

## PREDIKSI POLUSI UDARA KOTA JAKARTA MENGGUNAKAN RECURRENT NEURAL NETWORK-GATED RECURRENT UNITS

I.M.N. Yudiskara<sup>1</sup>, I.B.G. Dwidasmara<sup>2</sup>, dan I.M. Widiartha<sup>3</sup>.

### ABSTRAK

Udara bersih memiliki peran penting bagi kehidupan manusia dalam melakukan aktivitas, terutama aktivitas di luar ruangan. Informasi tentang pencemaran udara sangat bermanfaat untuk berbagai bidang, termasuk bidang kesehatan khususnya organ pernapasan manusia. Prakiraan polusi udara yang akurat diperlukan untuk mengurangi masalah ini. Polusi udara secara langsung berdampak pada kesehatan manusia melalui pembuangan polutan dan partikulat. Di Indonesia, terutama di kota-kota berpenduduk padat seperti Jakarta, polusi udara menjadi perhatian lebih. Jurnal ini menjelaskan proses pembuatan model machine learning dengan Gated Recurrent Units dan dilatih menggunakan data polusi udara di Jakarta dari tahun 2010 sampai tahun 2021 dengan, tingkat keakuratan prediksi dihitung menggunakan perhitungan RMSE. Hasil prediksi diharapkan dapat digunakan sebagai acuan dalam menentukan kebijakan pemerintah kota untuk menangani polusi udara kedepan.

**Kata kunci :** Polusi, Prediksi, Kesehatan, Gated Recurrent Units, Lingkungan .

### ABSTRACT

Clean air plays an important role in human life when conducting activities, especially outdoor activities. Air pollution information is very useful in a variety of areas, including the health sector, especially the human respiratory system. Accurate air pollution forecasts are needed to improve the performance of this sector. Air pollution has had a direct impact on human health by releasing pollutants and particles. Air pollution is a source of concern in Indonesia, especially in densely populated cities such as Jakarta. This paper describes the process of making machine learning models with Gated Recurrent Units which then trained using air pollution data in Jakarta from 2010 to 2021, the level of prediction accuracy is calculated using RMSE calculations. The prediction hopefully to be used as a reference in determining city government policies to address air pollution in the future.

**Keywords:** Pollution, Prediction, Health, Gated Recurrent Units, Environment.

### 1. PENDAHULUAN

Udara bersih memiliki peran penting bagi kehidupan manusia dalam melakukan aktivitas, terutama aktivitas di luar ruangan. Informasi tentang pencemaran udara sangat bermanfaat untuk berbagai bidang, termasuk bidang kesehatan khususnya organ pernapasan manusia. Prakiraan polusi udara yang akurat diperlukan untuk mengurangi masalah ini. Polusi udara secara langsung berdampak pada kesehatan manusia melalui pembuangan polutan dan partikulat. Di Indonesia, terutama di kota-kota berpenduduk padat seperti Jakarta, polusi udara menjadi perhatian lebih. Terdapat banyak risiko akibat polusi udara, termasuk risiko kanker. Dengan demikian, dengan meningkatnya masalah yang berasal dari polusi, prakiraan polusi udara

---

<sup>1</sup> Mahasiswa Prodi Informatika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Udayana Badung 80362 Bali yudiskara@gmail.com

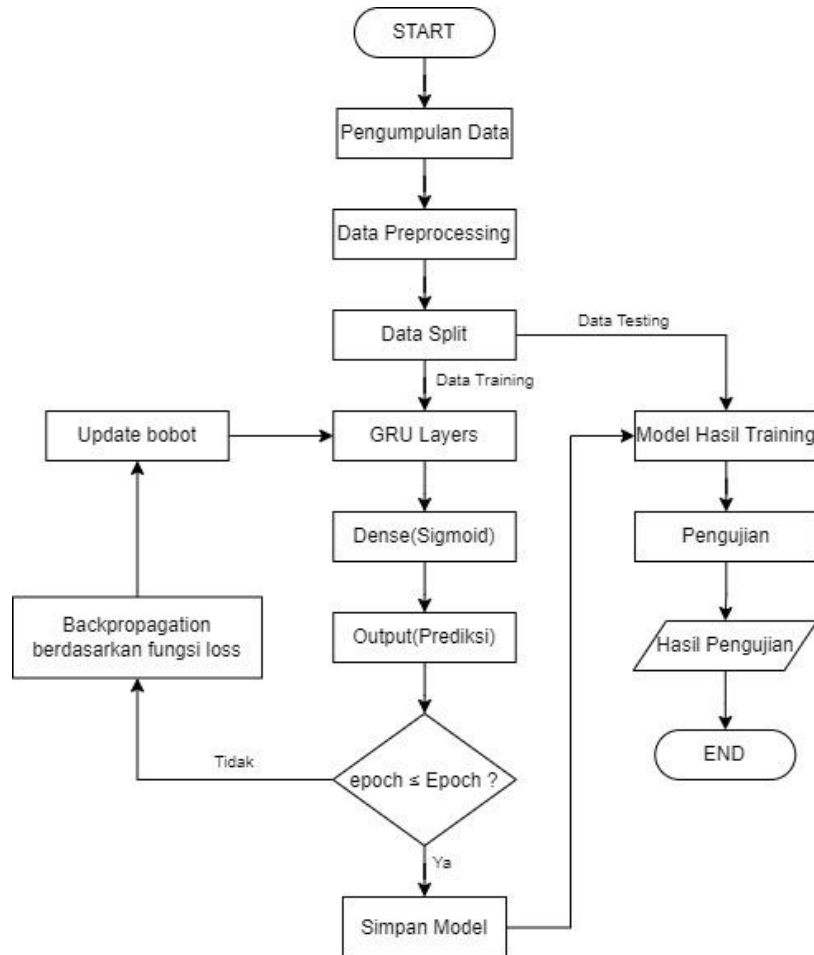
<sup>2</sup> Staf Pengajar Prodi Informatika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Udayana Badung 80362 Bali dwidasmara@unud.ac.id

<sup>3</sup> Staf Pengajar Prodi Informatika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Udayana Badung 80362 Bali madewidiartha@unud.ac.id

yang akurat memainkan peran penting dalam manajemen kualitas udara dan pencegahan populasi terhadap polusi. Sebuah sistem yang dapat menghasilkan peringatan berdasarkan kualitas udara dapat memainkan peran penting dalam peringatan kesehatan ketika tingkat polusi udara melebihi tingkat tertentu. Sistem ini juga diharapkan dapat memberi para regulator lingkungan pilihan untuk melakukan pengurangan emisi, atau bahkan tanggap darurat jika prakiraan menunjukkan polusi yang lebih tinggi di masa depan.

## 2. METODE PELAKSANAAN

Untuk proses pembuatan model machine learning dilakukan dengan tiga tahapan yaitu pengumpulan data, preprocessing data dan proses pelatihan model machine learning



**Gambar 2.1.** Flowchart Metode Pelaksanaan

### 2.1. Pengumpulan Data

Data diperoleh dari website [data.jakarta.go.id](http://data.jakarta.go.id), dataset yang digunakan merupakan data dari tahun 2010 sampai tahun 2021 dataset ini bersumber dari Dinas Lingkungan Hidup Provinsi DKI Jakarta. Dataset ini berisi mengenai Indeks Standar Pencemar Udara (ISPU) yang diukur dari 5 stasiun pemantau kualitas udara (SPKU) yang ada di Provinsi DKI Jakarta untuk setiap harinya.

### 2.2. Data Preprocessing

Data preprocessing merujuk pada manipulasi atau penghapusan data sebelum digunakan untuk memastikan atau meningkatkan kinerja data, dan merupakan langkah penting dalam proses data mining dan machine learning. Berikut merupakan tampilan awal data yang diperoleh

	tanggal	pm10	so2	co	o3	no2	max	critical	kategori	lokasi_spku
0	1/1/2010	60.0	4.0	73.0	27.0	14.0	73.0	CO	SEDANG	JAKARTA
1	1/2/2010	32.0	2.0	16.0	33.0	9.0	33.0	O3	BAIK	JAKARTA
2	1/3/2010	27.0	2.0	19.0	20.0	9.0	27.0	PM10	BAIK	JAKARTA
3	1/4/2010	22.0	2.0	16.0	15.0	6.0	22.0	PM10	BAIK	JAKARTA
4	1/5/2010	25.0	2.0	17.0	15.0	8.0	25.0	PM10	BAIK	JAKARTA

**Gambar 2.2.** Dataset Indeks Standar Pencemar Udara

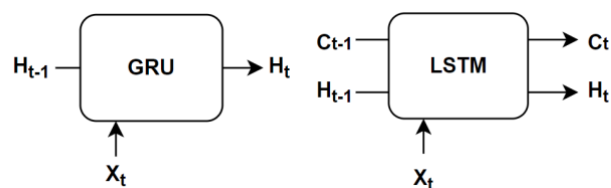
Pada dataset tersebut yang pertama dilakukan adalah menghapus kolom “critical”, “kategori”, “max”, dan “lokasi\_spku”. Kolom critical dihapus karena hanya merupakan kolom dengan nilai maksimal pada baris data, kolo, kategori dihapus karena model machine learning kali ini digunakan untuk prediksi partikel udara setiap harinya dan tidak digunakan untuk melakukan klasifikasi, kolom max dihapus karena merupakan hanya merupakan nilai paling besar pada setiap kolom sehingga tidak digunakan untuk prediksi, kemudian yang terakhir lokasi\_spku dihapus karena semua lokasi\_spku sama yaitu Jakarta. Setelah menghapus empat kolom tersebut selanjutnya dilakukan proses tranfomasi data yang pertama mengubah format pada kolom tanggal dengan format yang diterima oleh library tensorflow yaitu “yyyy-mm-dd” selanjutnya, nilai pada data di normalisasi dengan min-max scaler dengan menggunakan rumus:

Dimana  $x'$  merupakan data yang telah di normalisasi dan  $x$  merupakan nilai data awal

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)}$$

### 2.3. Gated Recurrent Unit (GRU)

GRU atau Gated Recurrent Unit adalah peningkatan dari RNN atau Recurrent Neural Network, yang pertama kali diperkenalkan oleh Kyunghyung Cho et al pada tahun 2014. GRU sangat mirip dengan Long Short Term Memory (LSTM) dan relatif baru dibandingkan dengan LSTM. Sama seperti LSTM, GRU menggunakan gates untuk mengontrol aliran informasi. GRU menawarkan beberapa peningkatan dari LSTM dan memiliki arsitektur yang lebih sederhana. Perbedaan GRU dengan LSTM adalah GRU tidak memiliki Cell State ( $C_t$ ), dan hanya memiliki hidden state ( $H_t$ ). Karena arsitektur yang sederhana ini, proses training GRU lebih cepat dibandingkan dengan LSTM



**Gambar 2.3.** Perbandingan arsitektur layer GRU dan LSTM

Pada setiap timestamp ( $t$ ), GRU mengambil input  $X_t$  dan hidden state  $H_{t-1}$  dari timestamp  $t-1$  sebelumnya, kemudian mengeluarkan sebuah hidden state baru  $H_t$  yang nantinya akan diteruskan ke timestamp berikutnya. GRU memiliki dua gate yaitu reset gate dan update gate.

- Reset Gate

Reset gate bertanggung jawab pada bagian short term memory pada jaringan neural network yaitu hidden state untuk menentukan seberapa banyak informasi di masa lalu ( $t-1$ ) yang perlu diabaikan. Berikut merupakan persamaan dari Reset Gate.

$$r_t = \sigma(x_t * U_r + H_{t-1} * W_r)$$

Nilai  $r_t$  akan memiliki rentang nilai dari 0 sampai 1 karena  $r_t$  merupakan fungsi sigmoid.  $U_r$  dan  $W_r$  adalah nilai bobot dari reset gate.

- Update Gate

Update Gate bertanggung jawab untuk menentukan seberapa banyak informasi di masa lalu ( $t-1$ ), yang perlu diteruskan ke state berikutnya, berikut merupakan persamaan dari Update Gate.

$$u_t = \sigma(x_t * U_u + H_{t-1} * W_u)$$

Nilai  $u_t$  akan memiliki rentang nilai dari 0 sampai 1 karena  $u_t$  merupakan fungsi sigmoid.  $U_u$  dan  $W_u$  adalah nilai bobot dari update gate.

Untuk mendapatkan hidden state ( $H_t$ ) pada GRU, perlu mengikuti dua langkah, yang pertama menentukan kandidat hidden state dengan mengambil input dan hidden state dari timestamp  $t-1$  sebelumnya yang kemudian dikalikan dengan output reset gate  $r_t$ . Kemudian meneruskan informasi ini ke fungsi tahn nilai hasilnya adalah kandidat hidden state.

$$\hat{H}_t = \text{tahn}(x_t * U_g + (r_t \circ H_{t-1}) * W_g)$$

Pada tahap ini reset gate berfungsi untuk menentukan pengaruh hidden state sebelumnya pada kandidat hidden state. Jika nilai  $r_t$  adalah 1 maka semua informasi pada hidden state  $H_{t-1}$  dapat dipertimbangkan, sebaliknya jika 0 maka semua informasi pada hidden state sebelumnya diabaikan. Selanjutnya kandidat hidden state digunakan untuk menghasilkan hidden state  $H_t$ . Persamaan hidden state dipengaruhi oleh Update Gate

$$H_t = u_t \circ H_{t-1} + (1-u_t) \circ \hat{H}_t$$

Misalkan nilai dari  $u_t$  berada pada kisaran 0, maka suku pertama dari persamaan akan hilang.

$$H_t = \mathbf{u}_t \circ \mathbf{H}_{t-1} + (1-u_t) \circ \hat{H}_t$$

Yang berarti hidden state yang baru tidak akan memiliki banyak informasi dari hidden state sebelumnya. Sebaliknya suku kedua akan mendekati 1 yang berarti hidden state pada saat ini hanya akan terdiri dari informasi dari kandidat hidden state. Jika suku ke dua mendekati 0 maka hidden state saat ini akan tergantung pada suku pertama yaitu informasi hidden state pada timestamp  $t-1$  sebelumnya

## 2.4. Training Model

Setelah melakukan preprocessing data maka selanjutnya dilakukan proses training menggunakan data yang telah di preprocess selanjutnya akan dibagi menjadi data training dan data testing yang mana data testing ini akan digunakan untuk menguji performa prediksi dari model machine learning yang telah dibuat, 80% data akan digunakan sebagai data training dan sisanya akan digunakan sebagai data testing. Berikut merupakan arsitektur model machine learning yang telah dibuat

Layer (type)	Output Shape	Param #
gru (GRU)	(None, None, 256)	202752
gru_1 (GRU)	(None, None, 512)	1182720
gru_2 (GRU)	(None, None, 256)	591360
dense (Dense)	(None, None, 5)	1285
Total params: 1,978,117		
Trainable params: 1,978,117		
Non-trainable params: 0		

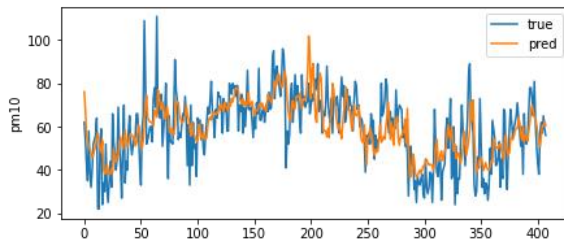
**Gambar 2.4.** Arsitektur model machine learning

Model machine learning ini dibuat menggunakan library tensorflow dan keras, arsitektur machine learning ini memiliki tiga layer GRU(Gated Recurrent Unit). GRU merupakan cel dengan kandungan 2 gate dan 3 fungsi aktivasi. Dengan gate dan fungsi aktivasi yang minim ini tentunya akan mempercepat proses pengolahan data yang umumnya berjumlah sangat besar. Kemampuan GRU dirancang untuk menjadi lebih baik dari LSTM terutama untuk dataset yang jumlahnya sedikit. Model ini dilatih dengan 20 epoch

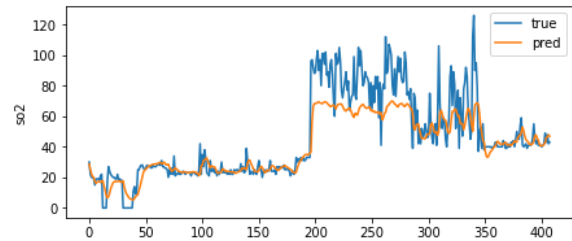
### 3. HASIL DAN PEMBAHASAN

#### 3.1 Tahap Testing

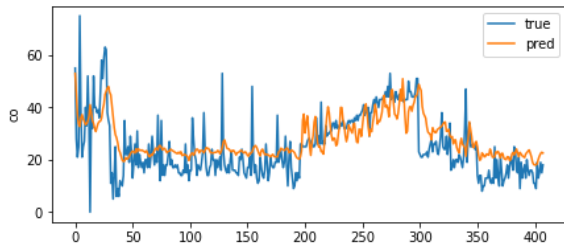
Pada tahap ini dilakukan pengujian model yang sudah di training terhadap data test. Data testing merupakan data yang belum pernah di prediksi oleh model sehingga akan dilakukan pengujian pada tersebut dengan membandingkan hasil prediksi dan nilai sebenarnya pada hari tersebut. Berikut merupakan plot perbandingan nilai prediksi dan nilai sebenarnya pada data



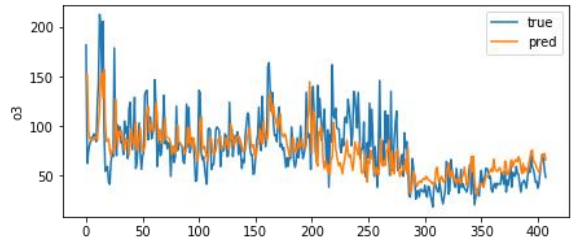
Gambar 3.1. Plot prediksi partikulat pm10



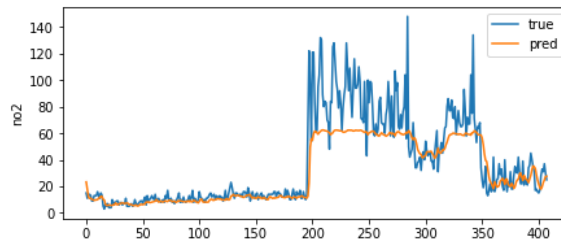
Gambar 3.2. Plot prediksi partikulat so2



Gambar 3.3. Plot prediksi partikulat co



Gambar 3.4. Plot prediksi partikulat o3



Gambar 3.5. Plot prediksi partikulat no2

Berikut merupakan nilai RMSE dari setiap partikulat yang diprediksi:

pm10	so2	co	o3	no2
9.80446512	8.77282145	7.24068196	18.02030243	10.63599659

### 3.2 Integrasi Model Machine Learning Pada Aplikasi

Model machine learning yang telah dibuat kemudian diupload ke google cloud, kemudian API yang sudah dibuat sebelumnya dibuat untuk menghubungkan model machine learning dengan aplikasi android menggunakan flask, dan data diletakan di big query agar dapat di proses oleh model Machine learning berikut merupakan tampilan output get request dari untuk prediksi 7 hari ke depan

```
[{'tanggal': Timestamp('2021-06-09 00:00:00'),
  'pm10': 60.78,
  'so2': 45.52,
  'co': 24.51,
  'o3': 70.95,
  'no2': 25.88,
  'max': 70.95,
  'city': 'JAKARTA'},
 {'tanggal': Timestamp('2021-06-10 00:00:00'),
  'pm10': 60.49,
  'so2': 44.99,
  'co': 23.63,
  'o3': 85.96,
  'no2': 26.28,
  'max': 85.96,
  'city': 'JAKARTA'},
 {'tanggal': Timestamp('2021-06-11 00:00:00'),
  'pm10': 58.04,
  'so2': 44.28,
  'co': 22.94,
  'o3': 90.09,
  'no2': 25.31,
  'max': 90.09,
  'city': 'JAKARTA'},
 {'tanggal': Timestamp('2021-06-12 00:00:00'),
  'pm10': 55.12,
  'so2': 43.18,
  'co': 22.32,
  'o3': 90.16,
  'no2': 23.84,
  'max': 90.16,
  'city': 'JAKARTA'},
 {'tanggal': Timestamp('2021-06-13 00:00:00'),
  'pm10': 52.44,
  'so2': 41.88,
  'co': 21.81,
  'o3': 89.72,
  'no2': 22.3,
  'max': 89.72,
  'city': 'JAKARTA'},
 {'tanggal': Timestamp('2021-06-14 00:00:00'),
  'pm10': 50.17,
  'so2': 40.51,
  'co': 21.46,
  'o3': 89.6,
  'no2': 20.81,
  'max': 89.6,
  'city': 'JAKARTA'},
 {'tanggal': Timestamp('2021-06-15 00:00:00'),
  'pm10': 48.28,
  'so2': 39.13,
  'co': 21.24,
  'o3': 89.76,
  'no2': 19.38,
  'max': 89.76,
  'city': 'JAKARTA'}]
```

Gambar 3.6. Output GET request

## 4. KESIMPULAN

Model machine learning yang telah dibuat telah berhasil memprediksi partikulat udara di kota jakarta dengan cukup baik. Model ini juga sudah berhasil diintegrasikan pada sebuah aplikasi android sehingga memudahkan pemantauan kadar udara

## UCAPAN TERIMA KASIH

Atas selesai hingga terbitnya artikel ini, ucapan terima kasih dan penghargaan yang setinggi-tingginya diberikan kepada Bapak Ida Bagus Dwidasmara selaku Pembimbing Akademik atas segala bantuannya selama ini, serta atas kebersamaan dan dukungan yang begitu luar biasa dari teman-teman Mahasiswa Program Studi Informatika. Selain itu saya juga ingin berterima kasih Ditjen Dikti menyelenggarakan program Bangkit dan memberikan kesempatan kepada saya untuk mengikuti program tersebut.

## DAFTAR PUSTAKA

Mauro C., Fabiana M. C. , Aleš P., Sara S., Leonardo V. (2020). A Machine Learning Approach to Predict Air Quality in California, Complexity, **Vol. 2020**, pp. 1-23.  
Jan K. D., Rasa Z., Mario G., Yves R. (2017), Modeling PM2.5 Urban Pollution Using Machine Learning and Selected Meteorological Parameters, *Journal of Electrical and Computer Engineering*, **Vol. 2017**, pp. 1-14.  
Madhuri V.M., Samyama G. G. H., Savitha K. (2020), Air Pollution Prediction Using Machine Learning Supervised Learning Approach, *International Journal Of Scientific & Technology Research*, **Vol. 9**, pp. 118-123.

- U. A. Hvidtfeldt, M. Ketzel, M. Sørensen et al. (2018), Long-term residential exposure to PM<sub>2.5</sub>, PM<sub>10</sub>, black carbon, NO<sub>2</sub>, and ozone and mortality in a Danish cohort, *Environment International*, **Vol. 123**, pp. 265-272.
- L. Pimpin, L. Retat, D. Fecht et al. (2018), Estimating the costs of air pollution to the National Health Service and social care: an assessment and forecast up to 2035, *PLoS Medicine*, **Vol. 15**, pp. 1–16.
- Verma, Ishan, Rahul Ahuja, Hardik Meisheri, and Lipika Dey (2018). Air pollutant severity prediction using Bi-directional LSTM Network, *IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, **Vol. 2018**, pp. 651-654.
- X. Ni, H. Huang, and W. Du (2017), Relevance analysis and short-term prediction of PM<sub>2.5</sub> concentrations in Beijing based on multi-source data, *Atmospheric Environment*, **Vol. 150**, pp. 146–161.
- J. Chen, H. Chen, Z. Wu, D. Hu, and J. Z. Pan (2017), Forecasting smog-related health hazard based on social media and physical sensor, *Information Systems*, **Vol. 64**, pp. 281–291.
- W. Sun and J. Sun (2017), Daily PM<sub>2.5</sub> concentration prediction based on principal component analysis and LSSVM optimized by cuckoo search algorithm, *Journal of Environmental Management*, **Vol. 188**, pp. 144–152.
- W. Li, D. Kong, and J. Wu (2017), A new hybrid model FPA-SVM considering cointegration for particular matter concentration forecasting: a case study of Kunming and Yuxi, China, *Computational Intelligence and Neuroscience*, **vol. 2017**, pp. 1-11.
- Y. Liu, N. Luo, S. Wu (2019), Nonlinear Effects of Environmental Regulation on Environmental Pollution, *Discrete Dynamics in Nature and Society*, **vol. 2019**, pp. 1-10.
- X. Li (2021), Dynamic Combination Evaluation Method of Rural Environmental Pollution Control Effect, *Complexity*, **vol. 2021**, pp. 1-10.
- X. Li, S. Li, J. Gao, P. Zhao, H. Li (2018), Human Settlement Quality Evaluation Based on Air Quality in Major Cities of China, *Advances in Meteorology*, **vol. 2018**, pp. 1-9.
- P. Oyjinda, N Pochai (2017), Numerical Simulation to Air Pollution Emission Control near an Industrial Zone, *Advances in Mathematical Physics*, **vol. 2017**, pp. 1-7.
- G. Lancia, F. Rinaldi, P. Serafini (2018), A Facility Location Model for Air Pollution Detection, *Mathematical Problems in Engineering*, **vol. 2018**, pp. 1-8.
- S. S. Tasci, C. Kavalci, A. E. Kayipmaz (2018), Relationship of Meteorological and Air Pollution Parameters with Pneumonia in Elderly Patients, *Emergency Medicine International*, **vol. 2018**, pp. 1-8.
- S. Michaelides, D. Paronis, A. Retalis, F. Tymvios (2017), Monitoring and Forecasting Air Pollution Levels by Exploiting Satellite, Ground-Based, and Synoptic Data, Elaborated with Regression Models, *Advances in Meteorology*, **vol. 2017**, pp. 1-17.

*Halaman ini sengaja dikosongkan*