

Demo Day Final Report

Machine Learning with TensorFlow Training

Professional Academy Digital Talent Scholarship 2022

Group Number	AUTO_6
Dataset	Grocery Data - Automatic checkout, warehouse, or stock inventory system
Name - DTS ID	1. Mayasari Dwi Partono - 152236035100-308 2. Ulfa Illiyina - 152236035100-538 3. Yulia Ratningsari - 152236035100-180 4. Mufita Rahmadina - 152236035100-880

Selected Theme: Automatic checkout, warehouse, or stock inventory system

Title of the Project: Seleksi fitur dan modelling total pendapatan dari data automatic checkout, warehouse dan inventory system (grocery_data)

Background Summary:

Toko kelontong adalah industri senilai \$ 1 triliun di Amerika Serikat saja. Terlepas dari ukurannya, toko kelontong lambat mengadopsi teknologi baru. hal ini mulai berubah dengan munculnya machine learning dan data science. Machine learning adalah jenis kecerdasan buatan yang memungkinkan komputer belajar dari data tanpa diprogram secara eksplisit. Data science adalah studi tentang bagaimana mengekstrak informasi dari data. Machine learning dan data science penting dalam pengelolaan toko kelontong. Machine learning dapat digunakan untuk memprediksi permintaan pelanggan dan mengoptimalkan inventaris. Ilmu data dapat digunakan untuk memahami perilaku pelanggan dan target kampanye pemasaran. Penggunaan machine learning dan data science di toko kelontong masih dalam tahap awal. Namun, jelas bahwa teknologi ini akan berdampak besar pada industri di tahun-tahun mendatang.

Pada bidang bisnis diperlukan kebijakan yang baik untuk meningkatkan hasil, salah satunya dari data. Data bisnis tidak berguna jika dalam bentuk data saja melainkan harus dianalisis dan direpresentasikan dalam bentuk yang dapat digunakan oleh pelaku bisnis. Analisis data bisnis berguna dalam membantu perencanaan dan strategi bisnis, misalnya prediksi pembelian yang bertujuan untuk memprediksi jumlah harga pembelian dari data series. Data pembelian saat ini menjadi data yang berlimpah dan perlu dilakukan pengolahan lebih lanjut untuk dimanfaatkan pelaku bisnis.

Grocery automatic checkout system merupakan sistem yang digunakan untuk memproses data di toko kelontong. Sistem ini digunakan untuk manajemen inventory toko dan menghitung total pendapatan toko, sistem ini juga menghitung pajak toko dan menyediakan struk penerimaan dari bagi toko, Sistem ini juga bisa digunakan untuk menghitung stok barang dalam toko. Dalam demo kali ini kami memiliki beberapa hal yang kami ingin pecahkan yaitu bagaimanakah profile penjualan produknya, penjualan barang apa aja yang paling banyak memberikan pendapatan, bagaimana potensi kedepan pendapatan toko ini apakah akan naik atau turun (prediksi pendapatan), barang apakah yang akan naik permintaan nya, dan bagaimana modeling pendapatan toko tersebut terhadap waktu. Untuk itu digunakan fitur seleksi untuk menemukan maksimal 30 macam barang yang dominan terhadap pendapatan, dan time series modeling untuk membuat pemodelan total pendapatan.

Please specifically mention what you've done:

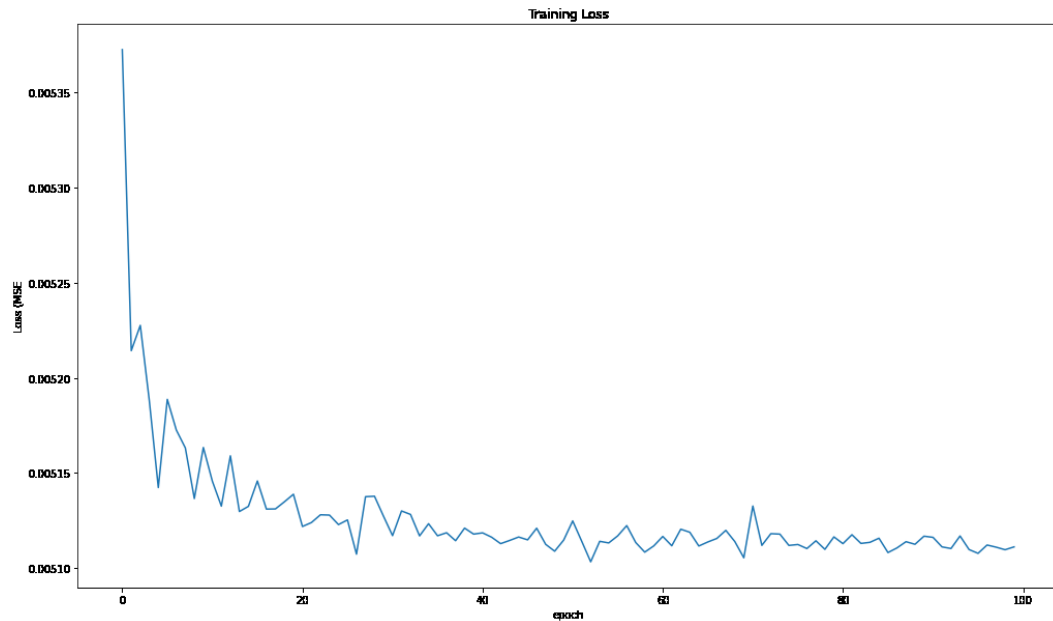
- [1] Mayasari Dwi Partono - 152236035100-308 : (membuat file Auto6_FeatureSelection_Modelling.ipynb) - Mengerjakan pembuatan program machine learning mulai dari Data engineering, cleaning, exploratory data analysis, feature selection dengan Genetic algorithm estimator decision tree, pembuatan model prediksi time series pendapatan dari 9 data pendapatan sebelumnya, pembuatan model time series pendapatan terhadap waktu, pembuatan github, pembuatan video dan pembuatan laporan akhir.
- [2] Ulfa Illiyina - 152236035100-538, (membuat file Update prediction_data_Automatic_checkout_AOUTO6.ipynb) pembuatan data untuk modelling, Plot chart, pembuatan modeling Prediksi total barang yang akan dibeli konsumen menggunakan Bidirectional LSTM, learning, prediksi dengan 2 macam model, pembuatan video dan penjelasan model.
- [3] Yulia Ratningsari - 152236035100-180, mengerjakan pembuatan proposal dan video
- [4] Mufita Rahmadina - 152236035100-880, pembuatan video Exploratory data Analysis

Screenshot Output Model :

Dari file [Auto6_FeatureSelection_Modelling.ipynb](#)

Hasil modelling prediksi total pendapatan dari 9 data total pendapatan sebelumnya.

Grafik Training Loss



```
#membuat prediksi output model dari data training (input dfx)
prediksi = model.predict(dfx)
```

```
#Hitung score hasil prediksi model (Evaluasi)
#Diperoleh hasil score evaluasi 0.5% error
scores = model.evaluate(dfx, dfy, verbose=1)
print("Hasil tes evaluasi model : {} %".format(scores * 100))
```

```
156/156 [=====] - 2s 6ms/step - loss: 0.0051
Hasil tes evaluasi model : 0.5110546946525574 %
```

```
#Hitung score error hasil prediksi MAE, MSE, RMSE
from sklearn import metrics

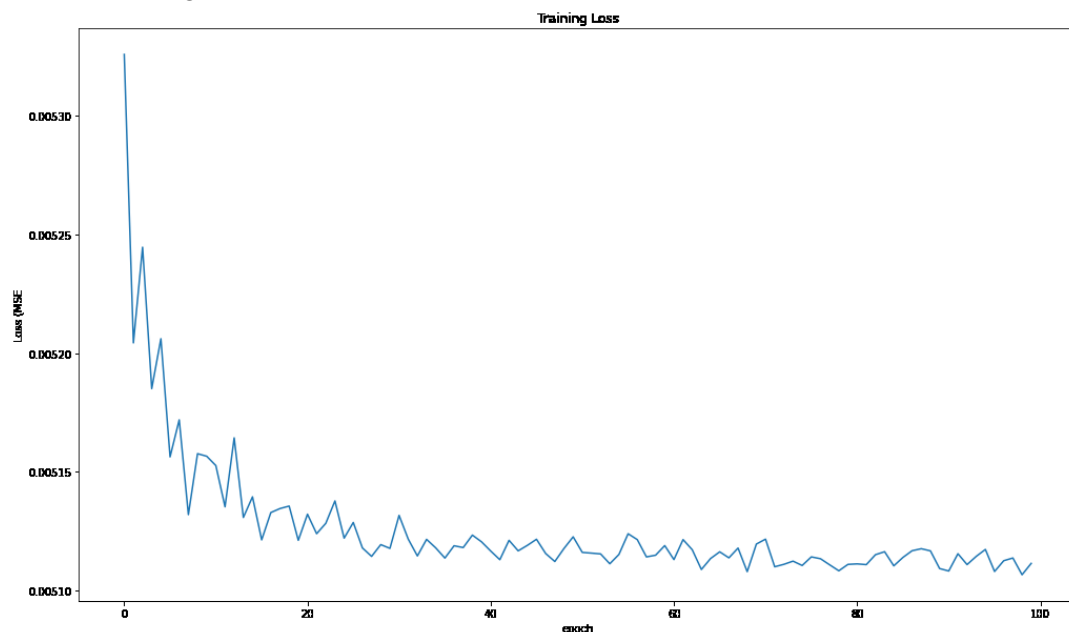
print('Mean Absolute Error:', np.round(metrics.mean_absolute_error(dfy, prediksi),3)) #MAE
print('Mean Squared Error:', np.round(metrics.mean_squared_error(dfy, prediksi),3)) #MSE
print('Root Mean Squared Error:', np.round(metrics.mean_squared_error(dfy, prediksi, squared=False),3)) #RMSE
```

```
Mean Absolute Error: 0.052
Mean Squared Error: 0.005
Root Mean Squared Error: 0.071
```

```
#menampilkan grafik hubungan antar actual Total_Price dan Prediksi total_price
plt.scatter(prediksi, dfy)
plt.xlabel("Predicted Total price with RNN model",fontsize=20)
plt.ylabel("Actual Total price",fontsize=20)
```

Hasil modeling Total pendapatan terhadap waktu kedatangan

Grafik Training Loss



```
# Prediksi total_price dari data training
prediksi01 = model.predict(dfx1)
```

```
#Evaluasi model dengan data training
scores = model.evaluate(dfy1, dfy1, verbose=1)
print("Hasil tes evaluasi model : {} %".format(scores * 100))
```

```
156/156 [=====] - 4s 7ms/step - loss: 0.0051
Hasil tes evaluasi model : 0.5110181868076324 %
```

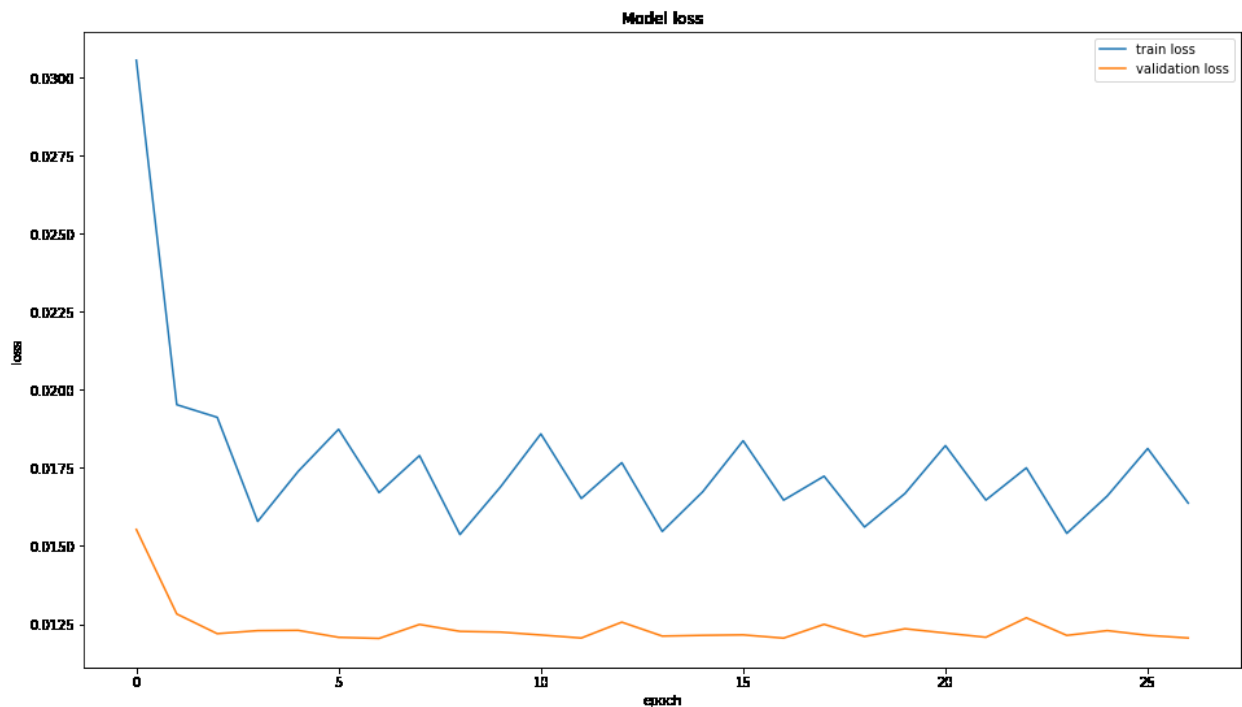
```
#Cek nilai error MAE, MSE dan RMSE
print('Mean Absolute Error:', np.round(metrics.mean_absolute_error(dfy1, prediksi01),3)) #MAE
print('Mean Squared Error:', np.round(metrics.mean_squared_error(dfy1, prediksi01),3)) #MSE
print('Root Mean Squared Error:', np.round(metrics.mean_squared_error(dfy1, prediksi01, squared=False),3)) #RMSE

Mean Absolute Error: 0.053
Mean Squared Error: 0.005
Root Mean Squared Error: 0.071
```

Dari file [prediction_data Automatic checokut AOUTO6.ipynb](#), Prediksi total barang yang akan dibeli konsumen menggunakan Bidirectional LSTM.

Hasil model ke 1:

Grafik loss training dan validation

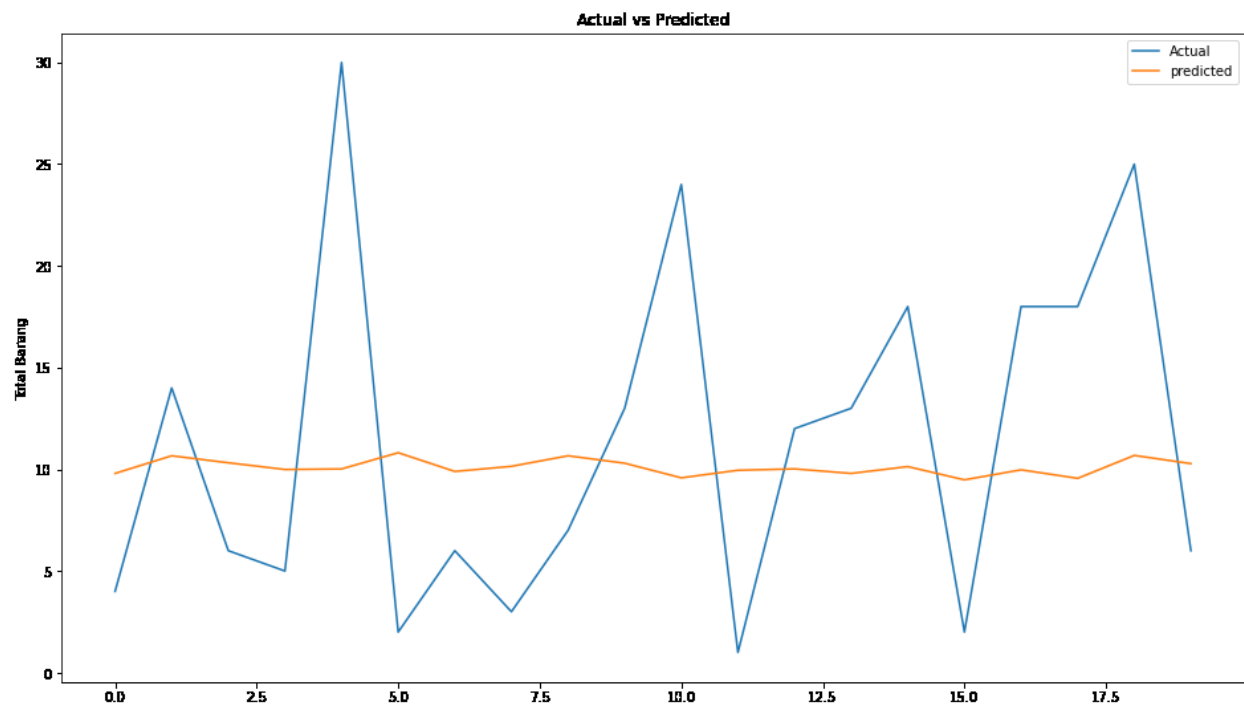


```
from sklearn import metrics

#Melakukan evaluasi performa
print(f"MSE: {metrics.mean_squared_error(validate['total_barang'], prediksi_inverse[0])}") #MSE
print(f"MAE: {metrics.mean_absolute_error(validate['total_barang'], prediksi_inverse[0])}") #MAE
print(f"MAPE: {metrics.mean_absolute_percentage_error(validate['total_barang'], prediksi_inverse[0])}") #MAPE
print(f"RMSE: {np.sqrt(metrics.mean_squared_error(validate['total_barang'], prediksi_inverse[0]))}") #RMSE

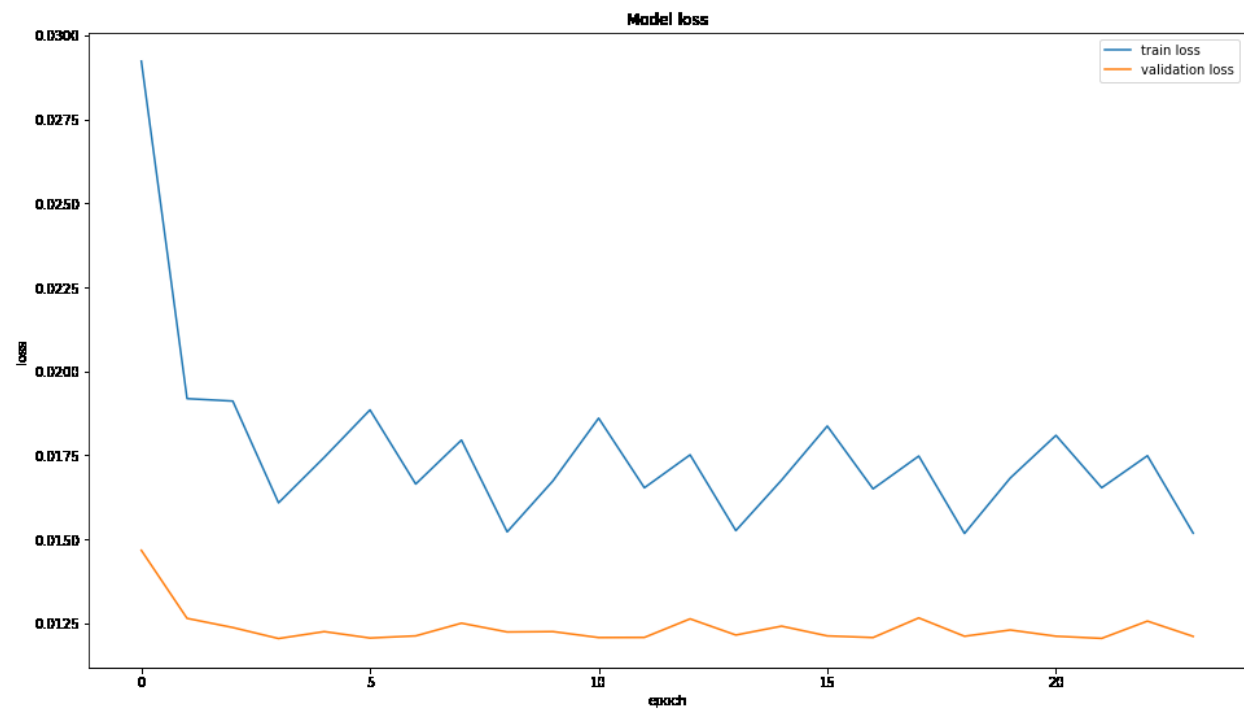
MSE: 71.57074209535381
MAE: 7.180915880203247
MAPE: 1.42930053552312
RMSE: 8.459949296263767
```

Grafik actual vs predicted



Hasil model ke 2 :

Grafik training loss dan validation loss

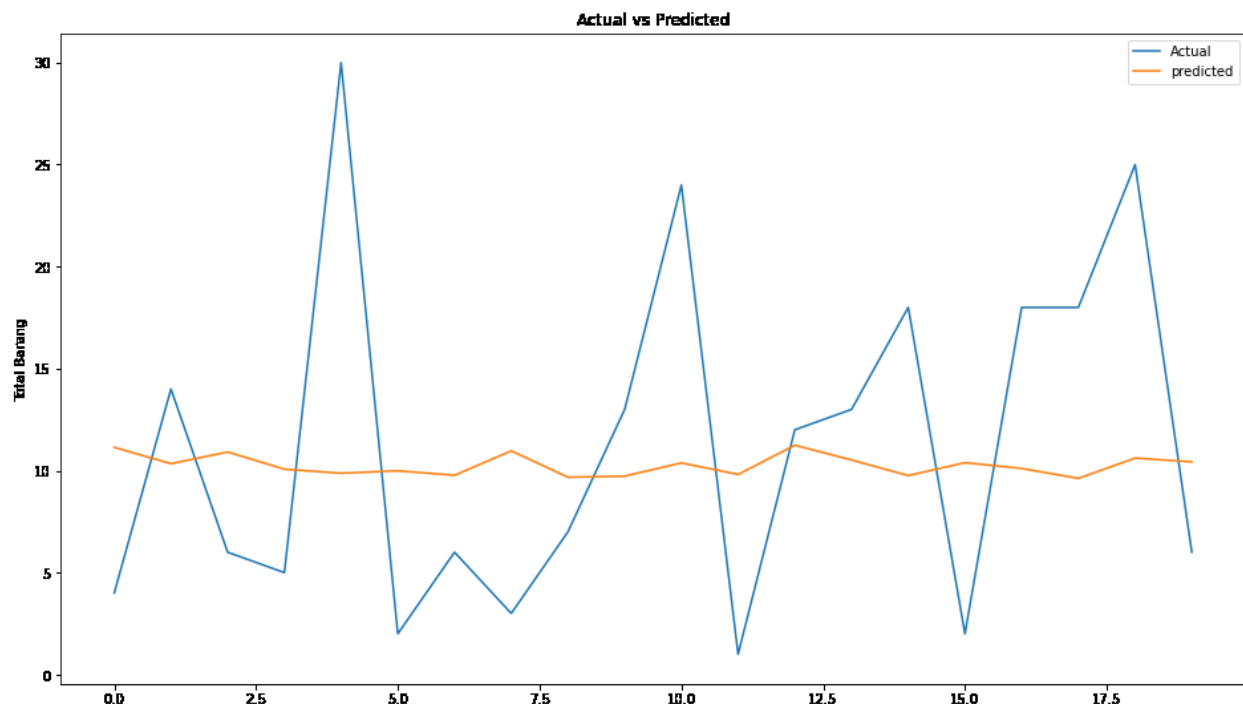


```
from sklearn import metrics

#melakukan evaluasi performa
print(f"MSE: {metrics.mean_squared_error(validate['total_barang'], prediksi_inverse[0])}") #MSE
print(f"MAE: {metrics.mean_absolute_error(validate['total_barang'], prediksi_inverse[0])}") #MAE
print(f"MAPE: {metrics.mean_absolute_percentage_error(validate['total_barang'], prediksi_inverse[0])}") #MAPE
print(f"RMSE: {np.sqrt(metrics.mean_squared_error(validate['total_barang'], prediksi_inverse[0]))}") #RMSE
```

MSE: 72.31182767807695
MAE: 7.198863983154297
MAPE: 1.4483718019378928
RMSE: 8.50363614450177

Grafik actual vs predicted



Penjelasan Hasil :

Prediksi time series multivariate: Prediksi total barang yang akan dibeli konsumen menggunakan Bidirectional LSTM (kolom 'total_barang')

Pada sesi ini menggunakan bidirectional LSTM pada model yang digunakan. Bidirectional LSTM dapat meningkatkan performa dan meningkatkan kecepatan learning. Contoh dalam masalah di mana semua time steps merupakan input sequence, Bidirectional LSTM melakukan training dua. Pertama pada input sequence dan kedua pada salinan inverse dari input sequence. Ini dapat memberikan konteks tambahan ke jaringan dan menghasilkan learning yang lebih cepat dan lebih lengkap tentang masalah tersebut.

Kasus ini menggunakan window 25 dan memprediksi total barang 20 data ke depan. Dalam prediksi total barang ini, dilakukan dua kali training dengan model yang berbeda. Untuk model pertama pada training pertama menggunakan dua layer bidirectional LSTM, pertama dengan unit 100 dan kedua dengan unit 80. Sedangkan model kedua pada training kedua menggunakan dua layer bidirectional LSTM, pertama dengan unit 150 dan kedua dengan unit 120. Unit bidirectional LSTM yang digunakan model kedua lebih besar dari model pertama. Grafik loss train data pada training pertama dan training kedua menunjukkan penurunan nilai loss pada lima data pertama.

Grafik nilai total barang actual vs prediksi untuk training pertama dan training kedua, menunjukkan ada kemiripan pada kedua grafik tersebut. Untuk model pertama MSE sebesar 71.57 lebih kecil dibanding MSE model kedua sebesar 72.31. MAE model pertama sebesar 7.18 lebih kecil dibanding MAE model kedua sebesar 7.19. Untuk model pertama RMSE sebesar 8.45 lebih kecil dibanding RMSE model kedua sebesar 8.50. MAPE untuk model pertama sebesar 1.42 lebih kecil dibanding MAPE model kedua sebesar 1.44. Kedua hasil MAPE untuk model pertama dan kedua nilainya dibawah 10%, menunjukkan model bekerja dengan sangat baik.

Github Repo Link:

- [Automatic Checkout Auto6/Auto6 FeatureSelection Modelling.ipynb at main · Dtscience80/Automatic_Checkout_Auto6 \(github.com\)](https://github.com/Dtscience80/Automatic_Checkout_Auto6)
- https://github.com/Dtscience80/Automatic_Checkout_Auto6

10-Min Video Presentation Link:

Video [Final Video AUTO 6 Dts Kominfo](https://www.youtube.com/watch?v=cHjM_2F6eVs) : https://www.youtube.com/watch?v=cHjM_2F6eVs



