



BAB II

TINJAUAN PUSTAKA

BAB II

TINJAUAN PUSTAKA

II.1 Prediksi Kelulusan Mahasiswa

Sebagai salah satu ikhtiar untuk memelihara dan mengembangkan kepribadian manusia baik secara spiritual maupun jasmani, pendidikan merupakan proses yang dapat didefinisikan sebagai proses untuk mengubah sikap dan perilaku seseorang atau kelompok melalui pengajaran dan pelatihan (Fitri, 2021). Sebagai salah satu pilar penting di dalam pendidikan di Indonesia, perguruan tinggi merupakan institusi yang memberikan layanan pembelajaran kepada mahasiswa, khususnya dalam hal penguasaan keilmuan tingkat tinggi (Sihite & Saleh, 2019).

Mahasiswa, sebagai salah satu komponen penting di dalam sebuah perguruan tinggi, yang menimba ilmu pada perguruan tinggi dengan harapan dapat meningkatkan tingkat intelektualitas, kecerdasan dalam berfikir, dan memiliki perencanaan dalam bertindak, sehingga dapat memposisikan diri secara harmonis di lingkungan pendidikan atau masyarakat yang lebih luas dan kompleks (Hulukati & Djibrani, 2018). Dalam proses menimba ilmu di perguruan tinggi, ketepatan waktu kelulusan mahasiswa merupakan indikator penilaian terhadap kualitas pendidikan di sebuah perguruan tinggi, dimana perbandingan antara jumlah pendaftar yang lulus seleksi di sebuah program studi haruslah seimbang dengan jumlah mahasiswa yang lulus tepat waktu di satu tahun ajaran (Agwil et al., 2020). Ada beberapa faktor yang mempengaruhi kelulusan mahasiswa tepat

waktu, seperti nilai Indeks Prestasi Kumulatif (IPK), jumlah Satuan Kredit Semester (SKS), status pekerjaan mahasiswa, status sipil mahasiswa, waktu kuliah, dan status keaktifan mahasiswa tersebut (Suriana et al., 2017).

Melihat pentingnya tingkat kelulusan mahasiswa tepat waktu di perguruan tinggi, masalah peramalan atau bagaimana memprediksi mahasiswa yang lulus tepat waktu merupakan topik penelitian yang populer, khususnya di bidang *machine learning*, seperti yang ditunjukkan pada prediksi kelulusan mahasiswa Universitas Pendidikan Ganesha (Bendesa Subawa, 2019), prediksi kelulusan mahasiswa program studi Informatika di Universitas Prof. Dr. Hamka (Anugrah Putra & Kamayani, 2020), dan prediksi kelulusan mahasiswa fakultas Teknik di Universitas Jabal Gafur (Yuswardi et al., 2022). Dari penelitian-penelitian ini, terlihat jelas bahwa penelitian mengenai prediksi kelulusan mahasiswa merupakan masalah yang populer untuk diteliti, walaupun sudah banyak penelitian-penelitian terdahulu yang melakukannya.

II.2 Artificial Neural Network (Jaringan Saraf Tiruan)

Artificial Neural Network (Jaringan Syaraf Tiruan) adalah model non-linear yang kompleks, dibangun dari komponen yang secara individu berperilaku mirip dengan model regresi. Jaringan syaraf tiruan dapat divisualisasikan sebagai grafik, dan beberapa sub-grafik mungkin ada perilaku yang sama dengan gerbang logika. Meskipun struktur dari jaringan saraf secara eksplisit dirancang terlebih dahulu, pengolahan bahwa jaringan tidak untuk menghasilkan hipotesis (berbagai gerbang logika dan pengolahan lainnya terstruktur dalam jaringan) berkembang selama

proses pembelajaran. Hal ini memungkinkan *neuron* yang membentuk jaringan akan digunakan sebagai pemecahan masalah dari "program itu sendiri".(Amin, 2018).

Menurut (Wanto, 2018) Jaringan Saraf Tiruan merupakan approximator universal fungsi kompleks, yang dapat menangkap hubungan samar antara SNP (Single Nucleotide Polymorphisms) dan nilai fenotipik tanpa memerlukan definisi model genetik secara eksplisit.

Menurut (Hasanati & Dwiny Meidelfi, 2020) Jaringan Saraf Tiruan atau JST merupakan sistem yang memproses informasi yang punya ciri serta karakteristik mirip dengan jaringan saraf yang ada pada tubuh manusia. Pembuatan struktur jaringan biologi yang diilhami dari struktur jaringan saraf tiruan, manusia memiliki kemampuan yang otak sangat luas biasa khususnya yang kompleks serta berstruktur.

Neuron-neuron adalah bagian penghubung otak yang disebut sinapsis. *Neuron* bekerja berdasarkan impuls/sinyal yang diberikan pada *neuron*, *neuron* meneruskannya pada *neuron* lain. Diperkirakan manusia memiliki 10^{13} *neuron* dan 6.10^{18} sinapsis. Begitu banyak kemampuan otak dalam melakukan dan mengenali pola organ-organ tubuh serta dari komputer digital perhitungan kecepatannya lebih tinggi (Satria, 2021).

Metode Jaringan Saraf Tiruan (JST) adalah pemrosesan informasi yang terinspirasi oleh sistem saraf Biologi. JST mampu mengenali kegiatan dengan berbasis data pada masa lalu. Data masa lalu akan dipelajari oleh JST sehingga mempunyai kemampuan untuk memberi keputusan terhadap data yang belum pernah dipelajari (Hasanah & Permatasari, 2020).

Model jaringan saraf tiruan merupakan model yang dianggap baik dalam melakukan peramalan. JST merupakan model yang dapat menjelaskan masalah-masalah peramalan yang kompleks, salah satu metode dari jaringan saraf tiruan yang paling populer dan masih digunakan hingga saat ini adalah Metode Backpropagation (Setialaksana et al., 2020). Model Jaringan Syaraf Tiruan terbaik hanya dapat dihasilkan dengan cara *trial-error*, yakni mengubah nilai-nilai setiap parameter secara acak hingga dihasilkan model terbaik dengan akurasi tertinggi (Syamsiah, 2020).

II.3 Konsep Dasar Jaringan Saraf Tiruan

Kerangka kerja Jaringan Syaraf Tiruan bisa dilihat dari jumlah lapisan (*layer*) dan jumlah *Neuron*. *Neuron-neuron* tersebut terkumpul di dalam lapisan-lapisan yang disebut *neuron layers*. Lapisan-lapisan penyusun Jaringan Syaraf Tiruan tersebut dapat dibagi menjadi 3, yaitu (Zola et al., 2018):

1. Lapisan *Input*, node-node di dalam lapisan *input* disebut unit-unit. Unit-unit *input* menerima *input* dari dunia luar. *Input* yang dimasukkan merupakan penggambaran dari suatu masalah.
2. Lapisan Tersembunyi, node-node dalam lapisan ini disebut unit-unit tersembunyi. Di mana keluarannya tidak dapat secara langsung diamati.
3. Lapisan *Output*, node-node pada lapisan *output* disebut unit-unit output. Keluaran atau *output* dari lapisan ini merupakan *output* Jaringan Syaraf Tiruan terhadap suatu permasalahan.

II.4 Metode Backpropagation

Metode Backpropagation pertama kali diperkenalkan oleh Paul Werbos pada tahun 1974, kemudian dikemukakan kembali oleh David Parker di tahun 1982 dan kemudian dipopulerkan oleh Rumelhart dan McClland pada tahun 1986. terdiri dari tiga *layer*, yaitu *input layer*, *hidden layer* dan *output layer*. Pada *input layer* tidak terjadi proses komputasi, namun pada *input layer* terjadi pengiriman sinyal *input X* ke *hidden layer*. Pada *hidden* dan *output layer* terjadi proses komputasi terhadap bobot dan bias dan dihitung pula besarnya *output* dari *hidden* dan *output layer* tersebut berdasarkan fungsi aktivasi tertentu. Adapun, fungsi aktivasi biasanya digunakan sigmoid biner dengan *output* [0-1] (Simanungkalit et al., 2020).

Backpropagation adalah model JST yang paling banyak digunakan. Topologi khas backpropagation melibatkan tiga lapisan (Wanto, 2018). Pada metode backpropagation terdiri dari alur maju (*forward propagation*) dan alur mundur (*backward propagation*). Alur maju dilakukan terlebih dahulu untuk mengaktifkan *neuron-neuron* dengan menggunakan fungsi aktivasi, kemudian *error output* digunakan untuk mengubah nilai bobot-bobotnya ke arah alur mundur (Arianto & P, 2020).

Langkah-langkah yang perlu dilakukan untuk pelatihan pada algoritma Backpropagation adalah:

1. Inisialisasi bobot dengan bilangan nilai acak kecil.
2. Selama kondisi berhenti tidak terpenuhi, kerjakan langkah 3 – 8.

Tahap 1: Umpan Maju (Feed forward)

3. Tiap unit masukan $x_i, i = 1, \dots, n$ menerima isyarat masukan x_i dan diteruskan ke unit-unit tersembunyi (*hidden layer*).
4. Tiap unit tersembunyi $Z_j, j = 1, \dots, p$ menjumlahkan bobot sinyal *input* dengan persamaan.

$$Z_{injk} = v_{0j} + \sum_{i=1}^n x_i v_{ij} \quad (\text{II.1})$$

Dimana:

Z : *neuron hidden*.

v_{0j} : bobot bias *neuron input* ke-j

x_i : *neuron input* ke-i

v_{ij} : bobot *neuron input* ke *neuron hidden*.

Dengan menerapkan fungsi aktivasi yang dihitung dengan persamaan

$$Z_j = f(Z_{inj}) \quad (\text{II.2})$$

Dimana:

Z_j : unit ke j pada lapisan tersembunyi.

Z_{inj} : keluaran untuk unit Z_j

Jika fungsi aktiva yang di gunakan yaitu sigmoid dengan persamaan:

$$Y = f(x) = \frac{1}{1 + e^{-x}} \quad (\text{II.3})$$

Dan mengirimkan isyarat ini ke semua unit pada unit keluaran.

5. Tiap unit keluaran $y_k, k = 1, \dots, m$ menjumlahkan isyarat masukan berbobot dengan menggunakan persamaan.

$$Y_{in_k} = W_{0k} + \sum_{j=1}^p Z_j W_{jk} \quad (II.4)$$

Dimana:

Y_{in_k} : keluaran untuk unit y_k .

W_{0k} : bobot bias untuk *neuron hidden* ke-k

Z_j : unit ke-j

W_{jk} : bobot *neuron hidden* ke *neuron output*.

Dengan menerapkan fungsi aktivasi yang dihitung dengan persamaan.

$$Y_k = f(Y_{in_k}) \quad (II.5)$$

Dimana Y_{in_k} adalah keluaran untuk unit Y_k

Tahap 2: Perambatan Galat Mundur (*Backward*)

6. Tiap unit keluaran $y_k, k = 1, \dots, m$ menerima pola pelatihan masukannya.

Hitung galat (*error*) setiap *layer* dengan persamaan

$$\delta_k = (t_k - y_k) f'(y_{in_k}) \quad (II.6)$$

Dimana:

δ_k : faktor koreksi bobot W_{jk}

t : adalah target

y_k : *neuron output* ke-k

y_{in_k} : keluaran untuk unit y_k

Hitung koreksi bobot dan biasnya dengan persamaan

$$\Delta W_{jk} = \alpha \delta_k x_j \quad (II.7)$$

$$\Delta W_{0k} = \alpha \delta_k \quad (II.8)$$

Dimana:

ΔW_{jk} : selisih antara $W_{jk}(t)$ dengan $\Delta W_{jk}(t + 1)$

ΔW_{0k} : bobot bias untuk *neuron hidden* ke- k

α : *learning rate*

δ_k : faktor koreksi bobot W_{jk}

x : *input*

7. Tiap unit tersembunyi $Z_j, j = 1, \dots, p$ menjumlahkan *delta* masukannya (dari unit-unit yang berada pada lapisan atasannya) dengan persamaan.

$$\delta_{in_j} = \sum_{k=1}^m \delta_k W_{jk} \quad (\text{II.9})$$

Dimana:

δ_k : faktor koreksi bobot W_{jk}

W_{jk} : bobot *neuron hidden* ke *neuron output*

Hitung galat (*error*) setiap *layer* dengan persamaan.

$$\delta_j = \delta_{in_j} f(x_{in_j}) \quad (\text{II.10})$$

Dimana:

δ_j : faktor koreksi bobot V_{ij}

δ : faktor koreksi

x : *input*

Hitung koreksi bobot dan bias-nya dengan persamaan

$$\Delta V_{ij} = \alpha \delta_j x_i \quad (\text{II.11})$$

Dimana:

ΔV_{ij} : bobot *neuron input* ke *neuron hidden*

α : *learning rate*

δ_j : faktor korelasi bobot V_{ij}

x_i : *neuron input* ke- i

Tahap 3: Perbaiki bobot dan bias (Update Weight)

8. Tiap unit keluaran $y_j, k = 1, \dots, m$ memperbaharui bobot dan bias $j = 0.1, \dots, p$ dihitung dengan persamaan

$$W_{jk}(\text{baru}) = W_{jk}(\text{lama}) + \Delta W_{jk} \quad (\text{II.12})$$

Dimana:

W_{jk} : bobot *neuron hidden* ke *neuron output*

ΔW_{jk} : selisih bobot *neuron hidden* ke *neuron output*

δ_j : faktor korelasi bobot V_{ij}

x_i : *neuron input* ke- i

Tiap unit tersembunyi $Z_j, j = 1, \dots, p$ memperbaharui bobot dan bias-nya $i = 0.1, \dots, n$ dihitung dengan persamaan

$$V_{ij}(\text{baru}) = V_{ij}(\text{lama}) + \Delta V_{ij} \quad (\text{II.13})$$

Dimana:

V_{ij} : bobot *neuron input* ke *neuron hidden*

ΔV_{ij} : selisih bobot *neuron input* ke *neuron hidden*

9. Uji syarat berhenti

II.5 Penelitian Terkait

Penelitian dilakukan oleh (Yalidhan, 2018) memprediksi kelulusan mahasiswa dengan algoritma Backpropogation, dengan menggunakan 97 sampel data angkatan 2011, 2012 dan 2013 didapatkan akurasi sebesar 98.97 %.

Penelitian yang dilakukan oleh (Riyanto et al., 2019), memprediksi kelulusan mahasiswa dengan membandingkan tiga algoritma, yaitu Decision Tree (DT), Support Vector Machine (SVM) dan Neural Network (NN). Fitur yang digunakan dalam penelitian ini adalah jurusan di SMA, jenis kelamin, asal SMA, dan nilai IPK dari semester 1 sampai semester 6. Target prediksi yang digunakan di dalam penelitian ini adalah Lulus Tepat Waktu dan Lulus Tidak Tepat Waktu. Hasil penelitian ini menunjukkan bahwa algoritma DT menghasilkan akurasi sebesar 84.96%, algoritma SVM sebesar 85.18%, dan algoritma NN sebesar 84.68%.

Penelitian yang dilakukan oleh (Rismayati et al., 2022), memprediksi kelulusan mahasiswa dengan membandingkan delapan algoritma, yaitu Logistic Regression (LR), DT, Naïve Bayes (NB), Random Forest (RF), Ada Bost, SVM, K-Nearest Neighbors (KNN) and Multilayer Perceptron (MLP). Fitur yang digunakan dalam penelitian ini nilai IPK dari semester 1 sampai semester 4. Target prediksi yang digunakan di dalam penelitian ini adalah Lulus Tepat Waktu dan Lulus Tidak Tepat Waktu. Hasil penelitian ini menunjukkan bahwa algoritma LR menghasilkan rata-rata akurasi sebesar 91.46%, algoritma DT sebesar 83.58%, algoritma NB sebesar 82.99%, algoritma RF sebesar 90.45%, algoritma Ada Boost sebesar 90.87%, algoritma SVM sebesar 91.64%, algoritma KNN sebesar 89.8%, dan algoritma MLP sebesar 91.87%.

Penelitian yang dilakukan oleh (Ridwan et al., 2020), mengimplementasikan metode *constant learning rate* pada prediksi tingkat kelulusan mahasiwa menggunakan algoritma NN. Dengan menggunakan nilai *learning rate* sebesar 0.1

dan empat *training cycles* berbeda (200, 400, 600, 800), rata-rata akurasi yang diperoleh adalah 98.1375 %.

Penelitian yang dilakukan oleh (Yuhelmi et al., 2022), mengimplementasikan metode *constant learning rate* pada prediksi prestasi akademik mahasiswa yang bekerja paruh waktu. Dengan menggunakan nilai *learning rate* sebesar 0.42, 0.74, dan 0.9, akurasi yang diperoleh pada nilai *learning rate* 0.42 mencapai nilai 89.29%, pada nilai 0.74 mencapai akurasi sebesar 90%, dan pada nilai 0.9 mencapai akurasi sebesar 89.63%.

Penelitian yang dilakukan oleh (Firzatullah, 2021), mengimplementasikan metode *constant learning rate* pada penentuan penerima beasiswa Uang Kuliah Tunggal (UKT) dengan menggunakan algoritma Backpropagation. Dengan menggunakan nilai *learning rate* sebesar 0.1, 0.25, 0.5, dan 0.75, akurasi yang diperoleh pada nilai *learning rate* 0.1 mencapai nilai 46.625%, pada nilai 0.25 mencapai akurasi sebesar 46.625%, pada nilai 0.5 mencapai akurasi sebesar 46.625%, dan pada nilai 0.75 mencapai akurasi sebesar 38.875%.

Penelitian yang dilakukan oleh (Ge et al., 2019), mengimplementasikan metode *step decay* pada klasifikasi dataset CIFAR-10 menggunakan algoritma Stochastic Gradient Descent (SGD). Dengan menggunakan nilai *decay step* sebesar 0.0005, akurasi tertinggi yang diperoleh adalah 99.3%.

Penelitian yang dilakukan oleh (Martínez-Fernandez et al., 2023), mengimplementasikan metode *step decay* pada pendeteksian kanker kolorektal berdasarkan citra hispatologi menggunakan arsitektur VGG-19. Dengan

menggunakan nilai *minimum learning rate* sebesar 0.01 dan *maximum learning rate* sebesar 0.1, akurasi tertinggi yang diperoleh adalah 96.4%.

Penelitian yang dilakukan oleh (Nakamura et al., 2021), mengimplementasikan metode *step decay* pada Deep Neural Network dengan menggunakan dataset MNIST, Fashion-MNIST, SVHN, CIFAR-10, dan CIFAR-100. Dengan menggunakan nilai *decay step* sebesar 0.0005, rata-rata akurasi tertinggi yang diperoleh adalah 98.58%.

Penelitian yang dilakukan oleh (Li et al., 2021), mengimplementasikan metode *cosine annealing* pada klasifikasi dataset Fashion-MNIST, CIFAR-10, dan CIFAR-100 menggunakan algoritma CNN. Dengan menggunakan nilai *weight decay* sebesar 0.0001, akurasi yang diperoleh pada dataset Fashion-MNIST adalah 92.85%, akurasi pada dataset CIFAR-10 sebesar 91.99%, dan akurasi pada dataset CIFAR-100 sebesar 74.97%.

Penelitian yang dilakukan oleh (Pradiptatmaka et al., 2021), mengimplementasikan metode *cosine annealing* pada pendeteksian gerakan bibir menggunakan algoritma CNN dan Recurrent Neural Network (RNN). Dengan menggunakan nilai *weight decay* sebesar 0.01, akurasi tertinggi yang diperoleh adalah sebesar 82.8%.

Penelitian yang dilakukan oleh (Z. Wang et al., 2022), mengimplementasikan metode *cosine annealing* pada pengenalan suara otomatis menggunakan arsitektur VGG-16. Dengan menggunakan nilai *weight decay* sebesar 0.1, akurasi tertinggi yang diperoleh adalah sebesar 92.6%.

Dari penelitian-penelitian terkait di atas, penulis merangkum hasil analisis yang diperoleh pada Tabel II.1 sebagai acuan di dalam penelitian prediksi tingkat kelulusan mahasiswa STMIK Time.

Tabel II.1 Penelitian Terkait

Penulis	Topik	Hasil	Analisis
(Yalidhan, 2018)	Prediksi kelulusan mahasiswa menggunakan Backpropagation	Akurasi = 98.97%. Menggunakan 97 data.	Menggunakan 6 fitur yaitu Jenis Kelamin, SKS dari semester 1 sampai semester 4 dan nilai IPK dari semester 1 sampai semester 4. Sampel data yang digunakan terlalu kecil, karena belum mencapai nilai 100.
(Riyanto et al., 2019)	Membandingkan tiga algoritma pada prediksi kelulusan mahasiswa	Akurasi DT = 84.96%, SVM = 85.18%, NN = 84.68%.	Menggunakan 9 fitur, yaitu Jenis Kelamin, Jurusan SMA, Asal SMA, dan nilai IPK dari semester 1 sampai semester 6. Hanya menggunakan fitur nilai IPK sebagai acuan prestasi akademik mahasiswa, sehingga belum cukup untuk merepresentasikan kemampuan akademik mahasiswa yang diteliti.
(Rismayati et al., 2022)	Membandingkan delapan algoritma pada prediksi kelulusan mahasiswa	Akurasi LR = 91.46%, DT = 83.58%, NB = 82.99%, RF = 90.45%, Ada Boost = 90.87%, SVM = 91.64%, KNN = 89.8%, MLP = 91.87%.	Menggunakan 4 fitur, yaitu nilai IPK dari semester 1 sampai semester 4. Hanya menggunakan fitur nilai IPK dari semester 1 sampai 4, sehingga belum cukup untuk merepresentasikan kemampuan akademik mahasiswa yang diteliti.

Penulis	Topik	Hasil	Analisis
(Ridwan et al., 2020)	Implementasi <i>constant learning rate</i> pada prediksi kelulusan mahasiswa menggunakan algoritma NN	Rata-rata Akurasi = 98.1375%.	Menggunakan nilai <i>learning rate</i> 0.1 dengan variasi 200, 400, 600, dan 800 <i>training cycles</i> . Hanya menggunakan satu nilai <i>learning rate</i> , sehingga belum membandingkan performa model dengan <i>learning rate</i> yang berbeda.
(Yuhelmi et al., 2022)	Implementasi <i>constant learning rate</i> dengan tiga variasi nilai yang berbeda	Akurasi <i>learning rate</i> 0.42 = 89.29%, <i>learning rate</i> 0.74 = 90%, <i>learning rate</i> 0.9 = 89.63%.	Sudah menunjukkan pengaruh nilai <i>learning rate</i> yang berbeda terhadap performa model. Pemilihan nilai <i>learning rate</i> belum memiliki pola, sehingga perlu diperdalam lebih lanjut.
(Firzatullah, 2021)	Implementasi <i>constant learning rate</i> dengan empat variasi nilai yang berbeda.	Akurasi <i>learning rate</i> 0.1 = 46.625%, <i>learning rate</i> 0.25 = 46.625%, <i>learning rate</i> 0.5 = 46.625%, <i>learning rate</i> 0.75 = 38.875%..	Sudah menunjukkan pengaruh nilai <i>learning rate</i> yang berbeda terhadap performa model serta pemilihan <i>learning rate</i> sudah memiliki pola. Akurasi model terlalu rendah (belum mencapai nilai 50%), dan nilai yang dihasilkan hanya menunjukkan perbedaan performa pada satu <i>learning rate</i> saja.
(Ge et al., 2019)	Implementasi <i>step decay</i> pada klasifikasi dataset CIFAR-10 pada algoritma SGD.	Akurasi tertinggi = 99.3%.	Metode <i>step decay</i> berhasil diimplementasikan pada algoritma SGD dengan nilai akurasi yang sangat baik. Menarik untuk diteliti bagaimana performa metode <i>learning rate</i> tersebut pada algoritma Backpropagation.
(Martínez - Fernandez et al., 2023)	Implementasi <i>step decay</i> pada pendeteksian kanker kolorektal menggunakan arsitektur VGG-19.	Akurasi tertinggi = 96.4%.	Metode <i>step decay</i> berhasil diimplementasikan pada model <i>pre-trained</i> CNN, yaitu VGG-19 dengan nilai akurasi yang sangat baik. Menarik untuk diteliti bagaimana performa metode <i>learning rate</i> tersebut pada algoritma Backpropagation.

Penulis	Topik	Hasil	Analisis
(Nakamura et al., 2021)	Implementasi <i>step decay</i> pada lima dataset yang berbeda, dengan menggunakan algoritma Deep Neural Network.	Rata-rata Akurasi = 98.58%.	Metode <i>step decay</i> berhasil diimplementasikan pada algoritma Deep Neural Network dengan nilai akurasi yang sangat baik. Menarik untuk diteliti bagaimana performa metode <i>learning rate</i> tersebut pada algoritma Backpropagation.
(Li et al., 2021)	Implementasi <i>cosine annealing</i> pada tiga dataset yang berbeda, menggunakan algoritma CNN.	Akurasi dataset Fashion-MNIST = 92.85%, dataset CIFAR-10 = 90%, dataset CIFAR-100 = 74.97%.	Metode <i>cosine annealing</i> yang diimplementasikan pada algoritma CNN menunjukkan akurasi yang sangat baik pada dataset Fashion-MNIST dan CIFAR-10 dan hasil yang cukup baik pada dataset CIFAR-100. Menarik untuk diteliti bagaimana performa metode <i>learning rate</i> tersebut pada algoritma Backpropagation.
(Pradiptatama et al., 2021)	Implementasi <i>cosine annealing</i> pada dua algoritma yang berbeda, yaitu CNN dan RNN.	Akurasi tertinggi = 82.8.	Metode <i>cosine annealing</i> yang diimplementasikan pada algoritma CNN dan RNN berhasil mencapai akurasi yang sangat baik, sehingga menarik untuk diteliti bagaimana performa metode <i>learning rate</i> tersebut pada algoritma Backpropagation.
(Z. Wang et al., 2022)	Implementasi <i>step decay</i> pada pengenalan suara otomatis menggunakan arsitektur VGG-16.	Akurasi tertinggi = 92.6%.	Metode <i>cosine annealing</i> yang diimplementasikan pada arsitektur VGG-16 berhasil mencapai akurasi yang sangat baik, sehingga menarik untuk diteliti bagaimana performa metode <i>learning rate</i> tersebut pada algoritma Backpropagation.

Berdasarkan referensi penelitian pada Tabel II.1 di atas, penulis memilih tiga *learning rate scheduler*, yaitu Constant Learning Rate, Step Decay, dan Cosine

Annealing untuk diterapkan pada model yang dibangun. Selain itu, penulis menggunakan rentang nilai 0.001 sampai 0.1 sebagai nilai *learning rate* yang diteliti, untuk kemudian dianalisa *learning rate* mana yang menghasilkan performa model terbaik berdasarkan hasil evaluasi model.

II.6 Ringkasan Bab II

Beberapa indikator kelulusan mahasiswa seperti IPK, SKS, waktu kuliah dan status keaktifan dapat digunakan sebagai fitur dalam perancangan sebuah model prediksi. Jaringan syaraf tiruan dapat dimanfaatkan dalam merancang sebuah model prediksi, dengan cara memproses informasi pada sebuah dataset melalui *neuron* pada *input layer* dan *hidden layer*, sehingga menghasilkan output prediksi yang memiliki tingkat akurasi yang tinggi. Sebagai salah satu jenis algoritma yang menggunakan arsitektur jaringan syaraf tiruan, Backpropagation melakukan komputasi bobot dan bias pada *hidden layer* dengan bantuan sebuah fungsi aktivasi berdasarkan nilai *learning rate* yang diinputkan. Algoritma ini sudah terbukti memiliki tingkat akurasi yang baik jika diimplementasikan pada masalah prediksi, khususnya yang berhubungan dengan tingkat kelulusan siswa/mahasiswa. Dengan variasi jenis-jenis *learning rate* seperti *constant learning rate*, *step decay*, dan *cosine annealing*, Backpropagation beberapa penelitian terdahulu sudah membuktikan bagaimana pemilihan jenis *learning rate* mempengaruhi tingkat akurasi dari model prediksi yang dihasilkan.