid search*, *decision tree pruning*, *updating learning rate*, *updating weight hidden layer*. Hasil menunjukkan setelah dilakukan *fine-tuning* pada *pre-processing* data maka didapatkan nilai F-Measure $0,92 = 92\%$ dan tanpa *fine-tuning* *f-measure* $0,83 = 83\%$ mampu memprediksi pelajar yang gagal perkuliahan [26].

Penelitian berikutnya dengan judul *On the prediction of students' quiz score by recurrent neural network* [16], pada penelitian ini mengeksplorasi faktor yang mempengaruhi peningkatan kinerja Prediksi nilai kuis pelajar menggunakan *reccurent neural network* (RNN), menggunakan Log 15 jenis perkuliahan dengan mengambil data dari sistem *e-Learning Moodle*, sistem *mahara*, dan sistem *e-book Book Roll*, yang kemudian memprediksi keberhasilan nilai pelajar,

Setiap minggu data dari seorang pelajar dimasukkan ke dalam RNN. Skor kuis pelajar dari 0 sampai 100 diregresikan sebagai output. Data deret waktu dimasukkan ke dalam RNN, dan setiap waktu diprediksi oleh RNN yang dilatih. Untuk *Training data* RNN, menerapkan *back propagation through time (BPTT)* untuk memperbarui parameter jaringan berulang kali dan mempelajari parameter optimal [10].

Untuk penelitian berikut dengan judul *a deep learning classification framework for early prediction of team-based academic performance* [27], pada penelitian ini membahas tentang prediksi kinerja kelompok pelajar menggunakan kerangka *Deep Neural Network (DNN)* dengan dua *hidden layer* untuk memprediksi kinerja awal kelompok. Kerangka DNN kemudian di evaluasi dengan fungsi aktivasi yang berbeda (Sigmoid, ReLu, and Tanh) dan mengoptimalkan kerangka DNN dengan *Adagrad* dan *Adadelata*. Hasil dari penelitian ini menunjukkan bahwa akurasi prediksi dari kerangka kerja DNN, ketika dioptimalkan dengan *Adadelata* dan *Adagrad*, akurasi menjadi 76,73% dan 82,39%, sedangkan prestasi belajarnya secara keseluruhan adalah 80,76% dan 86,57%, artinya mampu memprediksi kinerja tim secara memadai dan akurat [27].

Penelitian selanjutnya dengan judul *artificial neural networks in academic performance prediction: systematic implementation and predictor evaluation* [14], penelitian ini menguji prosedur sistematis untuk mengimplementasikan jaringan saraf tiruan dalam memprediksi

prestasi akademik di perguruan tinggi. Tujuan kedua adalah untuk menganalisis pentingnya beberapa prediktor terkait kinerja akademik di pendidikan tinggi. Temuan menunjukkan bahwa mungkin untuk secara sistematis mengimplementasikan jaringan syaraf tiruan dalam mengklasifikasikan prestasi akademik siswa baik “tinggi” (akurasi 82%) atau “rendah” (akurasi 71%). Jaringan saraf tiruan mengungguli algoritma *machine learning* lainnya dalam evaluasi metrik seperti recall dan skor F1. Selanjutnya, ditemukan bahwa prestasi akademik sebelumnya, kondisi sosial ekonomi, dan karakteristik sekolah menengah atas, merupakan prediktor penting dari kinerja akademik siswa dalam pendidikan tinggi. pada penelitian ini membahas rekomendasi untuk mengimplementasikan jaringan saraf tiruan untuk analisis kinerja akademik di perguruan tinggi [14].

Penelitian selanjutnya dengan judul *predicting students at risk of academic failure using ensemble model during pandemic in a distance learning system* [12], pada penelitian ini memprediksi performa pelajar pada masa transisi mekanisme pembelajaran pada masa pandemi, penelitian ini mengambil data aktifitas pembelajaran *synchronous* dan *asynchronous*, data yang digunakan memperhatikan jenis-kelamin, jumlah catatan perkuliahan, waktu sesi online, materi kuliah yang di unduh, jumlah kehadiran, dan nilai kuis. memprediksi dengan menggabungkan algoritma *machine learning* atau dengan *ensemble model* dan mendapatkan hasil akurasi sebesar 90,34% [12].

Penelitian berikutnya dengan judul *week-wise student performance early prediction in virtual learning environment using a deep explainable artificial intelligence* [8], dalam penelitian ini, memprediksi kemungkinan kegagalan dalam perkuliahan sejak dini dari mekanisme pembelajaran online, fitur yang diambil dari pemanfaatan *virtual learning environment*, kemudian membandingkan performa prediksi dari Algoritma SVM, Linear Regresi, LSTM, Conv-LSTM, CNN-LSTM, hasil penelitian menunjukkan bahwa model berbasis LSTM (yaitu, LSTM, CNN-LSTM, dan Conv-LSTM) dapat secara efektif mengekstrak pola temporal di dalam kumpulan data VLE dengan skor F1 masing-masing 0,84, 0,88, dan 0,91 [8].

Mengacu pada kajian literatur yang telah dipaparkan, maka usulan penelitian dengan judul **“Evaluasi Model Pada Pengembangan Lingkungan Pembelajaran Berbasis Learning Analytics”** akan menjadi solusi untuk melakukan prediksi tingkat keberhasilan peserta didik untuk mengoptimalkan tingkat pencapaian akademik pada model evaluasi pembelajaran yang dikembangkan.

C. STATE OF THE ART

Pada Tabel 2.2 menunjukkan rangkaian penelitian yang berhubungan dengan *academic performance prediction* untuk mengetahui prediksi pada evaluasi pembelajaran sebagai berikut :

Tabel 2. 2. State Of The Art

No	Penulis	Judul	Permasalahan	Algoritma	Potensi Penelitian (PP)
1.	Wanli Xing, Rui Guo, Eva Petakovic, Sean Goggins, (2015)	Participation-based student final performance prediction model through interpretable Genetic Programming: Integrating learning analytics, educational data mining and theory [28]	Mensintesis pendekatan pembelajaran <i>learning analytic</i> , EDM, dan HCI teory. Secara online menggunakan <i>Collaborative learning tool</i> Mengukur secara kuantitatif partisipasi pelajar dalam kursus CSCL (Computer-supported Collaborative Learning).	Menggunakan genetic programming/G P-ICRM, Decision-tree, Logistic regression, Perceptron, Naïve Bayes untuk memprediksi model dan resiko pelajar yang bermasalah.	Tools: Virtual Math Teams with Geogebra (VMTwG) Data: Upaya pemecahan masalah, kepatuhan terhadap aturan, mahir menggunakan tools, pemanfaatan media komunikasi, kerja sama, objek yang diselesaikan. PP1: Menerapkan Platform untuk mengumpulkan data pembelajaran untuk diolah pada proses prediksi
2	Farshid Marbouti, Heidi A. Diefes-Dux, Krishna Madhavan, (2016)	Models for early prediction of at-risk students in a course using standards-based grading [21]	Mengidentifikasi pelajar yang beresiko gagal atau lulus lebih awal pada tahap perkuliahan dengan memberi peringatan dini kepada pelajar Membandingkan algoritma Klasifikasi untuk Mengetahui performa dalam model prediksi	<i>Logistic Regression, Support Vector Machine, Decision Tree, Multi layer Preceptron, Naive Bayes Classifier, dan K-Nearest Neighbor</i>	Belum menggunakan fitur pembelajaran online, dan pada tahap pengolahan data belum menggunakan dashboard yang dapat bekerja secara adaptif. PP2: Membuat Dashboard Student Performance
3	Raheela Asif, Agathe Merceron, Syed Abbas Ali, Najmi Ghani Haider, (2017)	Analyzing undergraduate students' performance using educational data mining [19]	Memprediksi prestasi akademik pelajar pada akhir program studi empat tahun. Apakah kualitasnya rendah atau tinggi Hasilnya menunjukkan bahwa dengan memfokuskan	Perbandingan algoritma (decision trees, rule induction, artificial neural networks, k-nearest neighbours, Naive Bayes, random forest trees artificial neural networks, k-nearest	Hanya menggunakan data Transkrip nilai SMA, nilai tahun 1 dan tahun ke 2, tidak menggunakan data economic keluarga dan demograph pelajar PP3: Menggunakan Data diawal perkuliahan agar mampu memprediksi performa pelajar dengan waktu yang lebih banyak

No	Penulis	Judul	Permasalahan	Algoritma	Potensi Penelitian (PP)
			pada sejumlah kecil perkuliahan yang merupakan indikator kinerja yang sangat baik atau buruk, dimungkinkan untuk memberikan peringatan dan dukungan tepat waktu kepada pelajar yang berprestasi rendah, dan saran serta peluang bagi pelajar berprestasi.	neighbours)	tanpa harus menunggu akhir masa perkuliahan
4	Evandro B. Costa, Balduino Fonseca, Marcelo Almeida Santana, Fabrícia Ferreira de Araújo, Joilson Rego, (2017)	Evaluating the effectiveness of educational data mining techniques for early prediction of students' academic failure in introductory programming courses [26]	Membanding tentang efektivitas teknik data mining untuk memprediksi lebih awal Siswa yang cenderung gagal dalam kuliah pengantar pemrograman Mengevaluasi teknik Mining pada data yang telah diproses <i>fine-tuning</i> sebelumnya dan data tanpa <i>fine-tuning</i> Menggunakan metode <i>Grid Search</i> , <i>decision tree pruning</i> , <i>updating learning rate</i> , <i>weight hidden layer</i> .	SVM, Decision Tree J48, NN, Naive Bayes	Tools: Java, WEKA tool Data: Dari 2 kampus berbeda, dan Kinerja Pada Mata kuliah Pengantar Pemrograman. PP4: Merancang <i>early warning system</i> untuk Mengoptimalkan Kinerja Pelajar diawal perkuliahan
5.	Fumiya Okubo, Takayoshi Yamashita	On the Prediction of Students' Quiz Score by	Menemukan pelajar "beresiko" yang kemungkinan	Recurrent neural network (RNN)	Tools: sistem e-learning Moodle, e-portfolio sistem Mahara, dan sistem e-book BookRoll,

No	Penulis	Judul	Permasalahan	Algoritma	Potensi Penelitian (PP)
	, Atsushi Shimada, Yuta Taniguchi and Shin'ichi Konomi, (2018)	Recurrent Neural Network [16]	besar akan gagal dalam perkuliahan. Mengeksplorasi faktor untuk meningkatkan kinerja prediksi nilai kuis pelajar dengan menggunakan RNN		Data :Log dalam LMS(Kehadiran, Kelengkapan Tugas, Aktifitas pengumpulan Tugas, Tampilan Kursus, Penanda Dalam Konten Perkuliahan, Catatan dalam Aplikasi Perkuliahan, Aktifitas dalam Aplikasi, Jumlah Kata dalam Tugas. PP5: Dapat Meningkatkan Kelemahan Algoritma RNN dalam Hal Proses Data Vektor Skala Besar Dengan Algoritma Lainnya. Dan menggunakan sumberdata lain dari sistem pembelajaran
6.	Gabriela Czibula, Andrei Mihai, Liana Maria Crivei, (2019)	S PRAR: A novel relational association rule mining classification model applied for academic performance prediction [29]	Masalah memprediksi apakah seorang pelajar akan lulus atau gagal pada disiplin akademik tertentu berdasarkan nilai pelajar yang diterima selama semester	Memodifikasi (DT,ANN, SVM) untuk memprediksi model pembelajaran yang baik untuk pelajar.	Tools: Membuat Website, dan quisioner Data: rincian mata pelajaran, rincian nilai pelajar, rincian kehadiran pelajar. PP6: Menggunakan Data Demografi Pelajar untuk memprediksi performa pelajar pada dashboard yang dirancang
7.	Libor Juhanak Jirí Zounek Lucie Rohlíková, (2019)	Using process mining to analyze students' quiz-taking behavior patterns in a learning management system [30]	mengeksplorasi perilaku pelajar dan pola interaksi dalam berbagai jenis kegiatan kuis online dalam sistem manajemen pembelajaran (LMS). Pembelajaran online yang berlangsung apakah bermanfaat terhadap penerimaan materi	Alpha Miner dan Heuristic Miner	Tools: Moodle LMS, Virtual Learning Environment Data: Log Aktivitas perilaku pelajar dari Moodle LMS, Data Kuis Pelajar. Test type, No. Of students, Time limit, Random question selection, Number Of attempts, Feedback, Quiz viewed, quiz attempt started PP7: Membuat Platform yang mampu mengumpulkan Log dari proses Pembelajaran

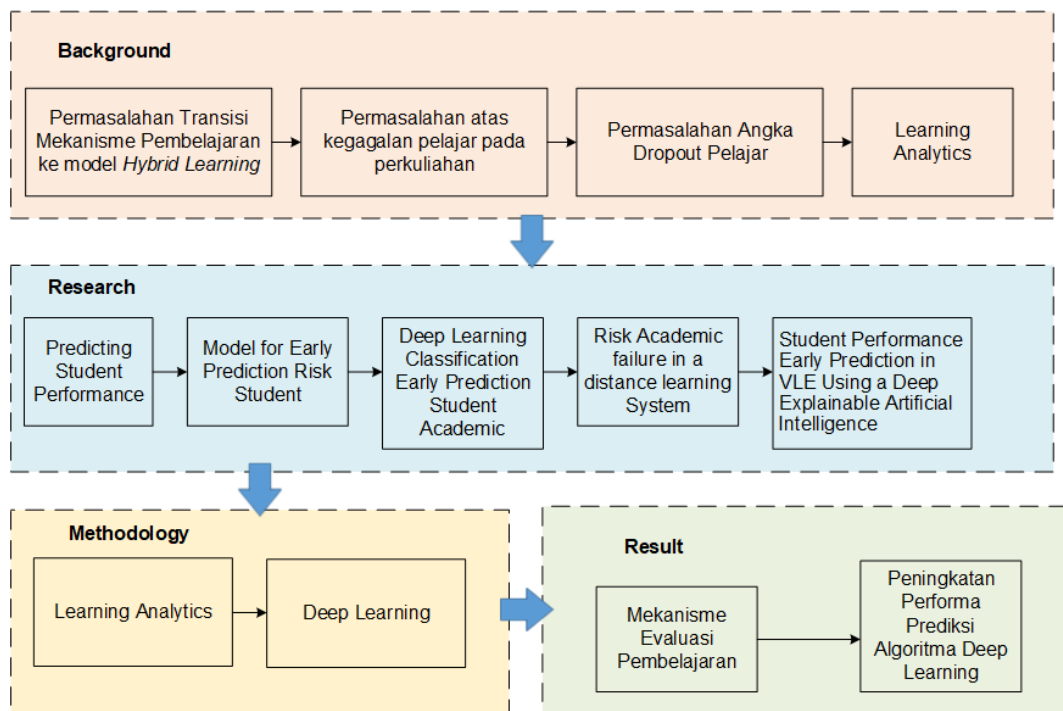
No	Penulis	Judul	Permasalahan	Algoritma	Potensi Penelitian (PP)
8.	Aderibigbe Israel Adekitan, Odunayo Salau, (2019)	The impact of engineering students' performance in the first three years on their graduation result using educational data mining [31]	Bagaimana Melakukan Pengawasan regulasi terkait kualitas pendidikan dalam sistem pendidikan, memastikan lulusan sesuai standar kelulusan tepat waktu atau terlambat, Memprediksi sejauh mana pengaruh kinerja pelajar dalam tiga tahun pertama terhadap hasil kelulusan mereka. Memberikan <i>warning</i> dan saran kepada pelajar yang terindikasi bermasalah melalui sistem	predictive models using both KNIME and Membandingkan Probabilistic Neural Network (PNN) based on the DDA (Dynamic Decay Adjustment), the Random Forest Predictor, the Decision Tree Predictor, the Naïve Bayes Predictor, the Tree Ensemble Predictor, and the Logistic Regression	Algoritma: membandingkan algoritma data mining, Tool: KNIME analytics platform, MATLAB DATA: indeks prestasi semester, dan indeks prestasi kumulatif pelajar tiap tahun PP8: Menggunakan Kombinasi data History Pembelajaran, data keterlibatan pembelajaran dalam pembelajaran dan data demografi pada proses prediksi
9	Nikola Tomasevic, Nikola Gvozdenovic, Sanja Vranes, (2020)	An overview and comparison of supervised data mining techniques for student exam performance prediction [32]	<i>Machine learning</i> yang diterapkan untuk menyelesaikan tugas prediksi kinerja ujian pelajar, yaitu menemukan pelajar pada "risiko tinggi" keluar (<i>drop out</i>) dari proses pembelajaran, dan memprediksi pencapaian mereka kedepan, seperti misalnya, final nilai ujian. Menggunakan teknik	K-Nearest Neighbors (k-NN), Support Vector Machines (SVM), Jaringan Syaraf Tiruan (JST), Pohon Keputusan, Naïve Bayes, dan Regresi Logistik.	Data: Data Demografi, Data Keterlibatan Dalam Pembelajaran PP9: Menemukan Fitur terpenting pada Kumpulan Fitur yang digunakan dalam prediksi

No	Penulis	Judul	Permasalahan	Algoritma	Potensi Penelitian (PP)
			<i>Classification dan regression.</i>		
10	F. Giannakas, C. Troussas, I. Voyiatzis, C. Sgouropoulou, (2021)	<i>A deep learning classification framework for early prediction of team-based academic performance</i> [27]	Menguji prosedur sistematis untuk mengimplementasikan jaringan saraf tiruan dalam memprediksi prestasi akademik pada model pembelajaran kelompok.	Memaksimalkan Kinerja DNN dengan algoritma optimasi <i>Adadelata</i> dan <i>Adagrad</i> dengan akurasi sebesar 76,73% dan 82,39%, sedangkan prestasi belajarnya secara keseluruhan adalah 80,76% dan 86,57%, artinya mampu memprediksi kinerja tim secara memadai dan akurat.	Data: Data Pembelajaran kelompok pelajar PP10: Membandingkan algoritma DNN yang diusulkan dengan teknik pembelajaran mesin lainnya, dan lebih jauh mengeksplorasi prediksi algoritma untuk membantu pembentukan kelompok, serta menyediakan aktivitas kolaboratif yang ditargetkan kepada tim untuk memaksimalkan hasil pembelajaran.
11	Carlos Felipe Rodríguez-Hernandez, Mariel Musso, Eva Kyndt, Eduardo Cascallar, (2021)	Artificial neural networks in academic performance prediction: Systematic implementation and predictor evaluation [14]	Menganalisis kepentingan prestasi akademik sebelumnya, status sosial ekonomi, karakteristik sekolah menengah, dan status kerja ketika memprediksi.	Mengupdate <i>Hyperparameter</i> pada JST untuk melihat kinerja model prediksi BayesNet, Decision tree, KNN, Logistic Regresion, Naive Bayes, NN, Random Forest, random tree.	Data: Prior academic achievement, Tuition fees, Students' socioeconomic status, Students' home characteristics, Working Status, Students' background information, High school characteristics, Academic performance in higher education. PP11: Eksperimen dengan <i>Hyperparameter</i> pada Algoritma yang dioptimalkan untuk menambah akurasi prediksi Kontribusi yang diharapkan adalah mengidentifikasi indikator-indikator terbaik, yang berkinerja rendah, yang dapat memberikan informasi yang berguna sebagai sistem peringatan dini di bidang pendidikan.
12	Halit Karalar, Ceyhun Kapucu,	Predicting students at risk of academic	Memprediksi siswa yang beresiko gagal	Gradient Boosting (GB), Quadratic	Data : Data Pembelajaran <i>synchronous</i> dan

No	Penulis	Judul	Permasalahan	Algoritma	Potensi Penelitian (PP)
	Hüseyin Gürüler, (2021)	failure using ensemble model during pandemic in a distance learning system [12]	dalam masa transisi perkuliahan jarak jauh di masa pandemi. Mengambil aktivitas log pembelajaran <i>synchronous</i> dan <i>asynchronous</i> dan memprediksi dengan kombinasi algoritma <i>Machine learning</i> dan teknik <i>Ansemble learning</i>	Discriminant Analysis (QDA), Extra Trees (ET), Decision Tree (DT), Random Forest (RF), Logistic Regression (LR), and Artificial Neural Network (ANN)	<i>asynchronous</i> (Jenis-Kelamin, Jumlah Catatan Perkuliahan, Waktu Sesi Online, Materi Kursus Yang Di Unduh, Jumlah Kehadiran, Dan Nilai Kuis.) PP12: dimungkinkan untuk menguji ensemble learning yang dibuat dengan kombinasi algoritma yang berbeda menggunakan lebih banyak data karakteristik siswa, dan aktivitas pembelajaran <i>synchronous</i> & <i>asynchronous</i>.
13	Hsing-Chung Chen, Eko Prasetyo, Shian-Shyong Tseng, Karisma Trinanda Putra, Prayitno, Sri Suning Kusumawardani, Chien-Erh Weng, (2022)	Week-Wise Student Performance Early Prediction in Virtual Learning Environment Using a Deep Explainable Artificial Intelligence [8]	Mengusulkan model ESPP untuk Memprediksi kinerja pembelajaran Online dalam masa Pandemi Covid-19, untuk mengetahui tingkat kegagalan pelajar pada masa perkuliahan.	Mengalisa fitur terpenting dalam pembelajaran online yang mempengaruhi tingkat keberhasilan siswa. Kombinasi convolutional neural network (CNN) dan long short-term memory (LSTM) digunakan untuk mengekstrak fitur spatiotemporal dan mengembangkan model <i>deep learning</i> (DL) prediksi awal perkuliahan.	Algoritma: LSTM, CNN-LSTM, Conv-LSTM, SVM, Linear Regresi. Input: Data Pembelajaran Online: Assignment, File, Forum, homepage, label, page, quis, URL. PP13: menambah fitur intervensi pengajar pada proses prediksi Mengoptimalkan performa dengan dengan eksplorasi jumlah gate dan jumlah fungsi aktivasi.

B. KERANGKA PIKIR

Kerangka pikir pada penelitian ini terdiri dari latar belakang, pendekatan penelitian yang ada, methodology dan hasil yang diharapkan, yang menunjukkan alur pemikiran penulis pada penelitian yang akan dilaksanakan.



Gambar 2. 14. Kerangka Pikir Penelitian

1. masalah yang terjadi dimulai dari mengidentifikasi pelajar yang beresiko "gagal" atau "lulus" pada tahap perkuliahan sejak dini, dengan memberi peringatan kepada pelajar, mengontrol kemampuan belajar mahasiswa dan menyesuaikan model belajar dengan profile mahasiswa, mengurangi angka *drop out* pelajar, dikarenakan tingginya angka mahasiswa yang menyelesaikan masa studi tidak tepat waktu.

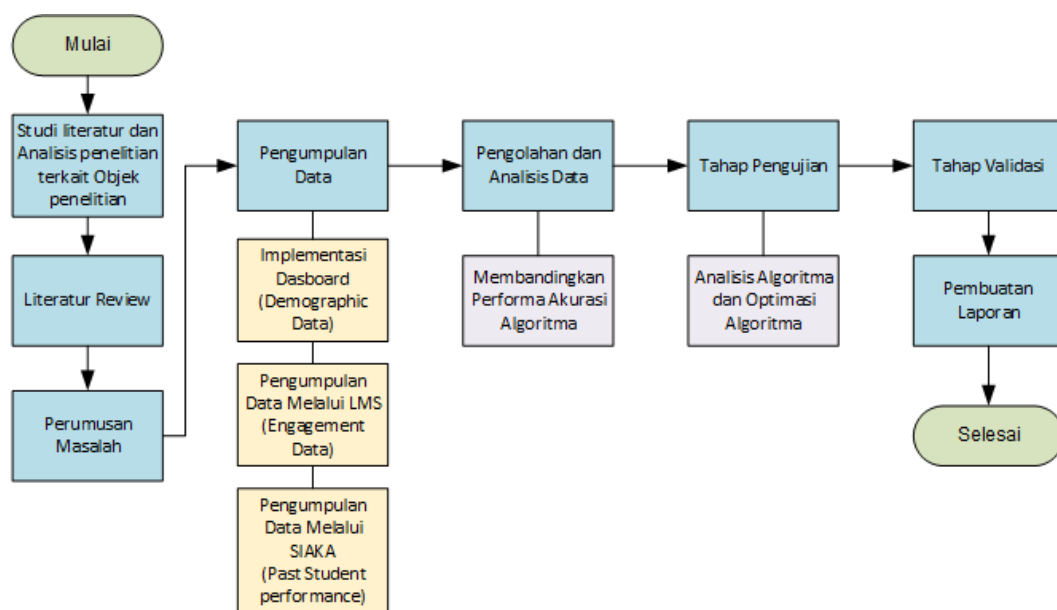
2. Merancang dan membuat mekanisme Evaluasi untuk pengolahan data *predicting student performance*, melalui aplikasi *student dashboards* mengumpulkan data awal, kemudian menganalisa hasil proses belajar mengajar mahasiswa. Merancang mekanisme evaluasi untuk menyesuaikan kinerja proses belajar mengajar sehari-hari, dengan memperhatikan data demografi, data keterlibatan dalam perkuliahan, dan data prestasi masa lalu pelajar.
3. Menggunakan pendekatan *learning analytic* untuk proses prediksi. Mengoptimalkan kinerja algoritma *deep learning* serta mengoptimalkan proses *prediksi* untuk mendapatkan akurasi prediksi terbaik dari perbandingan beberapa algoritma *deep learning* serta meningkatkan akurasi dengan *ensemble method* dan *algoritma optimasi*.
4. Menemukan Mekanisme evaluasi pembelajaran dan algoritma prediksi dengan akurasi yang tinggi dalam proses klasifikasi dan prediksi, yang menjadi pilihan dalam melakukan proses evaluasi akademik mahasiswa dalam meningkatkan sdm mahasiswa, sehingga dengan adanya mekanisme evaluasi terbaru untuk *academic prediction performance* ini maka diharapkan dapat menjadi pengambil keputusan (sistem pendukung keputusan) dalam mengambil kebijakan tentang skema pembelajaran guna meningkatkan kualitas belajar mengajar bagi mahasiswa dan pengajar.

BAB III

METODE PENELITIAN

A. TAHAPAN PENELITIAN

Untuk menghasilkan luaran penelitian, maka perlu dilakukan perencanaan dalam bentuk tahapan penelitian, sebagai mana ditunjukkan pada gambar 3.1. sebagai berikut:

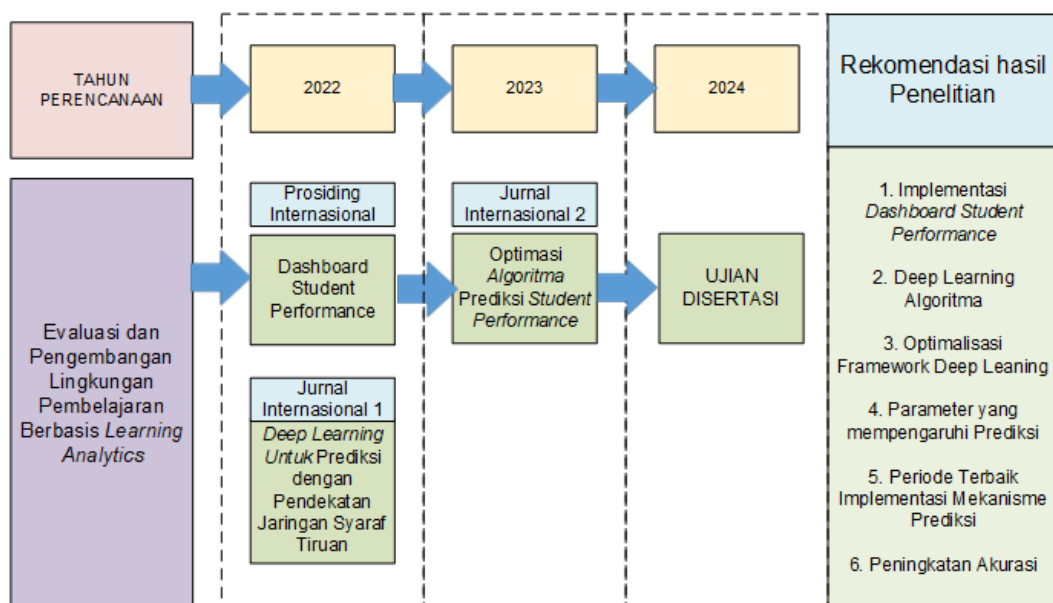


Gambar 3. 1. Tahapan Penelitian

Adapun pada tahap penelitian pertama dilakukan studi literatur terkait topik *academic performance prediction*, setelah itu dilakukan pemetaan terkait paper yang didapatkan untuk mencari rekomendasi hasil penelitian, setelah itu dilakukan perumusan masalah terkait rencana penelitian. Pada tahap pengumpulan data akan dikumpulkan data terkait demografi pelajar dari dashboard yang dirancang, untuk melengkapi fitur

yang akan diolah juga diambil data dari aktifitas pada lms, dan untuk menunjang kekurangan data untuk fitur yang diolah maka diambil beberapa fitur dari sistem informasi akademik perguruan tinggi. Data yang terkumpul kemudian diolah untuk mencari mekanisme prediksi, dengan memanfaatkan *ensemble method* untuk mencari akurasi terbaik dari perbandingan beberapa algoritma jaringan syaraf tiruan, kemudian setelah didapatkan algoritma terbaik, maka di lakukan optimasi dengan memanfaatkan algoritma optimasi terkait algoritma yang didapatkan. Pada tahap pengujian, setelah model terbentuk maka akan dilakukan mekanisme validasi terhadap pelajar yang sementara melakukan proses perkuliahan sampai pada masa semester berjalan berakhir, untuk mengetahui performa algoritma yang di dapatkan.

B. ROADMAP PENELITIAN



Gambar 3. 2. Roadmap Penelitian

Pada roadmap penelitian digambarkan rencana publikasi yang akan dilakukan terkait topik yang akan di angkat, pada tahun pertama akan dilakukan publikasi terkait hasil implementasi *dashboard student performance* yang dijadikan sistem untuk mengintegrasikan parameter yang akan di analisis, ditahun yang sama juga akan direncanakan menerbitkan jurnal internasional dengan topik *deep learning prediction* dengan pendekatan jaringan syaraf tiruan, kemudian pada tahun kedua akan diterbitkan jurnal internasional mengenai *optimasi algoritma prediction for academic performance prediction*.

C. WAKTU DAN LOKASI PENELITIAN

Penelitian ini akan dilaksanakan dengan durasi waktu selama 24 bulan yang dimulai pada bulan februari 2022 sampai februari 2024, penelitian ini dilaksanakan pada laboratorium *cloud computing and system information*, departemen teknik informatika, fakultas teknik, universitas hasanuddin. Pengambilan data dilakukan pada kampus stmik handayani, stmik akba, stmik profesional, dan universitas dipa makassar.

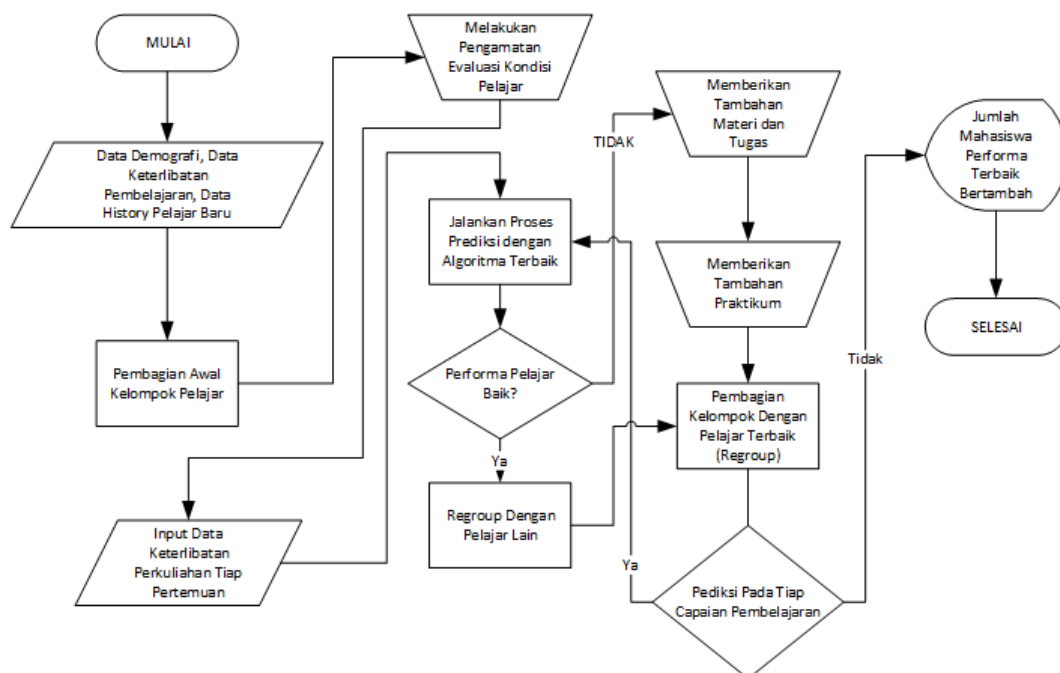
D. JENIS PENELITIAN

Jenis penelitian ini merupakan penelitian eksperimental, yang pada tahap perumusan masalah dilakukan dengan metode studi pustaka, pengambilan data dengan penggunaan aplikasi, kemudian dilakukan tahap pengolahan dan analisis data, untuk mengimplementasikan mekanisme

prediksi, yang kemudian dilakukan pengujian dan validasi dari model yang didapatkan.

E. PERANCANGAN MEKANISME PREDIKSI

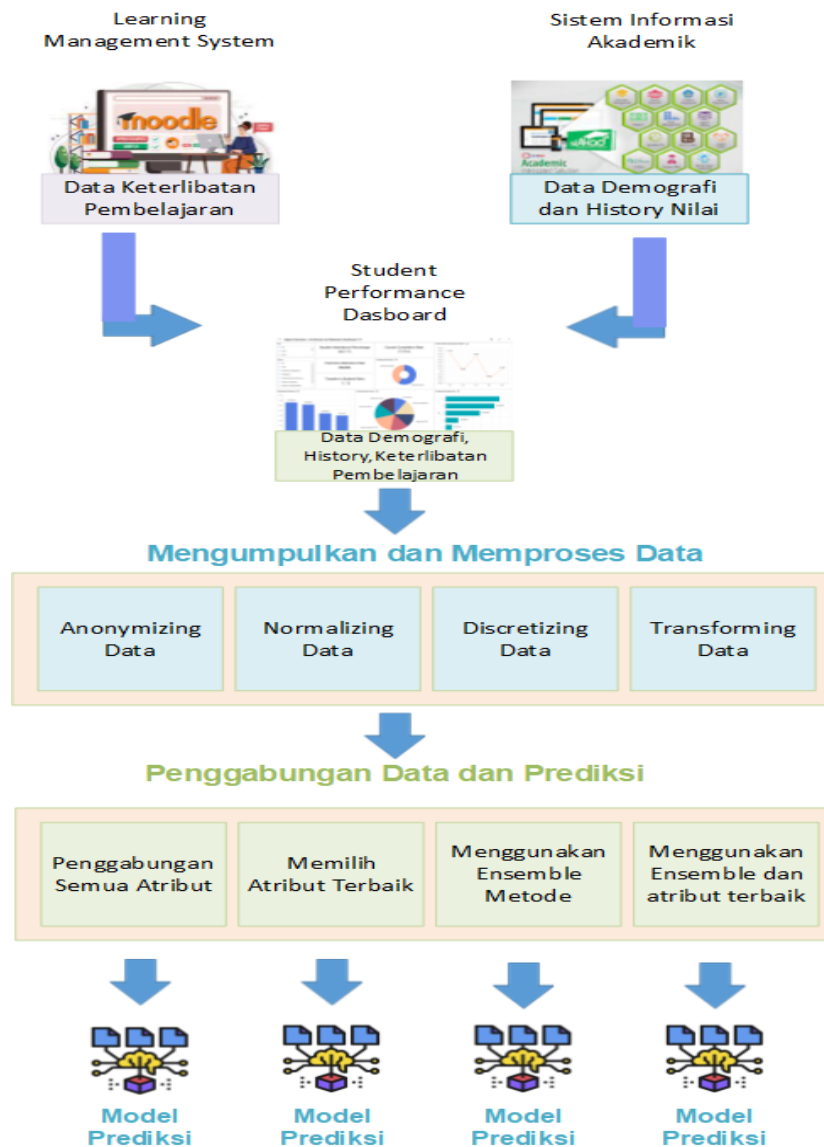
Mekanisme evaluasi yang diusulkan ini secara bersamaan memprediksi kinerja siswa dan menghasilkan interpretasi dari sisi pelajar dari sudut pandang pembelajaran *hybrid*. Teknik interpretabilitas ini berguna untuk menjelaskan hasil yang telah didapatkan dari algoritma prediksi. Di sisi lain, dari sudut pandang sistem VLE, teknik interpretabilitas menjadi bahan evaluasi umum untuk memperbaiki fitur-fitur yang tidak berperan dalam menyeimbangkan proses pembelajaran, sehingga sistem pembelajaran dapat direvisi dan ditingkatkan untuk pembelajaran pada tahun akademik berikutnya.



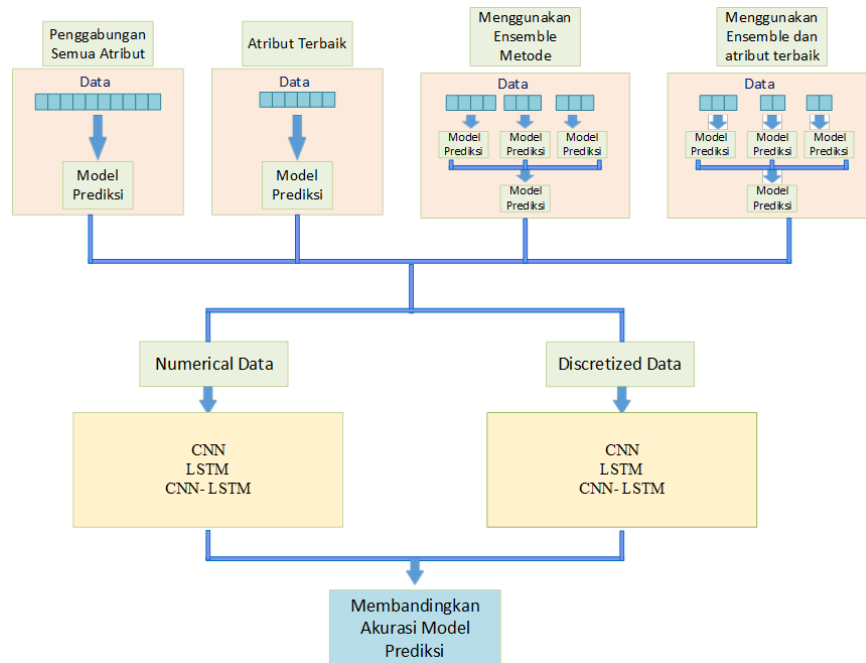
Gambar 3. 3. Flowchart Sistem Yang Diusulkan

Pada Tahap awal untuk meningkatkan performa pelajar dengan mengeksplorasi proses pembelajaran, dengan mekanisme penambahan tugas juga dapat dilakukan peningkatan minat belajar dengan cara melakukan pembagian kelompok pelajar (*team-based*) dengan maksud bahwa peran pelajar yang minat belajarnya rendah akan digabungkan dengan pelajar yang mempunyai performa baik ke dalam kelompok, setiap periode yang ditentukan pada akhir capaian pembelajaran, dilakukan evaluasi untuk melihat kembali performa untuk masing-masing pelajar, kemudian dilakukan perubahan anggota kelompok ketika masih ditemukan performa pelajar yang kurang, proses lainnya juga dilakukan dengan memasukkan pelajar yang memiliki performa rendah ke dalam kelompok [27], (*multi-group*) sehingga pelajar mampu bekerja dari beberapa sumber kelompok, dengan mekanisme tersebut maka diharapkan hasilnya mampu meningkatkan kualitas pembelajaran, dan minat belajar dari siswa yang kurang menjadi semakin baik performanya, dilakukan perancangan sistem yang akan digunakan untuk mengumpulkan data demografi pelajar, kemudian dilakukan kombinasi data dari dalam LMS terkait data aktifitas pelajar, serta dilakukan pengumpulan data dari SIAKAD, dan dari *platform* yang akan dirancang untuk mengambil *log* pembelajaran dari interpersi pengajar, terkait fitur yang berkaitan dengan durasi penjelasan materi pengajar. Fitur yang ada kaitannya dengan durasi log pembelajaran *hybrid* dalam lingkungan pembelajaran baik dari sisi pelajar maupun sisi pengajar,

juga digunakan untuk menambah fitur yang akan dianalisis. Sebagaimana ditunjukkan pada gambar 3.4 berikut:



Gambar 3. 4. Rencana Mekanisme Prediksi



Gambar 3. 5. Skema Tahapan Eksperimen

Pada gambar 3.5. rencana eksperimen untuk mendapatkan algoritma prediksi terbaik, dilakukan dengan mengumpulkan seluruh fitur atau atribut dari sumber data, baik data aktivitas perkuliahan, *history* nilai, dan dari kondisi demografi pelajar melalui *platform dashboard* yang dirancang. Pada skenario eksperimen berikutnya dilakukan prediksi menggunakan data dari atribut terbaik saja, terkait data yang memiliki *record* lengkap pada *field* yang akan di *training*. Skenario berikutnya memilih data *training* dari algoritma terbaik yang didapatkan, hasil dari kedua tahapan eksperimen sebelumnya, kemudian masuk ke tahap pre-processing, AI modeling, dan performance analysis, dengan melakukan prediksi menggunakan algoritma dari atribut terbaik dan algoritma yang didapatkan dari *ensemble method*, dari hasil prediksi maka didapatkan

algoritma terbaik, dimana algoritma yang digunakan diantaranya CNN, LSTM, dan CNN-LSTM. Kemudian dilanjutkan pada tahap optimasi performa algoritma, dengan mengkombinasikan algoritma terbaik yang didapatkan dari proses *analysis performance*, dengan algoritma optimasi yang direncanakan sebelumnya, kemudian dilakukan proses pengujian dengan menggunakan *cross validation* dan dikombinasikan dengan *confusion matrix*.

F. SUMBER DATA

Sumber data yang akan digunakan pada penelitian ini di klasifikasikan menjadi, diantaranya:

Tabel 3. 1. Tabel Fitur Data

Data Demografi	Data Keterlibatan Proses Pembelajaran	Data History
1. Jumlah Mata kuliah yang di program [14] 2. Jenis kelamin [14] 3. Usia [14] 4. Lokasi tempat tinggal [14] 5. Laptop/PC/smartphone [14] 6. Pendapatan keluarga perbulan [14] 7. Beasiswa [14] 8. Pendidikan ayah [14] 9. Pendidikan ibu [14] 10. Pekerjaan ayah [14] 11. Pekerjaan ibu [14] 12. Jenis SMA [14] 13. Tanggungan keluarga 14. Bisnis keluarga [14] 15. Gelar akademik ayah [14] 16. Gelar akademik ibu [14]	31. Jumlah Akses Materi [27] 32. Lama Akses Materi [27] 33. Kesamaan Tugas 34. Ketepatan Waktu Upload tugas [27] 35. Jumlah Tugas Yang dikumpul [27] 36. Keaktifan dalam Forum [27] 37. Jumlah Matakuliah yang diikuti [27] 38. Catatan Pada LMS [27] 39. Total akses LMS [27] 40. Tugas kelompok [27] 41. durasi online pada pembelajaran daring	47. Kehadiran [12] 48. Nilai Pekerjaan Rumah [12] 49. Nilai Tugas [12] 50. Nilai MID [12] 51. Nilai Final [12] 52. Nilai Mata pelajaran SMA [33]

Data Demografi	Data Keterlibatan Proses Pembelajaran	Data History
17. Jenis pekerjaan mahasiswa [14] 18. Kerja mingguan [14] 19. Penghasilan mahasiswa [14] 20. Kegiatan lain orang tua 21. Bisnis orang tua [14] 22. Kendaraan operasional 23. Jenis hunian [14] 24. Kondisi hunian [14] 25. Akses internet di rumah [14] 26. Besar bandwidth [14] 27. Total Waktu Online [14] 28. Total Waktu Offline [14] 29. Akses Konten Video [14] 30. Social Media [14]	42. jumlah off kamera dan durasi online pembelajaran daring 43. durasi penjelasan pengajar setiap materi pembelajaran daring 44. log off dan on kamera pelajar 45. log off dan on kamera pengajar pada saat pembelajaran online 46. durasi bimbingan/ pembelajaran online dan offline	

G. INSTRUMEN PENELITIAN

Terdapat dua bagian instrumen yang digunakan pada penelitian ini, sebagai berikut :

1. Perangkat Keras (*Hardware*)
 - a. Processor Intel Core i7
 - b. RAM 8 GB
 - c. HDD Storage 1 TB
2. Perangkat Lunak (*Software*)
 - a. OS Windows 8
 - b. MatLab R2021a
 - c. Rapid Miner Studio
 - d. Python – 3.7.1

- e. XAMP
- f. Visual Studio

H. JADWAL PENELITIAN

Tabel 3. 2.Tabel Jadwal Penelitian

No	Tahapan dan Kegiatan	Semester						Target Publikasi
		1	2	3	4	5	6	
1	Persiapan dan Studi Literatur							International Conference, Submit Journal International.
2	Pra Penelitian dan Seminar Proposal							
3	Penelitian 1 Publikasi Karya Ilmiah 1							
4	Penelitian 2 Publikasi Karya Ilmiah 2							
5	Penelitian 3 Publikasi Karya Ilmiah 3							
6	Penulisan Laporan Disertasi							
7	Ujian Tutup							Submit Journal International.
8	Ujian Terbuka							

DAFTAR PUSTAKA

- [1] P. A. N. Prestes, T. E. V. Silva, and G. C. Barroso, "Correlation analysis using teaching and learning analytics," *Heliyon*, vol. 7, no. 11, p. e08435, 2021, doi: 10.1016/j.heliyon.2021.e08435.
- [2] P. Kostagiolas, C. Lavranos, and N. Korfiatis, "Learning analytics: Survey data for measuring the impact of study satisfaction on students' academic self-efficacy and performance," *Data Br.*, vol. 25, pp. 0–6, 2019, doi: 10.1016/j.dib.2019.104051.
- [3] D. J. Lemay, C. Baek, and T. Doleck, "Computers and Education : Artificial Intelligence Comparison of learning analytics and educational data mining : A topic modeling approach," *Comput. Educ. Artif. Intell.*, vol. 2, no. March, p. 100016, 2021, doi: 10.1016/j.caeai.2021.100016.
- [4] C. R. Henrie, R. Bodily, R. Larsen, and C. R. Graham, "Exploring the potential of LMS log data as a proxy measure of student engagement," *J. Comput. High. Educ.*, 2017, doi: 10.1007/s12528-017-9161-1.
- [5] C. M. Stracke, "Quality frameworks and learning design for open education," *Int. Rev. Res. Open Distance Learn.*, vol. 20, no. 2, pp. 180–203, 2019, doi: 10.19173/irrodl.v20i2.4213.
- [6] W. Chango and R. Cerezo, "Multi-source and multimodal data fusion for predicting academic performance in blended learning university courses," vol. 89, no. October 2020, 2021, doi: 10.1016/j.compeleceng.2020.106908.
- [7] X. Bai *et al.*, "Educational Big Data : Predictions , Applications and Challenges," *Big Data Res.*, vol. 26, p. 100270, 2021, doi: 10.1016/j.bdr.2021.100270.
- [8] H. C. Chen *et al.*, "Week-Wise Student Performance Early Prediction in Virtual Learning Environment Using a Deep Explainable Artificial Intelligence," *Appl. Sci.*, vol. 12, no. 4, pp. 1–16, 2022, doi: 10.3390/app12041885.
- [9] R. Conijn, C. Snijders, A. Kleingeld, and U. Matzat, "Predicting student performance from LMS data: A comparison of 17 blended courses using moodle LMS," *IEEE Trans. Learn. Technol.*, vol. 10, no. 1, pp. 17–29, 2017, doi: 10.1109/TLT.2016.2616312.
- [10] F. Okubo, T. Yamashita, A. Shimada, and S. Konomi, "Students' performance prediction using data of multiple courses by recurrent neural network," in *Proceedings of the 25th International Conference on Computers in Education, ICCE 2017 - Main Conference*

Proceedings, 2017, pp. 439–444.

- [11] J. W. You, “Internet and Higher Education Identifying significant indicators using LMS data to predict course achievement in online learning,” *Internet High. Educ.*, vol. 29, pp. 23–30, 2016, doi: 10.1016/j.iheduc.2015.11.003.
- [12] H. Karalar, C. Kapucu, and H. Gürüler, “Predicting students at risk of academic failure using ensemble model during pandemic in a distance learning system,” *Int. J. Educ. Technol. High. Educ.*, 2021, doi: 10.1186/s41239-021-00300-y.
- [13] P. Paule-ruíz, F. Ortin, and M. Riestra-gonz, “Computers & Education Massive LMS log data analysis for the early prediction of course-agnostic student performance,” vol. 163, no. July 2020, 2021, doi: 10.1016/j.compedu.2020.104108.
- [14] C. F. Rodríguez-Hernández, M. Musso, E. Kyndt, and E. Cascallar, “Artificial neural networks in academic performance prediction: Systematic implementation and predictor evaluation,” *Comput. Educ. Artif. Intell.*, vol. 2, no. March, 2021, doi: 10.1016/j.caeai.2021.100018.
- [15] T. Devasia, T. P. Vinushree, and V. Hegde, “Prediction of students performance using Educational Data Mining,” *Proc. 2016 Int. Conf. Data Min. Adv. Comput. SAPIENCE 2016*, pp. 91–95, 2016, doi: 10.1109/SAPIENCE.2016.7684167.
- [16] F. Okubo, T. Yamashita, A. Shimada, Y. Taniguchi, and K. Shin’ichi, “On the prediction of students’ quiz score by recurrent neural network,” in *CEUR Workshop Proceedings*, 2018, vol. 2163, pp. 1–6.
- [17] C. C. Gray and D. Perkins, “Computers & Education Utilizing early engagement and machine learning to predict student outcomes,” *Comput. Educ.*, vol. 131, no. December 2018, pp. 22–32, 2019, doi: 10.1016/j.compedu.2018.12.006.
- [18] M. F. Musso, E. C. Cascallar, N. Bostani, and M. Crawford, “Identifying Reliable Predictors of Educational Outcomes Through Machine-Learning Predictive Modeling,” no. July, 2020, doi: 10.3389/feduc.2020.00104.
- [19] R. Asif, A. Merceron, S. A. Ali, and N. G. Haider, “Analyzing undergraduate students’ performance using educational data mining,” *Comput. Educ.*, vol. 113, pp. 177–194, 2017, doi: 10.1016/j.compedu.2017.05.007.
- [20] A. Jokhan, B. Sharma, and S. Singh, “Studies in Higher Education Early warning system as a predictor for student performance in higher education blended courses,” *Stud. High. Educ.*, vol. 0, no. 0, pp. 1–

12, 2018, doi: 10.1080/03075079.2018.1466872.

- [21] F. Marbouti, H. A. Diefes-Dux, and K. Madhavan, "Models for early prediction of at-risk students in a course using standards-based grading," *Comput. Educ.*, vol. 103, pp. 1–15, 2016, doi: 10.1016/j.compedu.2016.09.005.
- [22] W. Xing, R. Guo, E. Petakovic, and S. Goggins, "Participation-based student final performance prediction model through interpretable Genetic Programming: Integrating learning analytics, educational data mining and theory," *Comput. Human Behav.*, vol. 47, pp. 168–181, 2015, doi: 10.1016/j.chb.2014.09.034.
- [23] P. L. and G. Siemens, "Penetrating the Fog: Analytics in Learning and Education," *Educ. Rev.*, pp. 31–40, 2011, doi: 10.1177/0894318416647779.
- [24] J. Wira and G. Putra, "Pengenalan Konsep Pembelajaran Mesin dan Deep Learning," vol. 4, 2020.
- [25] Christopher Olah, "Understanding LSTM Networks," *August 27, 2015*, 2015. [Online]. Available: <https://colah.github.io/posts/2015-08-Understanding-LSTMs/>.
- [26] E. B. Costa, B. Fonseca, M. Almeida, F. Ferreira, D. Araújo, and J. Rego, "Computers in Human Behavior Evaluating the effectiveness of educational data mining techniques for early prediction of students' academic failure in introductory programming courses," *Comput. Human Behav.*, vol. 73, pp. 247–256, 2017, doi: 10.1016/j.chb.2017.01.047.
- [27] F. Giannakas, C. Troussas, I. Voyiatzis, and C. Sgouropoulou, "A deep learning classification framework for early prediction of team-based academic performance," *Appl. Soft Comput.*, vol. 106, p. 107355, 2021, doi: 10.1016/j.asoc.2021.107355.
- [28] N. Erdogan and C. Stuessy, "Examining the role of inclusive stem schools in the college and career readiness of students in the United States: A multi-group analysis on the outcome of student achievement," *Kuram ve Uygulamada Egit. Bilim.*, vol. 15, no. 6, pp. 1517–1529, 2015, doi: 10.12738/estp.2016.1.0072.
- [29] A. Mihai and L. Maria, "ScienceDirect S PRAR PRAR : : A A novel novel relational relational association association rule rule mining mining classification classification model model applied applied for for academic academic performance performance prediction prediction," *Procedia Comput. Sci.*, vol. 159, pp. 20–29, 2019, doi: 10.1016/j.procs.2019.09.156.
- [30] L. Juhaňák, J. Zounek, and L. Rohlíková, "Using process mining to

analyze students' quiz-taking behavior patterns in a learning management system," *Comput. Human Behav.*, vol. 92, pp. 496–506, 2019, doi: 10.1016/j.chb.2017.12.015.

- [31] A. I. Adekitan and O. Salau, "The impact of engineering students' performance in the first three years on their graduation result using educational data mining," *Heliyon*, vol. 5, no. 2, p. e01250, 2019, doi: 10.1016/j.heliyon.2019.e01250.
- [32] N. Tomasevic, N. Gvozdenovic, and S. Vranes, "An overview and comparison of supervised data mining techniques for student exam performance prediction," *Comput. Educ.*, vol. 143, no. August 2019, p. 103676, 2020, doi: 10.1016/j.compedu.2019.103676.
- [33] J. Bravo-Agapito, S. J. Romero, and S. Pamplona, "Early prediction of undergraduate Student's academic performance in completely online learning: A five-year study," *Comput. Human Behav.*, vol. 115, p. 106595, 2021, doi: 10.1016/j.chb.2020.106595.