

Multivariate Time Series Forecasting pada Penjualan Barang Retail dengan Recurrent Neural Network

Robertus Bagaskara Radite Putra¹, Hendry²
Universitas Kristen Satya Wacana
Jl. Diponegoro 52-60, Salatiga, Indonesia
Email: 672018061@student.uksw.edu¹, hendry@uksw.edu²

Abstrack – Indonesian retail market is growth along with increasing of population and purchasing power. This opportunity needs to be utilized, but on retail business, sometimes there's a situation in store where it happens Out of Stock or Over Stock. To handle this problem, we can make a forecast or a prediction about the sales on the future. There are many methods on forecasting, but in general they are divide into statistical methods and computational intelligence methods. This research is aimed to forecast retail sales on each day using Recurrent Neural Network (RNN) as part of computational intelligence method. From this research, we can get a result that on retail forecasting sales case, the accuration performance of RNN is better than statistical method.

Keywords – Forecasting, Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU).

Intisari – Pasar ritel di Indonesia semakin berkembang seiring bertambahnya penduduk dan daya beli. Peluang ini harus dimanfaatkan, namun dalam bisnis ritel, kadangkala terjadi keadaan *Out of Stock* maupun *over stock* di dalam toko. Untuk mengatasi hal tersebut, kita bisa mengatasinya dengan melakukan peramalan atau prediksi penjualan yang akan terjadi di masa mendatang. Ada beberapa macam metode untuk melakukan peramalan, namun secara umum terbagi menjadi 2 jenis yaitu metode statistika dan juga *computational intelligence*. Penelitian ini mencoba untuk melakukan prediksi penjualan barang retail perhari menggunakan metode *Recurrent Neural Network* (RNN) sebagai bagian dari metode *computational intelligence*. Dari penelitian ini kita bisa dapatkan hasil bahwa dalam kasus prediksi penjualan ritel, performa akurasi RNN lebih baik dari metode statistika.

Kata Kunci - Forecasting, Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU).

I. PENDAHULUAN

Di Indonesia, industri ritel memiliki kontribusi yang cukup strategis terhadap perekonomian, dan potensi pasar retail Indonesia sendiri termasuk cukup besar dalam konteks global [1]. Berdasarkan penelitian yang dilakukan oleh Technavio, pasar retail Indonesia diprediksi akan naik sebesar 37.32 miliar USD dengan CAGR (*Compound Annual Growth Race*) sebesar 4% dari tahun 2021 hingga 2025 [2]. Melihat potensi seperti ini, sebuah perusahaan ritel terutama perusahaan ritel yang berskala besar, harus secara tepat memperkirakan berapa kebutuhan *inventory* barang yang ada berdasarkan kemungkinan penjualan yang terjadi.

Peramalan (*forecasting*) penjualan yang akurat sangat krusial terhadap keuntungan yang didapatkan perusahaan ritel karena berpengaruh terhadap manajemen stok barang perusahaan [3]. Manajemen stok barang yang baik akan meminimalisir adanya barang *Out of Stock* maupun *Over Stock* yang secara langsung akan berpengaruh terhadap keuntungan yang didapatkan oleh perusahaan. Untuk itu *forecasting* merupakan hal yang cukup penting terhadap jalanya proses bisnis ritel.

Dalam *forecasting*, umumnya data yang digunakan sebagai basis perhitungan adalah data masa lampau dalam *range* waktu tertentu atau biasa disebut dengan data runtun waktu atau *Time Series*. Secara garis besar, terdapat 2 jenis metode dalam *forecasting time series*, yaitu metode statistik (*statistical time series*) dan metode *machine learning* atau *computational intelligence* [4], [5]. Metode statistik yang umum digunakan seperti algoritma *Moving Average* dan *Exponential Smoothing* membutuhkan data yang sifatnya linier, namun dalam realitanya seringkali data yang didapat bersifat non-linier [5]. Metode *deep learning* yang merupakan subset dari *machine learning* memiliki kemampuan yang lebih baik dalam menangkap pola non-linier dalam data, selain itu apabila data yang dimiliki cukup besar, *deep learning* lebih mampu untuk meningkatkan performa apabila dibandingkan dengan metode statistik yang tidak meningkatkan hasil *forecasting* meski data yang tersedia lebih banyak [5], [6].

Salah satu algoritma yang termasuk dalam keluarga *deep learning* adalah RNN (*Recurrent Neural Network*). RNN adalah salah satu kelas dari ANN (*Artificial Neural Network*), dimana arsitektur *hidden layer* nya memperbolehkan output sebelumnya menjadi *input* saat ini dalam *states* [7]. RNN dapat memodelkan ketergantungan temporal dan cocok untuk melakukan prediksi terhadap data yang berurut [8]. Untuk itu RNN akan digunakan dalam penelitian ini karena dirasa cocok digunakan dalam memprediksi penjualan barang ritel dikarenakan data yang dianalisis merupakan data – data yang terurut berdasarkan waktu dan sangat mungkin untuk bersifat non-linear. Selain itu, dalam prakteknya, banyaknya penjualan juga disebabkan oleh beberapa faktor seperti ada tidaknya promo pada barang tertentu, hari – hari spesial dan juga harga barang. Untuk itu, perbedaan dalam penelitian ini adalah proses *forecasting* tidak hanya menggunakan satu variabel yaitu penjualan itu sendiri saja, namun juga mempertimbangkan beberapa faktor lain untuk melakukan prediksi atau disebut juga multivariat.

II. SIGNIFIKANSI STUDI

A. Studi Literatur

Penelitian yang berjudul “Penerapan *Long Short-Term Memory* pada Data *Time Series* Untuk Memprediksi Penjualan Produk PT. Metiska Farma menggunakan Algoritma LSTM”, membahas tentang penerapan algoritma LSTM untuk memprediksi penjualan produk farmasi. Dari penelitian tersebut, didapat bahwa model LSTM dengan pembagian dataset sebesar 90% *training* dan 10% *testing* mendapatkan hasil paling optimal berdasarkan perhitungan RMSE dan MAPE dari data *testing* [8].

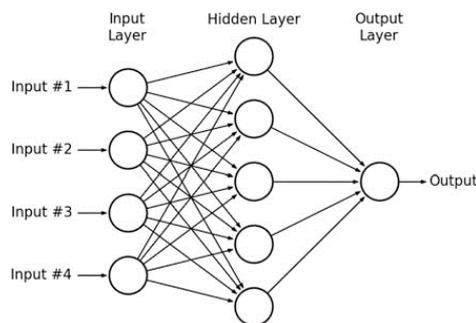
Pada penelitian berjudul “*Daily Retail Demand Forecasting using Machine Learning with Emphasis on calendric special days*”, membahas tentang penerapan beberapa algoritma berbasis *machine learning* untuk memprediksi permintaan (*demand*) berdasarkan data transaksi pada salah satu industri *bakery*, terutama di hari spesial (*special days*) dengan jumlah permintaan yang berbeda dibandingkan hari biasa. Dalam penelitian, didapatkan bahwa performa RNN lebih unggul dari ANN(MLPs), GBRTs, *Linear Regression based model*, *ETS based model*, dan LGBM [9].

Dalam penelitian yang berjudul “*Electricity Price Forecasting Using Recurrent Neural Network*”, dilakukan penelitian guna memprediksi harga listrik berdasarkan data univariat maupun multivariat menggunakan algoritma ANN, LSTM, dan GRU. Dalam penelitian tersebut didapatkan hasil bahwa LSTM dan GRU memiliki hasil yang lebih akurat daripada ANN. Sedangkan jika dilihat dari sisi performa, algoritma GRU lebih baik jika dibandingkan dengan LSTM [10].

Forecasting Multivariate Time Series sendiri adalah peramalan yang menggunakan lebih dari satu kriteria atau variabel yang berubah dari waktu ke waktu [11]. Dengan penggunaan data multivariat ini diharapkan hasil prediksi yang didapat akan lebih akurat daripada menggunakan data univariat, karena model multivariat menyimpan sifat yang berlaku antar

variabel [12]. Menurut jangka waktunya, *forecasting* dibedakan menjadi *forecasting* jangka pendek, menengah dan panjang. *Forecasting* jangka pendek adalah *forecasting* yang menggunakan periode waktu (harian, mingguan, bulanan) ke masa depan, jangka menengah menggunakan periode waktu 1-2 tahun, dan jangka panjang periode waktunya selama beberapa tahun [13]. Dalam penelitian ini, akan digunakan jenis *forecasting* jangka pendek.

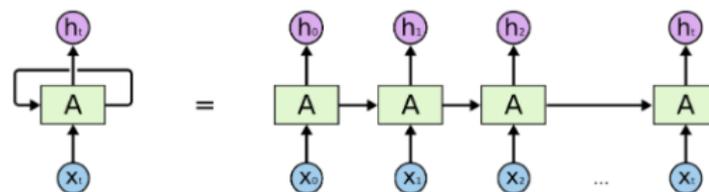
Jaringan syaraf tiruan (*Artificial Neural Network / ANN*) adalah sistem cerdas yang terinspirasi dari sistem saraf manusia yang mengirimkan informasi dari satu *neuron* menuju ke *neuron* yang lain. *Neural Network* menghubungkan beberapa *input nodes* $x_i = (x_1, x_2, \dots, x_n)$ dalam *input layer* dengan satu atau beberapa *output nodes* $y_i = (y_1, y_2, \dots, y_n)$ yang berada pada *output layer* melewati *hidden layer*. Gambaran model ANN kurang lebih seperti berikut:



Gambar 1. Diagram Struktur Arsitektur *Artificial Neural Network*

Dalam penelitian ini, model akan belajar dari pengalaman yang pernah terjadi sebelumnya. Untuk itu kita membutuhkan data masa lalu dimana *output* /prediksi kita nantinya akan dipengaruhi oleh data yang dipelajari [14]–[16].

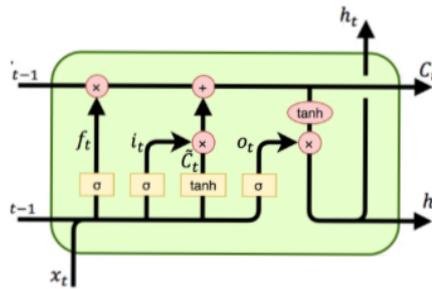
RNN atau disebut juga jaringan umpan balik merupakan pengembangan dari ANN, dimana didalamnya terdapat *loop* yang mengakomodasi *output* pada *network* untuk menjadi *input* pada *network* tersebut yang kemudian akan digunakan untuk menghasilkan *output* baru. RNN mempertahankan memori internal yang memungkinkan mereka untuk mengenali pola/*pattern* yang dinamis.



Gambar 2. Arsitektur *Recurrent Neural Network*

Dari Gambar 2, terlihat pola *looping* yang terkait dalam *sequences* yang menyebabkan RNN dapat menyimpan ingatan sementara, dan nanti dipakai dalam pemrosesan data berikutnya. Algoritma RNN memiliki beberapa variasi pengembangan seperti LSTM (*Long Short Term Memory*) dan GRU (*Gated Recurrent Unit*) [9], [14], [17], [18].

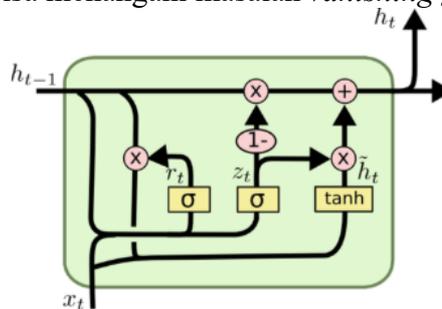
LSTM dan GRU keduanya merupakan modifikasi dari RNN (*Recurrent Neural Network*). LSTM adalah sebuah algoritma pengembangan yang ada untuk mengatasi permasalahan pada RNN yaitu *vanishing gradient*. Hal ini sering terjadi pada data yang memiliki *sequence* yang besar sehingga RNN tidak dapat menangkap *long term dependencies* [19]. Permasalahan yang terdapat pada RNN ini dapat menyebabkan akurasi dari prediksi menggunakan RNN berkurang [20]. Arsitektur LSTM tertuang dalam gambar dibawah ini.



Gambar 3. Arsitektur LSTM

LSTM memiliki 3 *gate* didalamnya, *gate* yang pertama adalah *Input Gate* yang memutuskan nilai mana yang akan diperbarui. Kedua adalah *Forget Gate* yaitu *gate* yang memutuskan informasi apa yang akan dibuang. Dan terakhir adalah *Output Gate* yaitu *gate* yang memutuskan bagian mana saja yang akan dihasilkan.

GRU juga sebuah pengembangan dari RNN, konsep komputasinya lebih sederhana daripada LSTM namun masih bisa menangani masalah *vanishing gradient* [20].



Gambar 4. Arsitektur GRU

GRU memiliki 2 *gate*, yaitu *reset gate* r dan *update gate* z . *Reset gate* adalah *gate* yang menentukan bagaimana cara menggabungkan *input* yang baru kedalam memori sebelumnya, dan *Update gate* menentukan banyaknya memori sebelumnya yang harus disimpan [20].

Ide dasar penggunaan mekanisme *gate* yaitu untuk mempelajari *long-term dependencies*. Kedua algoritma ini dipilih dalam penelitian ini karena menjadi algoritma yang cukup populer sebagai bentuk pengembangan dari RNN dan dari berbagai penelitian memiliki tingkat akurasi yang baik. Selain itu kedua algoritma ini cukup mudah diaplikasikan kedalam program karena sudah terdapat *library* untuk memudahkan penggunaannya dalam bahasa pemrograman *Python*.

Untuk membandingkan performa tiap model RNN, akan digunakan beberapa matriks yang sering dipakai sebagai indikator akurasi tiap model seperti MAE (*Mean Absolute Error*), MAPE (*Mean Absolute Percentage Error*), dan juga MSE (*Mean Squared Error*). Berikut ini ditampilkan rumus dari ketiga pengujian tersebut [9], [21].

$$MSE = \frac{1}{n} \sum (Y_t - Y'_t)^2 \tag{1}$$

$$MAPE = \frac{1}{n} \sum_{t=1}^n \frac{|Y_t - Y'_t|}{Y_t} \tag{2}$$

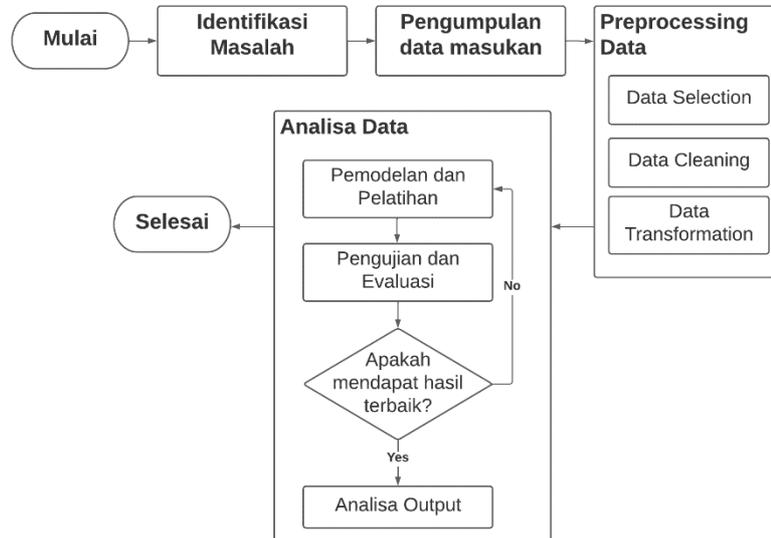
$$MAE = \frac{1}{n} \sum_{t=1}^n |Y_t - Y'_t| \tag{3}$$

Keterangan:

- Y_t = Nilai aktual periode t
- Y'_t = Nilai peramalan periode t
- n = Banyaknya periode

B. Tahap Penelitian

Diagram berikut menggambarkan proses atau alur dalam penelitian ini:



Gambar 5. Tahap Penelitian

Hal yang pertama dilakukan dalam penelitian ini adalah mengidentifikasi masalah yang terdapat dalam industri ritel terkait dengan penyediaan barang dan perencanaan stok barang. Dilanjutkan dengan mengumpulkan dataset yang akan dijadikan sumber *training* untuk model. Data yang sudah didapatkan akan dilanjutkan ke dalam proses *preprocessing* data. Selanjutnya dilakukan proses pembuatan model menggunakan data *training* yang ada. Terakhir, dilakukan proses perbandingan dan evaluasi hasil *forecasting* untuk tiap model dan SKU yang ada.

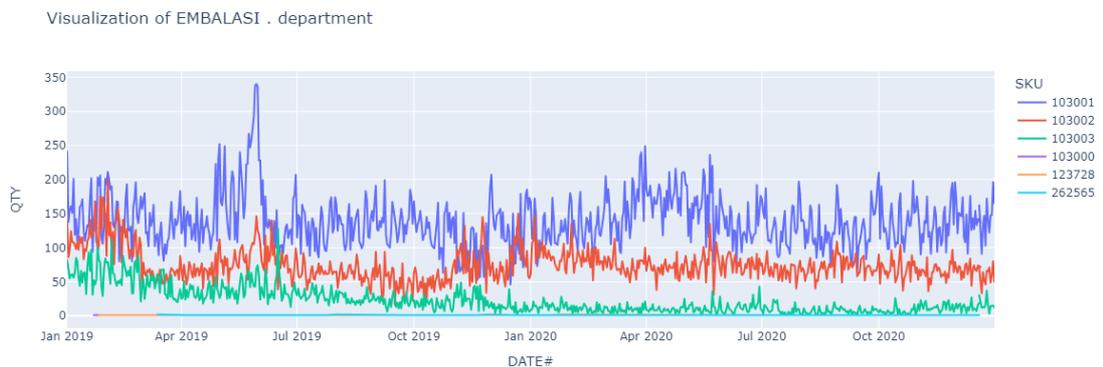
III. HASIL DAN PEMBAHASAN

Data yang digunakan dalam penelitian ini adalah data dalam format .csv berisikan *record* data penjualan salah satu store milik PT. XYZ dari tanggal 01 Januari 2019 hingga 31 Desember 2020. Program *Python* dengan GUI menggunakan *Jupyter Notebook* digunakan untuk membaca dan mem-visualisasikan data yang ada. *Python* sendiri dipilih karena merupakan sebuah bahasa pemrograman yang fleksibel dan banyak didukung oleh *library* yang sangat berguna dalam proses *data analyst*. Gambar 6 menunjukkan gambaran data menjadi sumber penelitian ini.

SKU	DATE#	STORE_CODE	QTY	PRICE	FLAG_PROMO	CTGR	DEPT	FLAG_FESTIVE	DESCP	DESCP_KATAGORI	DESCP_DEPT
103001	2019-01-01		241		1	9011	90	0		EMBALASI	EMBALASI .
103001	2019-01-02		138		1	9011	90	0		EMBALASI	EMBALASI .
103001	2019-01-03		156		1	9011	90	0		EMBALASI	EMBALASI .

Gambar 6. Gambaran Data yang digunakan

Lalu Gambar 7 menunjukkan grafik ilustrasi penjualan produk di salah satu departemen yang ada pada PT. XYZ yaitu departemen embalasi.



Gambar 7. Grafik Penjualan Departemen Embalasi

Terdapat 5.789 jenis produk yang berbeda dan terbagi di dalam 40 departemen produk. Dataset ini memiliki 12 kolom. Tabel 1 menunjukkan tentang detail kolom yang terdapat dalam data.

TABEL I
DETAIL DATA

Nama Kolom	Tipe Data	Deskripsi
SKU	Integer	ID Produk
DATE#	String	Tanggal Transaksi
STORE_CODE	String	Kode Toko
QTY	Integer	Jumlah Penjualan
PRICE	Integer	Harga
FLAG_PROMO	Integer	ID Flag Promo
CTGR	Integer	ID Kategori Produk
DEPT	Integer	ID Departemen Produk
FLAG_FESTIVE	Integer	ID Flag Festive
DESCP	String	Nama Produk
DESCP_KATAGORI	String	Nama Kategori
DESC_DEPT	String	Nama Departemen

Data ini diolah lebih dahulu serta dikonversi kedalam bentuk *dataframe* menggunakan *library* Pandas untuk memudahkan proses eksplorasi dan juga pemodelan menggunakan *Python*.

Tidak setiap hari barang yang ada dalam toko terjual, sehingga jika diurutkan berdasarkan tanggal transaksi lalu data dikelompokan berdasarkan SKU, ada banyak *record* yang nilai *null*. Data yang bernilai *null* ini sebenarnya bisa ditangani dengan melakukan interpolasi data untuk memberikan nilai 0 untuk kolom QTY dalam setiap tanggal yang tidak ada dalam *record*. Namun data yang sangat bergantung dengan kebijakan PT. XYZ sendiri seperti PRICE, FLAG_PROMO dan FLAG_FESTIVE tidak bisa kita lakukan interpolasi.

Sedangkan dalam penelitian *multivariate forecasting*, tidak diperbolehkan terdapat data bernilai *null*. Maka dari itu, kita hanya bisa menggunakan barang yang memiliki *record* dimana semua kolomnya tersedia secara lengkap dengan jumlah *row* yang juga lengkap yaitu sebanyak 731 hari. Gambar 8 adalah output dari program *Python* yang menunjukkan daftar SKU dengan *record* yang lengkap.

```
Product with SKU 8603 has complete 731 days of transactions
Product with SKU 8658 has complete 731 days of transactions
Product with SKU 103001 has complete 731 days of transactions
Product with SKU 103002 has complete 731 days of transactions
Product with SKU 154870 has complete 731 days of transactions
```

Gambar 8. Output Program Analisa SKU

Kelima SKU yaitu 8603, 8658, 103001, 103002 dan 154870 adalah SKU yang akan digunakan dalam penelitian *forecasting* ini untuk menjadi data *training* maupun *testing*.

Beberapa variabel akan digunakan sebagai *feature* untuk memprediksi jumlah penjualan perharinya yaitu PRICE, FLAG_PROMO, dan FLAG_FESTIVES. Variabel CTGR dan DEPT tidak digunakan dalam penelitian ini karena model dibuat untuk prediksi per-SKU, sehingga data *training* memiliki data dimensi SKU yang sama untuk setiap model. Tidak ada *feature* baku yang harus digunakan dalam *forecasting*, namun ada *basic feature* yang sering digunakan di dalam analisa *forecasting time series* berupa *date-based feature*. Dalam penelitian ini kita mengekstraksi *feature* berupa tahun, bulan, hari dalam tahun, hari dalam bulan, hari dalam minggu, dan minggu dalam tahun. Proses ini dinamai *feature engineering*. Gambar 9 menunjukkan hasil ekstraksi dari data tanggal ke dalam beberapa *features* tadi. Pembuatan *time features* ini sendiri memanfaatkan *library Python Pandas*.

DATE#	year	month	dayofyear	dayofmonth	dayofweek	weekofyear
2019-01-01	2019	1	1	1	1	1
2019-01-02	2019	1	2	2	2	1
2019-01-03	2019	1	3	3	3	1
2019-01-04	2019	1	4	4	4	1

Gambar 9. Date Feature Engineering

Proses *training* dalam penelitian ini dilakukan menggunakan *resource* CPU Intel Core i5-8250U. Model ini sendiri dibuat menggunakan dukungan *library* Keras dari Tensorflow. Untuk konfigurasi kedua model yaitu *Bidirectional* GRU dan LSTM dituangkan dalam potongan kode berikut ini:

Kode Program 1. Kode untuk Membuat tiap *Layer Model*

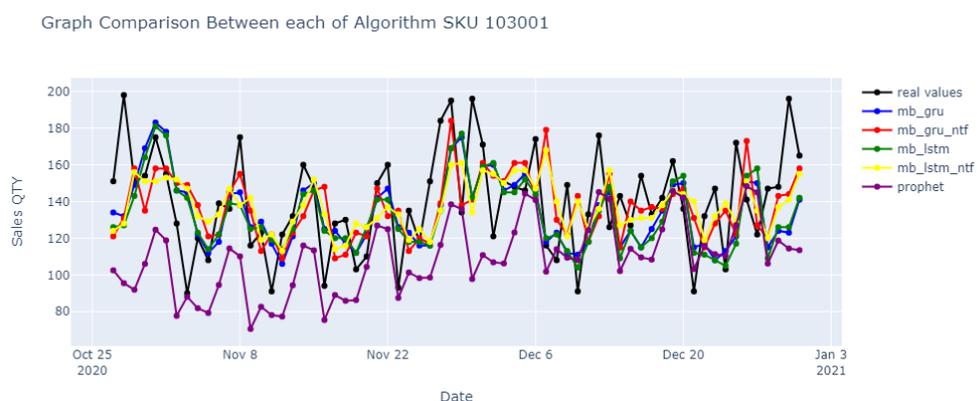
```

1. myModel = keras.Sequential()
2. myModel.add(
3.     keras.layers.Bidirectional(
4.         keras.layers.GRU( # ubah disini untuk mengganti model
5.             units = 128,
6.             input_shape = (XTrain.shape[1], XTrain.shape[2])
7.         )
8.     )
9. )
10. myModel.add(keras.layers.Dropout(rate = 0.2))
11. myModel.add(keras.layers.Dense(units = 1))
12. myModel.compile(loss = "mean_squared_error", optimizer = "adam")

```

Dalam proses *training* ini, digunakan konfigurasi *epoch* sebesar 50 dan juga *n_steps* sebesar 7. Penggunaan *epoch* yang tidak terlalu besar bertujuan untuk mengantisipasi terjadinya *overfitting* yang mungkin terjadi dalam proses *training*.

Dilakukan pembagian *dataset* sebanyak 90% untuk *training* dan 10% untuk *testing*. Untuk menganalisa performa, kita perlu indikator pembandingan dari algoritma lain terutama algoritma yang berbasis statistika. Maka dari itu dalam penelitian ini digunakan pula sebuah *library forecasting* yang cukup terkenal yaitu *Prophet* yang memiliki dasar algoritma statistika *additive model* yang juga tersedia secara *open source* guna implementasi dengan bahasa pemrograman Python [22]. Dalam penelitian ini, penulis juga menguji coba model LSTM dan GRU yang tidak menggunakan *time features* sebagai data *training* dan *testing*. Berikut ini disajikan perbandingan hasil *forecasting* setiap model yang diaplikasikan terhadap kelima SKU yang ada.



Gambar 10. Komparasi Prediksi tiap Model

Gambar 10 menunjukkan grafik hasil *forecasting* untuk SKU 103001. Model – model yang disajikan disini secara berurutan adalah *Multivariate Bidirectional LSTM* baik yang menggunakan *time features* sebagai data *training* dan *testing* maupun tidak, *Multivariate Bidirectional GRU* baik yang menggunakan *time features* sebagai data *training* dan *testing* maupun tidak, serta yang terakhir adalah *Prophet*.

Rekapitulasi tiap indikator *error* disajikan dalam beberapa tabel berikut ini.

TABEL II
KOMPARASI MAE

Algoritma	SKU					MAE (AVG)	RANK
	8603	8658	103001	103002	154870		
<i>Multivariate Bidirectional LSTM</i>	11,2	15,55	21,52	10,73	3,68	12,536	1
<i>Multivariate Bidirectional GRU</i>	12,89	17,74	20,83	10,39	3,71	13,112	3
<i>Multivariate Bidirectional LSTM-No TF</i>	13,18	14,62	22,83	9,88	3,73	12,848	2
<i>Multivariate Bidirectional GRU-No TF</i>	20,3	16,45	23,05	9,83	4,23	14,772	4
<i>Prophet</i>	10,52	14,51	33,28	36,2	3,32	19,566	5

Tabel II merupakan hasil perhitungan MAE dari prediksi setiap model dari tiap – tiap SKU yang ada. Model LSTM memiliki MAE paling kecil daripada model yang lain dan *Prophet* memiliki hasil terbesar untuk MAE.

TABEL III
KOMPARASI MSE

Algoritma	SKU					MSE (AVG)	RANK
	8603	8658	103001	103002	154870		
<i>Multivariate Bidirectional LSTM</i>	221,08	404,76	742,91	192,58	22,8	316,826	1
<i>Multivariate Bidirectional GRU</i>	296,29	674,53	695,05	179,12	22,53	373,504	3
<i>Multivariate Bidirectional LSTM-No TF</i>	265,97	366,17	794,53	178,39	23,97	325,806	2
<i>Multivariate Bidirectional GRU-No TF</i>	657,73	586,27	801,59	168,95	35,17	449,942	4
<i>Prophet</i>	188,23	312,3	1578,78	1595,33	21,9	739,308	5

Tabel III merupakan hasil perhitungan MSE dari prediksi setiap model dari tiap – tiap SKU yang ada. Model LSTM memiliki MAE paling kecil daripada model yang lain dan *Prophet* juga memiliki hasil terbesar untuk MSE.

TABEL IV
KOMPARASI MAPE

Algoritma	SKU					MAPE (AVG)	RANK
	8603	8658	103001	103002	154870		
<i>Multivariate Bidirectional LSTM</i>	48,52	38,82	15,32	18,17	37,03	31,572	1
<i>Multivariate Bidirectional GRU</i>	60,16	47,17	14,97	17,44	37,44	35,436	2
<i>Multivariate Bidirectional LSTM-No TF</i>	63,47	39,75	17,26	16,88	40,41	35,554	3
<i>Multivariate Bidirectional GRU-No TF</i>	87,2	47,33	17,53	16,78	44,85	42,738	5
<i>Prophet</i>	53,17	44,55	22,58	62,17	28,7	42,234	4

Tabel III merupakan hasil perhitungan MAPE dari prediksi setiap model dari tiap – tiap SKU yang ada. Model LSTM memiliki MAPE paling kecil daripada model yang lain dan dalam perhitungan MAPE, GRU tanpa *time features* adalah model dengan tingkat MAPE paling besar, disusul dengan *Prophet*.

Dari ketiga tabel diatas, bisa kita dapatkan hasil bahwa model dengan tingkat *error* terkecil dalam penelitian ini adalah model dengan algoritma LSTM diikuti dengan model GRU dan selanjutnya model yang menggunakan algoritma Prophet apabila dilihat dari skor MAE, MSE dan juga MAPE nya. Semakin kecil tingkat *error* berarti semakin baik akurasi dari model, untuk itu algoritma RNN yang dalam kasus ini adalah LSTM dan GRU cukup mengungguli *Prophet* yang memiliki basis model statistika. Lebih spesifik lagi, algoritma terbaik ya didapat dari hasil penelitian ini adalah LSTM jika dilihat dari akurasi yang didapatkan.

Selain itu, dari penelitian ini, bisa kita peroleh kesimpulan bahwa penggunaan *time features* juga berpengaruh positif terhadap akurasi dari model yang dibuat untuk kasus model RNN.

Akan tetapi performa sebuah model *forecasting* juga ditentukan pula dari kecepatan eksekusi program. Tabel V menunjukkan hasil lama eksekusi dalam satuan detik (*seconds*) dari tiap – tiap model yang ada menggunakan *resource* komputer yang sama.

TABEL V
KOMPARASI LAMA WAKTU EKSEKUSI

Algoritma	SKU					execution times (seconds)
	8603	8658	103001	103002	154870	
Multivariate Bidirectional LSTM	18,8	18,22	17,3	17,07	14,68	17,214
Multivariate Bidirectional GRU	15,12	16,15	15,23	16,8	14,35	15,53
Multivariate Bidirectional LSTM-No TF	15,96	15,82	18,22	16,22	14,62	16,168
Multivariate Bidirectional GRU-No TF	24,39	27,33	27,56	15,96	14,96	22,04
Prophet	0,86	1,34	1,11	1,2	1,36	1,174

Salah satu kekurangan dari metode dengan basis *machine learning* adalah proses *training* yang membutuhkan *resource* lebih besar daripada metode berbasis statistic dikarenakan proses komputasi yang lebih kompleks. Sehingga, apabila dilakukan proses *training* dengan *resource* yang sama, metode *machine learning* yang salah satunya adalah *Recurrent Neural Network* akan membutuhkan proses yang lebih lama daripada metode statistika.

IV. KESIMPULAN

Dari hasil penelitian ini dapat disimpulkan bahwa performa *forecasting* data ritel menggunakan *Recurrent Neural Network* yang pada kasus ini adalah LSTM dan GRU, dari segi akurasi memiliki hasil yang lebih baik daripada metode statistika yang pada kasus ini adalah Prophet. Penggunaan variabel *time features* yang didapatkan dari proses *feature engineering* juga cukup berpengaruh positif terhadap performa akurasi model. Lalu terbukti bahwa performa LSTM merupakan algoritma dengan performa yang paling baik. Namun meski performa RNN lebih baik daripada metode statistika dalam segi akurasi, RNN membutuhkan waktu *training* yang lebih lama daripada metode statistika. Untuk saran penelitian selanjutnya, diharapkan sebelum melakukan proses pemodelan, dilakukan percobaan lebih lanjut terkait parameter yang diberikan terhadap model. Serta terlebih dahulu dilakukan penelitian untuk memilih *feature*

mana yang paling cocok menjadi sumber prediksi terutama untuk data yang tidak linier sesuai dengan dataset dan *use case* masing – masing.

REFERENSI

- [1] M. Yanto, E. Praja Wiyata Mandala, and D. Eka Putri, “Peramalan Penjualan Pada Toko Retail Menggunakan Algoritma Backpropagation Neural Network,” *MEDIA INFORMATIKA BUDIDARMA*, vol. 2, no. 3, pp. 110–117, 2018.
- [2] “Indonesia Retail Market to grow at a CAGR of 4% | CT Corp. and PT Erajaya Swasembada Tbk emerge as Some Key Contributors to growth | 17000 + Technavio Reports.” <https://www.prnewswire.com/news-releases/indonesia-retail-market-to-grow-at-a-cagr-of-4--ct-corp-and-pt-erajaya-swasembada-tbk-emerge-as-some-key-contributors-to-growth--17000--technavio-reports-301399662.html> (accessed Nov. 10, 2021).
- [3] S. Ren, H. L. Chan, and T. Siqin, “Demand forecasting in retail operations for fashionable products: methods, practices, and real case study,” *Annals of Operations Research*, vol. 291, no. 1–2, pp. 761–777, Aug. 2020, doi: 0.1007/s10479-019-03148-8.
- [4] T. Ma, C. Antoniou, and T. Toledo, “Hybrid machine learning algorithm and statistical time series model for network-wide traffic forecast,” *Transportation Research Part C: Emerging Technologies*, vol. 111, pp. 352–372, Feb. 2020, doi: 0.1016/j.trc.2019.12.022.
- [5] H. Abbasimehr, M. Shabani, and M. Yousefi, “An optimized model using LSTM network for demand forecasting,” *Computers and Industrial Engineering*, vol. 143, May 2020, doi: 10.1016/j.cie.2020.106435.
- [6] A. Cecaj, M. Lippi, M. Mamei, and F. Zambonelli, “Comparing Deep Learning and Statistical Methods in Forecasting Crowd Distribution from Aggregated Mobile Phone Data,” *Applied Sciences*, vol. 10, no. 18, p. 6580, Sep. 2020, doi: 10.3390/app10186580.
- [7] “CS 230 - Recurrent Neural Networks Cheatsheet.” <https://stanford.edu/~shervine/teaching/cs-230/cheatsheet-recurrent-neural-networks> (accessed Nov. 16, 2021).
- [8] L. Wiranda and M. Sadikin, “PENERAPAN LONG SHORT TERM MEMORY PADA DATA TIME SERIES UNTUK MEMPREDIKSI PENJUALAN PRODUK PT. METISKA FARMA,” *Prodi Pendidikan Teknik Informatika Universitas Pendidikan Ganesha*, vol. 8, no. 3, pp. 184–196, 2019, [Online]. Available: <https://ejournal.undiksha.ac.id/index.php/janapati/article/view/19139>
- [9] J. Huber and H. Stuckenschmidt, “Daily retail demand forecasting using machine learning with emphasis on calendric special days,” *International Journal of Forecasting*, vol. 36, no. 4, pp. 1420–1438, Oct. 2020, doi: 10.1016/j.ijforecast.2020.02.005.
- [10] U. Ugurlu, I. Oksuz, and O. Tas, “Electricity Price Forecasting Using Recurrent Neural Networks,” *Energies (Basel)*, vol. 11, no. 5, p. 1255, May 2018, doi: 10.3390/en11051255.
- [11] S. Zahara and Sugianto, “Peramalan Data Indeks Harga Konsumen Berbasis Time Series Multivariate Menggunakan Deep Learning,” *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 5, no. 1, pp. 24–30, Feb. 2021, doi: 10.29207/resti.v5i1.2562.
- [12] D. A. Lusia and A. Ambarwati, “PERBANDINGAN PERAMALAN UNIVARIAT DAN MULTIVARIAT ARIMA PADA INDEKS HARGA SAHAM GABUNGAN,” *Jurnal Statistika Universitas Muhammadiyah Semarang*, vol. 6, no. 2, pp. 78–84, 2018, [Online]. Available: <http://finance.yahoo.com/q/hp?s=%5JKS>

- [13] A. Nurlifa and S. Kusumadewi, "Sistem Peramalan Jumlah Penjualan Menggunakan Metode Moving Average Pada Rumah Jilbab Zaky," *INOVTEK Polbeng - Seri Informatika*, vol. 2, no. 1, p. 18, Jun. 2017, doi: 10.35314/isi.v2i1.112.
- [14] M. Abdul Dwiyanto Suyudi, E. C. Djamal, A. Maspupah Jurusan Informatika, and F. Sains dan Informatika Universitas Jenderal Achmad Yani Cimahi, "Prediksi Harga Saham menggunakan Metode Recurrent Neural Network," *Seminar Nasional Aplikasi Teknologi Informasi (SNATi)*, pp. 1907–5022, 2019.
- [15] A. Chawla, A. Singh, A. Lamba, N. Gangwani, and U. Soni, "Demand forecasting using artificial neural networks—A case study of american retail corporation," in *Advances in Intelligent Systems and Computing*, vol. 697, Springer Verlag, 2019, pp. 79–89. doi: 10.1007/978-981-13-1822-1_8.
- [16] R. Khaldi, A. el Afia, R. Chiheb, and R. Faizi, "Artificial Neural Network Based Approach for Blood Demand Forecasting: Fez Transfusion Blood Center Case Study," in *BDCA'17: Proceedings of the 2nd international Conference on Big Data, Cloud and Applications*, Mar. 2017, vol. Part F129474. doi: 10.1145/3090354.3090415.
- [17] D. Tarkus, S. R. U. A. Sompie, and A. Jacobus, "Implementasi Metode Recurrent Neural Network pada Pengklasifikasian Kualitas Telur Puyuh," *Jurnal Teknik Informatika*, vol. 15, no. 2, pp. 137–144, 2020.
- [18] S. Alhirmizy and B. Qader, "Multivariate Time Series Forecasting with LSTM for Madrid, Spain pollution," in *2019 International Conference on Computing and Information Science and Technology and Their Applications (ICCISTA)*, 2019, pp. 1–5. doi: 10.1109/ICCISTA.2019.8830667.
- [19] A. Saxena and T. R. Sukumar, "Predicting bitcoin price using lstm And Compare its predictability with arima model," *International Journal of Pure and Applied Mathematics*, vol. 119, no. 17, Feb. 2018, doi: 10.13140/RG.2.2.15847.57766.
- [20] A. Hanifa, S. A. Fauzan, M. Hikal, and M. B. Ashfiya, "PERBANDINGAN METODE LSTM DAN GRU (RNN) UNTUK KLASIFIKASI BERITA PALSU BERBAHASA INDONESIA," *Dinamika Rekayasa*, vol. 17, no. 1, pp. 33–39, 2021, doi: 10.20884/1.dr.2021.17.1.436.
- [21] G. Ayuni and D. Fitrihanah, "Penerapan Metode Regresi Linear Untuk Prediksi Penjualan Properti pada PT XYZ," *Jurnal Telematika*, vol. 14, no. 2, pp. 79–86, 2020, [Online]. Available: <https://journal.ithb.ac.id/telematika/article/view/321>
- [22] S. J. Taylor and B. Letham, "Forecasting at Scale," *The American Statistician*, vol. 72, no. 1, pp. 37–45, Jan. 2018, doi: 10.1080/00031305.2017.1380080.

UCAPAN TERIMA KASIH

Terima kasih disampaikan kepada Tim *Jurnal Informatika Polbeng* yang telah meluangkan waktu untuk mengevaluasi jurnal ini dan juga kepada semua pihak yang ikut berperan dalam proses penelitian ini.