

Nama: M.Rizky Fadillah (1103210259)

Kelas: TK-45-GAB

LAPORAN TUGAS 14 PEMBELAJARAN MESIN

- **Markov**

Pada eksperimen ini, dilakukan pelatihan model menggunakan Recurrent Neural Network (RNN) yang dimaksudkan untuk meniru pendekatan model Hidden Markov Model (HMM) dalam kasus klasifikasi multi-kelas. Dataset yang digunakan diproses dengan teknik standarisasi dan pembagian data ke dalam set pelatihan dan pengujian. Eksperimen bertujuan untuk mengevaluasi pengaruh beberapa faktor terhadap kinerja model, termasuk ukuran hidden layer, jenis pooling, algoritma optimizer, dan jumlah epoch pelatihan.

2. Persiapan Data

- **Dataset:** Data yang digunakan adalah dataset yang diunggah melalui Google Colab. Dataset ini kemudian dibaca ke dalam DataFrame menggunakan pustaka Pandas. Dataset tersebut memiliki fitur dan target yang dipisahkan sebagai `X_train` untuk fitur dan `y_train` untuk label.
- **Preprocessing:** Fitur dalam dataset distandarisasi menggunakan `StandardScaler` agar model menerima data yang ternormalisasi. Proses ini penting untuk memastikan bahwa setiap fitur memiliki skala yang seragam, yang dapat mempercepat konvergensi dalam pelatihan model.
- **Pembagian Data:** Data dibagi menjadi set pelatihan dan pengujian dengan rasio 80:20 menggunakan fungsi `train_test_split` dari `scikit-learn`. Pembagian ini dilakukan dengan menggunakan stratifikasi untuk memastikan distribusi label pada kedua set tersebut tetap seimbang.

3. Desain Model

Model yang digunakan adalah **MarkovRNN**, sebuah implementasi dari RNN yang mencakup beberapa fitur penting:

- **RNN Layer:** Layer RNN pertama kali mengolah data urutan untuk menangkap pola temporal atau sekuensial dalam data.
- **Pooling:** Setelah output dari RNN, dilakukan pooling menggunakan metode tertentu. Ada dua jenis pooling yang diuji dalam eksperimen ini:
 - **Max Pooling:** Mengambil nilai maksimum dari output di sepanjang dimensi waktu.
 - **Average Pooling:** Mengambil rata-rata dari output di sepanjang dimensi waktu.
- **Fully Connected Layer:** Output dari pooling diteruskan ke lapisan fully connected (linear) untuk menghasilkan prediksi akhir.

4. Fungsi Pelatihan dan Evaluasi

- **Pelatihan:** Fungsi `train_model` digunakan untuk melatih model dengan menggunakan data pelatihan. Dalam setiap epoch, model menerima batch data, menghitung loss menggunakan fungsi `CrossEntropyLoss`, dan kemudian melakukan pembaruan bobot dengan optimizers seperti Adam.
- **Evaluasi:** Fungsi `evaluate_model` digunakan untuk mengukur akurasi model pada data pengujian. Model dievaluasi dengan menghitung jumlah prediksi yang benar dibandingkan dengan jumlah total data.

5. Eksperimen dan Hasil

Eksperimen dilakukan dengan mengubah beberapa parameter untuk menguji pengaruhnya terhadap kinerja model. Parameter yang diuji meliputi ukuran hidden layer, jenis pooling, optimizer yang digunakan, dan jumlah epoch.

5.1. Pengaruh Ukuran Hidden Layer

Eksperimen pertama dilakukan dengan dua ukuran hidden layer: 32 dan 64. Hasilnya adalah sebagai berikut:

- Ukuran hidden layer 32: **Akurasinya 96.99%**
- Ukuran hidden layer 64: **Akurasinya 98.04%**

Dengan ukuran hidden layer yang lebih besar, akurasi model meningkat.

5.2. Pengaruh Jenis Pooling

Eksperimen kedua menguji dua jenis pooling: max pooling dan average pooling. Hasilnya menunjukkan bahwa kedua jenis pooling memberikan hasil yang sama:

- **Max Pooling:** Akurasi **96.73%**
- **Average Pooling:** Akurasi **96.73%**

Dengan demikian, jenis pooling tidak memiliki pengaruh yang signifikan terhadap akurasi model.

5.3. Pengaruh Optimizer

Eksperimen ketiga menguji tiga optimizer yang berbeda: SGD, RMSProp, dan Adam. Hasilnya adalah sebagai berikut:

- **SGD:** Akurasi **90.07%**
- **RMSProp:** Akurasi **96.99%**
- **Adam:** Akurasi **97.39%**

Optimizer Adam menunjukkan hasil terbaik dalam hal akurasi, diikuti oleh RMSProp dan SGD.

5.4. Pengaruh Jumlah Epoch

Eksperimen terakhir menguji berbagai jumlah epoch, yaitu 5, 50, 100, 250, dan 350. Hasilnya adalah sebagai berikut:

- **5 Epoch:** Akurasi **97.25%**
- **50 Epoch:** Akurasi **96.60%**
- **100 Epoch:** Akurasi **96.86%**
- **250 Epoch:** Akurasi **97.52%**
- **350 Epoch:** Akurasi **97.12%**

Jumlah epoch yang lebih banyak (250 epoch) memberikan akurasi terbaik.

6. Kesimpulan

Dari eksperimen yang dilakukan, dapat disimpulkan bahwa:

- **Ukuran Hidden Layer:** Semakin besar ukuran hidden layer, semakin tinggi akurasi yang diperoleh.
- **Jenis Pooling:** Tidak ada perbedaan signifikan antara max pooling dan average pooling dalam hal akurasi.
- **Optimizer:** Optimizer Adam memberikan kinerja terbaik, diikuti oleh RMSProp dan SGD.
- **Jumlah Epoch:** Lebih banyak epoch memberikan akurasi yang lebih baik, dengan 250 epoch menghasilkan akurasi terbaik.