

**MODEL PREDIKSI PENJUALAN MULTI-ITEM *TIME SERIES* BERBASIS
MACHINE LEARNING MENGGUNAKAN METODE *AUTOREGRESSIVE
INTEGRATED MOVING AVERAGE* DAN *LONG SHORT-TERM MEMORY*
PADA PRODUK *PERISHABLE*
(STUDI KASUS: RETAIL SAYUR TOSAGA)**

TUGAS AKHIR

**Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh Gelar Sarjana Strata-1
Pada Jurusan Teknik Industri Fakultas Teknologi Industri**



Nama : Shelly Elvina Salsabila

No. Mahasiswa : 16522173

**PROGRAM STUDI TEKNIK INDUSTRI
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA
YOGYAKARTA
2020**

PERNYATAAN KEASLIAN

Demi Allah, saya akui karya ini adalah karya hasil kerja saya sendiri kecuali nukilan dan ringkasan yang setiap satunya telah saya jelaskan sumbernya. Jika dikemudian hari ternyata terbukti pengakuan saya ini tidak benar dan melanggar peraturan yang sah dalam karya tulis dan hak kekayaan intelektual maka saya bersedia ijazah yang telah saya terima untuk ditarik kembali oleh Universitas Islam Indonesia.

Yogyakarta, 12 November 2020



Shelly Elvina Salsabila

16522173

**MODEL PREDIKSI PENJUALAN MULTI-ITEM *TIME SERIES* BERBASIS
MACHINE LEARNING MENGGUNAKAN METODE *AUTOREGRESSIVE
INTEGRATED MOVING AVERAGE* DAN *LONG SHORT-TERM MEMORY*
PADA PRODUK *PERISHABLE*
(STUDI KASUS: RETAIL SAYUR TOSAGA)**

TUGAS AKHIR



Nama : Shelly Elvina Salsabila

No. Mahasiswa : 16522173

Yogyakarta, 12 November 2020

Pembimbing

A handwritten signature in black ink, appearing to be 'Abdullah 'Azzam', is written over a faint circular stamp.

(Abdullah 'Azzam, S.T., M.T)

**MODEL PREDIKSI PENJUALAN MULTI-ITEM *TIME SERIES* BERBASIS
MACHINE LEARNING MENGGUNAKAN METODE *AUTOREGRESSIVE
INTEGRATED MOVING AVERAGE* DAN *LONG SHORT-TERM MEMORY*
PADA PRODUK *PERISHABLE*
(STUDI KASUS: RETAIL SAYUR TOSAGA)**

TUGAS AKHIR

Oleh

Nama : Shelly Elvina Salsabila

No. Mahasiswa : 16522173

Telah dipertahankan di depan sidang penguji sebagai satu syarat untuk memperoleh
gelar Sarjana Strata-1 Teknik Industri

Yogyakarta, 3 Desember 2020

Tim Penguji

Abdullah 'Azzam, S.T., M.T.

Ketua

Annisa Uswatun Khasanah, S.T., M.Sc.

Anggota I

Suci Miranda, S.T., M.Sc.

Anggota II



Mengetahui
Ka.Prodi Teknik Industri
Fakultas Teknologi Industri
Universitas Islam Indonesia



Abdullah 'Azzam, S.T., M.T.

HALAMAN PERSEMBAHAN

Alhamdulillahirabbil' alamin

Kepada kedua orang tua yang saya cintai, Dwi Hariyanto dan Heni Nur Amrina, terima kasih atas doa dan dukungan yang selama ini selalu diberikan.

Kepada dosen pembimbing yang saya hormati, Abdullah 'Azzam, S.T., M.T., terima kasih atas bimbingan dan ilmu yang diberikan sejak saya menjadi asisten di Laboratorium Sistem Manufaktur.

Kepada teman, kakak, dan adik Teknik Industri Industri UII, terima kasih atas ilmu, kebaikannya, dan dukungannya.

Serta semua yang memberikan saya kepercayaan,
Jazakumullah Khairan Katsira.

Dan yang terakhir untuk diri saya sendiri,
terima kasih sudah berusaha. *Nothing to lose.*

HALAMAN MOTTO

“Karena sesungguhnya sesudah kesulitan ada kemudahan, sesungguhnya sesudah kesulitan ada kemudahan”

Al-Insyirah: 5-6

“Janganlah engkau bersedih, sesungguhnya Allah bersama kita”

At-Taubah: 40



HALAMAN PENGANTAR

Asalamualaikum Warahmatullahi Wabarakatuh

Puji syukur atas kehadiran Allah SWT dengan berbagai limpahan karunia yang kita terima hingga saat ini. Shalawat serta salam tidak lupa kita ucapkan kepada junjungan kita Nabi Muhammad SAW, keluarga, dan para sahabat beliau. Semoga kita termasuk umat beliau yang senantiasa giat dalam mengamalkan ajaran yang beliau sampaikan. Amin.

Alhamdulillahirabbil' alamin atas karunia-Nya dan doa serta semangat orang terdekat, Laporan Tugas Akhir ini dapat diselesaikan. Dalam mengolah dan menganalisis ilmu di laporan ini tentunya tidak lepas dari bantuan beberapa cendekiawan dan dermawan. Diantaranya penulis berterimakasih kepada:

1. Hari Purnomo, Prof., Dr., Ir., M.T. selaku Dekan Fakultas Teknologi Industri Universitas Islam Indonesia.
2. M. Ridwan Andi Purnomo, S.T., M.Sc., Ph.D.. selaku Kepala Jurusan Teknik Industri Universitas Islam Indonesia.
3. Dr. Taufiq Immawan, S.T., M.M. selaku Kepala Program Studi S1 Teknik Industri Universitas Islam Indonesia.
4. Abdullah 'Azzam, S.T., M.T. selaku dosen pembimbing yang selalu membimbing, mendukung, serta menyediakan waktu untuk penulis sampai akhirnya dapat menyelesaikan Laporan Tugas Akhir.
5. Mama, Papa, dan Aldi selaku keluarga atas doa, semangat, dan dukungan yang selalu diberikan.

6. Kepala Laboratorium, Laboran dan seluruh Asisten Laboratorium Sistem Manufaktur Terintegrasi Universitas Islam Indonesia yang telah memberikan kesempatan bagi penulis untuk mengembangkan diri dan ilmu.
7. Teman dekat penulis, atas dukungan, doa dan keikhlasan bantuan serta waktu yang diberikan.
8. Semua yang tidak dapat penulis sebutkan satu per satu yang membantu dalam penyelesaian Laporan Tugas Akhir ini.

Meskipun banyak kekurangan dalam penulisan, terdapat harapan semoga laporan ini dapat bermanfaat dan mendatangkan berkah ilmu bagi pembaca pada umumnya dan mahasiswa Teknik Industri Universitas Islam Indonesia pada khususnya. Sebagai refleksi, ilmu tanpa disebar dan dikembangkan hanya akan menuju sia-sia. Mulai saat yang tepat, saya selaku penulis berkeinginan lebih keras berbagi dan belajar, agar nilai luhur Ulil Albab dapat tertanam dalam diri saya, dan kemudian dapat memberikan kontribusi kepada lingkungan

Wasalamualaikum Warahmatullahi Wabarakatuh

Yogyakarta, 12 November 2020



(Shelly Elvina Salsabila)

16522173

ABSTRAK

Perishable product merupakan peroduk yang kualitasnya akan menurun seiring dengan bertambahnya umur produk yang tidak lebih dari 14 hari. Hal ini akan mempengaruhi kepuasan konsumen dalam pemilihan produk. Kesalahan peramalan pada *perishable product* dapat menyebabkan kerugian akibat hal tersebut. Retail Sayur Tosaga merupakan toko yang menjual berbagai macam kebutuhan masak, salah satunya adalah *perishable product*. Peramalan permintaan yang selama ini dilakukan berdasarkan dengan *expert judgement*, akibatnya hampir setiap hari akan ada produk sisa. Penelitian ini akan berfokus pada pembuatan model prediksi menggunakan metode ARIMA dan LSTM yang diproses menggunakan pemrograman Python. Terdapat 2 dataset yang dimodelkan, yaitu Dataset 1 untuk produk dengan satuan kilogram (KG), dan Dataset 2 untuk produk dengan satuan unit (PCS). Performa kedua metode tersebut akan diukur menggunakan RMSE, pada metode ARIMA dihasilkan nilai 7,4% untuk Dataset 1 dan 34,3% untuk Dataset 2. Sedangkan untuk metode LSTM menunjukkan nilai RMSE yang lebih kecil pada kedua dataset, namun kedua model tersebut memiliki *learning curve* yang termasuk dalam kategori *overfit*. Sehingga metode yang baik untuk digunakan adalah metode ARIMA, hal ini dikarenakan metode LSTM *overfit* yang artinya hasil dari pelatihan data baik namun tidak dapat menggeneralisasi data baru sehingga tidak bisa digunakan untuk melakukan prediksi secara berkala.

Keyword: ARIMA, LSTM, Perishable Product, Prediksi

DAFTAR ISI

HALAMAN JUDUL.....	i
PERNYATAAN KEASLIAN.....	ii
LEMBAR PENGESAHAN DOSEN PEMBIMBING.....	iii
LEMBAR PENGESAHAN TIM PENGUJI.....	iv
HALAMAN PERSEMBAHAN	v
HALAMAN MOTTO.....	vi
HALAMAN PENGANTAR.....	vii
ABSTRAK.....	ix
DAFTAR ISI.....	x
DAFTAR TABEL.....	xiii
DAFTAR GAMBAR.....	xiv
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	3
1.3 Batasan Masalah	4
1.4 Tujuan Penelitian.....	4
1.5 Manfaat Penelitian.....	4
1.6 Sistematika Penulisan	5
BAB II KAJIAN LITERATUR.....	6
2.1 Kajian Literatur Deduktif	6
2.1.1 <i>Perishable Product</i>	6
2.1.2 <i>Time series technique</i>	7
2.1.3 <i>System modelling</i>	8
2.1.4 <i>Data Mining</i>	9
2.1.5 <i>Machine learning</i>	10
2.1.6 Pengukuran Kesalahan Peramalan	11
2.2 Kajian Literatur Induktif.....	13
2.2.1 <i>From Predictive to Prescriptive Analytics: A Data-driven Multi-item Newsvendor Model</i>	13
2.2.2 <i>Hospital Daily Outpatient Visits Forecasting Using A Combinatorial Model Based on ARIMA and SES Models</i>	14
2.2.3 <i>Sales Demand Forecast in E-commerce Using a Long Short-Term Memory Neural Network Methodology</i>	14
2.2.4 <i>DeepAR: Probabilistic Forecasting with Autoregressive Recurrent Networks</i>	15
2.2.5 <i>CNN-LSTM Coupled Model for Prediction of Waterworks Operation Data</i>	15
2.2.6 <i>The Forecasting of PM2.5 Using a Hybrid ModelBased on Wavelet Transform and an ImprovedDeep Learning Algorithm</i>	16

2.2.7	<i>Short-term Load Forecasting by Using a Combined Method of Convolutional Neural Networks and Fuzzy Time Series</i>	17
2.2.8	<i>A Hybrid Neural Network Model for Sales Forecasting Based on ARIMA and Search Popularity of Article Titles</i>	17
2.2.9	<i>Travel Time Prediction with LSTM Neural Network</i>	18
2.2.10	<i>Forecasting of Particulate Matter Time Series using Wavelet Analysis and Wavelet-ARMA/ARIMA Model in Taiyuan, China</i>	18
2.2.11	<i>Cluster-based Hierarchical Demand Forecasting for Perishable Goods</i>	19
2.2.12	<i>Single-hidden Layer Neural Networks for Forecasting Intermittent Demand</i>	19
2.2.13	<i>Bayesian Dynamic Modeling and Forecasting of Count Time Series</i>	19
2.2.14	<i>Forecasting the Forecastability Quotient for Inventory Management</i>	20
2.2.15	<i>Performance of State Space and ARIMA Models for Consumer Retail Sales Forecasting</i>	20
BAB III METODE PENELITIAN		27
3.1	Kerangka Rencana Penelitian.....	27
3.2	Objek Penelitian.....	28
3.3	Jenis Data Penelitian.....	28
3.4	Metode Pengumpulan Data.....	28
3.5	Metode Pengolahan Data.....	29
3.5.1	<i>Data Preprocessing Machine Learning</i>	29
3.5.2	<i>Pemodelan Auto-Regressive Integrated Moving Average (ARIMA)</i>	29
3.5.3	<i>Long Short-term Memory (LSTM)</i>	29
3.5.4	Pengujian Error.....	30
3.6	Analisis dan Pembahasan.....	30
3.7	Kesimpulan dan Saran.....	31
3.8	Flowchart Kerangka Penelitian.....	31
BAB IV PENGUMPULAN DAN PENGOLAHAN DATA		32
4.1	Pengumpulan Data.....	32
4.2	<i>Preprocessing Data</i>	33
4.2.1	<i>Data Integration</i>	33
4.2.2	<i>Data Cleaning</i>	34
4.2.3	Pemisahan Data.....	40
4.2.4	Penjumlahan Harian.....	41
4.3	<i>Autoregressive Integrated Moving Average (ARIMA)</i>	42
4.3.1	Pemeriksaan Kestasioneran Data.....	43
4.3.2	Plotting ACF dan PACF.....	47
4.3.3	Pemodelan ARIMA.....	48
4.3.4	Perhitungan Nilai Error ARIMA.....	48
4.4	<i>Long Short-Term Memory (LSTM)</i>	49
4.4.1	Persiapan Data.....	50
4.4.2	Pembelajaran Data.....	50
4.4.3	Pemodelan LSTM.....	52
4.4.4	Perhitungan Nilai Error LSTM.....	54
BAB V HASIL DAN PEMBAHASAN		55
5.1	<i>Analisis Preprocessing Data</i>	55

5.1.1 Analisis <i>Data Integration</i>	55
5.1.2 Analisis <i>Data Cleaning</i>	56
5.1.3 Analisis Pemisahan Data.....	58
5.1.4 Analisis Penjumlahan Harian	58
5.2 Analisis Pemodelan ARIMA	59
5.2.1 Analisis Pemeriksaan Kestasioneran Data	59
5.2.2 Analisis Plotting ACF dan PACF	59
5.2.3 Analisis Pemodelan ARIMA.....	60
5.3 Analisis Pemodelan LSTM.....	62
5.4 Analisis Performa Metode	65
BAB VI PENUTUP	69
6.1 Kesimpulan.....	69
6.2 Saran	71
DAFTAR PUSTAKA	72



DAFTAR TABEL

Tabel 1.1 Karakteristik Ulasan Penelitian.....	22
Tabel 4.1 Plotting ACF dan PACF.....	47
Tabel 5.1 Perbandingan RMSE ARIMA.....	61
Tabel 5.2 Perbandingan RMSE LSTM.....	63
Tabel 5.1 <i>Forecast</i> Dataset 1 (KG) ARIMA.....	66
Tabel 5.2 <i>Forecast</i> Dataset 2 (PCS) ARIMA.....	67



DAFTAR GAMBAR

Gambar 1.1 Pengaruh Jumlah Stok <i>Perishable Product</i> Terhadap Pemasukan.....	2
Gambar 3.1 <i>Flowchart</i> Alur Penelitian.....	31
Gambar 4.1 Data Historis Bulan Januari 2019	32
Gambar 4.1 Data Historis Bulan Januari 2019	32
Gambar 4.2 <i>Pseudocode Data Integration</i>	33
Gambar 4.3 Hasil Penggabungan Data	34
Gambar 4.4 <i>Pseudocode</i> Penghapusan Data Kosong	34
Gambar 4.5 Penghapusan Data Kosong	35
Gambar 4.6 <i>Pseudocode</i> Penghapusan <i>Non-perishable</i>	35
Gambar 4.7 Daftar Nama Produk	35
Gambar 4.8 Data <i>Perishable Product</i>	36
Gambar 4.9 <i>Pseudocode</i> Penggantian Nama Produk.....	36
Gambar 4.10 Penamaan Ulang Data	37
Gambar 4.11 <i>Pseudocode</i> Visualisasi Setiap Jenis Barang	37
Gambar 4.12 Data Transaksi Kosong	38
Gambar 4.13 Hasil Penghapusan Data Transaksi Kosong	38
Gambar 4.14 <i>Pseudocode</i> Pemberian ID	39
Gambar 4.15 Hasil Pemberian ID	39
Gambar 4.16 <i>Pseudocode</i> Penyamaan Satuan	40
Gambar 4.17 Hasil Penyamaan Satuan	40
Gambar 4.18 <i>Pseudocode</i> Pemisahan Dataset KG dan PCS	40
Gambar 4.19 <i>Pseudocode</i> Penjumlahan Harian.....	41
Gambar 4.20 Hasil Pengelompokkan	41
Gambar 4.21 Hasil Pengurutan Tanggal Data	42
Gambar 4.22 <i>Pseudocode</i> Penggunaan Modul ARIMA.....	42
Gambar 4.23 <i>Pseudocode</i> Visualisasi Dataset.....	44
Gambar 4.24 Grafik Permintaan Dataset 1 (KG)	44
Gambar 4.25 Grafik Permintaan Dataset 2 (PCS)	44
Gambar 4.26 <i>Pseudocode</i> Visualisasi Stasioner	45
Gambar 4.27 Visualisasi Pengecekan Stasioner Dataset 1 (KG).....	45
Gambar 4.28 Visualisasi Pengecekan Stasioner Dataset 1 (PCS)	46
Gambar 4.29 <i>Pseudocode</i> Dickey Fuller Test	46
Gambar 4.30 Hasil Pengaplikasian ADF pada Dataset 1 (KG).....	47
Gambar 4.31 Hasil Pengaplikasian ADF pada Dataset 2 (PCS).....	47
Gambar 4.32 <i>Pseudocode</i> Memunculkan ACF dan PACF.....	48
Gambar 4.33 <i>Pseudocode</i> Pemodelan ARIMA	48
Gambar 4.34 <i>Pseudocode</i> RMSE ARIMA	49
Gambar 4.35 <i>Pseudocode</i> Penggunaan Modul ARIMA.....	49
Gambar 4.36 <i>Pseudocode</i> Persiapan Data	50
Gambar 4.37 <i>Pseudocode</i> Persiapan Pembelajaran Data.....	51

Gambar 4.38 Pseucode Penentuan Jumlah Pembelajaran Data	51
Gambar 4.39 Urutan Prediksi Bahan Pembelajaran Dataset 2 (PCS).....	51
Gambar 4.40 Pseucode Penghapusan ID dan Pembelajaran Data	52
Gambar 4.41 Pseucode Pemrosesan LSTM.....	53
Gambar 4.42 Pseucode Memunculkan Grafik LSTM	53
Gambar 4.43 Pseucode RMSE LSTM.....	54
Gambar 5.1 Data Tidak Lengkap.....	56
Gambar 5.2 Hasil Plotting ACF dan PACF Dataset 1 (KG)	60
Gambar 3.3 Hasil Plotting ACF dan PACF Dataset 2 (PCS)	60
Gambar 5.4 Hasil Pembelajaran Data ARIMA pada Dataset 1 (KG)	62
Gambar 5.5 Hasil Pembelajaran Data ARIMA pada Dataset 2 (PCS)	62
Gambar 5.4 <i>Learning Curve</i> Dataset 1 (KG).....	64
Gambar 5.5 <i>Learning Curve</i> Dataset 2 (PCS)	64

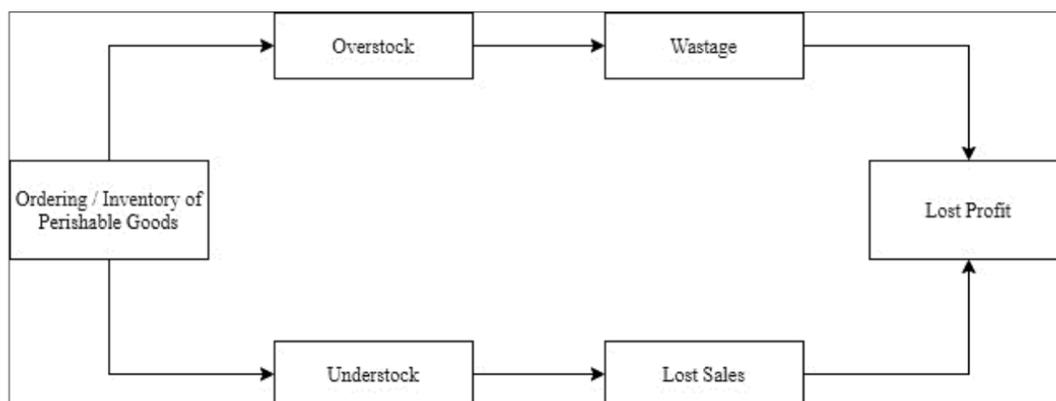


BAB I

PENDAHULUAN

1.1 Latar Belakang

Produk yang mudah rusak atau biasa disebut dengan *perishable product* memiliki siklus hidup yang singkat akibat dari keterbatasan waktu simpan produk tersebut (Tekin & Erol, 2017), contohnya seperti buah dan sayuran (Raghav et al., 2016). Ketahanan produk merupakan masalah penting bagi perusahaan industri tertentu karena produk yang mudah rusak apabila tidak dimanajemen dengan baik pada setiap langkah rantai pasokan dari produksi hingga klien akan menimbulkan biaya sistem yang tinggi, termasuk biaya pemesanan, biaya kekurangan, biaya penanganan persediaan, dan biaya rugi akibat produk yang sisa (Azadeh et al., 2017). Selain itu, kesegaran produk akan mempengaruhi kepuasan pelanggan seiring dengan nilai atau kualitas produk yang menurun (Wang et al., 2016), dengan demikian dapat dikatakan bahwa peramalan penjualan yang akurat dan tepat waktu mempengaruhi jumlah profit yang didapat dan manajemen stok yang lebih baik (Arunraj & Ahrens, 2015). Hal serupa juga disebutkan pada penelitian lain dimana stok yang tidak dikelola dengan baik dapat menimbulkan kerugian akibat dari adanya sisa stok yang tidak terjual maupun kekurangan stok yang mengakibatkan hilangnya kesempatan penjualan (Pandey et al., 2012), sehingga penentuan jumlah stok *perishable* produk menjadi penting untuk dipertimbangkan.



Gambar 1.1 Pengaruh Jumlah Stok *Perishable Product* Terhadap Pemasukan

Prediksi penjualan makanan memiliki beberapa tantangan, yaitu pola musim dan kebiasaan konsumen yang bersifat acak sehingga berakibat terjadinya beberapa kali puncak permintaan yang perlu untuk diprediksi dibanding dengan keakuratan sisa musim (Zliobaite, 2012). Namun, penentuan jumlah pesanan *perishable* produk pada retail sering kali didasari oleh pengalaman pemilik atau manager retail tersebut (Syntetos et al., 2010), hal ini dapat dipercaya jika keahlian peramal memadai dan kredibilitas sistem peramalan terpenuhi (Alvarado-Valencia, 2017). Pada Retail Sayur Tosaga, peramalan permintaan dilakukan oleh pemilik retail dalam melakukan pemesanan produk. Selain itu juga terdapat supplier yang menitipkan barangnya dan melakukan peramalan berdasarkan *expert judgment* untuk menentukan jumlah produk yang akan dititipkan. Seringkali perkiraan jumlah permintaan ini tidak sesuai dengan yang diprediksi sehingga terdapat produk sisa dan jarang terjadi kukurangan produk. Keakuratan peramalan dapat ditingkatkan dengan menggunakan lebih sedikit sumber daya untuk meningkatkan efisiensinya (Gilliland, 2011), yaitu dengan mempertimbangkan kompleksitas sistem dalam pemrosesan dan pemodelan peramalan (Ali et al., 2009). Sistem peramalan memerlukan pengkajian dalam identifikasi metode peramalan, pemilihan metode terbaik sesuai dengan masalah yang ada, serta evaluasi dan pemilihan metode yang sesuai (Hyndman & Athanasopoulos, 2018). Dengan demikian, setiap objek peramalan memiliki metode terbaik yang dapat diaplikasikan dalam melakukan peramalan.

Jumlah permintaan *perishable* produk dipengaruhi oleh berbagai faktor sehingga menghasilkan bentuk data berupa *time series*. Peramalan dengan menggunakan pola data

time series perlu untuk diperhatikan karena memiliki potensi peramalan yang luas (Zhang et al., 2017), penggunaan *machine learning* telah diusulkan dalam literatur akademik sebagai alternatif dalam peramalan *time series* (Makridakis et al., 2018). *Machine learning* digunakan untuk analisis data dan pembentukan pola data sehingga dapat digunakan untuk keperluan *data mining* (Bose & Mahapatra, 2001), dengan demikian *machine learning* dapat memiliki akurasi yang tinggi dalam menentukan peramalan suatu permintaan. Berdasarkan penjabaran sebelumnya, ditemukan sebuah masalah dalam melakukan peramalan *perishable* produk yaitu tingginya resiko kerugian apabila peramalan tidak dilakukan dengan akurat.

Penelitian ini ditujukan untuk membuat model permintaan *perishable product* pada Retail Sayur Tosaga. Pemodelan dilakukan menggunakan dua metode, yaitu *Autoregression Integrated Moving Average* (ARIMA) dan *Long Short-term Memory* (LSTM). Kedua metode ini dipilih karena merupakan metode yang sering digunakan untuk melakukan prediksi, hal ini dibuktikan dari banyaknya peneliti yang menggunakan metode tersebut dalam melakukan prediksi multi-item. ARIMA merupakan model prediksi yang sangat baik ketepatannya untuk peramalan jangka pendek dan untuk data *time series non stasioner pada saat linier* (Munarsih, 2011). Namun, ARIMA tidak mampu memodelkan *time series yang nonlinier* (Zhang, 2003), sehingga akan menurunkan tingkat keakuratannya apabila terdapat pola nonlinier didalamnya. Sedangkan LSTM merupakan pengembangan dari *recurrent neural network* (RNN), metode ini memiliki kemampuan melatih data secara efektif dengan cara memperbarui bobot secara cepat dengan meminimalisasi gradient hilang yang mana tidak diakomodasi oleh RNN (Brownlee, 2017). Setelah dilakukan pemodelan, akan dilakukan uji tingkat eror untuk membandingkan performa metode pemodelan sehingga akan didapatkan suatu metode terbaik dalam studi kasus permintaan *perishable product* Retail Sayur Tosaga.

1.2 Rumusan Masalah

Rumusan masalah pada penelitian ini adalah bagaimana model prediksi multi-item berbasis *machine learning*, dengan menggunakan metode ARIMA dan LSTM untuk *perishable product* di Retail Sayur Tosaga?

1.3 Batasan Masalah

Adapun batasan-batasan pada penelitian yang dilakukan di Retail Sayur Tosaga adalah sebagai berikut:

1. Objek penelitian yang digunakan hanya *perishable* produk di Retail Sayur Tosaga
2. Data historis yang digunakan adalah data pada bulan Januari 2019 - April 2020
3. Pemodelan hanya dilakukan pada produk yang selalu tersedia dan tidak musiman
4. Pemodelan hanya dilakukan berdasarkan data historis penjualan tanpa dipengaruhi faktor lain seperti harga dan musim

1.4 Tujuan Penelitian

Tujuan dari penelitian ini adalah menghasilkan model prediksi multi-item berbasis *machine learning* dengan menggunakan metode ARIMA dan LSTM sehingga mengurangi ketidakpastian dalam melakukan peramalan produk.

1.5 Manfaat Penelitian

Hasil penelitian ini diharapkan dapat memberikan manfaat bagi almamater, penulis, pembaca serta UKM Tosaga. Adapun manfaat penelitian yang didapat dari penelitian ini adalah sebagai berikut:

1. Bagi Keilmuan
Membantu pengembangan ilmu pengetahuan dalam kajian keilmuan dan perkembangan teknologi, serta hasil penelitian ini diharapkan dapat menjadi pembelajaran yang berguna bagi penelitian lainnya.
2. Bagi Perusahaan
Membantu perusahaan dalam menyelesaikan permasalahan yang ada yaitu dalam melakukan peramalan permintaan dan menjadi faktor pertimbangan dalam pengambilan keputusan.

1.6 Sistematika Penulisan

Sistematika penulisan tugas akhir ini dibagi menjadi enam bab, masing-masing memiliki pembahasan yang berbeda. Pada Bab I atau pendahuluan, dilakukan penentuan masalah sebagai dasar dilakukannya penelitian ini. Tahapan yang dilakukan diantaranya adalah menentukan subjek penelitian, kemudian dilakukan pengamatan untuk mengidentifikasi masalah yang ada. Apabila telah menemukan masalah, tahapan selanjutnya adalah merumuskan masalah serta menentukan batasan penelitian yang akan dilakukan.

Pada Bab II atau kajian literatur, dilakukan untuk mendapatkan landasan teoritik sebagai pedoman dalam melakukan penelitian. Tahapan yang dilakukan diantaranya adalah melakukan kajian deduktif dan induktif untuk mengetahui dasar teori yang akan digunakan dalam penelitian.

Bab III berisi metode penelitian, pada bab ini dilakukan pembahasan mengenai alur penelitian dari awal hingga akhir. Tahapan dimulai dari pengumpulan data, kemudian ditentukan instrumen penelitian yang akan digunakan dalam pemrosesan data.

Bab IV merupakan pengumpulan dan pengolahan data. Pada bab ini dilakukan pelatihan data (*data training*) menggunakan metode-metode yang telah terpilih. Kemudian dilakukan prediksi berdasarkan hasil dari data yang telah dilatih atau disebut juga dengan *data testing*. Jika pada setiap metode telah melalui kedua proses tersebut maka dilakukan analisis eror untuk mengetahui besaran tingkat kesalahannya.

Bab V berisi pembahasan berupa perbandingan besaran tingkat eror pada setiap metode untuk mengetahui metode mana yang paling baik digunakan UKM Tosaga. Selain itu juga akan dianalisis mengenai karakteristik metode-metode yang digunakan berdasarkan hasil implementasi pada data *time series* UKM Tosaga.

Bab VI atau penutup. Pada bab ini berisi kesimpulan dari penelitian, yaitu hasil metode terpilih yang paling optimal untuk digunakan beserta dengan alasannya. Kemudian akan dipaparkan saran untuk penelitian yang dapat dilakukan selanjutnya.

BAB II

KAJIAN LITERATUR

Pada bagian ini akan dijelaskan mengenai landasan teori yang digunakan dalam melakukan penelitian. Selain itu, pada bab ini juga akan dilakukan kajian empiris mengenai penelitian-penelitian sebelumnya yang sudah pernah dilakukan dan serupa dengan penelitian penulis untuk menentukan state of the art penelitian.

2.1 Kajian Literatur Deduktif

Kajian deduktif merupakan kajian teoritis yang akan dijadikan landasan teori sebagai acuan dalam memecahkan masalah penelitian. Kajian ini ditujukan untuk membangun konseptual yang bersifat umum dalam penelitian.

2.1.1 *Perishable Product*

Perishable product adalah produk yang tidak dapat dipakai lagi setelah jangka waktu tertentu. Masa penggunaan *perishable product* hanya selama kualitas produk masih bagus dan tergantung pada waktu bertahan produk tersebut. Jika waktu bertahan telah dilampaui maka produk tidak dapat lagi digunakan dan harus dibuang, contoh produk yang sering ditemui adalah makanan, darah, dan obat (Prastacos, 1981), dengan demikian produk akan mengalami penurunan kualitas seiring dengan pengurangan waktu umurnya yang mengakibatkan turunnya nilai produk tersebut (Dewi et al, 2018).

Umumnya *perishable product* yang dijual memiliki panjang umur tidak lebih dari 14 hari, hal ini biasanya diakibatkan oleh kerusakan mikroba serta terjadinya proses biomekanikal seperti reaksi perubahan enzim. Terdapat produk yang termasuk dalam *semi perishable product*, yaitu produk yang memiliki panjang umurnya hingga 6 bulan, contohnya keju dan makanan beku. Sedangkan untuk produk yang memiliki panjang umur stabil atau *non-perishable* produk memiliki panjang umur lebih dari 6 bulan hingga 3 tahun selama dalam masa penyimpanan (Labuza, 2000).

2.1.2 *Time series technique*

Time series adalah kumpulan nilai yangurut dan didokumentasikan dalam suatu waktu tertentu. Tujuan dari pengumpulan data ini adalah untuk memperoleh struktur serta pola data yang akan diamati dan dapat juga digunakan untuk mengaplikasikan suatu model pada data sehingga dapat dibuat suatu prediksi di masa mendatang. Analisis *time series* dapat digunakan dalam berbagai hal, seperti peramalan ekonomi dan pengontrolan kualitas. Hal ini dikarenakan nilai yang ada pada *time series* didokumentasikan secara *real time* dan memuat data-data rekaman masa lalu yang dapat digunakan untuk analisis-analisis penting. Terdapat 3 bentuk uraian *time series*, diantaranya adalah (Cleveland and Tiao, 1976):

1. Trend

Pola data trend merupakan pola umum yang ditunjukkan selama periode pengamatan tanpa mempertimbangkan pola lain seperti seasonal dan noise.

2. Seasonal

Pada pola data ini terdapat fluktuasi nilai yang berkala dalam waktu tertentu, hal ini menunjukkan adanya kestabilan data seiring waktu.

3. Noise

Noise merupakan data yang tersisa, yaitu bagian yang tidak dipakai karena pola datanya tidak dapat dijelaskan dalam pola data trend maupun seasonal akibat perbedaan data yang cukup jauh.

Analisis *time series* dibagi menjadi dua, yaitu univariat dan multivariat. *Time series univariate* berisi pengamatan tunggal dari waktu ke waktu, sedangkan *time series*

multivariate digunakan ketika terdapat lebih dari satu aspek yang saling terkait dan perlu dipertimbangkan dalam menganalisis data *time series* tersebut (Deb, 2017). Suatu data dapat memiliki pola yang berbeda-beda sehingga akan menghasilkan analisis yang berbeda pula.

2.1.3 *System modelling*

Pemodelan sistem atau *system modelling* merupakan tindakan pendekatan terhadap suatu model, yaitu skema teoritis dalam menyederhanakan suatu sistem nyata yang kompleks dengan tujuan agar lebih mudah untuk dipahami. Dengan demikian sebuah sistem nyata dapat dijelaskan, dikontrol, disimulasikan, diprediksi, serta ditingkatkan (Cassilas et al., 2003). Pemodelan sistem memiliki peran yang penting dalam bidang komunikasi, sistem pakar, serta pengontrolan dikarenakan tujuan dari pemodelan itu sendiri yaitu memodelkan operasi sistem dari input menjadi output yang terukur. Melalui model sistem yang telah disimulasikan, pengamat dapat lebih mudah memahami sifat yang mendasari suatu sistem dan memberikan penanganan dengan benar, terlebih saat sistem yang ada merupakan sistem yang kompleks dan nonlinier (Lee and Ouyang, 2003).

Model yang berbeda dapat digabungkan untuk membentuk suatu sistem, namun perlu pertimbangan masing-masing model ini memiliki entitas yang berbeda. Pengembangan dan penggunaan model harus mengikuti skema pemodelan, contohnya pemodelan yang berorientasi pola. Menguji dan menggunakan model biasanya akan ditujukan untuk reformulasi model hingga model dirasa cukup baik untuk digunakan. Langkah-langkah pemodelan adalah sebagai berikut (EFSA Panel, 2014):

1. Definisi masalah
2. Perencanaan konseptual model
3. Pembuatan model
4. Pengimplementasian model
5. Analisis model
6. Penggunaan model
7. Evaluasi model

2.1.4 *Data Mining*

Data mining berasal dari kata penambangan, yaitu proses penambangan data dari analisis hingga mendapatkan suatu pengetahuan baru (Han et al., 2011). *Data mining* memiliki tujuh fase, dimana empat fase awal digunakan untuk proses *pre-processing* dan tiga lainnya digunakan untuk mendapatkan informasi dari data yang telah diproses. Ketujuh fase tersebut adalah (Agarwal, 2013):

1. *Data cleansing*

Data cleansing digunakan untuk memisahkan dan menghilangkan data yang tidak diperlukan serta data yang tidak konsisten pada inputan data.

2. *Data integration*

Integrasi data digunakan untuk mengintegrasikan atau menggabungkan data yang diperoleh dari sumber yang berbeda.

3. *Data warehouse*

Data warehouse merupakan tempat penyimpanan semua data yang telah melalui proses *data cleansing* dan *data integration* sehingga data telah bersih dan terintegrasi.

4. *Data selection*

Tahap ini digunakan untuk memilih data mana yang cocok untuk dilakukan penambangan data.

5. *Data transformation*

Transformasi data digunakan untuk mengubah data menjadi format lain sesuai dengan format penggalian data.

6. *Data mining*

Proses ini merupakan proses penambangan data yang digunakan untuk menghasilkan suatu pengetahuan baru menggunakan metode-metode yang sesuai.

7. Evaluasi

Tahap ini dilakkan evaluasi terhadap hasil pengetahuan atau pola yang telah dihasilkan, data akan disajikan dalam format yang mudah dipahami oleh pembaca.

2.1.5 *Machine learning*

Machine learning mengacu pada bidang ilmu komputer dimana akan terjadi proses pembelajaran untuk memahami input yang tidak diketahui sebelumnya, hasilnya adalah akan menjadi sebuah pola yang dapat dimengerti. *Machine learning* termasuk dalam *artificial intelligence* dan statistik sehingga dapat memproses algoritma pembelajaran dengan jumlah informasi yang banyak. Seperti halnya otak manusia, *machine learning* juga dapat memproses hal-hal seperti mengidentifikasi gambar serta suara, mengenali gambit, dan mengoptimasi suatu strategi (Schuld et al., 2015).

Machine learning menggunakan teori statistika dalam membangun model matematis, hal ini dikarenakan fungsi utamanya adalah membuat sebuah kesimpulan dari sampel input yang tersedia. Komputer memiliki dua peran dalam *machine learning*, yang pertama yaitu melakukan *training* atau pelatihan, pada proses ini diperlukan algoritma yang efisien untuk dapat mengoptimasi suatu masalah, dalam hal ini merupakan proses penyimpanan dan pemrosesan data yang bersifat umum dan berjumlah banyak. Sedangkan yang kedua adalah ketika sebuah model telah dipelajari, model perlu ditampilkan dengan cara yang juga harus efisien. Yang menjadi parameter efisien disini adalah dari segi kompleksitas serta waktu pemrosesan namun tetap memperhatikan tingkat keakuratan (Alpaydin, 2020).

Saat ini *machine learning* mulai banyak digunakan, hal ini dikarenakan *machine learning* dapat diimplementasikan dalam berbagai bidang seperti penemuan pengetahuan dalam basis data, pemrosesan bahasa, control robot, optimasi, dan masalah-masalah umum seperti pengenalan suara, pengenalan tulisan tangan, analisis data medis, serta permainan game. Penggunaan *machine learning* banyak pula diterapkan pada masalah-masalah yang baru (Dietterich, 1997).

2.1.6 Pengukuran Kesalahan Peramalan

Dalam melakukan peramalan, dibutuhkan sebuah pengukuran yang berbentuk angka untuk menggambarkan performa dari hasil peramalan (Jolliffe and Stephendon, 2012). Berbagai kriteria dapat digunakan untuk mengukur kualitas peramalan, seperti pengukuran nilai eror, kecepatan dalam perhitungan, kemudahan, dan lain-lain (Tyukov et al., 2012).

Terdapat berbagai metode dalam melakukan pengukuran kesalahan model untuk peramalan, berikut merupakan metode-metode pengukuyan yang telah dirangkum oleh Prestwich dkk (2014):

2.1.7.1 *Scale dependent measures*

1. Mean [Signed] Error (ME)

$$\text{Mean} (e_t)$$

$$e_t = y_t - \hat{y}_t \quad \dots\dots\dots (1)$$

2. Mean Square Error (MSE)

$$\text{Mean} (e_t^2) \quad \dots\dots\dots (2)$$

3. Root-Mean Square Error (RMSE)

$$\sqrt{MSE} \quad \dots\dots\dots (3)$$

4. Mean Absolute Error (MAE)

$$\text{Mean} (|e_t|) \quad \dots\dots\dots (4)$$

5. Median Absolute Error (MdAE)

$$\text{Median} (|e_t|) \quad \dots\dots\dots (5)$$

Pengujian ini dapat digunakan secara bersamaan untuk mendapatkan metode terbaik dalam melakukan peramalan. Namun, MAE dikatakan kurang optimal untuk melakukan pengukuran terhadap peramalan *short-term*.

2.1.7.2 *Percentage errors*

1. Mean Absolute Percentage Error (MAPE)

$$\text{Mean} (|p_t|) \quad \dots\dots\dots (6)$$

$$p_t = \frac{100 e_t}{y_t}$$

2. Median Absolute Percentage Error (MdAPE)

$$\text{Median} (|p_t|) \quad \dots\dots\dots (7)$$

3. Root-Mean Square Percentage Error (RMSPE)

$$\sqrt{\text{mean} (p_t^2)} \quad \dots\dots\dots (8)$$

4. Root-Median Square Percentage Error (RMdSPE)

$$\sqrt{\text{median} (p_t^2)} \quad \dots\dots\dots (9)$$

5. Symmetric Mean Absolute Percentage Error (sMAPE)

$$\text{Mean} \left(\frac{200 |e_t|}{y_t - \hat{y}_t} \right) \quad \dots\dots\dots (10)$$

6. Symmetric Median Absolute Percentage Error (sMdAPE)

$$\text{Median} \left(\frac{200 |e_t|}{y_t - \hat{y}_t} \right) \quad \dots\dots\dots (11)$$

Terdapat banyak penelitian yang membandingkan metode pengukuran ini, dari hasil tersebut diketahui bahwa MAPE dan MdAPE memiliki hasil yang baik dalam melakukan pengukuran.

2.1.7.3 *Relative error based measuremet*

1. Mean Relative Absolute Error (MRAE)

$$\text{Mean} (|r_t|) \quad \dots\dots\dots (12)$$

$$r_t = \frac{e_t}{e_t^*}$$

2. Median Relative Absolute Error (MdRAE)

$$\text{Median} (|r_t|) \quad \dots\dots\dots (13)$$

2.1.7.4 *Relative measurement*

Metode pengukuran ini didapatkan dari metode dasar lainnya yaitu MAE, MSE, dan RMSE.

1. Relative Mean Absolute Error (RelMAE)

$$\frac{MAE}{MAE_b} \dots\dots\dots (14)$$

2. Relative Mean Squared Error (RelMSE)

$$\frac{MSE}{MSE_b} \dots\dots\dots (15)$$

3. Relative Root Mean Squared Error (RelRMSE)

$$\frac{RMSE}{RMSE_b} \dots\dots\dots (16)$$

Metode pengukuran ini didapatkan dari metode dasar lainnya yaitu MAE, MSE, dan RMSE.

Penggabungan metode belum tentu dapat menghasilkan hasil yang lebih baik, hal ini dibuktikan dari berbagai penelitian yang menunjukkan performa dari masing-masing pengukuran. Dengan demikian pemilihan metode pengukuran akan lebih baik didasari dengan bukti perbandingan performa yang telah diuji sebelumnya.

2.2 Kajian Literatur Induktif

Kajian induktif ini merupakan sekumpulan dari beberapa penelitian terkait penggunaan metode dalam melakukan peramalan *time series*. Dengan demikian diharapkan mampu menjadi acuan dan referensi guna memahami fokus penelitian baik metode maupun penggunaannya.

2.2.1 *From Predictive to Prescriptive Analytics: A Data-driven Multi-item Newsvendor Model*

Penelitian yang dilakukan oleh Sushil Punia, Surya Prakash Singh, Jitendra K. Madaan dan dipublikasi pada tahun 2020, yang berjudul "*From predictive to prescriptive analytics: A data-driven multi-item newsvendor model*" meneliti tentang masalah prediksi multi-item dengan suatu kendala kapasitas (Z). Kendala yang ditemukan adalah cara yang digunakan untuk memprediksi suatu permintaan hanya dengan menggunakan asumsi atau *expert judgement* sehingga terjadi kesalahan dalam pengoptimalan inventaris. Penelitian ini ditujukan untuk mengusulkan suatu prediksi dengan berdasarkan data yang ada,

pendekatan yang diusulkan dengan menggunakan data historis sebagai input dan *machine learning* untuk mengurangi resiko empiris dalam menentukan jumlah pesanan. Hierarki produk ritel dikembangkan untuk mengoptimalkan jumlah penyimpanan multi-item dengan batasan kapasitas penyimpanan tertentu. Metode perakiraan permintaan berbasis *mmachine learning* dan *deep learning* (bagian dari pendekatan yang diusulkan) memberikan hasil yang lebih baik daripada neural network, regresi berganda, arima, dan lain-lain. Hasilnya, perbandingan total biaya inventaris dari yang diusulkan, maks-min, dan metode pengoptimalan inventaris empiris dilakukan, dan diamati bahwa pendekatan berbasis data yang diusulkan mengarah pada pengurangan biaya inventaris yang signifikan.

2.2.2 Hospital Daily Outpatient Visits Forecasting Using A Combinatorial Model Based on ARIMA and SES Models

Penelitian yang dilakukan oleh Li Luo, Le Luo, Xinli Zhang, Xiaoli He dan dipublikasikan pada tahun 2017, yang berjudul "*Hospital daily outpatient visits forecasting using a combinatorial model based on ARIMA and SES models*" bertujuan untuk meramalkan kunjungan rawat jalan rumah sakit untuk merencanakan alokasi sumber daya perawatan medis yang optimal. Pada kasus yang ada, jumlah kunjungan pasien dapat dipengaruhi oleh jadwal dokter serta waktu kedatangan sehingga perlu adanya perkiraan yang baik untuk mengalokasikan sumber daya. Pada penelitian ini dilakukan pemodelan dengan menggunakan Seasonal ARIMA (SARIMA) dan *Single Exponential Smoothing* (SES) serta keduanya pada data historis pasien di dua rumah sakit besar di Chengdu, China untuk meramalkan jumlah kunjungan rawat jalan harian sekitar 1 minggu kedepan. Hasilnya adalah kedua model tunggal serta kombinasi yang dilakukan menunjukkan hasil yang lebih optimal pada model kombinasi.

2.2.3 Sales Demand Forecast in E-commerce Using a Long Short-Term Memory Neural Network Methodology

Penelitian yang dilakukan oleh Kasun Bandara, Peibei Shi, Christoph Bergmeir, Hansika Hewamalage, Quoc Tran, Brian Seaman dan dipublikasikan pada tahun 2019, yang

berjudul "*Sales Demand Forecast in E-commerce Using a Long Short-Term Memory Neural Network Methodology*" bertujuan untuk membuat perkiraan penjualan pada bisnis e-commerce yang akurat. Umumnya, peramalan dilakukan berdasarkan data historis single produk, namun pada e-commerce terdapat banyak data *time series* untuk setiap produk. Selain itu terdapat beberapa produk yang sejenis dan berhubungan dimana pola permintaan produk tersebut dapat dikorelasikan. Penelitian yang dilakukan pada penelitian ini adalah penggabungan informasi seri dan melatih dengan metode LSTM untuk mengakomodasi hubungan non-linier dalam bermacam-macam produk di e-commerce. Selain peramalan juga diusulkan kerangkakerangka *preprocessing* yang sistematis untuk menghadapi tantangan dalam bisnis e-commerce. Sumber data yang digunakan adalah dari situs Walmart.com dan hasilnya metode yang diusulkan cukup kompetitif pada tingkat kategori maupun super-departemen.

2.2.4 *DeepAR: Probabilistic Forecasting with Autoregressive Recurrent Networks*

Penelitian yang dilakukan oleh David Salinas, Valentin Flunkert, Jan Gasthaus, Tim Januschowski dan dipublikasikan pada tahun 2019, yang berjudul "*DeepAR: Probabilistic forecasting with autoregressive recurrent networks*" bertujuan untuk melakukan perkiraan probabilitas *time series* di waktu yang akan datang untuk mengoptimalkan proses bisnis. Subjek yang digunakan adalah bisnis ritel dimana probabilitas perkiraan permintaan menjadi sangat penting agar produk yang dimiliki dapat tersedia dalam waktu dan tempat yang tepat. Penelitian ini menggunakan metode DeepAR, yaitu metode yang dapat digunakan untuk memprediksi probabilitas secara akurat dengan menggunakan *autoregressive recurrent neural network* pada data yang banyak. Melalui evaluasi yang dilakukan oleh peneliti, dapat diketahui bahwa metode ini lebih baik dan tidak banyak memerlukan pengerjaan manual.

2.2.5 *CNN-LSTM Coupled Model for Prediction of Waterworks Operation Data*

Penelitian yang dilakukan oleh Kerang Cao, Hangyung Kim, Chulhyun Hwang, and Hoekyung Jung dan dipublikasikan pada tahun 2018, yang berjudul "*CNN-LSTM*

Coupled Model for Prediction of Waterworks Operation Data" berisi mengenai usulan metode untuk memprediksi permintaan air pada perusahaan air. Model prediksi statistik seperti model regresi linier berganda menunjukkan tingkat prediksi yang tidak cukup untuk untuk permintaan fluktuatif dan prediksi jangka panjang. Untuk model deep learning seperti model *long-short term memory* (LSTM) telah diterapkan, namun kurang mencerminkan korelasi antara berbagai faktor terkait, serta memerlukan model tambahan meningkatkan performa prediksi jangka panjang. Dalam penelitian ini, model *convolutional neural network* (CNN) digunakan untuk memilih berbagai variabel input yang memiliki korelasi untuk meningkatkan kecepatan prediksi jangka panjang, sehingga meningkatkan kecepatan prediksi LSTM. Selain itu, model regresi linier berganda diterapkan untuk menyusun data prediksi CNN dan LSTM, yang kemudian mengkonfirmasi data tersebut sebagai hasil prediksi akhir.

2.2.6 *The Forecasting of PM2.5 Using a Hybrid ModelBased on Wavelet Transform and an ImprovedDeep Learning Algorithm*

Penelitian yang dilakukan oleh Weibiao Qiao, Wencai Tian, Yu Tian, Quan Yang, Yining Wang, Jianzhuang Zhang dan dipublikasikan pada tahun 2019, yang berjudul "*The Forecasting of PM2.5 Using a Hybrid ModelBased on Wavelet Transform and an ImprovedDeep Learning Algorithm*" bertujuan untuk memprediksi jumlah emisi PM2.5 sebagai pertimbangan dalam membuat kebijakan peraturan terkait hal tersebut. Karena deret waktu PM2.5 non-linear menyebabkan sulit untuk memprediksi secara akurat. Meskipun beberapa penelitian tentang PM2.5 sedang diusulkan, masalah hilangnya gradien LSTM (memori jangka pendek) dan pemilihan acak urutan dan lapisan wavelet masih belum terpecahkan. Dalam studi ini, model baru berbasis WT (wavelet transform) -SAE (stacked autoencoder) -LSTM akan diusulkan. Hasilnya menyiratkan bahwa: (1) kinerja peramalan SAE-LSTM lebih baik daripada model lain (misalnya, BP (back propagation)) yang digunakan untuk perbandingan; (2) untuk enam sampel PM 2.5 yang berbeda, empat *orders* lima lapisan, lima *orders* enam lapisan, lima *orders* tujuh lapisan, tiga *orders* enam lapisan, lima *orders* tujuh lapisan, dan lima *orders* enam lapisan adalah yang paling sesuai. Kesimpulan bahwa model baru seperti itu dapat membantu meningkatkan akurasi prediksi PM 2.5.

2.2.7 *Short-term Load Forecasting by Using a Combined Method of Convolutional Neural Networks and Fuzzy Time Series*

Penelitian yang dilakukan oleh Hossein Javedani Sadaei, Petronio Candido de Lima e Silvaa, Frederico Gadelha Guimaraesa, Muhammad Hisyam Lee dan dipublikasikan pada tahun 2019, yang berjudul "*Short-term load forecasting by using a combined method of convolutional neural networks and fuzzy time series*" mengusulkan metode fuzzy time series (FTS) dan convolutional neural networks (CNN) untuk melakukan prediksi jangka pendek. Pada metode yang diusulkan, data deret waktu multi-variasi dan deret waktu versi fuzzified, diubah menjadi multi-channel untuk dilatihkan pada model CNN. Dengan demikian, model CNN yang dapat menentukan dan mengekstrak parameter penting yang terkait, secara implisit dan otomatis. Metode ini dinilai lebih praktis dibanding metode tradisional, hal ini dapat dilihat sebagai salah satu perbedaan besar antara metode yang diusulkan dan beberapa metodologi mutakhir peramalan jangka pendek. Selain itu, penggunaan logika fuzzy memiliki kontribusi besar untuk mengontrol over-fitting dengan mengekspresikan satu dimensi deret waktu dengan ruang fuzzy.

2.2.8 *A Hybrid Neural Network Model for Sales Forecasting Based on ARIMA and Search Popularity of Article Titles*

Penelitian yang dilakukan oleh Hani Omar, Van Hai Hoang, and Duen-Ren Liu dan dipublikasikan pada tahun 2016, yang berjudul "*A Hybrid Neural Network Model for Sales Forecasting Based on ARIMA and Search Popularity of Article Titles*" bertujuan untuk memprediksi penjualan suatu majalah berdasarkan data historis yang dipengaruhi oleh judul yang diangkat. *Backpropagation Neural Networks* (BPNN) telah berhasil digunakan untuk mengembangkan model prediksi untuk perkiraan penjualan. Dalam studi ini, diusulkan model jaringan saraf tiruan baru untuk peramalan penjualan berdasarkan hasil prediksi peramalan deret waktu dan popularitas judul artikel. Model yang diusulkan menggunakan data penjualan historis, popularitas judul artikel, dan hasil prediksi dari suatu time series, metode peramalan Autoregressive Integrated Moving Average (ARIMA) untuk mempelajari model peramalan berbasis BPNN. Model perkiraan dievaluasi secara eksperimental dengan membandingkan dengan teknik

prediksi penjualan konvensional. Hasil percobaan menunjukkan bahwa metode peramalan yang diusulkan mengungguli teknik konvensional.

2.2.9 *Travel Time Prediction with LSTM Neural Network*

Penelitian yang dilakukan oleh Yanjie Duan, Yisheng Lv and Fei-Yue Wang dan dipublikasikan pada tahun 2016, yang berjudul "*Travel Time Prediction with LSTM Neural Network*" meneliti tentang model pembelajaran mendalam, model jaringan saraf LSTM, untuk prediksi waktu tempuh pada industri travel. Waktu perjalanan adalah salah satu perhatian utama diantara para pelancong sebelum memulai perjalanan dan juga merupakan indikator penting dari kondisi lalu lintas. Namun, waktu tempuh biasanya tidak teratur. Dengan menggunakan data waktu perjalanan yang disediakan oleh Highways England, dibuat 66 jaringan saraf LSTM prediksi seri untuk 66 tautan dalam kumpulan data. Melalui pelatihan dan validasi model, didapatkan struktur optimal dalam rentang pengaturan untuk setiap tautan. Melalui percobaan ini, model *deep learning* yang mempertimbangkan relasi urutan cukup menjanjikan dalam prediksi data deret lalu lintas.

2.2.10 *Forecasting of Particulate Matter Time Series using Wavelet Analysis and Wavelet-ARMA/ARIMA Model in Taiyuan, China*

Penelitian yang dilakukan oleh Hong Zhang, Sheng Zhang, Ping Wang, Yuzhe Qin & Huifeng Wang dan dipublikasikan pada tahun 2017, yang berjudul "*Forecasting of particulate matter time series using wavelet analysis and wavelet-ARMA/ARIMA model in Taiyuan, China*" mengusulkan model wavelet-ARMA / ARIMA untuk meramalkan rangkaian jangka pendek konsentrasi PM10. Model dievaluasi dengan eksperimen menggunakan kumpulan data 10 tahun konsentrasi PM10 harian dari 4 stasiun yang berlokasi di Taiyuan, Cina. Dibandingkan dengan metode ARMA / ARIMA tradisional, metode ARMA / ARIMA wavelet ini secara efektif dapat mengurangi kesalahan peramalan, meningkatkan akurasi prediksi, dan mewujudkan prediksi skala waktu ganda. Model yang diusulkan dapat secara efisien dan berhasil diterapkan pada bidang peramalan PM10.

2.2.11 Cluster-based Hierarchical Demand Forecasting for Perishable Goods

Penelitian yang dilakukan oleh Jakob Huber, Alexander Gossmannb, Heiner Stuckenschmidt dan dipublikasikan pada tahun 2017, yang berjudul "*Cluster-based hierarchical demand forecasting for perishable goods*" bertujuan untuk melakukan peramalan permintaan suatu retail yang menjual *perishable product* dimana *demand underestimation* dan *overestimation* berdampak negatif terhadap pendapatan. Diusulkan DSS yang mendukung operasi sehari-hari dengan memberikan perkiraan hierarki pada tingkat organisasi yang berbeda berdasarkan data titik penjualan terbaru. Model ARIMA multivariat digunakan memperkirakan permintaan harian untuk mendukung keputusan operasional. Evaluasi pendekatan tersebut didapat dengan data titik penjualan dari rantai toko roti industri dan menunjukkan bahwa dimungkinkan untuk meningkatkan ketersediaan sambil membatasi kerugian pada saat yang bersamaan.

2.2.12 Single-hidden Layer Neural Networks for Forecasting Intermittent Demand

Penelitian yang dilakukan oleh F. Lolli, R. Gamberini, A. Regattieri, E. Balugani, T. Gatos, S. Gucci dan dipublikasikan pada tahun 2017, yang berjudul "*Single-hidden layer neural networks for forecasting intermittent demand*" meneliti mengenai penggunaan *extreme learning machine* untuk mengelola item dalam jumlah besar, dan mengoptimalkan jaringan pada level item untuk menyesuaikan parameter selama tahap pelatihan. Dalam penelitian ini, jaringan saraf yang dilatih oleh *backpropagation* dan *extreme learning machine* dibandingkan dengan *neural networks*, serta metode prakiraan standar untuk permintaan intermiten pada rangkaian waktu nyata, dengan menggabungkan pola input dan arsitektur yang berbeda. Akhirnya, beberapa wawasan bagi praktisi disajikan untuk meningkatkan potensi jaringan saraf untuk diterapkan di lingkungan nyata.

2.2.13 Bayesian Dynamic Modeling and Forecasting of Count Time Series

Penelitian yang dilakukan oleh Lindsay R. Berry dan dipublikasikan pada tahun 2019, yang berjudul "*Bayesian Dynamic Modeling and Forecasting of Count Time Series*"

bertujuan untuk menghasilkan prakiraan multi-step secara online untuk ribuan rangkaian individu dengan cara yang efisien dan fleksibel. Disertasi ini memperkenalkan kelas model baru untuk mengatasi efisiensi dan keakuratan model dinamis berdasarkan konsep decouple / recouple yang diterapkan pada beberapa seri yang secara individual diwakili melalui model ruang keadaan univariat baru. Model multivariansi baru kemudian memungkinkan berbagi informasi dalam konteks dimana data pada tingkat yang lebih teragregasi memberikan kesimpulan yang lebih tajam tentang pola bersama seperti tren dan musim. Beberapa studi kasus di banyak item, prakiraan penjualan supermarket multi-step menunjukkan peningkatan kinerja peramalan menggunakan model yang diusulkan, dengan diskusi metrik akurasi prakiraan dan manfaat penilaian akurasi prakiraan probabilistik.

2.2.14 Forecasting the Forecastability Quotient for Inventory Management

Penelitian yang dilakukan oleh Arthur V. Hill, Weiyong Zhang, Gerald F. Burch dan dipublikasikan pada tahun 2015, yang berjudul "*Forecasting the forecastability quotient for inventory management*" mengembangkan dan menguji secara empiris model untuk mengestimasi keuntungan ekonomi menggunakan sistem *time phased order point* (TPOP) dengan peramalan time series dibanding dengan sistem pemesanan ulang sederhana dalam konteks manajemen inventaris permintaan independen. Ditetapkan definisi hasil bagi perkiraan (Q) untuk mendukung analisis ekonomi tersebut. TPOP diterapkan dalam analisis empiris melalui *double exponential smoothing* dengan tren dan menerapkan ROP melalui rata-rata sederhana. Penelitian ini memberikan kontribusi baik untuk literatur penelitian maupun untuk manajer yang perlu memutuskan apakah item permintaan independen harus dikelola dengan TPOP atau sistem titik pemesanan ulang.

2.2.15 Performance of State Space and ARIMA Models for Consumer Retail Sales Forecasting

Penelitian yang dilakukan oleh Patrícia Ramos, Nicolau Santos, Rui Rebelo dan dipublikasikan pada tahun 2015, yang berjudul "*Performance of state space and ARIMA models for consumer retail sales forecasting*" bertujuan untuk membandingkan kinerja

state space model dan model ARIMA melalui studi kasus penjualan ritel dari lima kategori alas kaki wanita yang berbeda: Boots, Booties, Flats, Sandals dan Shoes. Pada kedua metodologi, model dengan nilai minimum Kriteria Informasi untuk periode sampel dipilih dari semua model yang dapat diterima untuk evaluasi lebih lanjut diluar sampel. Hasilnya menunjukkan bahwa ketika algoritme otomatis *state space* dan model ARIMA yang dievaluasi, disimpulkan bahwa *state space* dan ARIMA menghasilkan probabilitas cakupan yang mendekati tarif nominal untuk prakiraan suatu waktu dan multi waktu.



Tabel 2.1 Karakteristik Ulasan Penelitian

No	Penulis	Karakteristik Data					Pemodelan															Uji Nilai Error	
		Time Series	Multi-item	Stasioner	Univariate	ARIMA	DSS	Bayes	State Space	SAE	DNN	RF	FTS	ES	BPNN	LSTM	Wavelet	Autoregresi	RNN	CNN	Hibrid		
1	Sushil Punia, Surya Prakash Singh, Jitendra K. Madaan (2020)	✓	✓	✓	✓	✓					✓	✓										✗	RMSE, MAPE
2	Li Luo, Le Luo, Xinli Zhang and Xiaoli He (2017)	✓	✓	✓	✓	✓								✓								✓	MAPE
3	Kasun Bandara, Peibei Shi, Christoph Bergmeir, Hansika Hewamalage, Quoc Tran, and Brian Seaman (2019)	✓	✓	✗	✓											✓						✗	mMAPE
4	David Salinas, Valentin Flunkert, Jan Gasthaus, Tim Januschowski (2019)	✓	✓	✓	✓													✓	✓		✓		ND, RMSE

No	Penulis	Karakteristik Data				Pemodelan															Uji Nilai Error
		Time Series	Multi-item	Stasioner	Univariate	ARIMA	DSS	Bayes	State Space	SAE	DNN	RF	FTS	ES	BPNN	LSTM	Wavelet	Autoregresi	RNN	CNN	
5	Kerang Cao, Hangyung Kim, Chulhyun Hwang, and Hoekyung Jung (2018)	✓	✓	✓	✓											✓			✓	✓	RMSE
6	Weibiao Qiao, Wencai Tian, Yu Tian, Quan Yang, Yining Wang, Jianzhuang Zhang (2019)	✓	✗	✗	✓					✓						✓				✓	MAE
7	Hossein Javedani Sadaei, Petronio Candido de Lima e Silvaa, Frederico Gadelha Guimaraesa, Muhammad Hisyam Lee (2019)	✓	✗	✓	✗							✓							✓	✗	MAPE
8	Hani Omar, Van Hai Hoang, and	✓	✓	✓	✓	✓										✓				✓	RMSE

No	Penulis	Karakteristik Data								Pemodelan											Uji Nilai Error
		Time Series	Multi-item	Stasioner	Univariate	ARIMA	DSS	Bayes	State Space	SAE	DNN	RF	FTS	ES	BPNN	LSTM	Wavelet	Autoregresi	RNN	CNN	
9	Duen-Ren Liu (2016) Yanjie Duan, Yisheng Lv and Fei-Yue Wang (2016)	✓	✗	✓	✓										✓					✗	MAE, RMSE, MRE
10	Hong Zhang, Sheng Zhang, Ping Wang, Yuzhe Qin & Huifeng Wang (2017)	✓	✓	✓	✗	✓										✓				✓	RMSE, NRMSE
11	Jakob Huber, Alexander Gossmannb, Heiner Stuckenschmidt (2017)	✓	✓	✓	✗	✓	✓													✓	MAPE, RMSE
12	F. Lolli, R. Gamberini, A. Regattieri, E. Balugani, T. Gatos, S. Gucci (2017)	✓	✗	✓	✓						✓			✓						✗	MAPE, MAD, ME

No	Penulis	Karakteristik Data								Pemodelan											Uji Nilai Error
		Time Series	Multi-item	Stasioner	Univariate	ARIMA	DSS	Bayes	State Space	SAE	DNN	RF	FTS	ES	BPNN	LSTM	Wavelet	Autoregresi	RNN	CNN	
13	Lindsay R. Berry (2019)	✓	✓	✓	✗			✓												✗	MAPE, WAPE, WAFE
14	Arthur V. Hill, Weiyong Zhang, Gerald F. Burch (2015)	✓	✓	✓	✗								✓							✗	MAPE, MASE
15	Patrícia Ramos, Nicolau Santos, Rui Rebelo (2015)	✓	✓	✓	✗	✓		✓												✗	RMSE, MAE, MAPE
16	Shelly Elvina Salsabila (2020)	✓	✓	✓	✓	✓									✓					✗	RMSE

Berdasarkan ulasan penelitian yang telah disajikan, dapat diketahui bahwa terdapat berbagai macam metode pemodelan yang dapat digunakan untuk melakukan peramalan. Masing-masing metode memiliki kelebihan serta kekurangan yang berbeda, sehingga terdapat beberapa penelitian yang menggunakan *hybrid method* dengan tujuan untuk menutupi kekurangan suatu metode dengan keunggulan metode lainnya. Penggunaan suatu metode perlu untuk disesuaikan dengan bentuk karakteristik data yang dimiliki sehingga perlu diketahui karakteristik data terlebih dahulu sebelum melakukan pemilihan metode untuk menyelesaikan suatu masalah.

Karakteristik data yang dimiliki oleh UKM Tosaga yaitu berupa pola data deret waktu, memiliki berbagai macam item penjualan (multi-item), serta permintaan tidak menentu sehingga grafik berbentuk linier. Selain itu juga pemodelan hanya dilakukan berdasarkan data historis penjualan tanpa mempertimbangkan faktor lain sehingga model akan bersifat *univariate*. Berdasarkan karakteristik data yang telah disebutkan, metode pemodelan yang dapat digunakan sesuai dengan ulasan yang ada adalah *Auto-Regressive Integrated Moving Average (ARIMA)*, *Deep Neural Network (DNN)*, *Exponential Smoothing*, *Auto-regression*, *Recurrent Neural Network (RNN)*, *Long Short-term Memory (LSTM)*, *Convolutional Neural Network (CNN)*, dan *Back Propagation Neural Network (BPNN)*.

Pada penelitian ini, metode yang akan diujicobakan adalah *Long Short-term Memory (LSTM)* yang merupakan bagian dari RNN serta telah mencakup BPNN. Selain itu juga akan diujikan pada *Auto-Regressive Integrated Moving Average (ARIMA)*. Kemudian model akan diukur tingkat erornya menggunakan MAPE serta RMSE paling banyak digunakan berdasarkan ulasan tersebut.

BAB III

METODE PENELITIAN

Pada bab ini akan dijelaskan mengenai metode penelitian yang akan digunakan dalam menyelesaikan masalah yang telah dirumuskan. Metode penelitian akan mencakup kerangka rencana penelitian, objek penelitian, jenis data penelitian, metode pengumpulan data, metode pengolahan data, metode analisis data, dan diagram alir penelitian.

3.1 Kerangka Rencana Penelitian

Jumlah permintaan *perishable product* yang tidak menentu menimbulkan resiko bagi UKM Tosaga, yaitu kerugian akibat adanya produk sisa yang tidak terjual maupun *opportunity lost* akibat tidak tersedianya produk. Hal ini menyebabkan peneliti ingin melakukan peramalan terkait permintaan pasar terhadap *perishable product* yang ada di UKM Tosaga. Dalam hal ini, peneliti ingin menguji coba beberapa metode pemodelan yang dapat digunakan untuk melakukan peramalan berdasarkan data permintaan yang ada. Setelah itu akan dilakukan evaluasi tingkat kesalahan pada setiap metode dan dilakukan perbandingan untuk mendapatkan metode yang paling optimal untuk diterapkan di UKM Tosaga.

Adapun data yang diperlukan yakni data historis penjualan UKM Tosaga dan waktu ketahanan produk. Data tersebut akan didapatkan dari basis data yang dimiliki oleh UKM Tosaga serta wawancara dengan narasumber terkait. Gambaran penelitian yang dilakukan secara keseluruhan akan dijelaskan pada Tabel 3.1, untuk menggambarkan objek, subjek, waktu, tempat, alasan, dan langkah penelitian.

3.2 Objek Penelitian

Objek penelitian ini adalah penentuan metode pemodelan yang optimal pada *perishable product* yang memiliki umur tidak lebih dari 14 hari, dengan demikian model dapat digunakan untuk melakukan peramalan permintaan oleh UKM Tosaga. Adapun aspek yang diukur adalah nilai eror antara model data historis *perishable product* dengan model yang dihasilkan dari masing-masing metode.

3.3 Jenis Data Penelitian

Sumber data pada penelitian ini hanya berasal dari sumber data sekunder, yaitu data yang diperoleh dari informasi berupa dokumentasi termasuk arsip lembaga yang berkaitan dengan masalah dalam penelitian (Usman et al., 2009). Data sekunder yang digunakan berupa data historis permintaan *perishable product* pada UKM Tosaga sejak bulan Januari 2019 – April 2020.

3.4 Metode Pengumpulan Data

Proses pengumpulan data yang dilakukan dalam penelitian ini adalah dengan menggunakan teknik wawancara, yaitu merupakan suatu teknik pengumpulan data dengan cara lisan terhadap responden yang dapat memberikan informasi yang relevan dengan masalah yang sedang diteliti (Nazir, 1988). Penggunaan metode wawancara bertujuan untuk memperoleh informasi berupa umur ketahanan produk, penetapan satuan berat (*pack*) untuk setiap produk, dan penjelasan produk akibat penamaan yang tidak konsisten pada data historis. Selain itu juga dilakukan studi literatur penelitian terdahulu guna mencari informasi terkait penelitian.

3.5 Metode Pengolahan Data

3.5.1 Data Preprocessing Machine Learning

Pemrosesan awal data atau *data preprocessing* adalah langkah umum di banyak aplikasi *machine learning* (Johnson, 2016). *Data preprocessing* telah diklaim oleh banyak peneliti sebagai tahap dasar metode ML dan secara signifikan memengaruhi prediksi akhir (Huang, 2015). Untuk mencapai hasil yang lebih baik dari model yang diterapkan dalam Machine Learning, format data harus sesuai. Beberapa model *machine learning* membutuhkan informasi dalam format tertentu, kumpulan data harus diformat sedemikian rupa sehingga lebih dari satu algoritma *machine learning* dan *deep learning* dapat dijalankan dalam satu kumpulan data.

3.5.2 Pemodelan Auto-Regressive Integrated Moving Average (ARIMA)

Autoregressive integrated moving average (ARIMA) adalah salah satu model linier paling populer untuk peramalan deret waktu karena sifat statistiknya yang baik dan fleksibilitas yang tinggi (Liu, 2016). Metode forecasting ARIMA merupakan metode yang sangat tepat untuk mengatasi kerumitan deret waktu dan situasi peramalan lainnya (Sutanto, 201). Model ARIMA terbagi menjadi 3 model yaitu autoregressive dengan rumus perhitungan:

$$Y_t = \theta_0 + \theta_1 Y_{t-1} + \dots + \theta_p Y_{t-p} + \varepsilon_t \quad \dots\dots (17)$$

Model moving average dengan rumus perhitungan:

$$Y_t = \mu + \varepsilon_t - \omega_1 \varepsilon_{t-1} - \dots - \omega_q \varepsilon_{t-q} \quad \dots\dots (18)$$

Model autoregressive and moving average dengan rumus perhitungan:

$$Y_t = \theta_0 + \theta_1 Y_{t-1} + \dots + \theta_p Y_{t-p} + \varepsilon_t - \omega_1 \varepsilon_{t-1} - \dots - \omega_q \varepsilon_t \quad \dots\dots (19)$$

3.5.3 Long Short-term Memory (LSTM)

Long Short-term Memory (LSTM) dapat menyelesaikan banyak tugas yang tidak dapat diselesaikan oleh pembelajaran sebelumnya, yaitu jaringan saraf berulang (RNN) (Gers, 1999).

Cara kerja LSTM adalah membuat perubahan pada RNN dengan menambahkan memory cell yang dapat menyimpan informasi untuk jangka waktu yang lama (Zaman, 2019). Dalam LSTM, terdapat sebuah *cell state* yang berfungsi sebagai memori atau ingatan untuk sebuah layer. Terdapat tiga jenis unit gerbang berbeda yang digunakan dalam LSTM, yaitu *input gate*, *forget gate*, dan *output gate*. *Input gate* berfungsi untuk menentukan sebuah masukan akan ditambahkan ke dalam memori *cell state* saat itu atau tidak. *Forget gate* berguna untuk menentukan sebuah memori pada waktu sebelumnya harus dilupakan atau tidak. Sedangkan *Output gate* berguna untuk menentukan seberapa besar pengaruh memori cell state terhadap hasil prediksi yang akan dihasilkan (Karpathy, 2015).

3.5.4 Pengujian Error

Dalam melakukan peramalan, dibutuhkan sebuah pengukuran yang berbentuk angka untuk menggambarkan performa dari hasil peramalan. Terdapat pengukuran berupa angka yang menggambarkan performa dari metode yang digunakan, nilai ini dapat digunakan untuk mengembangkan cara pelatihan data serta metode yang digunakan untuk meningkatkan keakuratan hasilnya. Dengan demikian penyusunan strategi dapat lebih tepat (Jolliffe and Stephendon, 2012). Dalam hal ini, pemodelan yang telah dilakukan akan diuji tingkat eror atau kesalahannya menggunakan *root mean square error* (RMSE).

3.6 Analisis dan Pembahasan

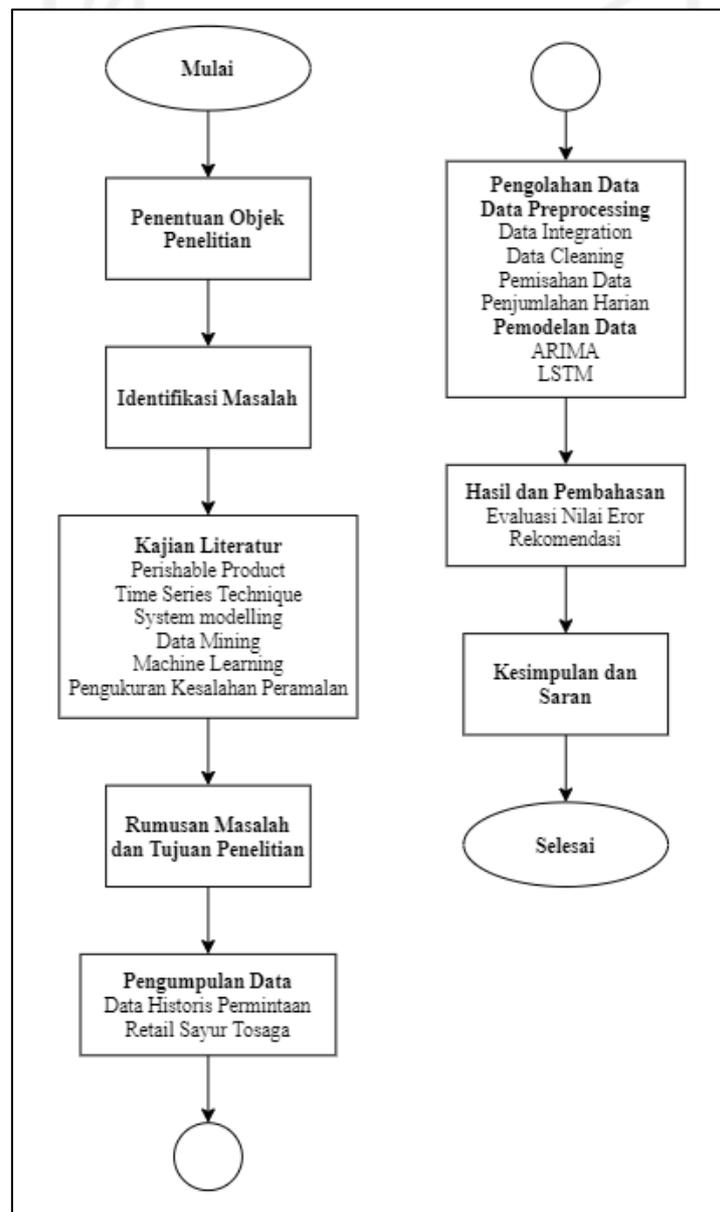
Analisis dan pembahasan dimaksudkan untuk mengkaji penggunaan metode-metode yang ada pada pola data historis. Hasil analisis dan pembahasan akan menyediakan informasi mengenai evaluasi metode-metode yang digunakan serta rekomendasi metode yang optimal untuk dapat digunakan oleh UKM Tosaga dalam melakukan prediksi permintaan pada masa yang akan datang

3.7 Kesimpulan dan Saran

Bagian ini akan membahas mengenai kesimpulan yang merupakan jawaban rumusan masalah tujuan dari penelitian ini, serta akan ada saran terkait metode atau penggunaannya untuk digunakan oleh peneliti selanjutnya maupun pada perusahaan sendiri.

3.8 Flowchart Kerangka Penelitian

Berikut merupakan alur penelitian yang akan dilakukan:



Gambar 3.1 *Flowchart* Alur Penelitian

BAB IV

PENGUMPULAN DAN PENGOLAHAN DATA

4.1 Pengumpulan Data

Data yang digunakan pada penelitian ini merupakan data sekunder, yaitu data historis permintaan yang dimiliki oleh UKM Tosaga. Pengumpulan data dilakukan dengan cara mengunduh data yang telah tersedia dalam sebuah database pada suatu periode tertentu, pengunduhan dilakukan pada data historis penjualan bulan Januari 2019 – Maret 2020 dalam format Microsoft Excel. Total data penjualan yang terekam adalah sebanyak 261.995. Terdapat beberapa *field* yang didokumentasikan seperti yang tampak pada Gambar 4.1 berikut.

	tanggal	jam	kasir	pelanggan	no_notas	no_urut	kode	nama_barang	qty	satuan
0	02-Jan-19	07:24:59	ADM	CASH	A90102001	1	100042	TAOGE PANJANG	1.00	PCS
1	02-Jan-19	07:24:59	ADM	CASH	A90102001	2	100004	SAWI SENDOK	0.32	KG
2	02-Jan-19	07:24:59	ADM	CASH	A90102001	3	500028	KECAP BANGGO 220ML	1.00	PCS
3	02-Jan-19	07:24:59	ADM	CASH	A90102001	4	500074	BAWANG GORENG SPM 30G	1.00	PCS
4	02-Jan-19	07:30:37	ADM	CASH	A90102002	1	500036	BAWANG PUTIH	0.10	KG
...
16224	31-Jan-19	20:07:48	ADM	CASH	A90131140	1	100034	BUNCIS	1.00	PCS
16225	31-Jan-19	20:07:48	ADM	CASH	A90131140	2	100033	KACANG PANJANG	1.00	PCS
16226	31-Jan-19	20:07:48	ADM	CASH	A90131140	3	100073	CABE PACK	1.00	PCS
16227	31-Jan-19	20:07:48	ADM	CASH	A90131140	4	100084	RAWIT PACK	2.00	PCS
16228	31-Jan-19	20:07:48	ADM	CASH	A90131140	5	400035	TEMPE AULIA	1.00	PCS

16229 rows × 10 columns

Gambar 4.1 Data Historis Bulan Januari 2019

Gambar 4.1 Data Historis Bulan Januari 2019

Pada database dapat diketahui bahwa item yang dijual oleh UKM Tosaga terdiri dari *perishable product* dan *non-perishable product*. Serta penjualan dilakukan dalam satuan yang berbeda-beda, yaitu per buah (*pcs*), gram, dan kilogram. Pengumpulan informasi mengenai jenis produk serta satuan yang digunakan didapati dengan cara wawancara pada penjual di UKM Tosaga, hal ini untuk mengetahui umur ketahanan produk sehingga dapat dikategorikan dalam *perishable product* ataupun *non-perishable product*.

4.2 Preprocessing Data

Preprocessing data bertujuan untuk menyiapkan data mentah menjadi data yang dapat dijadikan input pemodelan, sehingga model yang dihasilkan berkualitas. *Preprocessing data* dilakukan menggunakan *software* Python dengan beberapa *library* yaitu modul Pandas dan modul OS. Modul Pandas (*Python for Data Analysis*) digunakan sebagai pengolah data tabular yang tersimpan dalam baris dan kolom, pengolahan data tersebut seperti manipulasi data, persiapan data, dan pembersihan data. Sedangkan Modul OS (*Operating System*) dapat digunakan untuk mengoperasikan berbagai macam hal, seperti membuat dan menghilangkan dokumen, serta mengubah dan mengidentifikasi penulisan. *Preprocessing data* dilakukan dalam beberapa langkah, yaitu:

4.2.1 Data Integration

Data historis penjualan yang dimiliki oleh Retail Sayur Tosaga tersimpan secara bulanan, sedangkan penelitian yang akan dilakukan membutuhkan data historis pada bulan Januari 2019 – April 2020 sehingga perlu untuk data bulanan tersebut perlu untuk digabungkan. Hasil *data integration* dapat dilihat pada Gambar 4.3 berikut.

```
Aktifkan OS
Aktifkan Pandas

Munculkan semua dokumen
Pilih dokumen = Januari 2019 - April 2020
Gabungkan dokumen

Simpan pada Excel|
```

Gambar 4.2 Pseudocode Data Integration

	tanggal	jam	kasir	pelanggan	no_nota	no_urut	kode	nama_barang	qty	satuan
0	02-Jan-19	07:24:59	ADM	CASH	A90102001	1	100042	TAOGE PANJANG	1.00	PCS
1	02-Jan-19	07:24:59	ADM	CASH	A90102001	2	100004	SAWI SENDOK	0.32	KG
2	02-Jan-19	07:24:59	ADM	CASH	A90102001	3	500028	KECAP BANGGO 220ML	1.00	PCS
3	02-Jan-19	07:24:59	ADM	CASH	A90102001	4	500074	BAWANG GORENG SPM 30G	1.00	PCS
4	02-Jan-19	07:30:37	ADM	CASH	A90102002	1	500036	BAWANG PUTIH	0.10	KG
...
261990	30-Apr-20	19:15:20	ADM	CASH	B00430163	3	700012	TEMPE MUCHLAR PCS	2.00	PCS
261991	30-Apr-20	20:06:59	ADM	CASH	B00430164	1	100019	SAWI PUTIH BOX CURAH	245.00	GR
261992	30-Apr-20	20:06:59	ADM	CASH	B00430164	2	800002	BAWANG BOMBAY CURAH	95.00	GR
261993	30-Apr-20	20:06:59	ADM	CASH	B00430164	3	600007	BAKSO BUNGKUS	1.00	PCS
261994	30-Apr-20	20:06:59	ADM	CASH	B00430164	4	500028	SONICE SISIS-15 375G	1.00	PCS

261995 rows × 10 columns

Gambar 4.3 Hasil Penggabungan Data

4.2.2 Data Cleaning

Data cleaning atau pembersihan data dilakukan untuk menghilangkan kesalahan informasi pada data (Rahm & Do, 2000). Pembersihan dilakukan pada data yang tidak sesuai, seperti tidak lengkap, *noise*, dan tidak konsisten. Pada penelitian ini, terdapat beberapa proses pembersihan data, yaitu:

1. Data tidak lengkap

Data yang tidak lengkap atau kosong akan menimbulkan ketidakpastiaan ketika dilakukan pemodelan pada tahap selanjutnya, sehingga dilakukan pengidentifikasian data tidak lengkap atau *null* pada data historis.

```
Masukkan dataset gabungan
Munculkan informasi data

Hapus baris data tidak lengkap
```

Gambar 4.4 Pseudocode Penghapusan Data Kosong

Setelah dilakukan identifikasi data kosong atau tidak lengkap selanjutnya dilakukan penghapusan untuk mengurangi ketidakpastian pemodelan. Gambar 4.5 menunjukkan adanya 315 data kosong pada kolom “nama_barang” yang kemudian dihapus barisnya sehingga tidak ada lagi data kosong pada semua kolom.

	Total	%		Total	%
nama_barang	315	0.1	satuan	0	0.0
satuan	0	0.0	qty	0	0.0
qty	0	0.0	nama_barang	0	0.0
kode	0	0.0	kode	0	0.0
no_urut	0	0.0	no_urut	0	0.0

Gambar 4.5 Penghapusan Data Kosong

2. Non-perishable Product

Data historis yang dimiliki oleh retail sayur Tosaga tidak membedakan antara *perishable* dan *non-perishable product*. Pada penelitian ini, hanya item *perishable product* yang akan digunakan, sehingga perlu dipisahkan antara *perishable* dan *non-perishable product*.

```
Munculkan semua daftar nama
Hapus baris yang memiliki kata "BAWANG GORENG"
Hapus baris yang memiliki kata "TERI NASI"
Hapus baris yang memiliki kata "..."
```

Gambar 4.6 Pseudocode Penghapusan *Non-perishable*

```
[ 'TAOGE PANJANG' 'SAWI SENDOK' 'BAWANG GORENG SPM 30G' 'BAWANG PUTIH'
'BAWANG MERAH' 'SINGKONG POLAH 500 GR' 'TEMPE MUCHLAR' 'RAWIT PACK'
'BAYAM' 'RESE PACK' 'TELOR PUYUH MATENG' 'KETUMBAR BUBUK 6,5G'
'BUMBU DAPUR' 'BAWANG MERAH SUPER' 'AYAM PACK' 'WORTEL' 'KANGKUNG'
'KRUPUK AYU' 'TOMAT MERAH' 'TIMUN ACAR' 'PINDANG SEDANG' 'CABE PACK'
'AYAM NEGRI' 'ATI AMPELA NEGRI' 'BAKSO BUGKUS' 'DAUN BAWANG' 'BROKOLI'
'KACANG KAPRI' 'TIMUN BEBY' 'SALEM BESAR' 'SOP-SOPAN' 'BLUE BAND'
'TEMPE MUCHLAR BESAR' 'JIPANG' 'TERASI UDANG' 'TERONG UNGGU' 'JAGUNG'
'GULA MERAH' 'ASEM PACK' 'BERAS MENTHIK' 'KELAPA PARUT' 'KEMANGI'
'IKAN ASIN LAYUR' 'DAUN JERUK' 'TAHU TAKWA' 'ROYCO AYAM 9G'
'LADAKU BUBUK 4G' 'KUNYIT CURAH' 'ROYCO SAPI' 'TERIGU SEGITIGA CURAH 500G'
'TEMPE GEMBUS' 'IKAN LELE SEGAR' 'TAHU PUTIH' 'GULA BATU' 'TELOR NEGRI'
'SAWI IJO' 'KACANG PANJANG' 'SANTAN KARA 65ML' 'TERONG IJO'
'KRUPUK SUMBER' 'ROTI BESAR' 'KEMIRI PACK' 'BOMBAY' 'KENTANG BESAR'
'PEPAYA' 'PARE' 'OPAK WONOSOBO' 'CHAM SOSIS AYAM 150 GR'
'CHAM SOSIS SAPI 150 GR' 'BOLA UDANG 200 GR' 'SELEDRI' 'FREN FREIS 250 GR'
'KOBIS' 'TEPUNG KRISPY' 'SAORI TIRAM 23ML' 'GARAM REVINA'
'BUMBU PECEL SUGIH WARAS' 'BAKPAO COKLAT 500 GR' 'PEPAYA SERUT' 'JANGGEL'
'KEMBANG KOL' 'KEMBANG WARU' 'TERI NASI' 'KACANG TANAH' 'GORI CACAH'
'KRIPIK SUKUN' 'IKAN ASIN PEDO BESAR' 'MINYAK HEMAT 500ML'
```

Gambar 4.7 Daftar Nama Produk

Pemisahan ini berdasarkan umur produk, dimana *perishable product* memiliki panjang umur tidak lebih dari 14 hari, hal ini biasanya diakibatkan oleh kerusakan mikroba serta terjadinya proses biomekanikal seperti reaksi perubahan enzim (Labuza, 2000). Kemudian dilakukan penghapusan pada data *non-perishable product*.

	tanggal	jam	kasir	pelanggan	no_nota	no_urut	kode	nama_barang	qty	satuan
0	02-Jan-19	07:24:59	ADM	CASH	A90102001	1	100042	TAOGE PANJANG	1.00	PCS
1	02-Jan-19	07:24:59	ADM	CASH	A90102001	2	100004	SAWI SENDOK	0.32	KG
2	02-Jan-19	07:30:37	ADM	CASH	A90102002	4	400033	TEMPE MUCHLAR	1.00	PCS
3	02-Jan-19	07:30:37	ADM	CASH	A90102002	5	100084	RAWIT PACK	1.00	PCS
4	02-Jan-19	07:39:26	ADM	CASH	A90102004	5	500043	BUMBU DAPUR	1.00	PCS
...
139584	30-Apr-20	18:51:08	ADM	CASH	B00430161	9	200002	BUNCIS PACK 150G	1.00	PCS
139585	30-Apr-20	19:13:25	ADM	CASH	B00430162	1	200042	TERONG UNGU CURAH	700.00	GR
139586	30-Apr-20	19:15:20	ADM	CASH	B00430163	1	200042	TERONG UNGU CURAH	300.00	GR
139587	30-Apr-20	19:15:20	ADM	CASH	B00430163	3	700012	TEMPE MUCHLAR PCS	2.00	PCS
139588	30-Apr-20	20:06:59	ADM	CASH	B00430164	1	100019	SAWI PUTIH BOX CURAH	245.00	GR

139589 rows × 10 columns

Gambar 4.8 Data *Perishable Product*

3. Inkonsisten Data

Nama item produk pada data historis Retail Sayur Tosaga terindikasi tidak konsisten, hal ini juga telah divalidasi oleh penjaga retail. Pada tahap ini, dilakukan penamaan ulang sehingga setiap item hanya akan memiliki satu nama.

```
Munculkan semua nama barang
Ganti nama "CABE MERAH PACK" menjadi "CABE MERAH"
Ganti nama "CABE MERAH PACK CMP" menjadi "CABE MERAH"
Ganti nama "CABE MERAH CURAH" menjadi "CABE MERAH"
Ganti nama "CABE MERAH CURAH" menjadi "CABE MERAH"
Ganti nama "CABE MERAH PACK 50G CMP" menjadi "CABE MERAH"
Ganti nama "..." menjadi "..."
```

Gambar 4.9 Pseudocode Penggantian Nama Produk

	tanggal	jam	kasir	pelanggan	no_nota	no_urut	kode	nama_barang	qty	satuan
0	02-Jan-19	07:24:59	ADM	CASH	A90102001	1	100042	TAOGE PANJANG	1.00	PCS
1	02-Jan-19	07:24:59	ADM	CASH	A90102001	2	100004	PAKCOY	0.32	KG
2	02-Jan-19	07:30:37	ADM	CASH	A90102002	4	400033	TEMPE KECIL	1.00	PCS
3	02-Jan-19	07:30:37	ADM	CASH	A90102002	5	100084	CABE RAWIT	1.00	PCS
4	02-Jan-19	07:39:26	ADM	CASH	A90102004	5	500043	BUMBU DAPUR	1.00	PCS
...
139584	30-Apr-20	18:51:08	ADM	CASH	B00430161	9	200002	BUNCIS	1.00	PCS
139585	30-Apr-20	19:13:25	ADM	CASH	B00430162	1	200042	TERONG UNGU	700.00	GR
139586	30-Apr-20	19:15:20	ADM	CASH	B00430163	1	200042	TERONG UNGU	300.00	GR
139587	30-Apr-20	19:15:20	ADM	CASH	B00430163	3	700012	TEMPE KECIL	2.00	PCS
139588	30-Apr-20	20:06:59	ADM	CASH	B00430164	1	100019	SAWI PUTIH	245.00	GR

139589 rows × 10 columns

Gambar 4.10 Penamaan Ulang Data

4. Penghapusan Transaksi Kosong

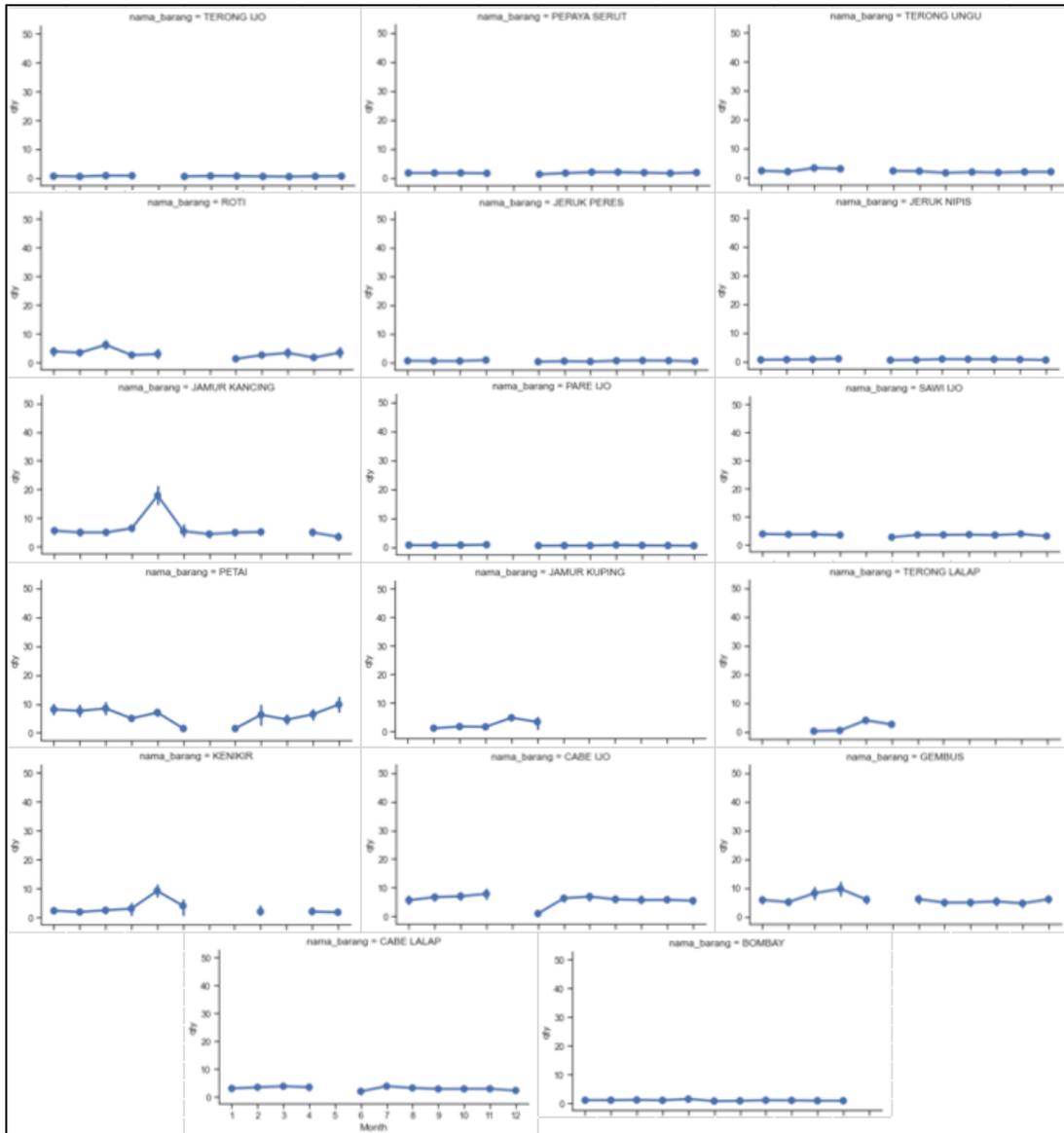
Data transaksi kosong merupakan data yang tidak ada akibat dari tidak adanya transaksi pada suatu periode. Visualisasi data dilakukan untuk mengetahui pola data. Pada tahap ini, beberapa *library* ditambahkan untuk visualisasi data. *Library* *datetime* digunakan untuk mendefinisikan waktu, dalam hal ini adalah tanggal pada dataset. Kemudian *Library* *seaborn* dan *matplotlib.pyplot* digunakan untuk memvisualkan data. Tahap setelahnya adalah penghapusan data untuk produk yang memiliki transaksi kosong pada suatu periode tertentu.

```
Aktifkan library datetime
Aktifkan library seaborn
Aktifkan library matplotlib.pyplot

Aplikasikan datetime pada bagian tanggal
Beri index tahunan
Beri index bulanan
Beri index mingguan
Beri index harian

Visualisasi data (sumbu x = bulan, sumbu y = jumlah, judul = nama barang, jenis = point)
Tampilkan visualisasi
```

Gambar 4.11 Pseudocode Visualisasi Setiap Jenis Barang



Gambar 4.12 Data Transaksi Kosong

	tanggal	jam	kasir	pelanggan	no_nota	no_urut	kode	nama_barang	qty	satuan
0	02-Jan-19	07:24:59	ADM	CASH	A90102001	1	100042	TAOGE PANJANG	1.00	PCS
1	02-Jan-19	07:24:59	ADM	CASH	A90102001	2	100004	PAKCOY	0.32	KG
2	02-Jan-19	07:30:37	ADM	CASH	A90102002	4	400033	TEMPE KECIL	1.00	PCS
3	02-Jan-19	07:30:37	ADM	CASH	A90102002	5	100084	CABE RAWIT	1.00	PCS
4	02-Jan-19	07:39:26	ADM	CASH	A90102004	5	500043	BUMBU DAPUR	1.00	PCS
...
119956	30-Apr-20	18:51:08	ADM	CASH	B00430161	3	200010	CABE RAWIT	1.00	PCS
119957	30-Apr-20	18:51:08	ADM	CASH	B00430161	4	200008	CABE MERAH	1.00	PCS
119958	30-Apr-20	18:51:08	ADM	CASH	B00430161	9	200002	BUNCIS	1.00	PCS
119959	30-Apr-20	19:15:20	ADM	CASH	B00430163	3	700012	TEMPE KECIL	2.00	PCS
119960	30-Apr-20	20:06:59	ADM	CASH	B00430164	1	100019	SAWI PUTIH	245.00	GR

119961 rows × 10 columns

Gambar 4.13 Hasil Penghapusan Data Transaksi Kosong

5. Pemberian ID

Pada tahap ini akan dilakukan penghapusan kode dan pemberian ID sesuai dengan nama barang untuk memudahkan pemrosesan selanjutnya. Selain itu juga sebagai pengganti kolom “kode” yang tidak konsisten sehingga tidak bisa dijadikan sebagai identitas produk.

```
Tampilkan semua nama barang
Jika nama barang = BANDENG, isi ID = 1
Jika nama barang = BAYAM, isi ID = 2
Jika nama barang = PEPAYA, isi ID = 3
Jika nama barang = ..., isi ID = ...
Selain itu, isi ID = 49
```

Gambar 4.14 Pseucide Pemberian ID

	tanggal	jam	kasir	pelanggan	no_nota	no_urut	id	nama_barang	qty	satuan
0	02-Jan-19	07:24:59	ADM	CASH	A90102001	1	40	TAOGE PANJANG	1.00	PCS
1	02-Jan-19	07:24:59	ADM	CASH	A90102001	2	30	PAKCOY	0.32	KG
2	02-Jan-19	07:30:37	ADM	CASH	A90102002	4	27	TEMPE KECIL	1.00	PCS
3	02-Jan-19	07:30:37	ADM	CASH	A90102002	5	10	CABE RAWIT	1.00	PCS
4	02-Jan-19	07:39:26	ADM	CASH	A90102004	5	5	BUMBU DAPUR	1.00	PCS
...
119956	30-Apr-20	18:51:08	ADM	CASH	B00430161	3	10	CABE RAWIT	1.00	PCS
119957	30-Apr-20	18:51:08	ADM	CASH	B00430161	4	9	CABE MERAH	1.00	PCS
119958	30-Apr-20	18:51:08	ADM	CASH	B00430161	9	6	BUNCIS	1.00	PCS
119959	30-Apr-20	19:15:20	ADM	CASH	B00430163	3	27	TEMPE KECIL	2.00	PCS
119960	30-Apr-20	20:06:59	ADM	CASH	B00430164	1	42	SAWI PUTIH	245.00	GR

119961 rows × 10 columns

Gambar 4.15 Hasil Pemberian ID

6. Penyamaan Satuan

Satuan yang digunakan pada Retail Sayur Tosaga ada 3 jenis, yaitu PCS (unit), KG (kilogram), dan GR (gram), hal ini didapatkan dari kolom ‘satuan’ pada data historis. Satuan jenis PCS menunjukkan produk dijual dalam bentuk *pack* dengan berat tertentu, sehingga antar produknya dapat memiliki berat yang berbeda-beda. Namun, terdapat jenis produk yang memiliki satuan ganda, yaitu KG dan GR. Untuk mengatasi hal itu, dilakukan pengecekan ulang terhadap ketidakkonsistenan data sejenis untuk menghilangkan ketidakkonsistenannya. Kemudian dilakukan penyesuaian sehingga setiap item hanya memiliki satu jenis satuan, yaitu PCS (unit) atau KG (kilogram).

```
Jika satuan = GR, kuantitas = penjualan*0.001
Ganti satuan = GR, menjadi = KG
```

Gambar 4.16 Pseudocode Penyesuaian Satuan

	tanggal	jam	kasir	pelanggan	no_nota	no_urut	id	nama_barang	qty	satuan
0	02-Jan-19	07:24:59	ADM	CASH	A90102001	1	40	TAOGE PANJANG	1.000	PCS
1	02-Jan-19	07:24:59	ADM	CASH	A90102001	2	30	PAKCOY	0.320	KG
2	02-Jan-19	07:30:37	ADM	CASH	A90102002	4	27	TEMPE KECIL	1.000	PCS
3	02-Jan-19	07:30:37	ADM	CASH	A90102002	5	10	CABE RAWIT	1.000	PCS
4	02-Jan-19	07:39:26	ADM	CASH	A90102004	5	5	BUMBU DAPUR	1.000	PCS
...
119956	30-Apr-20	18:51:08	ADM	CASH	B00430161	3	10	CABE RAWIT	1.000	PCS
119957	30-Apr-20	18:51:08	ADM	CASH	B00430161	4	9	CABE MERAH	1.000	PCS
119958	30-Apr-20	18:51:08	ADM	CASH	B00430161	9	6	BUNCIS	1.000	PCS
119959	30-Apr-20	19:15:20	ADM	CASH	B00430163	3	27	TEMPE KECIL	2.000	PCS
119960	30-Apr-20	20:06:59	ADM	CASH	B00430164	1	42	SAWI PUTIH	0.245	KG

119961 rows × 10 columns

Gambar 4.17 Hasil Penyesuaian Satuan

4.2.3 Pemisahan Data

Pada data historis terdapat dua macam satuan, yaitu PCS (unit) dan KG (kilogram) dimana satuan ini memiliki arti serta hubungan yang berbeda. Pada tahap ini akan dilakukan pemisahan data berdasarkan satuannya menggunakan *library* Pandas, sehingga akan ada dua dataset yang kemudian akan diproses masing-masing.

```
Pilih data dengan satuan = KG
Simpan dataset KG

Pilih data dengan satuan = PCS
Simpan dataset PCS
```

Gambar 4.18 Pseudocode Pemisahan Dataset KG dan PCS

4.2.4 Penjumlahan Harian

Pada tahap ini akan dilakukan penjumlahan permintaan produk secara harian untuk masing-masing produk menggunakan *library* Pandas. Selain itu juga akan dilakukan penghapusan beberapa kolom yang tidak akan digunakan pada tahap selanjutnya, yaitu 'jam', 'kasir', 'pelanggan', 'no_nota', 'no_urut', dan 'satuan'. Sehingga kolom yang tersisa adalah 'nama_barang', 'id', 'tanggal', dan 'qty'.

```
Satukan data yang memiliki kesamaan = nama barang, ID, tanggal, jumlahkan = kuantitas
Urutkan data berdasarkan = tanggal
Simpan
```

Gambar 4.19 Pseudocode Penjumlahan Harian

	nama_barang	id	tanggal	qty
0	BANDENG	1	01-Apr-20	10.0
1	BANDENG	1	01-Aug-19	7.0
2	BANDENG	1	01-Dec-19	5.0
3	BANDENG	1	01-Feb-19	1.0
4	BANDENG	1	01-Feb-20	8.0
...
11161	TEMPE POTONG	20	31-Jan-20	9.0
11162	TEMPE POTONG	20	31-Jul-19	10.0
11163	TEMPE POTONG	20	31-Mar-20	8.0
11164	TEMPE POTONG	20	31-May-19	18.0
11165	TEMPE POTONG	20	31-Oct-19	13.0

[11166 rows x 4 columns]

Gambar 4.20 Hasil Pengelompokkan

Penjumlahan item produk harian serta penghapusan kolom tertentu dilakukan pada kedua dataset. Setelah itu, dilakukan pengurutan berdasarkan tanggal untuk memudahkan dalam melihat data harian.

	nama_barang	id	tanggal	qty
1239	BUNCIS	6	2019-01-02	13.0
3049	JANGGEL	19	2019-01-02	4.0
8970	TAOGE PENDEK	35	2019-01-02	3.0
6397	TAHU KULIT	45	2019-01-02	5.0
9558	TEMPE BESAR	33	2019-01-02	11.0
7833	TAHU SOMAY	49	2019-01-02	12.0
5190	PAKET SOP	32	2019-01-02	6.0
13	BANDENG	1	2019-01-02	9.0
10399	TEMPE MENDOAN	21	2019-01-02	4.0
10762	TEMPE POTONG	20	2019-01-02	18.0

Gambar 4.21 Hasil Pengurutan Tanggal Data

4.3 Autoregressive Integrated Moving Average (ARIMA)

Autoregressive Integrated Moving Average (ARIMA) merupakan model peramalan yang menghasilkan ramalan-ramalan yang berdasarkan sintesis dari pola data secara historis. Metode ARIMA akan bekerja baik apabila data pada deret waktu yang digunakan bersifat dependen atau berhubungan satu sama lain secara statistik (Makridakis et al., 1999). Dalam pengaplikasian model menggunakan metode ARIMA terdapat beberapa langkah, seperti yang akan dijelaskan berikut.

```

Aktifkan numpy
Aktifkan pandas
Aktifkan datetime
Dari sklearn.metrics, aktifkan mean_squared_error
Dari numpy, aktifkan sqrt

Aktifkan matplotlib.pyplot
Dari matplotlib.pyplot, aktifkan rcParams
Aktifkan seaborn

Dari statsmodels.tsa.stattools, aktifkan adfuller
Dari statsmodels.tsa.stattools, aktifkan = acf, pacf
Dari statsmodels.tsa.seasonal, aktifkan seasonal_decompose
Dari statsmodels.tsa.ar_model, aktifkan AR
Dari statsmodels.tsa.arima_model, aktifkan ARIMA

Dari pandas.plotting, aktifkan autocorrelation_plot
Dari statsmodels.tsa.stattools, aktifkan = adfuller, acf, pacf, arma_order_select_ic
Aktifkan statsmodels.formula.api
Aktifkan statsmodels.tsa.api
Aktifkan statsmodels.api
Aktifkan scipy.stats

```

Gambar 4.22 Pseudocode Penggunaan Modul ARIMA

Terdapat beberapa *library* yang diperlukan untuk melakukan pemodelan dengan menggunakan metode ARIMA, diantaranya adalah:

1. Numpy : merupakan *library* Python yang digunakan untuk pemrosesan *array*. Selain itu juga dapat digunakan dalam *linear algebra* serta matriks.
2. Pandas : sebagai pengolah data tabular yang tersimpan dalam baris dan kolom, pengolahan data tersebut seperti manipulasi data, persiapan data, dan pembersihan data.
3. Datetime : digunakan untuk pemrosesan waktu pada objek data di Python dikarenakan tidak adanya tipe data tanggal pada Python.
4. Sklearn : *library* yang berfungsi untuk pemrosesan data maupun membangun model pembelajaran mesin (*training data*). *Library* ini memiliki berbagai algoritma pembelajaran, baik untuk regresi, pengelompokan, maupun klasifikasi.
5. Matplotlib: digunakan untuk membuat visualisasi statis, animasi, dan interaktif pada Python.
6. Seaborn : digunakan untuk visualisasi data pada Python berdasarkan matplotlib. Berguna untuk menggambarkan visual yang atraktif dan informatif.
7. Statsmodels : merupakan modul di Python yang menyediakan fungsi untuk mengestimasi berbagai model statistik serta melakukan uji statistik dan eksplorasi data.

4.3.1 Pemeriksaan Kestasioneran Data

Model ARIMA mengasumsikan data deret waktu yang dihasilkan berbentuk stasioner, namun tidak semua data berpola stasioner. Jika data tidak stasioner maka metode yang digunakan untuk membuat data stasioner adalah *differencing* agar data yang tidak stasioner dalam rata-rata dan proses transformasi untuk data yang tidak stasioner dalam varian (Mulyana, 2004).

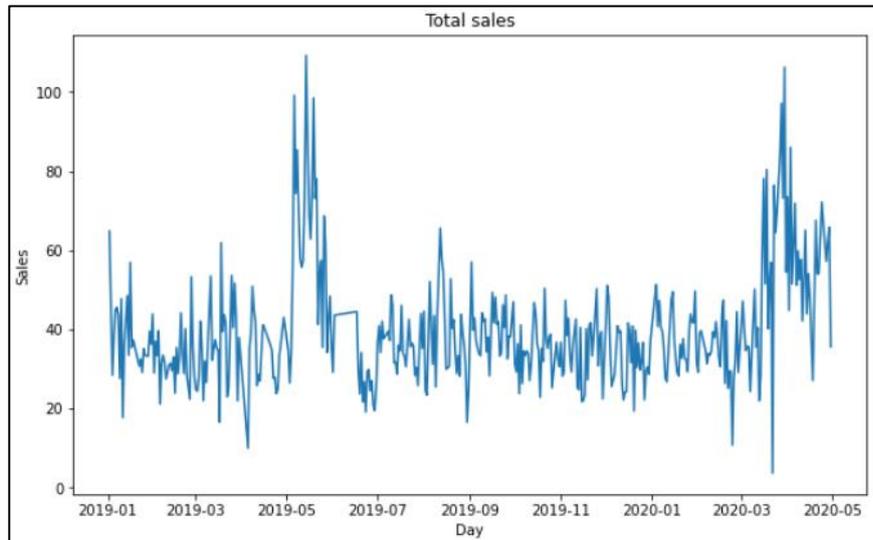
```

Masukkan dataset
Aplikasikan datetime pada tanggal

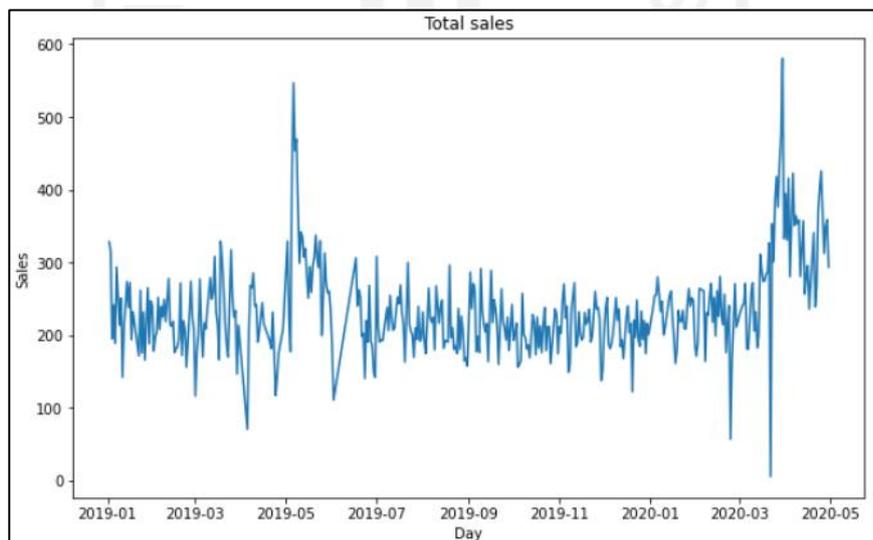
Kelompokkan = tanggal, jumlahkan = kuantitas
Visualisasikan data (sumbu x = waktu, sumbu y = kuantitas, judul = total sales)
Tampilkan visualisasi

```

Gambar 4.23 Pseudocode Visualisasi Dataset



Gambar 4.24 Grafik Permintaan Dataset 1 (KG)



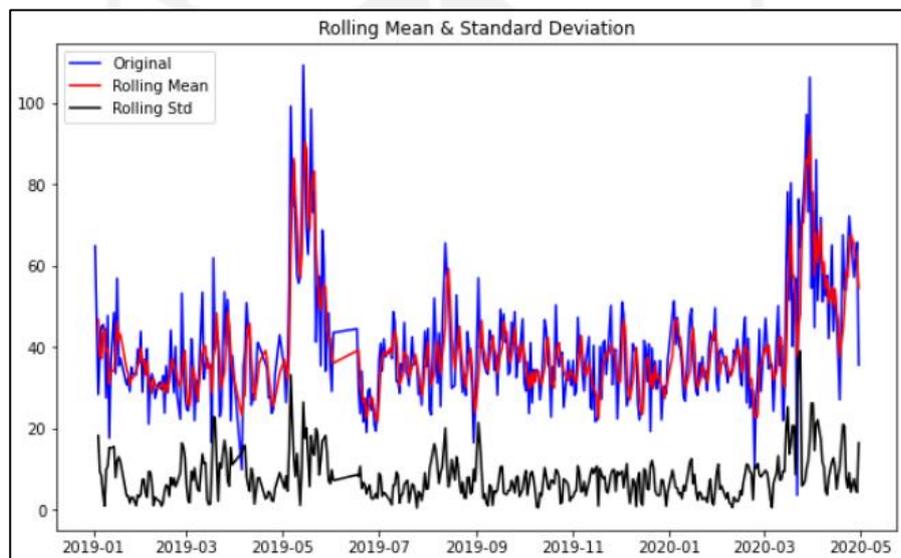
Gambar 4.25 Grafik Permintaan Dataset 2 (PCS)

Setelah diketahui pola data asli, selanjutnya adalah memunculkan rata-rata (*mean*) serta standar deviasi dari data historis. Hal ini diperlukan untuk pengecekan stasioner secara visual, yaitu jika tidak terdapat perubahan rata-rata maupun varian.

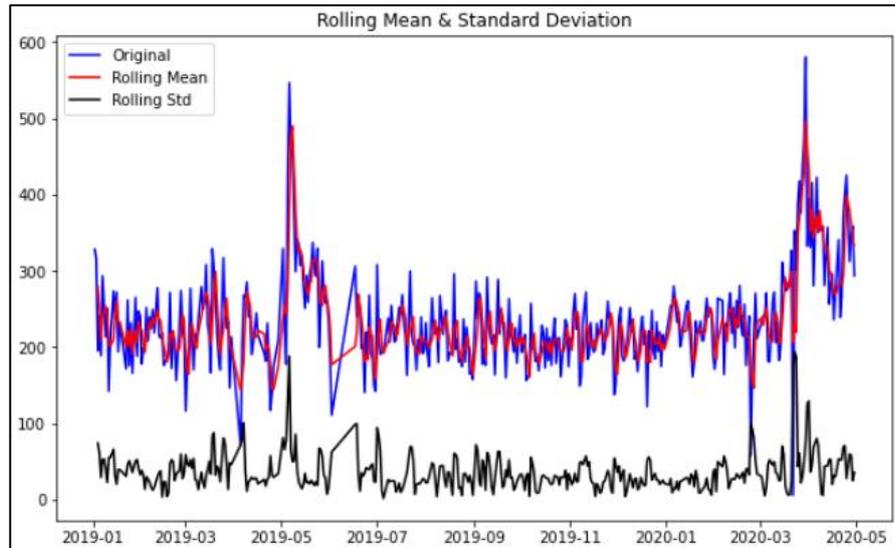
```
Tentukan, mean = mean dataset
Tentukan, std = std dataset

Plot data = kuantitas
Plot mean = mean
Plot std = std
Aktifkan legenda
Beri judul = Mean dan Std
Tampilkan visualisasi
```

Gambar 4.26 Pseudocode Visualisasi Stasioner



Gambar 4.27 Visualisasi Pengecekan Stasioner Dataset 1 (KG)



Gambar 4.28 Visualisasi Pengecekan Stasioner Dataset 1 (PCS)

Selain melihat secara visual, uji statistik dapat dilakukan menggunakan Augmented Dickey-Fuller test, yaitu salah satu bentuk dari *unit root test* yang menguji pengaruh *trend* dalam sebuah *time series*. Augmented Dickey-Fuller test merupakan pengembangan dari Dickey-Fuller test dengan memasukkan model regresi dalam pemrosesannya. Output yang dihasilkan diantaranya adalah *p-value*, *value of test statistic*, *number of lags for the test*, dan *critical value cutoffs*. Ketika *test statistic* menunjukkan angka yang lebih rendah dari *critical value* menandakan *time series* memiliki pola stasioner.

```
Tampilkan "Results of Dickey Fuller Test"
dfctest = panggil library adfuller (index sataset = kuantitas, lag otomatis = AIC)
dfoutput = Test statistik, p-value, jumlah lag, jumlah observasi
          untuk nilai di dataset:
            dfoutput = critical value untuk 1%, 5%, 10%

Tampilkan dfoutput
```

Gambar 4.29 Pseucode Dickey Fuller Test

Dalam mengimplementasikan Augmented Dickey-Fuller test, terdapat beberapa hal yang perlu untuk diperhatikan, yaitu *critical value cutoff* untuk menentukan batas kritis stasioner dimana *critical value* yang digunakan adalah 1%. Kemudian digunakan fungsi `autolag='AIC'` untuk menentukan jumlah lag paling. Hasil dari ADF pada kedua dataset adalah sebagai berikut:

```

Results of Dickey Fuller Test:
Test Statistic          -3.621183
p-value                 0.005369
#Lags Used              17.000000
Number of Observations Used  404.000000
Critical Value (1%)     -3.446640
Critical Value (5%)     -2.868721
Critical Value (10%)    -2.570595
dtype: float64

```

Gambar 4.30 Hasil Pengaplikasian ADF pada Dataset 1 (KG)

```

Results of Dickey Fuller Test:
Test Statistic          -2.641612
p-value                 0.084698
#Lags Used              15.000000
Number of Observations Used  406.000000
Critical Value (1%)     -3.446560
Critical Value (5%)     -2.868685
Critical Value (10%)    -2.570576
dtype: float64

```

Gambar 4.31 Hasil Pengaplikasian ADF pada Dataset 2 (PCS)

4.3.2 Plotting ACF dan PACF

Dalam mengidentifikasi model ARIMA (p,q,d) dapat dilakukan dengan melihat plot ACF (*Autocorrelation Function*) dan plot PACF (*Parsial Autocorrelation Function*) (Darsyah & Nur, 2016). Penentuan model ARIMA dapat dilakukan melalui model matematis maupun menggunakan *machine learning* dengan *library* tertentu, dengan penentuan nilai p dan q sebagai berikut:

Tabel 4.1 Plotting ACF dan PACF

ACF	PACF	Model
<i>Cutoff</i> setelah lag q	<i>Tail off</i>	MA (q)
<i>Tail off</i>	<i>Cutoff</i> setelah lag p	AR (p)
<i>Cutoff</i> setelah lag q	<i>Cutoff</i> setelah lag p	MA (q) atau AR (p), pilih model terbaik
<i>Tail off</i>	<i>Tail off</i>	ARMA (p,q), cek berbagai kombinasi p dan q kemudian pilih model terbaik

Pada Python, *library* statsmodels juga dapat membuat plot ACF serta FACF secara otomatis. Selain itu diperlukan *library* lain yaitu *library* matplotlib.pyplot untuk memunculkan visualisasi plot ACF serta FACF.

```
Aktifkan statsmodels.api

Gambar = ukuran (12,8)
Sumbu 1 = terdapat 2 gambar, posisi di atas
gambar = visual ACF (untuk data, jumlah lag = 40, sumbu = sumbu 1)
Sumbu 2 = terdapat 2 gambar, posisi di bawah
gambar = visual PACF (untuk data, jumlah lag = 40, sumbu = sumbu 2)
```

Gambar 4.32 Pseudocode Memunculkan ACF dan PACF

4.3.3 Pemodelan ARIMA

Autoregressive integrated moving average (ARIMA) adalah salah satu model linier paling populer untuk peramalan deret waktu karena sifat statistiknya yang baik dan fleksibilitas yang tinggi (Liu, 2016). Besar performa model ARIMA ditentukan oleh nilai (p,q,d) . Dalam hal ini nilai (p,q,d) yang optimal untuk kedua dataset adalah $(1,1,0)$.

```
model = ARIMA (untuk data, (p,q,r = 1,1,0))
model data = model data pas

Pilih bentuk visualisasi menggunakan 'seaborn-poster'
Grafik 1 = visualisasi kuantitas
Grafik 2 = visualisasi prediksi
Gunakan legenda
Beri judul 'ARIMA model'
Tampilkan visualisasi
```

Gambar 4.33 Pseudocode Pemodelan ARIMA

4.3.4 Perhitungan Nilai Error ARIMA

Perhitungan nilai error dilakukan untuk mengetahui perbandingan antara data asli dengan data hasil pembelajaran atau *data training*. Pengukuran dilakukan menggunakan *root mean square error* (RMSE).

```
Hitung RMSE = dari data ke-0 sampai data terakhir
Print persentase eror RMSE
```

Gambar 4.34 Pseudocode RMSE ARIMA

4.4 Long Short-Term Memory (LSTM)

Long Short-term Memory (LSTM) dapat menyelesaikan banyak tugas yang tidak dapat diselesaikan oleh pembelajaran sebelumnya, yaitu jaringan saraf berulang (RNN) (Gers, 1999). Cara kerja LSTM adalah membuat perubahan pada RNN dengan menambahkan memory cell yang dapat menyimpan informasi untuk jangka waktu yang lama (Zaman, 2019). Dalam pengaplikasian model menggunakan metode LSTM terdapat beberapa langkah, seperti yang akan dijelaskan berikut.

```
Aktifkan warnings
Aktifkan numpy
Aktifkan pandas
Aktifkan matplotlib.pyplot
Dari keras, aktifkan = optimizers
Dari keras.utils, aktifkan = plot_model
Dari keras.models, aktifkan = Sequential, Model
Dari keras.layers.convolutional, aktifkan = Conv1D, MaxPooling1D
Dari keras.layers, aktifkan = Dense, LSTM, RepeatVector, TimeDistributed, Flatten
Dari sklearn.metrics, aktifkan = mean_squared_error
Dari sklearn.model_selection, aktifkan = train_test_split
Aktifkan chart_studio.plotly
Aktifkan plotly.graph_objs
Dari plotly.offline, aktifkan = init_notebook_mode, iplot
Aktifkan tensorflow
```

Gambar 4.35 Pseudocode Penggunaan Modul ARIMA

Terdapat beberapa *library* yang diperlukan untuk melakukan pemodelan dengan menggunakan metode ARIMA, diantaranya adalah:

1. Numpy : merupakan *library* Python yang digunakan untuk pemrosesan *array*. Selain itu juga dapat digunakan dalam *linear algebra* serta matriks.
2. Pandas : sebagai pengolah data tabular yang tersimpan dalam baris dan kolom, pengolahan data tersebut seperti manipulasi data, persiapan data, dan pembersihan data.
3. Matplotlib: digunakan untuk membuat visualisasi statis, animasi, dan interaktif Pada Python.

4. Keras : digunakan untuk membangun dan melatih pembelajaran mendalam pada Python.
5. Sklearn : *library* yang berfungsi untuk pemrosesan data maupun membangun model pembelajaran mesin (*training data*). *Library* ini memiliki berbagai algoritma pembelajaran, baik untuk regresi, pengelompokan, maupun klasifikasi.
6. Plotly : digunakan untuk membuat grafik yang berkualitas, seperti *line plots*, *scatter plots*, *area chart*, *bar chart*, *error bars*, *box plot*, *histogram*, dan lain-lain.
7. Tensorflow : digunakan dalam *machine learning* atau proses pembelajaran mesin karena dapat mempelajari kecerdasan buatan secara mendalam.

4.4.1 Persiapan Data

Persiapan data ditujukan untuk menentukan data yang nantinya akan diolah menggunakan *Long Short-Term Memory* (LSTM). Tahapan ini merupakan tahapan awal sebelum data melalui proses pembelajaran mesin.

```
Masukkan dataset
Pilih data yang akan dipelajari, mulai '2019-01-01'
Urutkan data berdasarkan tanggal, satukan data berdasarkan = ID, tanggal
Pilih kolom yang akan dipelajari = ID, tanggal, kuantitas
```

Gambar 4.36 Pseudocode Persiapan Data

4.4.2 Pembelajaran Data

Machine learning atau pembelajaran mesin memiliki kemampuan untuk menghasilkan suatu model secara otomatis dari sekumpulan data. Pada tahap ini dilakukan proses pembelajaran data untuk mempelajari data historis yang telah dimiliki. Dalam melakukan prediksi suatu urutan, data tersebut perlu untuk disusun kembali sebagai bahan pembelajaran program (*supervised learning*).

Pada tahap ini, fungsi yang akan digunakan dalam mengubah bentuk data deret waktu (*time series*) menjadi data yang dapat dilakukan pembelajaran (*supervised learning*) adalah fungsi

shift () pada *library* Panda. Fungsi shift () dapat digunakan untuk menggandakan kolom data secara maju dengan menambahkan nilai “NaN” pada data awal, maupun secara mundur dengan menambahkan nilai “NaN” pada data akhir. Hal ini dibutuhkan dalam membuat kolom observasi *lag* sebagai kolom observasi peramalan pada data deret waktu dalam format *supervised learning*.

```

Definisikan data pembelajaran (jumlah waktu mundur = 1, jumlah waktu maju = 1, hapus data kosong)
Untuk nilai i dalam range (waktu mundur, 0, -1):
    Tambahkan kolom waktu mundur dengan
        Nama = nama kolom + (t - banyak waktu mundur)
Tambahkan kolom waktu maju dengan
        Nama = nama kolom + (t + banyak waktu maju)
Jadikan semua kolom dalam satu tabel
  
```

Gambar 4.37 Pseudocode Persiapan Pembelajaran Data

```

Tentukan jumlah waktu mundur = 15
Tentukan jumlah waktu maju = 1
Lakukan pembelajaran pada waktu mundur dan maju
  
```

Gambar 4.38 Pseudocode Penentuan Jumlah Pembelajaran Data

Jumlah *lag* yang akan digunakan pada proses ini adalah 15 *lag* untuk waktu mundur (*backward*) dan 1 *lag* untuk waktu maju (*forward*). Dalam hal ini, simbol waktu (t) melambangkan waktu sekarang, sedangkan waktu masa depan memiliki simbol (t+1, t+2) dan waktu lampau bersimbol (t-1, t-2).

	id(t-15)	qty(t-15)	id(t-14)	qty(t-14)	id(t-13)	qty(t-13)	id(t-12)	qty(t-12)	id(t-11)	qty(t-11)	...	id(t-3)	qty(t-3)	id(t-2)	qty(t-2)	id(t-1)	qty(t-1)	id(t)	qty(t)	id(t+1)	qty(t+1)
15	1.0	9.0	1.0	13.0	1.0	9.0	1.0	5.0	1.0	6.0	...	1.0	14.0	1.0	8.0	1.0	6.0	1	4.0	1.0	9.0
16	1.0	13.0	1.0	9.0	1.0	5.0	1.0	6.0	1.0	7.0	...	1.0	8.0	1.0	6.0	1.0	4.0	1	9.0	1.0	6.0
17	1.0	9.0	1.0	5.0	1.0	6.0	1.0	7.0	1.0	4.0	...	1.0	6.0	1.0	4.0	1.0	9.0	1	6.0	1.0	3.0
18	1.0	5.0	1.0	6.0	1.0	7.0	1.0	4.0	1.0	1.0	...	1.0	4.0	1.0	9.0	1.0	6.0	1	3.0	1.0	4.0
19	1.0	6.0	1.0	7.0	1.0	4.0	1.0	1.0	1.0	7.0	...	1.0	9.0	1.0	6.0	1.0	3.0	1	4.0	1.0	10.0
20	1.0	7.0	1.0	4.0	1.0	1.0	1.0	7.0	1.0	6.0	...	1.0	6.0	1.0	3.0	1.0	4.0	1	10.0	1.0	2.0
21	1.0	4.0	1.0	1.0	1.0	7.0	1.0	6.0	1.0	3.0	...	1.0	3.0	1.0	4.0	1.0	10.0	1	2.0	1.0	4.0
22	1.0	1.0	1.0	7.0	1.0	6.0	1.0	3.0	1.0	20.0	...	1.0	4.0	1.0	10.0	1.0	2.0	1	4.0	1.0	2.0
23	1.0	7.0	1.0	6.0	1.0	3.0	1.0	20.0	1.0	14.0	...	1.0	10.0	1.0	2.0	1.0	4.0	1	2.0	1.0	2.0
24	1.0	6.0	1.0	3.0	1.0	20.0	1.0	14.0	1.0	8.0	...	1.0	2.0	1.0	4.0	1.0	2.0	1	2.0	1.0	1.0

Gambar 4.39 Urutan Prediksi Bahan Pembelajaran Dataset 2 (PCS)

Setelah kolom pembelajaran tersedia, selanjutnya dilakukan penghapusan kolom ID dikarenakan tidak akan diproses dalam pemodelan LSTM. Dalam melakukan pemodelan, akan

digunakan 80% data *training* dan 20% data *testing*, sehingga untuk Dataset 1 (KG) jumlah perbandingan data *training* dan data *testing* adalah 5064 dan 1267, sedangkan untuk Dataset 2 (PCS) adalah 8560 dan 2140.

```
Pilih kolom waktu maju untuk 'ID'
Untuk nilai i dalam range (waktu mundur, 0, -1):
    Pilih kolom waktu mundur untuk 'ID'
Hapus kolom terpilih

Gunakan library train_test_split untuk dataset dan hasil pembelajaran (ukuran test = 0.2)
Tampilkan dimensi dataset (jumlah baris, jumlah kolom)
Tampilkan dimensi hasil pembelajaran (jumlah baris, jumlah kolom)
```

Gambar 4.40 Pseudocode Penghapusan ID dan Pembelajaran Data

4.4.3 Pemodelan LSTM

Pemodelan LSTM merupakan proses pemodelan setelah dilakukannya proses pembelajaran data. Pada tahap ini terdapat beberapa komponen dalam LSTM yang harus ditentukan nilainya, hal ini yang nantinya akan mempengaruhi performa pemodelan LSTM. Langkah pertama adalah mendefinisikan model yang akan dipakai, yaitu Sequential Class yang merupakan bagian dari *library* Keras sebagai pendefinisi layer dalam *neural network* yang dapat menghubungkan antar layer. Setelah model terdefinisi, aktifkan model LSTM () beserta dengan fungsi Dense () yang biasanya digunakan dalam melakukan prediksi LSTM. Dalam hal ini, jumlah output layer Dense adalah 1 neuron. Pada *hidden layer* LSTM, bentuk input layer harus berbentuk 3D, dalam hal ini dapat digunakan fungsi Reshape () pada *Library* Numpy yang dapat mengubah bentuk dataset 1D atau 2D menjadi dataset 3D. Dalam hal ini, kolom akan mendapat perlakuan dua kali *time steps* atau dua lag variabel pada observasi sebelumnya yang akan dipakai dalam proses selanjutnya. Kemudian digunakan fungsi aktivasi yang mengubah sinyal yang dijumlahkan dari setiap neuron pada sebuah layer menjadi objek layer pada Sequential. Dalam hal ini fungsi aktivasi yang digunakan adalah *Rectified Linear Unit* (ReLU).

Langkah kedua adalah penyusunan model yang telah terdefinisi, yaitu dengan mengubah urutan layer yang telah terdefinisi kedalam matrix yang akan dieksekusi oleh GPU atau CPU. Dalam menyusun model, dibutuhkan parameter spesifik yang nantinya akan digunakan dalam pelatihan jaringan syaraf, yaitu untuk mengoptimalkan pelatihan data yang akan dilakukan.

Dalam hal ini, algoritma optimasi yang digunakan adalah Adam dan evaluasi kesalahan data menggunakan *Mean Square Error* (MSE).

Langkah ketiga adalah pemasangan jaringan, dimana diperlukan penentuan pola input dan output yang akan dilakukan. Jaringan dilatih menggunakan algoritma *Backpropagation Through Time* dan dioptimalkan sesuai dengan algoritma optimasi dan evaluasi kesalahan yang telah ditentukan saat penyusunan model. Algoritma *backpropagation* akan melatih jaringan sebanyak jumlah *epoch*, setiap *epoch* dapat dibagi menjadi beberapa kelompok input-output yang disebut dengan *batch*. Hal ini akan menentukan jumlah pola pelatihan sebelum sebelum bobot baru diperbarui. Jumlah *epoch* yang digunakan pada pelatihan ini adalah 40 dan jumlah *batch* adalah 1. Selain itu juga dapat dilakukan penentuan jumlah informasi yang akan ditampilkan menggunakan fungsi *verbose*, dalam hal ini *verbose* yang digunakan adalah 2, artinya yang akan ditampilkan hanya jumlah data hilang pada setiap *epoch*.

```
#Pendefinisian Model
Menentukan model = sequential
Pada model LSTM, tambahkan (dense = 1)
Pada model LSTM, tambahkan (library pemodelan = LSTM(aktivasi = 'relu', data shape = 2)

#Optimasi dan Evaluasi
Pada model LSTM, satukan (data hilang = 'mse', optimasi = adam)

#Pemasangan Jaringan
Menentukan epoch = 40
Menentukan batch = 1
Menentukan kecepatan pembelajaran = 0.0003
LSTM = model LSTM (pembelajaran, validasi , epochs = epochs, verbose = 2)
```

Gambar 4.41 Pseudocode Pemrosesan LSTM

Setelah dilakukan pelatihan, selanjutnya dilakukan visualisasi data berupa perbandingan data yang tidak sama antara data asli dengan hasil pembelajaran. Visualisasi dilakukan dengan menggunakan *library* Matplotlib.

```
Atur ukuran gambar (20,15)
Tampilkan grafik histori hasil pembelajaran hilang, beri label = 'Train loss'
Tampilkan grafik histori hasil validasi hilang, beri label = 'Validation loss'
Tampilkan legenda
Beri judul grafik 'Reguler LSTM'
Tetapkan sumbu x = epoch
Tetapkan sumbu y = MSE
```

Gambar 4.42 Pseudocode Memunculkan Grafik LSTM

4.4.4 Perhitungan Nilai Error LSTM

Sama halnya seperti ARIMA, pada tahap ini juga akan dilakukan evaluasi model dengan memperhitungkan nilai eror LSTM pada *learning curve* untuk mengukur performa pemodelan LSTM. Visualisasi yang muncul pada tahap sebelumnya menunjukkan *learning curve* untuk menganalisis masalah saat dilakukan pembelajaran, seperti *underfit* atau *overfit*, serta apakah representatif data pelatihan dan data valisasi selesai. Dalam hal ini, pembelajaran yang baik adalah ketika nilai *train loss* > *validation loss*. Pengukuran dilakukan menggunakan *root mean square eror* (RMSE).

```
Prediksi hasil pembelajaran
Prediksi hasil validasi
Tampilkan nilai eror RMSE hasil pembelajaran
Tampilkan nilai eror RMSE hasil validasi
```

Gambar 4.43 Pseucode RMSE LSTM

BAB V

HASIL DAN PEMBAHASAN

Bab ini akan berisi pembahasan secara rinci mengenai penggunaan metode ARIMA serta LSTM pada dataset, serta akan dilakukan perbandingan hasil antara kedua metode pemodelan untuk mendapatkan performa terbaik yang nantinya dapat diimplementasikan oleh Retail Sayur Tosaga dalam melakukan peramalan produk.

5.1 Analisis *Preprocessing Data*

Sebelum dilakukan pembuatan model permintaan, data yang telah terkumpul terlebih dahulu dilakukan *preprocessing*. Terdapat beberapa langkah dalam *preprocessing* dikarenakan banyaknya data yang kosong maupun inkonsisten. Hasil dari proses ini adalah pembersihan data yang semula berjumlah 261.995 item pembelian dalam 1089 jenis item, menjadi 119.961 item pembelian dalam 49 jenis item yang siap untuk dimodelkan. Beberapa langkah yang dimaksud adalah sebagai berikut:

5.1.1 Analisis *Data Integration*

Data historis setiap perlu untuk digabungkan sebagai bahan latihan metode pemodelan menggunakan *machine learning*. Data setiap bulan memiliki komponen yang sama, seperti baris, kolom, serta isinya. Sehingga, semua data historis yang ada dapat digabungkan untuk memudahkan *preprocessing* serta pembuatan model setelahnya.

5.1.2 Analisis Data Cleaning

Pada data cleaning terdapat beberapa proses yang dilakukan untuk membersihkan data, diantaranya adalah:

1. Data tidak lengkap

Identifikasi data tidak lengkap ditujukan untuk mengetahui apakah terdapat kolom yang kosong atau tidak. Hal ini dikarenakan data kosong dapat mempengaruhi performa pemodelan. Hasil dari identifikasi adalah terdapat adanya data yang tidak lengkap, yaitu pada kolom “nama_barang” dimana jumlah data keseluruhan adalah 261.995 sedangkan data yang ada pada kolom tersebut hanya 261.680, seperti tampak pada Gambar 5.1. Hal ini menunjukkan terdapat sebanyak 315 tabel kosong atau setara dengan 0.1% dari keseluruhan data pada kolom “nama_barang”. Penghapusan baris dilakukan karena kolom tersebut merupakan salah satu kolom identitas produk selain kolom “kode”, ketika akan dilakukan pengisian nama produk secara manual berdasarkan kolom “kode”, ditemukan kode ganda atau satu kode dapat memiliki lebih dari satu nama produk. Sehingga, tidak ada acuan yang pasti mengenai isi tabel yang kosong dan baris yang memiliki kekosongan pada kolom “nama_barang” dihapus.

```

RangeIndex: 261995 entries, 0 to 261994
Data columns (total 10 columns):
tanggal      261995 non-null object
jam          261995 non-null object
kasir        261995 non-null object
pelanggan   261995 non-null object
no_nota      261995 non-null object
no_urut      261995 non-null int64
kode         261995 non-null int64
nama_barang  261680 non-null object
qty          261995 non-null float64
satuan       261995 non-null object
dtypes: float64(1), int64(2), object(7)

```

Gambar 5.1 Data Tidak Lengkap

2. Non-perishable Product

Penelitian hanya dilakukan pada *perishable product* dengan umur tidak lebih dari 14 hari saja, sehingga perlu dilakukan penghapusan data *non-perishable product* yang mana memiliki umur lebih dari 14 hari. Pendataan umur tersebut dilakukan dengan cara wawancara pada penjual Retail Sayur Tosaga untuk menanyakan umur setiap produk

berdasarkan cara penyimpanan yang biasanya dilakukan. Kemudian penghapusan dilakukan secara satu-persatu pada *non-perishable product* berdasarkan identitas nama produk. Hasilnya, dari 1089 jenis item yang ada, terdapat 260 item yang merupakan *perishable product*, sehingga item yang tidak termasuk dalam *perishable product* tersebut dihilangkan dan jumlah *dataframe* dari 261.680 menjadi 139.588.

3. Inkonsisten Data

Nama item produk pada data historis Retail Sayur Tosaga terindikasi tidak konsisten, hal ini juga telah divalidasi oleh penjaga retail. Ketidakkonsistenan nama item produk diakibatkan dari adanya perubahan yang dilakukan beberapa kali, baik oleh pemilik retail maupun penjaga retail pada waktu lampau. Sehingga, terdapat banyak produk yang memiliki 2-5 nama berbeda. Pendataan ulang nama produk tersebut dilakukan karena adanya perubahan jumlah satuan penjualan maupun kesalahan teknis pada mesin kasir. Namun, pendataan nama tersebut sering kali berbeda dari nama sebelumnya untuk menghindari kesamaan yang mengakibatkan banyaknya nama untuk setiap produk. Hal ini akan menyebabkan tidak optimalnya proses pemodelan pada suatu produk. Untuk menghindari hal tersebut dilakukan penamaan ulang setiap produk agar seragam dan lengkap dari bulan Januari 2019 – April 2020. Hasilnya, jumlah nama produk yang sebelumnya berjumlah 260 berubah menjadi 66 nama produk.

4. Penghapusan Transaksi Kosong

Data transaksi kosong merupakan data yang tidak ada akibat dari tidak adanya transaksi pada suatu periode. Hal ini dapat diakibatkan dari tidak datangnya supplier pada suatu periode ataupun akibat dari berbedanya nama saat pendataan ulang. Pemodelan hanya dilakukan pada data permintaan yang memiliki kelengkapan transaksi atau tidak musiman, sehingga item produk yang memiliki transaksi kosong pada suatu periode akan dihapus. Dari 66 jenis item produk, terdapat 17 item produk yang memiliki kekosongan transaksi. Selanjutnya dilakukan penghapusan data pada 17 item produk tersebut, sehingga total jenis item produk sebanyak 49 item dengan jumlah data yang semula sebanyak 139.588 menjadi 119.960.

5. Pemberian ID

Setiap jenis item memiliki identitas pada kolom “kode” dan kolom “nama_barang”, namun kode yang ada terindikasi tidak konsisten dikarenakan adanya kode yang memiliki nama barang ganda sehingga identitas kode tidak dapat digunakan. Pada tahap ini dilakukan pemberian ID sebagai pengganti kode yang sebelumnya telah ada, tujuannya adalah sebagai identitas item produk serta untuk memudahkan pemrosesan setelahnya yang membutuhkan kolom berisi *numeric*. Terdapat 49 jenis item produk sehingga ID yang diberikan merupakan angka diantara 1 dan 49.

6. Penyamaan Satuan

Setiap produknya dapat memiliki lebih dari satu satuan. Hal ini dikarenakan adanya percobaan strategi pada beberapa periode waktu serta permintaan jumlah pembelian tertentu. Selain itu juga terindikasi adanya kesalahan operator yaitu tidak mengembalikan format satuan yang telah ditetapkan, ketika terdapat permintaan jumlah pembelian tertentu. Sehingga, perlu dilakukan penyesuaian agar setiap item hanya memiliki satu jenis satuan, yaitu PCS (unit) atau KG (kilogram).

5.1.3 Analisis Pemisahan Data

Pada data historis terdapat dua macam satuan, yaitu PCS (unit) dan KG (kilogram) dimana satuan ini memiliki arti serta hubungan yang berbeda. Hal ini menyebabkan rasio angka yang digunakan berbeda pula, dimana biasanya jumlah pembelian dengan satuan PCS adalah bilangan bulat positif, sedangkan jumlah pembelian dengan satuan KG adalah bilangan desimal positif. Perbedaan selisih yang jauh ini menjadikan pengolahan data dibagi dalam dua dataset. Hasil pemisahan data menunjukkan terdapat 18 jenis item untuk satuan KG (kilogram) dan 31 jenis item untuk satuan PCS (unit).

5.1.4 Analisis Penjumlahan Harian

Perishable product adalah produk yang memiliki umur tahan tidak lebih dari 14 hari dan kualitasnya akan menurun seiring bertambahnya hari. Oleh karena itu, diperlukan prediksi secara harian untuk menjaga kualitas produk.

5.2 Analisis Pemodelan ARIMA

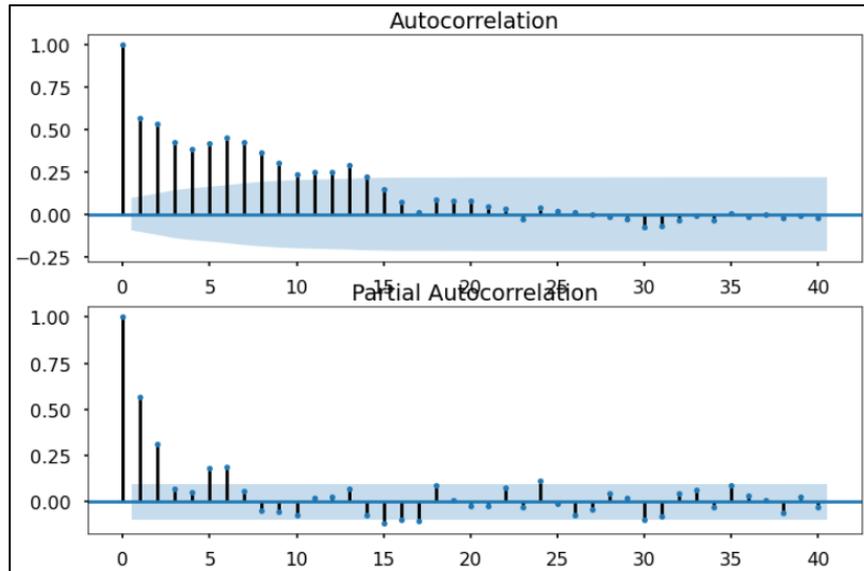
Pada pemodelan menggunakan metode ARIMA terdapat beberapa proses yang dilakukan, diantaranya adalah:

5.2.1 Analisis Pemeriksaan Kestasioneran Data

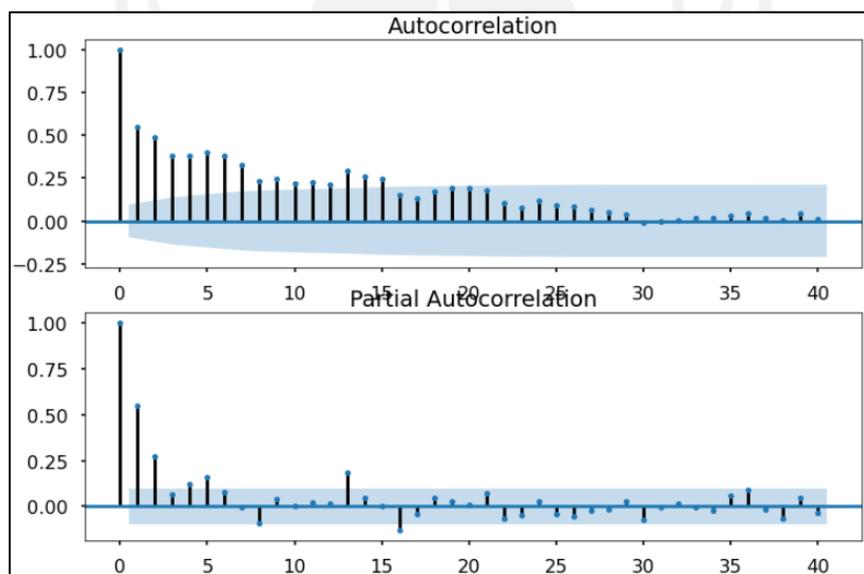
Pengecekan kestasioneran data dilakukan dengan menggunakan *library* *adfuller* pada Python. Dalam hal ini dipilih nilai *critical value* sebesar 1% sebagai parameter data dapat dikatakan stasioner. Hasil perhitungan yang didapatkan adalah Dataset 1 (KG) memiliki nilai *p-value* $(0,0054) < \text{critical value } 1\% (-3,4466)$. Sedangkan untuk Dataset 2 (PCS) memiliki nilai *p-value* $(0,0847) > \text{critical value } 1\% (-3,4466)$. Dengan demikian dapat dikatakan bahwa Dataset 1 berpola stasioner sedangkan Dataset 2 berpola non-stasioner sehingga perlu dilakukan proses *defferencing* agar menjadi stasioner.

5.2.2 Analisis Plotting ACF dan PACF

Selanjutnya dilakukan *plotting* ACF dan PACF untuk mendapatkan nilai (p,d,q) dalam pemrosesan ARIMA. Gambar 5.2 dan Gambar 5.3 merupakan hasil *plotting* ACF dan PACF. Pada kedua dataset terdeteksi adanya pola *seasonal* pada ACF, hal ini dapat diketahui dari pola *lay off* yang panjang dan *cut off* yang baru terjadi pada lag-23 untuk Dataset 1 (KG) dan lag-30 untuk Dataset 2 (PCS). Sehingga, diperlukan proses *defferencing* pada kedua dataset untuk mendapatkan performa model prediksi yang optimal. Dalam hal ini proses *defferencing* dilambangkan dengan d pada (p,d,q) .



Gambar 5.2 Hasil Plotting ACF dan PACF Dataset 1 (KG)



Gambar 2.3 Hasil Plotting ACF dan PACF Dataset 2 (PCS)

5.2.3 Analisis Pemodelan ARIMA

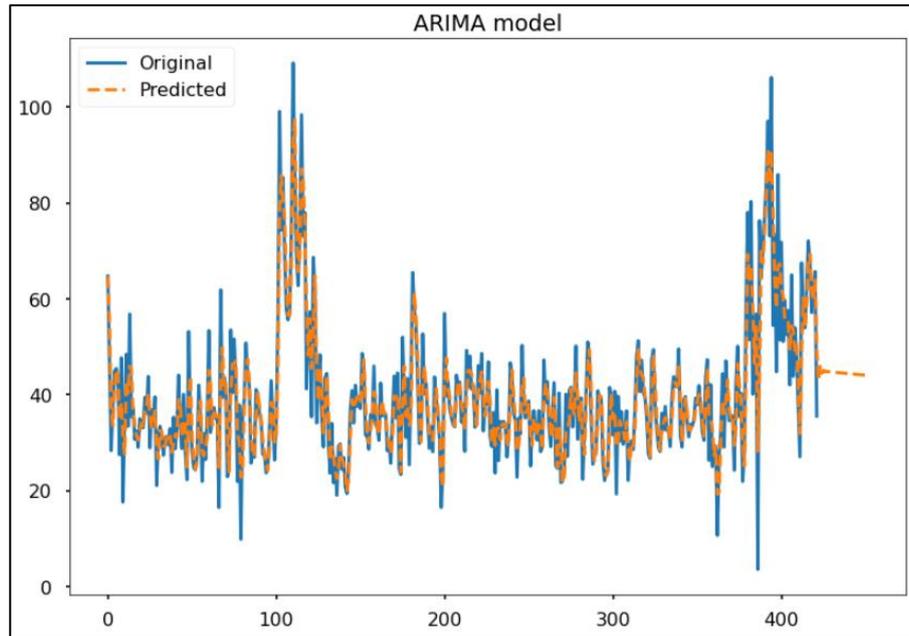
Pada tahap ini dilakukan pemodelan ARIMA dengan menggunakan variabel (p,d,q) . Seperti yang telah dijelaskan pada Tabel 4.1, bentuk *plotting cut off* ataupun *tail off* pada ACF dan PACF akan berpengaruh pada pada performansi model ARIMA. Berdasarkan *plotting* ACF dan PACF yang telah dilakukan, kedua dataset terindikasi memiliki pola *seasonal* sehingga perlu dilakukan *defferencing* yang menyebabkan nilai variabel d akan lebih besar dari 1 sesuai dengan

nilai *defferencing* yang dilakukan ($d > 0$). Untuk *plotting* ACF dan PACF, akan dipilih nilai *cut off* pertama yang akan menjadi nilai variabel p dan q sesuai dengan penjelasan pada Tabel 4.1. Untuk mengetahui performa dari perkiraan tersebut, dilakukan percobaan perhitungan RMSE pada beberapa kombinasi nilai (p,d,q) sebagai berikut:

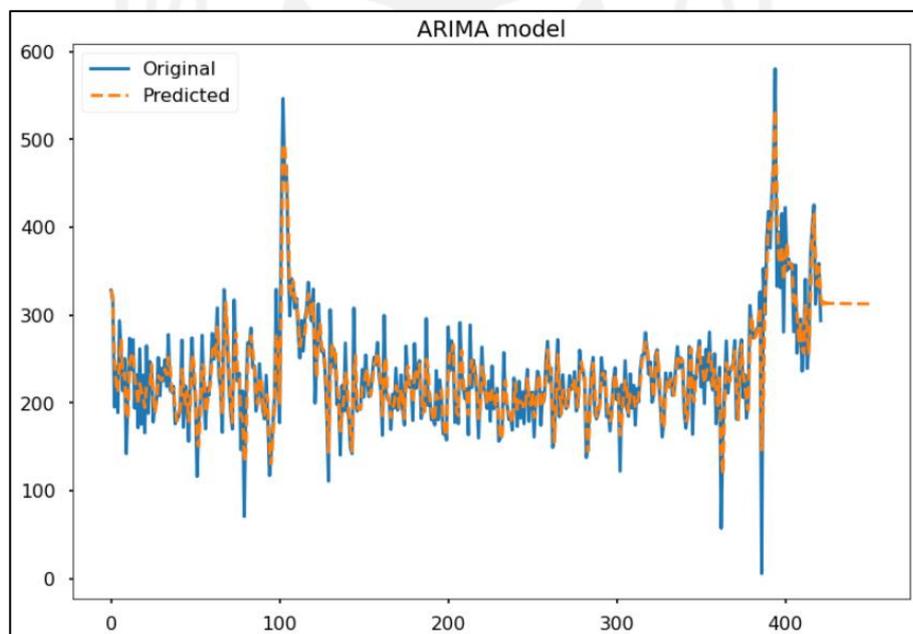
Tabel 5.2 Perbandingan RMSE ARIMA

(p,d,q)	RMSE	
	Dataset 1 (KG)	Dataset 2 (PCS)
(1,0,0)	12,2	53,9
(2,0,0)	8,3	39,0
(0,0,1)	14,2	62,3
(0,0,2)	10,6	49,5
(0,0,4)	9,9	45,5
(0,0,6)	9,5	43,2
(1,1,0)	7,4	34,3
(2,1,0)	7,6	35,6
(0,1,1)	8,6	38,5
(0,1,2)	9,1	42,6
(1,2,0)	16,1	73,1
(0,2,1)	14,0	61,9

Berdasarkan hasil percobaan yang telah dilakukan, didapatkan hasil kombinasi (p,d,q) yang paling optimal yaitu $(1,1,0)$ pada kedua dataset. Kombinasi tersebut menghasilkan nilai RMSE paling rendah dari semua kombinasi, yaitu 7,4% pada Dataset 1 (KG) dan 34,3% pada Dataset 2 (PCS). Nilai variabel p pada $(1,1,0)$ adalah 1 yang menunjukkan bahwa terjadi *cut off* pada Lag-1 dalam *plotting* PACF. Sedangkan untuk variabel d bernilai 1 yang menunjukkan hasil *defferencing* pada Lagi-1 serta variabel q bernilai 0 yang menunjukkan terjadi *lay off* pada *plotting* ACF. Hal ini sesuai dengan teori metode ARIMA dimana perlunya *defferencing* apabila data berpola *seasonal* serta pemilihan nilai *cut off* yang optimal adalah *cut off* paling awal antara *plotting* ACF ataupun PACF. Nilai RMSE tersebut menunjukkan besar kesalahan antara data *training* dan data *testing* dimana data *training* merupakan data asli yang dipakai untuk pembelajaran sedangkan data *testing* merupakan hasil prediksi dari pembelajaran model.



Gambar 5.4 Hasil Pembelajaran Data ARIMA pada Dataset 1 (KG)



Gambar 5.5 Hasil Pembelajaran Data ARIMA pada Dataset 2 (PCS)

5.3 Analisis Pemodelan LSTM

Pada LSTM, diperlukan beberapa langkah untuk menghasilkan suatu pemodelan yang optimal. Pertama adalah pendefinisian model, yaitu dengan mengaktifkan fungsi *Sequential Class*, fungsi *LSTM ()* serta *Dense ()* yang biasanya digunakan dalam LSTM. Selain itu juga diaktifkan

fungsi *Reshape* () untuk mengatur bentuk output dan aktivasi ReLU. Langkah kedua adalah penyusunan model yang akan di optimasi dengan fungsi Adam dan dievaluasi menggunakan fungsi MSE. Langkah ketiga adalah pemasangan jaringan untuk menentukan pola input dan output, dalam hal ini perlu didefinisikan bentuk *epoch*, *batch*, dan *verbose*.

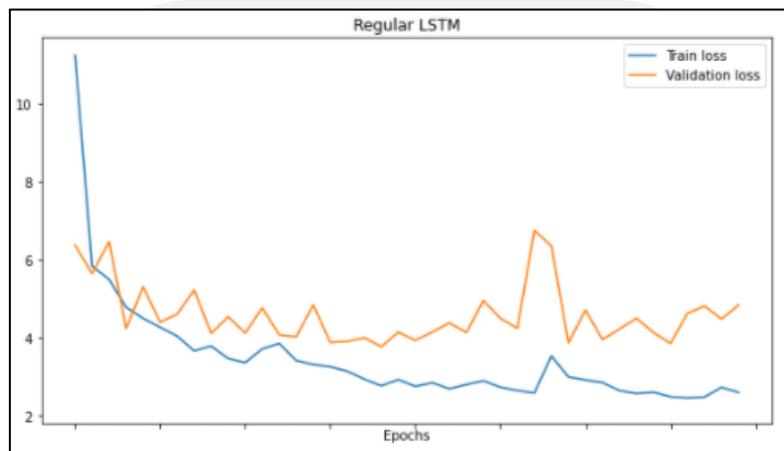
Seperti yang telah dijelaskan, terdapat beberapa fungsi yang perlu untuk didefinisikan bentuknya. Hal ini akan mempengaruhi performa dalam pelatihan data, yang mana dilakukan berdasarkan dengan nilai yang telah ditentukan. Fungsi yang perlu didefinisikan berkaitan dengan performa model diantaranya adalah Dense, Epoch, dan Batch. Variabel Dense akan menentukan jumlah neuron pada dense output layer, variabel Epoch menentukan jumlah pelatihan data, dan variabel Batch menentukan posisi pembaharuan bobot pada Epoch. Untuk mengetahui performa dari setiap parameter tersebut dilakukan evaluasi model menggunakan *root mean square error* (RMSE) yang didapat dari *learning curve* antara dataset pelatihan dan validasi. Berikut merupakan nilai RMSE yang didapatkan:

Tabel 5.3 Perbandingan RMSE LSTM

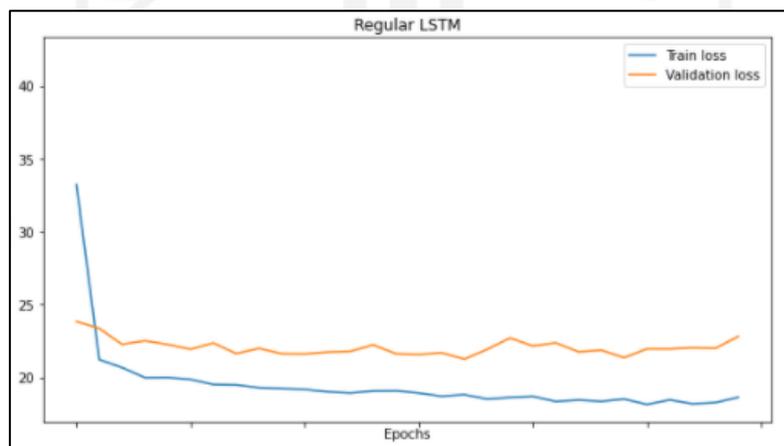
Dense	Epoch	Batch	Dataset 1 (KG)			Dataset 2 (PCS)		
			Train RMSE	Validation RMSE	Selisih	Train RMSE	Validation RMSE	Selisih
1	25	1	1.7503	1.9844	0.2341	4.4597	4.8222	0.3625
1	30	1	1.8012	2.101	0.2998	4.5966	4.9151	0.3185
1	30	6	1.6789	2.0376	0.3587	4.5669	4.9007	0.3338
2	30	1	1.8067	2.0659	0.2592	4.564	4.9169	0.3529
1	40	1	1.7714	2.0493	0.2779	4.4533	4.8215	0.3682
1	40	10	1.7304	2.0641	0.3337	4.3991	4.8203	0.4212
3	40	1	1.799	2.023	0.224	4.4593	4.8393	0.38
4	40	1	1.7234	2.0657	0.3423	4.3923	4.8105	0.4182
3	40	2	1.7424	2.033	0.2906	4.4371	4.8389	0.4018
3	64	1	1.4769	1.993	0.5161	4.2937	4.7618	0.4681
1	64	8	1.4721	2.1279	0.6558	4.3969	4.7878	0.3909

Berdasarkan Tabel 5.2, diketahui bahwa nilai Dense, Epoch, dan Batch yang berbeda akan menghasilkan performansi yang berbeda untuk setiap datanya, dimana dalam hal ini akan semakin sedikit data yang tidak sesuai. Kolom selisih menunjukkan banyaknya perbedaan jumlah *train* RMSE dan *validation* RMSE yang mana akan semakin baik jika selisihnya semakin sedikit. Pada Dataset 1 (KG), performa model paling optimal adalah dengan menggunakan nilai

Dense, Epoch, dan Batch masing- masing 3, 40, dan 1. Artinya pelatihan dilakukan sebanyak 40 kali dengan pembobotan ulang setelah dilakukan 1 kali pelatihan dan hanya terdapat 2 neuron pada output layer. Sedangkan Dataset 2 (PCS), performa model paling optimal adalah dengan menggunakan nilai Dense, Epoch, dan Batch masing- masing 1, 30, dan 1. Artinya pelatihan dilakukan sebanyak 30 kali dengan pembobotan ulang setelah dilakukan 1 kali pelatihan dan hanya terdapat 1 neuron pada output layer. Visualisasi yang didapatkan kedua dataset berdasarkan variabel tersebut adalah:



Gambar 5.3 *Learning Curve* Dataset 1 (KG)



Gambar 5.4 *Learning Curve* Dataset 2 (PCS)

Dari kedua visualisasi tersebut, Dataset 1 (KG) dan Dataset 2 (PCS) menunjukkan bahwa *learning curve* yang dihasilkan termasuk dalam kategori *overfit*. Hal ini banyak terjadi pada pemodelan menggunakan *neural network*. *Learning curve* yang *overfit* ditandai dari grafik *training loss* yang semakin lama semakin menurun dan grafik *validation loss* yang menurun

hingga epoch tertentu namun naik kembali. Artinya data telah terlatih dengan baik, namun kurang baik kemampuannya untuk menggeneralisasi data baru sehingga dapat meningkatkan tingkat kesalahan. Hal ini terjadi dikarenakan model mempelajari detail data dan *noise* secara keseluruhan, sehingga fluktuasi *noise* yang ada ikut dipelajari dan dianggap sebagai konsep dalam model sedangkan hal ini tidak berlaku pada data baru, dengan demikian dapat berdampak negatif pada kemampuan model untuk menggeneralisasi.

5.4 Analisis Performa Metode

Kedua metode yang digunakan telah dievaluasi performanya menggunakan RMSE. Pengevaluasian dilakukan dengan cara yang berbeda, dimana metode ARIMA yang merupakan metode statistik dilakukan dengan membandingkan antara data asli permintaan dengan hasil pemodelan menggunakan metode ARIMA. Sedangkan metode LSTM yang merupakan metode dalam *machine learning* dilakukan dengan membandingkan antara grafik *train loss* yang merupakan hasil pelatihan dengan *validation loss*.

Dari perbandingan nilai RMSE dapat diketahui bahwa metode ARIMA memiliki nilai RMSE terkecil untuk Dataset 1 (KG) sebesar 7,4% sedangkan untuk Dataset 2 (PCS) sebesar 34,3%. Kemudian untuk perbandingan metode LSTM memiliki nilai RMSE untuk *train loss* dan *validation loss* yang jauh lebih kecil, hal ini menandakan nilai kesalahan pembelajaran dan validasi yang dilakukan lebih sedikit. Namun, meskipun nilai RMSE yang dihasilkan metode LSTM kecil, terdapat kesalahan statistik yang menyebabkan *learning curve* pada kedua dataset *overfit* dimana bentuk yang baik adalah *good fit*. Dengan demikian perlu dilakukan pelatihan ulang dengan percobaan beberapa variabel ketika akan dimasukkan data baru

Berdasarkan penjelasan yang ada, diketahui bahwa metode ARIMA lebih optimal untuk digunakan pada kasus pemodelan *perishable product* dengan umur tidak lebih dari 14 hari. Hal ini dikarenakan model LSTM mengalami *overfit* sehingga tidak dapat digunakan untuk memprediksi jumlah stok secara berkelanjutan akibat dari tidak bisanya dimasukkan data baru pada model dikarenakan akan berdampak negatif pada performa model. Selanjutnya dilakukan peramalan untuk menguji performa metode ARIMA. Peramalan dilakukan pada produk Wortel yang merupakan produk dengan jumlah *demand* paling tinggi yaitu sebanyak 8104 transaksi

pada Dataset 1 (KG) dan produk Cabe Rawit dengan jumlah transaksi 9784 pada Dataset 2 (PCS), dalam jangka waktu Januari 2019 – April 2020. Peramalan dilakukan untuk bulan April 2020 dengan data pelatihan sejak Januari 2019 -Maret 2020.

Tabel 5.4 *Forecast* Dataset 1 (KG) ARIMA

Tanggal	<i>Demand</i> (kg)	<i>Forecast</i> (kg)	Selisih (<i>Demand - Forecast</i>)
1 April 2020	8,25	7,309	0,941
2 April 2020	4,98	6,452	-1,472
3 April 2020	6,945	6,354	0,591
4 April 2020	6,585	6,354	0,231
6 April 2020	8,15	7,938	0,212
7 April 2020	4,35	6,538	-2,188
8 April 2020	6,6	6,103	0,497
9 April 2020	4,755	5,875	-1,12
10 April 2020	7,675	5,689	1,986
11 April 2020	4,505	5,689	-1,184
13 April 2020	8,33	7,273	1,057
14 April 2020	4,09	6,306	-2,216
15 April 2020	4,005	5,871	-1,866
16 April 2020	5,085	5,643	-0,558
17 April 2020	3,745	5,457	-1,712
18 April 2020	3,45	5,457	-2,007
20 April 2020	8,285	7,04	1,245
21 April 2020	5,38	6,115	-0,735
22 April 2020	5,065	5,681	-0,616
23 April 2020	6,5	6,15	0,35
24 April 2020	8,17	6,385	1,785
25 April 2020	9,34	6,385	2,955

Tanggal	<i>Demand</i> (kg)	<i>Forecast</i> (kg)	Selisih (<i>Demand - Forecast</i>)
27 April 2020	6,715	8,022	-1,307
28 April 2020	8,925	7,287	1,638
29 April 2020	6,47	6,853	-0,383
30 April 2020	5,01	6,228	-1,218

Tabel 5.4 menunjukkan contoh peramalan untuk jenis produk Wortel pada Dataset 1 (KG). Pada pembuatan model multi-produk yang telah dilakukan sebelumnya menggunakan metode ARIMA, didapati nilai kesalahan atau nilai RMSE terendah pada Dataset 1 (KG) sebesar 7,4%, hal ini dapat dilihat pada Tabel 5.2. Selanjutnya dilakukan uji coba prediksi untuk *single product* pada salah satu produk yang ada pada Dataset 1 (KG) untuk menguji apakah pemilihan pemodelan *multi-product* dapat digunakan untuk peramalan *single product*. Dari Tabel 5.4 menunjukkan bahwa nilai kesalahan yang dihasilkan pada peramalan *single product* yaitu Wortel sebesar 5,1%, sehingga model ini dapat digunakan untuk melakukan peramalan produk secara terpisah atau *single product*.

Tabel 5.5 *Forecast* Dataset KG (PCS) ARIMA

Tanggal	Demand (pcs)	Forecast (pcs)	Selisih (Demand - Forecast)
1 April 2020	50	48,662	1,338
2 April 2020	37	47,878	-10,878
3 April 2020	63	47,878	14,772
4 April 2020	32	46,744	-14,744
6 April 2020	57	52,053	4,947
7 April 2020	34	47,528	-13,528
8 April 2020	51	47,528	3,472
9 April 2020	55	46,744	8,256
10 April 2020	54	46,744	7,256
11 April 2020	39	42,537	-4,037
13 April 2020	45	47,846	-2,846
14 April 2020	37	43,32	-6,82
15 April 2020	49	43,32	5,68

Tanggal	Demand (pcs)	Forecast (pcs)	Selisih (Demand - Forecast)
16 April 2020	44	42,537	1,463
17 April 2020	25	42,537	-17,137
18 April 2020	42	42,537	-0,537
20 April 2020	56	47,846	8,154
21 April 2020	28	43,32	-15,32
22 April 2020	44	43,32	0,68
23 April 2020	49	42,537	6,463
24 April 2020	40	42,537	-2,537
25 April 2020	52	42,537	9,463
27 April 2020	28	44,055	-16,055
28 April 2020	41	39,53	0,97
29 April 2020	48	39,53	7,97
30 April 2020	23	31,018	-8.,179

Sedangkan untuk Tabel 5.5 menunjukkan contoh peramalan untuk jenis produk Cabe Rawit pada Dataset 2 (PCS). Pada pembuatan model multi-produk yang telah dilakukan sebelumnya menggunakan metode ARIMA, didapati nilai kesalahan atau nilai RMSE terendah pada Dataset 2 (PCS) sebesar 34,3%, hal ini dapat dilihat pada Tabel 5.2. Selanjutnya dilakukan uji coba prediksi untuk *single product* pada salah satu produk yang ada pada Dataset 2 (PCS) untuk menguji apakah pemilihan pemodelan *multi-product* dapat digunakan untuk peramalan *single product*. Dari Tabel 5.5 menunjukkan bahwa nilai kesalahan yang dihasilkan pada peramalan *single product* yaitu Cabe Merah sebesar 31,57%, sehingga model ini dapat digunakan untuk melakukan peramalan produk secara terpisah atau *single product*.

BAB VI

PENUTUP

6.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan, diperoleh kesimpulan untuk menjawab rumusan masalah sebagai berikut:

Prediksi permintaan dapat dilakukan menggunakan *machine learning* karena memiliki kemampuan untuk melakukan *supervised learning*, yaitu kemampuan melatih dan mempelajari data untuk mendapatkan output yang ditargetkan. Pemodelan tidak hanya dapat dilakukan pada item tunggal saja, tetapi juga multi-item. Hal ini dibuktikan dari banyaknya penelitian mengenai pemodelan multi-item yang dijadikan model tunggal untuk mengefisienkan pemodelan, seperti halnya dengan Retail Sayur Tosaga yang memiliki 49 item *perishable product*. Dengan demikian, variabel-variabel pemodelan tunggal yang terpilih dapat digunakan pada masing-masing item.

Penggunaan pemrograman Python tidak hanya dapat digunakan pada metode dengan jaringan syaraf, tetapi juga dapat memproses metode statistik karena adanya *library* yang mengakomodasinya. Pada penelitian ini dilakukan pemodelan dengan metode ARIMA dan LSTM menggunakan pemrograman Python. Metode ARIMA merupakan metode statistik yang dapat digunakan untuk berbagai macam pola data, langkah pemodelan ARIMA menggunakan pemrograman Python sama seperti penggunaan konvensional, yaitu perlunya penstasioneran data terlebih dahulu, kemudian melakukan *plotting* ACF dan PACF, dan terakhir pemodelan ARIMA dengan menggunakan hasil dari *plotting* ACF dan PACF. Dengan menggunakan

pemrograman Python dapat dilakukan optimasi performa model berdasarkan nilai (p,q,d) yang berbeda untuk kemudian dievaluasi menggunakan RMSE. Metode ARIMA adalah metode statistik sehingga pengoptimalannya tidak menggunakan *library tertentu*, melainkan dengan nilai (p,q,d) yang dimiliki data tersebut.

Sedangkan metode LSTM merupakan salah satu metode jaringan syaraf yaitu *recurrent neural network* (RNN) yang juga digunakan untuk memproses data berpola stasioner maupun non-stasioner, sehingga tidak dilakukan pengidentifikasian pola. Pada metode ini diperlukan pengaktifan beberapa *library* untuk dapat melakukan pemodelan data. Langkah yang diperlukan dalam menggunakan metode LSTM adalah pendefinisian model terlebih dahulu, yaitu dengan mengaktifkan fungsi *Sequential Class*, fungsi *LSTM ()*, *Dense ()*, fungsi *Reshape ()* dan aktivasi ReLU. Kemudian penyusunan model dengan menggunakan optimasi Adam dan pengevaluasi MSE. Selanjutnya dilakukan pemasangan jaringan untuk menentukan pola input dan output, dalam hal ini perlu didefinisikan bentuk *epoch*, *batch*, dan *verbose*. Dengan demikian pengoptimalan model dapat dilakukan dengan mengubah variabel-variabel yang telah ditentukan sebelumnya untuk mendapatkan performa terbaik yang diukur menggunakan *learning curve* serta RMSE.

Kedua pemodelan telah dilakukan dan menunjukkan nilai RMSE yang berbeda. Metode ARIMA dengan variabel (p,q,d) yang paling optimal adalah (1,1,0) pada kedua dataset menunjukkan nilai RMSE sebesar 6,4% pada Dataset 1 (KG) dan 26,8% pada Dataset 2 (PCS). Sedangkan pada metode LSTM terdapat perbedaan nilai variabel Dense, Epoch, dan Batch optimal pada kedua dataset, yaitu masing-masing adalah 3, 40, dan 1 untuk Dataset 1 (KG) dengan nilai RMSE sebesar 2,023%. Sedangkan untuk Dataset 2 (PCS) masing-masing adalah 1, 30, dan 1 dengan nilai RMSE sebesar 4,915%. Namun meskipun nilai RMSE pada metode LSTM lebih rendah, *learning curve* yang dihasilkan termasuk dalam kategori *overfit*. Dengan demikian, pada kasus pemodelan *perishable product* Retail Sayur Tosaga lebih optimal digunakan metode ARIMA untuk melakukan peramalan permintaan dimasa yang akan datang dikarenakan nilai terendah RMSE pada metode LSTM tidak dapat dijadikan acuan ketika akan dimasukkan data baru akibat *learning curve* yang *overfit*.

6.2 Saran

Terdapat beberapa saran berdasarkan penelitian yang telah dilakukan, yaitu:

1. Untuk perusahaan, dapat dilakukan penamaan identitas yang tetap sehingga tidak terjadi perubahan pada waktu-waktu tertentu yang mengakibatkan adanya identitas ganda. Dengan demikian dapat dilakukan pengolahan data dengan mudah jika dibutuhkan sewaktu-waktu, termasuk untuk pembuatan model prediksi.
2. Untuk penelitian selanjutnya, dapat dilakukan pengidentifikasian dan pengurangan *noise* dengan mengembangkan model seperti simplifikasi model serta menambah fungsi *dropout layer* dan mengembangkan data menggunakan *feature selection* untuk lebih memperkecil nilai eror pada metode ARIMA serta menghindari terjadinya *overfit* pada metode LSTM.



DAFTAR PUSTAKA

- Agarwal, S., 2013, December. Data mining: data mining concepts and techniques. In *2013 International Conference on Machine Intelligence and Research Advancement* (pp. 203-207). IEEE.
- Ali, Ö.G., Sayın, S., Van Woensel, T. and Fransoo, J., 2009. SKU demand forecasting in the presence of promotions. *Expert Systems with Applications*, 36(10), pp.12340-12348.
- Alpaydin, E., 2020. *Introduction to machine learning*. MIT press.
- Alvarado-Valencia, J., Barrero, L.H., Önköl, D. and Dennerlein, J.T., 2017. Expertise, credibility of system forecasts and integration methods in judgmental demand forecasting. *International Journal of Forecasting*, 33(1), pp.298-313.
- Arunraj, N.S. and Ahrens, D., 2015. A hybrid seasonal autoregressive integrated moving average and quantile regression for daily food sales forecasting. *International Journal of Production Economics*, 170, pp.321-335.
- Azadeh, A., Elahi, S., Farahani, M.H. and Nasirian, B., 2017. A genetic algorithm-Taguchi based approach to inventory routing problem of a single perishable product with transshipment. *Computers & Industrial Engineering*, 104, pp.124-133.
- Bandara, K., Shi, P., Bergmeir, C., Hewamalage, H., Tran, Q., & Seaman, B., 2019. Sales demand forecast in e-commerce using a long short-term memory neural network methodology. In *International Conference on Neural Information Processing* (pp. 462-474). Springer, Cham.
- Berry, L. R., 2019. *Bayesian dynamic modeling and forecasting of count time series* (Doctoral dissertation, Duke University).
- Bose, I. and Mahapatra, R.K., 2001. Business data mining—a machine learning perspective. *Information & management*, 39(3), pp.211-225.
- Brownlee, J., 2017. *Long Short-term Memory Networks with Python: Develop Sequence Prediction Models with Deep Learning*. Machine Learning Mastery.
- Cao, K., Kim, H., Hwang, C., & Jung, H., 2018. CNN-LSTM coupled model for prediction of waterworks operation data. *Journal of Information Processing Systems*, 14(6), 1508-1520.

- Casillas, J., Cordón, O., Herrera, F. and Magdalena, L., 2003. Interpretability improvements to find the balance interpretability-accuracy in fuzzy modeling: an overview. In *Interpretability issues in fuzzy modeling* (pp. 3-22). Springer, Berlin, Heidelberg.
- Cleveland, W.P. and Tiao, G.C., 1976. Decomposition of seasonal time series: A model for the Census X-11 program. *Journal of the American Statistical Association*, 71(355), pp.581-587.
- Dietterich, T.G., 1997. Machine-learning research. *AI magazine*, 18(4), pp.97-97.
- Deb, C., Zhang, F., Yang, J., Lee, S.E. and Shah, K.W., 2017. A review on time series forecasting techniques for building energy consumption. *Renewable and Sustainable Energy Reviews*, 74, pp.902-924.
- Dewi, N.R., Susanti, E., Roflin, E., Octalia, T.B. and Novita, R., 2018. Implementasi Model Production Routing Problem with Perishable Inventory (PRPPI) dengan Kebijakan Optimize Delivery-Optimized Selling pada Produksi dan Distribusi Tempe. *Jurnal Matematika Vol*, 8(2), pp.104-113.
- Duan, Y., Yisheng, L. V., & Wang, F. Y., 2016. Travel time prediction with LSTM neural network. In *2016 IEEE 19th international conference on intelligent transportation systems (ITSC)* (pp. 1053-1058). IEEE.
- EFSA Panel on Plant Protection Products and their Residues (PPR), 2014. Scientific Opinion on good modelling practice in the context of mechanistic effect models for risk assessment of plant protection products. *Efsa Journal*, 12(3), p.3589.
- Gers, F. A., Schmidhuber, J., & Cummins, F., 1999. Continual prediction using LSTM with forget gates. In *Neural Nets WIRN Vietri-99* (pp. 133-138). Springer, London.
- Gilliland, M., 2011. Business forecasting effectiveness. *Analytics Magazine*, 16, pp.21-25.
- Han, J., Pei, J. and Kamber, M., 2011. *Data mining: concepts and techniques*. Elsevier.
- Hill, A. V., Zhang, W., & Burch, G. F., 2015. Forecasting the forecastability quotient for inventory management. *International Journal of Forecasting*, 31(3), 651-663.
- Huang, J., Li, Y. F., & Xie, M., 2015. An empirical analysis of data preprocessing for machine learning-based software cost estimation. *Information and software Technology*, 67, 108-127.
- Huber, J., Gossmann, A., & Stuckenschmidt, H., 2017. Cluster-based hierarchical demand forecasting for perishable goods. *Expert systems with applications*, 76, 140-151.
- Hyndman, R.J. and Athanasopoulos, G., 2018. *Forecasting: principles and practice*. OTexts.

- Johnson, A. E., Ghassemi, M. M., Nemati, S., Niehaus, K. E., Clifton, D. A., & Clifford, G. D., 2016. Machine learning and decision support in critical care. *Proceedings of the IEEE*, 104(2), 444-466.
- Jolliffe, I.T. and Stephenson, D.B. eds., 2012. *Forecast verification: a practitioner's guide in atmospheric science*. John Wiley & Sons.
- Karpathy, 2015. "The Unreasonable Effectiveness of Recurrent Neural Networks,".
- Labuza, T.P., 2000. Determination of shelf life of foods. *Department of Food Science and Nutrition, University*, p.32.
- Lee, S.J. and Ouyang, C.S., 2003. A neuro-fuzzy system modeling with self-constructing rule generation and hybrid SVD-based learning. *IEEE Transactions on Fuzzy Systems*, 11(3), pp.341-353.
- Liu, L., Luan, R. S., Yin, F., Zhu, X. P., & Lü, Q., 2016. Predicting the incidence of hand, foot and mouth disease in Sichuan province, China using the ARIMA model. *Epidemiology & Infection*, 144(1), 144-151.
- Lolli, F., Gamberini, R., Regattieri, A., Balugani, E., Gatos, T., & Gucci, S., 2017. Single-hidden layer neural networks for forecasting intermittent demand. *International Journal of Production Economics*, 183, 116-128.
- Luo, L., Luo, L., Zhang, X., & He, X., 2017. Hospital daily outpatient visits forecasting using a combinatorial model based on ARIMA and SES models. *BMC health services research*, 17(1), 469.
- Makridakis, S., Spiliotis, E. and Assimakopoulos, V., 2018. Statistical and Machine Learning forecasting methods: Concerns and ways forward. *PloS one*, 13(3).
- Makridakis, S., Wheelwright S.C, McGee V. dan McGee, E. 1999. Metode dan Aplikasi Peramalan. Alih Bahasa: Ir. Untung Sus Adriyanto, M.Sc dan Ir. Abdul Basith, M.Sc. Edisi Kedua Jilid Satu. Jakarta: Erlangga.
- Mulyana. 2004. Buku Ajar Analisis Deret Waktu. Universitas Padjajaran FMIPA Jurusan Statistika. Bandung
- Munarsih E. 2011. Penerapan Model ARIMANeural Network Hybrid untuk Peramalan Time Series. Thesis. S2 Matematika FMIPA Universitas Gadjah Mada. Yogyakarta
- Nazir, M., 1988. Metode Penelitian. *Jakarta: Ghalia Indonesia*.
- Omar, H., Hoang, V. H., & Liu, D. R., 2016. A hybrid neural network model for sales forecasting based on ARIMA and search popularity of article titles. *Computational intelligence and neuroscience*, 2016.

- Pandey, A., Chaubey, A., Garg, S., Siddiqui, S., & Srinivas, S., 2012. Demand Forecasting for Perishable Products. ISB 2012
- Prastacos, G.P., 1981. Allocation of a perishable product inventory. *Operations Research*, 29(1), pp.95-107.
- Prestwich, S., Rossi, R., Armagan Tarim, S. and Hnich, B., 2014. Mean-based error measures for intermittent demand forecasting. *International Journal of Production Research*, 52(22), pp.6782-6791.
- Punia, S., Singh, S. P., & Madaan, J. K., 2020. From predictive to prescriptive analytics: A data-driven multi-item newsvendor model. *Decision Support Systems*, 113340.
- Qiao, W., Tian, W., Tian, Y., Yang, Q., Wang, Y., & Zhang, J., 2019. The forecasting of PM2.5 using a hybrid model based on wavelet transform and an improved deep learning algorithm. *IEEE Access*, 7, 142814-142825.
- Raghav, P.K., Agarwal, N. and Saini, M., 2016. Edible coating of fruits and vegetables: a review. *International journal of scientific research and modern education*, 1(1), pp.188-204.
- Rahm, E., & Do, H. H., 2000. Data cleaning: Problems and current approaches. *IEEE Data Eng. Bull.*, 23(4), 3-13.
- Ramos, P., Santos, N., & Rebelo, R., 2015. Performance of state space and ARIMA models for consumer retail sales forecasting. *Robotics and computer-integrated manufacturing*, 34, 151-163.
- Sadaei, H. J., e Silva, P. C. D. L., Guimarães, F. G., & Lee, M. H., 2019. Short-term load forecasting by using a combined method of convolutional neural networks and fuzzy time series. *Energy*, 175, 365-377.
- Salinas, D., Flunkert, V., Gasthaus, J., & Januschowski, T., 2020. DeepAR: Probabilistic forecasting with autoregressive recurrent networks. *International Journal of Forecasting*, 36(3), 1181-1191.
- Schuld, M., Sinayskiy, I. and Petruccione, F., 2015. An introduction to quantum machine learning. *Contemporary Physics*, 56(2), pp.172-185.
- Sutanto, P., Setiawan, A., & Setiabudi, D. H., 2017. Perancangan Sistem Forecasting di Perusahaan Kayu UD. 3G dengan Metode ARIMA. *Jurnal Infra*, 5(1), 325-330.
- Syntetos, A.A., Nikolopoulos, K. and Boylan, J.E., 2010. Judging the judges through accuracy-implication metrics: The case of inventory forecasting. *International Journal of Forecasting*, 26(1), pp.134-143.

- Tekin, P. dan Erol, R., 2017. A new dynamic pricing model for the effective sustainability of perishable product life cycle. *Sustainability*, 9(8), p.1330.
- Tyukov, A., Brebels, A., Shcherbakov, M. and Kamaev, V., 2012, December. A concept of web-based energy data quality assurance and control system. In *Proceedings of the 14th International Conference on Information Integration and Web-Based Applications & Services* (pp. 267-271).
- Usman, H. and Purnomo, S., Akbar., 2009. *Metodologi Penelitian Sosial*. Jakarta: Bumi Aksara.
- Wang, X., Wang, M., Ruan, J. and Zhan, H., 2016. The multi-objective optimization for perishable food distribution route considering temporal-spatial distance. *Procedia Computer Science*, 96, pp.1211-1220.
- Zaman, L., Sumpeno, S. and Hariadi, M., 2019. Analisis Kinerja LSTM dan GRU sebagai Model Generatif untuk Tari Remo. *Jurnal Nasional Teknik Elektro dan Teknologi Informasi (JNTETI)*, 8(2), pp.142-150.
- Zhang G. 2003. Time series forecasting using a hybrid ARIMA and neural network model. *J.Neurocomputing* 50:159-175.
- Zhang, R., Ashuri, B. and Deng, Y., 2017. A novel method for forecasting time series based on fuzzy logic and visibility graph. *Advances in Data Analysis and Classification*, 11(4), pp.759-783.
- Zhang, H., Zhang, S., Wang, P., Qin, Y., & Wang, H., 2017. Forecasting of particulate matter time series using wavelet analysis and wavelet-ARMA/ARIMA model in Taiyuan, China. *Journal of the Air & Waste Management Association*, 67(7), 776-788.
- Žliobaitė, I., Bakker, J. and Pechenizkiy, M., 2012. Beating the baseline prediction in food sales: How intelligent an intelligent predictor is?. *Expert Systems with Applications*, 39(1), pp.806-815.