



# PENERAPAN LSTM DENGAN REGULARISASI UNTUK MENCEGAH OVERFITTING PADA MODEL PREDIKSI TINGKAT INFLASI DI INDONESIA

**Eko Verianto**

eko@unicimi.ac.id

Universitas Cendekia Mitra Indonesia

## Abstrak

Permasalahan inflasi dapat mempengaruhi pengembangan strategi, keputusan dan kebijakan pemerintah, sehingga diperlukan pemahaman mendalam mengenai tren inflasi di masa yang akan datang. Dalam menghadapi situasi ini diperlukan model prediksi yang dapat memodelkan tren inflasi di masa yang akan datang dengan tepat. Salah satu pendekatan yang dapat digunakan adalah *backpropagation*, namun penerapan *backpropagation* pada permasalahan prediksi seperti pada penelitian sebelumnya mendapatkan tantangan tersendiri, terutama pada data runtun waktu yang biasanya menghadirkan ketergantungan temporal. Penggunaan *backpropagation* dalam penelitian sebelumnya juga menunjukkan perilaku *overfitting*. Tujuan dari penelitian ini adalah mengatasi ketergantungan temporal pada data runtun waktu menggunakan *Long-Short Term Memory* (LSTM) dan penerapan *dropout* dalam arsitektur LSTM untuk mencegah terjadinya *overfitting* pada model prediksi tingkat inflasi di Indonesia. Hasil dari penelitian ini menunjukkan bahwa penerapan LSTM untuk mengatasi data dengan ketergantungan temporal menghasilkan kinerja yang cukup baik dan juga penggunaan *dropout* pada LSTM dapat mengatasi permasalahan *overfitting* pada prediksi tingkat inflasi di Indonesia.

**Kata kunci:** Prediksi, Inflasi, LSTM, Regularisasi, Dropout

## Abstract

*Inflation issues can influence the development of government strategies, decisions and policies, so a deep understanding of future inflation trends is needed. One approach that can be used is backpropagation, but applying backpropagation to prediction problems like in previous research has its challenges, especially in time-long data that typically presents temporal dependency. This research aims to solve temporal dependency in time series data using Long-Short Term Memory (LSTM) and the application of dropouts in the LSTM architecture to prevent overfitting on the model of inflation rate prediction in Indonesia. The results of this research show that applying LSTM to deal with temporal dependency data produces good performance and also use of dropouts on LSTM can solve the problem of overfitting on the prediction of the inflation rate in Indonesia.*

**Keywords:** Prediction, Inflation, LSTM, Regularization, Dropout

## 1. Pendahuluan

Teknologi sendiri dapat diartikan sebagai sarana atau sistem yang berfungsi memberikan kenyamanan dan kemudahan bagi manusia [1]. Teknologi tidak hanya cara bekerja dan berkomunikasi, tetapi juga telah memberikan pengaruh tentang bagaimana memahami dan merespon fenomena ekonomi seperti inflasi. Inflasi adalah kenaikan harga barang dan jasa secara umum dan terus menerus dalam jangka waktu tertentu [2]. Inflasi merupakan masalah yang terus menerus terjadi di setiap Negara yang mempengaruhi kemampuan pembuat kebijakan untuk mempertahankan biaya hidup yang rendah, pertumbuhan ekonomi yang pesat dan perekonomian yang pesat [3]. Agar dapat membuat kebijakan yang tepat sehingga dibutuhkan pemahaman yang mendalam mengenai tren inflasi. Salah satu teknik yang dapat digunakan untuk memahami tren inflasi adalah dengan menganalisa data historis. Analisis data historis memainkan peran penting dalam pemahaman tren inflasi.

Indonesia merupakan Negara berkembang dengan tantangan tersendiri dalam menghadapi masalah inflasi. Meskipun menunjukkan pertumbuhan ekonomi yang signifikan, namun bukan tidak mungkin

peningkatan inflasi terjadi di masa yang akan datang. Ini menyebabkan perlunya pemahaman yang mendalam mengenai tren inflasi saat ini untuk memprediksi tren inflasi di masa yang akan datang. Permasalahan inflasi juga akan mempengaruhi pengembangan strategi dan keputusan yang dikeluarkan oleh pemerintah dan pembuat kebijakan. Strategi dan keputusan yang dikeluarkan oleh pemerintah dan pembuat kebijakan ini ditujukan untuk menjaga pertumbuhan ekonomi yang berkelanjutan. Dalam menghadapi situasi seperti ini diperlukan model prediksi yang dapat memodelkan tren inflasi di masa yang akan datang dengan tepat.

Terdapat banyak algoritma yang dapat memodelkan tren inflasi dimasa yang akan datang, salah satunya adalah algoritma *backpropagation* pada arsitektur *multilayer perceptron*. *Backpropagation* merupakan metode dalam jaringan syaraf tiruan yang menggunakan pendekatan pembelajaran terawasi [4]. Adapun *multilayer perceptron* sendiri merupakan bentuk dasar dari jaringan syaraf tiruan *feed-forward*, jaringan syaraf tiruan memiliki satu lapisan masukan, satu atau lebih lapisan tersembunyi dan satu lapisan keluaran [5]. *Multilayer perceptron* dengan algoritma *backpropagation* dapat digunakan untuk memodelkan proses *non-linear* tanpa mengetahui penyebab dari data tersebut [6], seperti pada saat meramalkan tingkat inflasi. Salah satu penelitian yang berkaitan dengan prediksi tingkat inflasi seperti yang dilakukan oleh Wiranto, et al. pada tahun 2023 yang mengajukan algoritma *backpropagation neural network* untuk melakukan peramalan tingkat inflasi dengan hasil yang menunjukkan bahwa nilai MSE tahap pelatihan sebesar 0,00067535 serta MSE tahap pengujian sebesar 0,0767 pada jaringan dengan arsitektur 12-21-1 dengan pembagian data sebesar 50% untuk pelatihan dan 50% untuk pengujian [7].

Penelitian serupa lainnya juga pernah dilakukan oleh Anggoro pada tahun 2023 yang mengajukan arsitektur jaringan syaraf tiruan dengan algoritma pelatihan *backpropagation* untuk memprediksi perubahan saham LQ 45 yang menunjukkan empat model pelatihan dengan nilai MSE terkecil dari ke empat model tersebut sebesar 0,0000000974267 dan nilai MSE sebesar 0,005513 pada hasil pengujian [8]. Penelitian lainnya dilakukan oleh Saputra, et al. pada tahun 2023 yang mengajukan arsitektur *multilayer perceptron* dan algoritma *backpropagation* untuk memprediksi konsumsi energi listrik PT PLN UP3 Salatiga untuk 5 tahun ke depan, prediksi ini termasuk prediksi jangka panjang dengan hasil MAPE pada data latih sebesar 2,79%, sedangkan MAPE pada data uji sebesar 3,24% [9].

Meskipun penerapan algoritma *backpropagation* efektif dalam melakukan peramalan di bidang ekonomi [10], namun penerapan *backpropagation* pada permasalahan prediksi seperti pada penelitian sebelumnya mendapatkan tantangan tersendiri, terutama pada data runtun waktu. Hal ini dikarenakan data runtun waktu biasanya menghadirkan ketergantungan temporal, artinya suatu nilai bergantung pada nilai masa lalu [11]. Penggunaan *backpropagation* dengan arsitektur jaringan syaraf tiruan pada penelitian sebelumnya juga menunjukkan penggunaan *backpropagation* untuk pelatihan masih rentan terhadap *overfitting*, dimana kinerja pada pelatihan menunjukkan hasil yang lebih baik dibandingkan kinerja pada saat pengujian. Adapun rumusan masalah dalam penelitian ini diantaranya adalah bagaimana memprediksi data runtun waktu dengan ketergantungan temporal dan bagaimana mengatasi *overfitting* pada model prediksi tingkat inflasi di Indonesia.

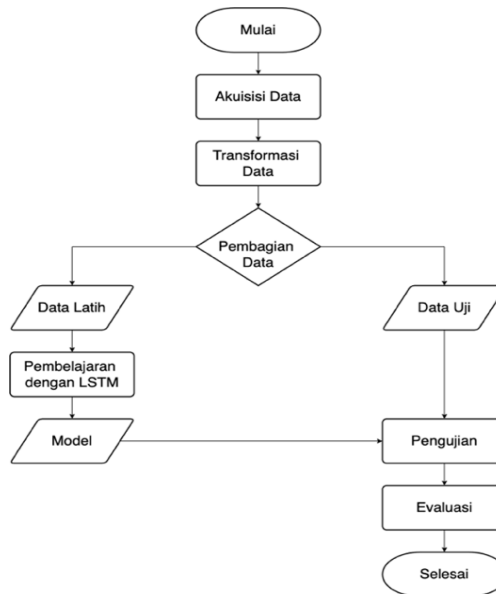
Berdasarkan hal tersebut, penelitian ini mengajukan *Long-Short Term Memory* (LSTM) sebagai solusi untuk mengatasi permasalahan ketergantungan temporal pada data runtun waktu. LSTM merupakan arsitektur jaringan dengan keunggulan mempelajari ketergantungan jangka panjang pada data, dimana LSTM dapat mengingat informasi jangka panjang [12], walaupun demikian LSTM masih memiliki kendala seperti sering kali mengalami *overfitting* karena strukturnya yang kompleks dan ekspresif [13]. Adapun untuk mengatasi permasalahan *overfitting*, penelitian ini menggunakan teknik regularisasi *dropout*. *Dropout* merupakan salah satu metode regularisasi paling dikenal untuk mencegah model mengalami *overfitting* dalam melakukan pelatihan [14]. Teknik ini akan digabungkan dengan LSTM yang dinilai dapat mengatasi ketergantungan pada data dengan sekuen yang panjang. Penelitian ini juga menerapkan *vector normalization* sebagai teknik untuk transformasi data. *Vector normalization* digunakan karena gabungan dari teknik ini dan *vector normalization* dapat memberikan hasil yang baik pada data runtun waktu [15].

## 2. Metode

Bagian ini membahas mengenai alur penelitian dan metode yang digunakan. Terdapat beberapa langkah yang dilakukan dalam penelitian ini untuk menghasilkan model prediksi tingkat inflasi di Indonesia.

### 2.1. Tahapan Penelitian

Tahapan dalam penelitian ini diawali dengan akuisisi data yang dilanjutkan dengan membagi data menjadi data pelatihan dan data pengujian. Data pelatihan dan data pengujian akan di transformasi ke skala yang lebih kecil. Selanjutnya data pelatihan akan digunakan untuk proses pembelajaran model dengan melakukan beberapa eksperimen. Model dengan hasil terbaik akan digunakan untuk memprediksi data pengujian. Adapun tahapan penelitian ini seperti yang tampak pada Gambar 1 berikut ini.



Gambar 1. Tahapan Penelitian

### 2.2. Akuisisi Data

Akuisisi data merupakan langkah awal yang sangat penting dalam proses penelitian untuk memastikan bahwa data yang diperoleh relevan. Data yang digunakan dalam penelitian ini merupakan data historis tingkat inflasi bulanan yang disediakan oleh Bank Indonesia dari periode Januari 2003 sampai dengan Desember 2023. Adapun bentuk dari data yang di akuisisi seperti yang tampak pada Gambar 2 berikut ini.

No	Periode	Inflasi
1	Januari 2003	8.68
2	Februari 2003	7.6
3	Maret 2003	7.17
4	April 2003	7.62
5	Mei 2003	7.15
248	Agustus 2023	3.27
249	September 2023	2.28
250	Oktober 2023	2.56
251	November 2023	2.86
252	Desember 2023	2.61

Gambar 2. Potongan Data Tingkat Inflasi di Indonesia

### 2.3. Transformasi Data

Transformasi data merupakan salah satu tahapan *pre-processing* dalam *data mining* yang digunakan untuk mengubah data menjadi bentuk yang sesuai dengan kebutuhan [16]. Transformasi data dilakukan sebelum data di latih menggunakan algoritma pembelajaran mesin, sehingga data yang akan di latih nantinya adalah data yang sudah di transformasi. Data yang digunakan dalam penelitian ini memiliki rentang yang berbeda-beda, sehingga data tersebut perlu di ubah ke skala baru menggunakan *vector normalization* [15]. Persamaan 1 berikut ini merupakan persamaan *vector normalization* yang digunakan dalam penelitian ini.

$$r_{ij} = \frac{x_{ij}}{\sum_{i=1}^m x_{ij}^2} \tag{1}$$

Dalam persamaan tersebut, terdapat  $r_{ij}$  yang merupakan data sampel setelah dilakukan transformasi, selanjutnya  $x_{ij}$  merupakan data sampel dan  $\sum_{i=1}^m x_{ij}^2$  merupakan hasil penjumlahan terhadap keseluruhan data sampel yang dikuadratkan. Dengan menggunakan persamaan ini, data inflasi yang telah diakuisisi sebelumnya ditransformasi ke bentuk skala yang lebih kecil seperti yang tampak pada Gambar 3 berikut ini.

Inflasi
0.08404073119015791
0.07358405035082952
0.06942074223887469
0.07377769258859486
0.06922710000110935
...
0.031660505874633225
0.022075215105248853
0.024786206433963627
0.027690840000443737
0.025270312028376977

Gambar 3. Transformasi Data Inflasi

### 2.4. Pembagian Data

Pembagian data merupakan tahapan penting dalam pembelajaran mesin. Pembagian data yang tepat juga dapat mempengaruhi kualitas model, terdapat dua teknik pembagian data yang dapat digunakan, yaitu *Holdout* dan *K-Fold Cross Validation*. *Holdout* pada prinsipnya akan memberikan setiap data kesempatan untuk menjadi data pelatihan dan data pengujian, sedangkan *K-Fold Cross Validation* akan membagi data sampel menjadi K subdivisi yang berbeda, dimana setiap subdivisi berfungsi sebagai subset pelatihan atau subset pengujian [17]. Penelitian ini menggunakan *holdout* untuk membagi data menjadi dua bagian, yaitu data pelatihan dan data pengujian. Adapun persentase pembagian data terdiri atas 60% data latih dan 40% data uji. Data yang digunakan pada tahapan ini adalah data dengan kondisi sudah ditransformasi yang selanjutnya akan di bagi menjadi dua bagian seperti pada Gambar 4 berikut ini.

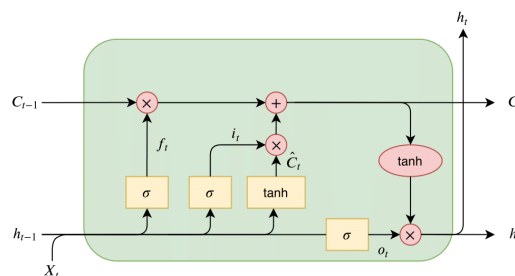
Training Data	Testing Data
0.08404073119015791	0.07029213230881871
0.07358405035082952	0.06951756335775736
0.06942074223887469	0.06612882419686389
0.07377769258859486	0.060513199301669006
0.06922710000110935	0.04734552713362583
...	...
0.06090048377719969	0.034081033846699985
0.06177187384714372	0.029820904615862488
0.06574153972133322	0.031660505874633225
0.06922710000110935	0.022075215105248853
0.07029213230881871	0.024786206433963627

Gambar 4. Pembagian Dataset Dengan *Holdout*

Setelah melakukan pembagian data menjadi data pelatihan dan data pengujian, tahapan selanjutnya adalah menentukan fitur dan target dari data, dimana dalam penelitian ini menggunakan teknik *sliding window*. Penerapan *sliding window* ditujukan untuk menjaga struktur temporal dari data, sehingga model nantinya dapat menangkap pola berurutan dari data. Setiap *window* mencakup sejumlah titik data yang berurutan yang kemudian digunakan untuk memprediksi data berikutnya. Dalam hal ini fitur akan diambil dari data sebelumnya dalam *window*, sedangkan target adalah titik akhir *window*. Dengan demikian, model dapat dilatih untuk mengenali pola temporal dari data.

### 2.5. Long Short-Term Memory (LSTM)

LSTM merupakan arsitektur *Recurrent Neural Network* (RNN) yang dimodifikasi untuk mengatasi masalah *vanishing gradient* dan *exploding gradient* serta memiliki kemampuan dalam melatih data dengan urutan panjang karena mampu mempertahankan informasi penting dalam jangka waktu yang lebih lama [18]. *Vanishing gradient* dan *exploding gradient* mengacu pada nilai dari suatu gradien yang sangat kecil (*vanishing*) atau sangat besar (*explode*). Cara kerja LSTM berfokus pada mekanisme pengelolaan memori yang terdiri dari *cell state* dan tiga gerbang utama, yaitu *input gate*, *forget gate* dan *output gate*. Adapun arsitektur LSTM seperti yang ditunjukkan pada Gambar 5 berikut ini.



Gambar 5. Arsitektur LSTM

Dalam arsitektur LSTM tersebut terdapat  $x_t$  yang merupakan *input* pada waktu  $t$ ,  $x_{t-1}$  merupakan *output* pada waktu  $t - 1$ ,  $c_{t-1}$  merupakan *cell state* pada waktu  $t - 1$ ,  $h_t$  adalah *hidden state* yang merupakan *output* dari LSTM pada waktu  $t$  dan  $c_t$  merupakan *cell state* pada waktu  $t$ ,  $h_{t-1}$  merupakan *output* dari LSTM pada waktu  $t - 1$ ,  $\hat{c}_t$  adalah *change gate* yang merupakan nilai baru pada waktu  $t$ ,  $f_t$  merupakan *forget gate*,  $i_t$  merupakan *input gate* serta terdapat  $O_t$  yang merupakan *output gate*. *Cell state*

menggunakan tiga gerbang untuk memperbarui informasi, diantaranya adalah gerbang masukan (*input gate*), gerbang lupa (*forget gate*) dan gerbang perubahan (*change gate*). Adapun dalam proses komputasi pada LSTM digunakan persamaan berikut:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (2)$$

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (3)$$

$$\hat{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \quad (4)$$

$$c_t = f_t * c_{t-1} + i_t * \hat{C}_t \quad (5)$$

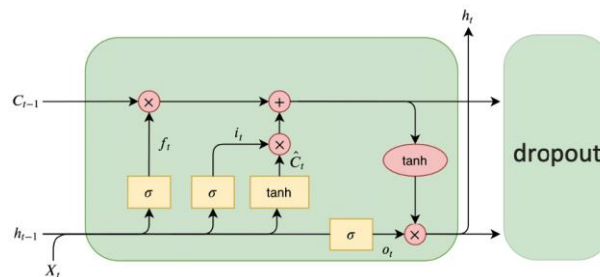
$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (6)$$

$$h_t = o_t * \tanh(c_t) \quad (7)$$

Penelitian ini menggunakan LSTM sebagai metode utama untuk memodelkan data tingkat inflasi di Indonesia. LSTM digunakan karena memiliki kinerja yang baik dalam menangani data runtun waktu dengan sekuen yang panjang [18]. Adapun data tingkat inflasi yang digunakan dalam penelitian ini berbentuk runtun waktu univariat, sehingga kelebihan yang ada pada LSTM diharapkan dapat memberikan kinerja yang baik dalam melatih data tingkat inflasi di Indonesia.

## 2.6. Dropout

*Dropout* merupakan salah satu teknik regularisasi yang paling efektif dan paling umum digunakan dalam jaringan syaraf tiruan [19]. Cara kerjanya dimana *dropout* akan menonaktifkan beberapa *neuron* dan *synapse weight* yang berkorespondensi dengan *neuron* tersebut [20]. Penelitian ini menerapkan *dropout* dalam LSTM sebagai solusi untuk mengatasi permasalahan *overfitting*. Gambar 6 berikut ini merupakan arsitektur LSTM dengan tambahan regularisasi *dropout*.



Gambar 6. Arsitektur LSTM dengan dropout

## 2.7. Evaluasi

Data hasil prediksi pada kasus peramalan biasanya tidak selalu sama dengan data aktual, sehingga terdapat selisih antara data hasil prediksi dengan data aktual. Selisih ini nantinya akan menjadi acuan dalam menentukan model mana yang memiliki kinerja yang lebih baik. Dalam kasus regresi seperti peramalan data akan menggunakan tingkat kesalahan terkecil sebagai acuan dalam menentukan model terbaik. Terdapat banyak metode yang dapat digunakan dalam menghitung selisih antara nilai hasil prediksi dengan nilai aktual, namun dalam penelitian ini akan menggunakan *Mean Squared Error* (MSE) sebagai metode untuk menghitung selisih antara nilai hasil prediksi dengan nilai aktual. Persamaan 2 berikut ini merupakan persamaan MSE yang digunakan dalam penelitian ini.

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \tilde{y}_i)^2 \quad (8)$$

Dimana  $n$  merupakan jumlah observasi,  $y_i$  merupakan nilai aktual dari observasi ke- $i$  dan  $\tilde{y}_i$  merupakan nilai prediksi dari observasi ke- $i$ . Hasil dari selisih antara nilai prediksi dan nilai aktual akan dikuadratkan.

## 3. Hasil dan Pembahasan

Sebagaimana yang dijelaskan sebelumnya bahwa arsitektur LSTM memiliki keunggulan data runtun waktu urutan panjang melalui mekanisme *gate* yang mengatur aliran informasi dan mencegah

masalah *vanishing gradient* dan *exploding gradient*. Penerapan arsitektur LSTM memerlukan penetapan nilai untuk setiap komponen LSTM. Penetapan nilai seperti ini dikenal dengan penetapan *hyperparameter*, tujuannya adalah untuk mendapatkan kinerja yang baik pada model. Adapun dalam penelitian ini menggunakan *hyperparameter* LSTM seperti yang tampak pada Tabel 3 berikut ini.

Tabel 3. *Hyperparameter* LSTM

Komponen	Hyperparameter	Value
	<i>units</i>	20
<i>LSTM Layer</i>	<i>input shape</i>	(15,1)
	<i>activation</i>	<i>sigmoid</i>
<i>Dropout Layer</i>	<i>rate</i>	[0.1,0.2,0.3]
	<i>unit</i>	1
<i>Dense Layer</i>	<i>activation</i>	<i>sigmoid</i>
	-	<i>optimizer</i>
-	<i>learning rate</i>	0.05
-	<i>loss function</i>	<i>mean_squared_error</i>
-	<i>epoch/iterasi</i>	200

Tabel 3 menunjukkan beberapa komponen pada arsitektur LSTM seperti *LSTM layer*, *dropout layer* dan *dense layer*. Adapun nilai *hyperparameter rate* untuk *dropout layer* menggunakan beberapa variasi untuk tujuan eksperimen. Penelitian ini juga menggunakan dua model lainnya sebagai pembanding model utama, diantaranya adalah model dengan pelatihan menggunakan *backpropagation* dan model LSTM tanpa regularisasi *dropout*.

Eksperimen pertama yang dilakukan dalam penelitian ini adalah eksperimen *overfitting*. Eksperimen ini dapat menunjukkan perilaku kinerja model terhadap data pembelajaran dan data pengujian. Adapun ukuran yang digunakan berupa nilai tingkat kesalahan rata-rata yang dilakukan terhadap data pelatihan dan data pengujian, baik pada model utama maupun model pembanding. Tabel 4 berikut ini menunjukkan hasil eksperimen *overfitting* pada model utama dan model pembanding, baik pada data pelatihan maupun data pengujian.

Tabel 4. Eksperimen *Overfitting*

Experiment	Model	Dropout Rate	MSE Training	MSE Testing	Random Seed [1-10]
1	Backpropagation	-	$1,8508 \times 10^{-4}$	$2,2142 \times 10^{-4}$	3
	LSTM	-	$1,9369 \times 10^{-4}$	$2,9238 \times 10^{-4}$	3
	LSTM+Dropout	0.1	$1,8858 \times 10^{-4}$	$1,2467 \times 10^{-4}$	3
2	Backpropagation	-	$1,6813 \times 10^{-4}$	$1,8475 \times 10^{-4}$	1
	LSTM	-	$1,8346 \times 10^{-4}$	$2,2355 \times 10^{-4}$	1
	LSTM+Dropout	0.2	$2,2327 \times 10^{-4}$	$8,8737 \times 10^{-5}$	1
3	Backpropagation	-	$1,6297 \times 10^{-4}$	$1,9285 \times 10^{-4}$	7
	LSTM	-	$2,2922 \times 10^{-4}$	$3,2666 \times 10^{-4}$	7
	LSTM+Dropout	0.3	$4,1282 \times 10^{-4}$	$2,8657 \times 10^{-4}$	7
4	Backpropagation	-	$1,5579 \times 10^{-4}$	$1,8404 \times 10^{-4}$	8
	LSTM	-	$2,0048 \times 10^{-4}$	$2,5207 \times 10^{-4}$	8
	LSTM+Dropout	0.1	$1,9826 \times 10^{-4}$	$7,2913 \times 10^{-5}$	8
5	Backpropagation	-	$1,3650 \times 10^{-4}$	$1,6209 \times 10^{-4}$	10
	LSTM	-	$1,7435 \times 10^{-4}$	$7,5863 \times 10^{-5}$	10
	LSTM+Dropout	0.2	$2,0416 \times 10^{-4}$	$3,2481 \times 10^{-5}$	10
6	Backpropagation	-	$1,8076 \times 10^{-4}$	$2,8007 \times 10^{-4}$	4
	LSTM	-	$1,7481 \times 10^{-4}$	$8,3360 \times 10^{-5}$	4
	LSTM+Dropout	0.3	$2,2534 \times 10^{-4}$	$4,1524 \times 10^{-5}$	4

Berdasarkan eksperimen *overfitting* yang telah dilakukan seperti yang ditunjukkan pada Tabel 4, dimana pada model *backpropagation* menunjukkan terjadinya *overfitting* pada semua eksperimen. Pada model LSTM tanpa regularisasi *dropout* menunjukkan terjadinya *overfitting* pada eksperimen pertama, kedua, ketiga dan keempat, sedangkan pada eksperimen kelima dan keenam menunjukkan tidak terjadinya *overfitting*. Adapun pada model LSTM dengan regularisasi *dropout* tidak menunjukkan terjadinya *overfitting* dalam semua eksperimen yang dilakukan.

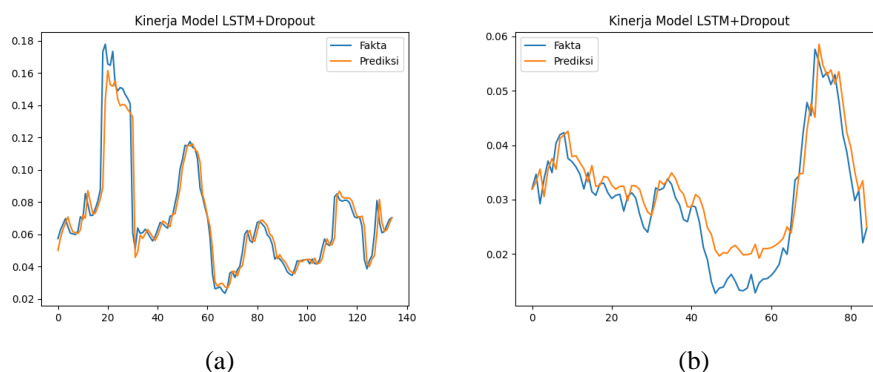
Hasil eksperimen ini menunjukkan bahwa penggunaan *dropout* pada arsitektur LSTM seperti yang terdapat pada Gambar 6 menunjukkan bahwa penerapan teknik regularisasi *dropout* pada LSTM dapat mengatasi permasalahan *overfitting* pada prediksi tingkat inflasi di Indonesia. Berdasarkan pada Gambar 6 menunjukkan proses komputasi diawali dengan data masukan yang diterima oleh lapisan LSTM, saat data di proses secara sekuensial, LSTM memproses informasi sambil mempertahankan memori dari langkah sebelumnya dengan menggunakan mekanisme *gate* yang mengatur aliran informasi. *Dropout* diterapkan pada setiap lapisan LSTM, dimana sejumlah unit di pilih secara acak untuk dinonaktifkan selama pelatihan dalam setiap iterasi. Ini membantu mengurangi kemungkinan terjadinya *overfitting* karena model tidak hanya mengandalkan pola pada unit-unit tertentu, melainkan harus belajar pada pola yang lebih umum yang berlaku untuk data yang belum pernah di lihat sebelumnya.

Selain melakukan eksperimen *overfitting*, penelitian ini juga melakukan eksperimen iterasi untuk menguji tingkat kinerja model berdasarkan jumlah iterasi. Eksperimen ini menggunakan salah satu model dalam eksperimen *overfitting* yang telah dilakukan sebelumnya. Hal ini tentu saja untuk memberikan pengalaman yang lebih banyak lagi bagi model yang di pilih untuk mempelajari pola dari suatu data, namun juga dengan tetap mempertimbangkan *overfitting*. Dalam eksperimen iterasi akan menggunakan variasi nilai iterasi 200, 400, 600, 800 dan 1000. Adapun hasil eksperimen ditunjukkan seperti pada Tabel 5 berikut ini.

Tabel 5. Eksperimen Iterasi

<i>Epoch/Iterasi</i>	<i>MSE Training</i>	<i>MSE Testing</i>
200	$2,0416 \times 10^{-4}$	$3,2481 \times 10^{-5}$
400	$1,6732 \times 10^{-4}$	$2,8289 \times 10^{-5}$
600	$1,6314 \times 10^{-4}$	$2,2121 \times 10^{-5}$
800	$1,4989 \times 10^{-4}$	$2,0257 \times 10^{-5}$
1000	$1,4266 \times 10^{-4}$	$2,0070 \times 10^{-5}$

Hasil eksperimen iterasi pada Tabel 5 menunjukkan bahwa tingkat kesalahan dari model LSTM dengan regularisasi *dropout* sangat bergantung pada jumlah iterasi yang digunakan dalam proses pembelajaran. Eksperimen ini juga menunjukkan bahwa penggunaan *dropout* pada arsitektur LSTM dapat menjaga perilaku *overfitting* pada model. Berdasarkan hasil eksperimen ini maka ditentukan bahwa model utama dalam penelitian ini akan dilatih menggunakan 1000 iterasi. Model ini akan digunakan untuk melakukan prediksi pada data latih dan data uji seperti yang ditunjukkan pada Gambar 6 berikut ini.



Gambar 7. Perbandingan Hasil Prediksi dan Fakta Pada Data Latih (a) dan Data Uji (b)



Gambar 7 menunjukkan kinerja dari model LSTM dengan regularisasi *dropout* yang berhasil mengikuti fluktuasi dan perubahannya pada data latih, sedangkan pada data uji terlihat tidak begitu tepat pada beberapa segmen. Dengan demikian, arsitektur LSTM dapat digunakan untuk memprediksi data dengan ketergantungan temporal pada data tingkat inflasi di Indonesia. Hasil ini juga menunjukkan bahwa tidak terjadinya *overfitting* pada model, baik pada data latih maupun data uji. Hal ini dikarenakan penerapan *dropout* yang berhasil melakukan generalisasi dengan baik, sehingga mampu menangkap pola yang ada pada data latih tidak secara berlebihan, dengan begini model mampu melakukan generalisasi dari pola yang dipelajari pada data latih untuk memahami data baru. Penerapan *dropout* ini sangat penting agar model tidak bergantung pada fitur yang terlalu spesifik dari data latih.

#### 4. Kesimpulan dan Saran

Berdasarkan analisa dari hasil eksperimen yang telah dilakukan dalam penelitian ini dan dengan mengacu pada rumusan masalah yang telah di uraikan sebelumnya, sehingga dapat diambil kesimpulan sebagai berikut:

1. Data runtun waktu dengan ketergantungan temporal dapat di prediksi menggunakan *Long-Short Term Memory* (LSTM). Penggunaan LSTM untuk memprediksi tingkat inflasi di Indonesia menghasilkan kinerja yang cukup baik, ini dikarenakan LSTM bekerja dengan mekanisme *gate* yang dapat mengatur aliran informasi secara sekuensial.
2. Penerapan regularisasi *dropout* pada arsitektur LSTM dapat mengatasi permasalahan *overfitting* dimana pada eksperimen *overfitting* menunjukkan model LSTM dengan regularisasi *dropout* tidak mengalami *overfitting* pada semua percobaan. Ini dapat terjadi karena model generalisasi dari pola yang dipelajari dan model tidak bergantung pada fitur yang terlalu spesifik.

Penelitian ini menggunakan data tingkat inflasi bulanan dengan periode yang terbatas, sehingga model kesulitan dalam menangkap variasi data. Pada penelitian selanjutnya diharapkan dapat menggunakan periode yang lebih lama, sehingga variasi data lebih beragam.

#### Daftar Pustaka

- [1] M. Naeruz, S. Afiffudin, D. Ruslan and M. Syafii, "The impact of economic growth on technological developments, emoneys and fluctuations interest rates and exchange rates in Indonesia," in *E3S Web of Conferences*, 2022.
- [2] F. A. Widya, F. L. Maula, M. D. D. L. A. Hakim, H. Salshabilla, D. N. Afifah and B. N. Asiyah, "Inflation and Its Contribution to Economic Stability in Indonesia in an Islamic Economic Perspective," in *Proceedings of Islamic Economics Business and Philanthropy*, 2023.
- [3] Y. Yahya and M. B. Pamuncak, "Long-Run Determinants of Inflation in Malaysia and Indonesia: Does Geopolitical risk matter?," *Journal of Sustainable Economics*, vol. 01, 2023.
- [4] V. Lestari, H. Mawengkang and Z. Situmorang, "Artificial Neural Network Backpropagation Method to Predict Tuberculosis Cases," *Sinkron : Jurnal dan Penelitian Teknik Informatika*, vol. 7, no. 1, 2023.
- [5] L. Karrach and E. Pivarciová, "Using Different Types of Artificial Neural Networks to Classify 2D Matrix Codes and Their Rotations - A Comparative Study," *Journal of Imaging*, 2023.
- [6] W. Cao, "Evaluating the Vocal Music Teaching Using Backpropagation Neural Network," *Mobile Information Systems*, 2022.
- [7] A. R. Wiranto, E. Setiawan, A. Nuryaman and M. Usman, "Implementasi Metode Backpropagation Neural Network dalam Meramalkan Tingkat Inflasi di Indonesia," *MATHunesa: Jurnal Ilmiah Matematika*, vol. 11, pp. 8-16, 2023.
- [8] A. Y. Anggoro, "Aplikasi Jaringan Syaraf Tiruan dengan Algoritma Backpropagasi untuk Memprediksi Perubahan Harga Saham LQ 45," in *SEMNAPTIKA 2023 : "Pembelajaran Matematika Kreatif & Inovatif dalam Implementasi Kurikulum Merdeka"*, Kupang, 2023.

- 
- [9] R. Saputra, S. Sunardiyo, A. Nugroho and Subiyanto, "Implementasi Multilayer Perceptron Artificial Neural Network untuk Prediksi Konsumsi Energi Listrik PT PLN (Persero) UP3 Salatiga," *Elektrika: Jurnal Ilmiah Teknik Elektro*, vol. 15, pp. 60-68, 2023.
- [10] A. R. Wiranto, E. Setiawan, A. Nuryaman and M. Usman, "Implementasi Metode Backpropagation Neural Network dalam Meramalkan Tingkat Inflasi di Indonesia," *Jurnal Ilmiah Matematika*, vol. 11, 2023.
- [11] J. Cuomo, H. Homayouni, I. Ray and S. Ghosh, "Detecting Temporal Dependencies in Data," in *BICOD21: British International Conference on Databases*, 2021.
- [12] M. N. A. P. Khano, D. R. S. Saputro, S. Sutanto and A. Wibowo, "Sentiment Analysis with Long-short Term Memory (LSTM) and Gated Recurrent Unit (GRU) Algorithms," *BAREKENG: Journal of Mathematics and Its Applications*, vol. 17, no. 4, pp. 2235-2242, 2023.
- [13] R. Kinoyama, E. A. M. Perez and H. Iba, "Preventing Overfitting of LSTMs using Ant Colony Optimization," in *International Congress on Advanced Applied Informatics (IIAI-AAI)*, Japan, 2021.
- [14] I. Salehin and D.-K. Kang, "A Review on Dropout Regularization Approaches for Deep Neural Networks within the Scholarly Domain," *Electronics*, 2023.
- [15] E. Verianto and B. S. D. Oetomo, "Artificial Neural Network Model with PSO as a Learning Method to Predict Movement of the Rupiah Exchange Rate against the US Dollar," *International Journal of Applied Information Technology*, vol. 04, 2021.
- [16] E. Haerani, F. Syafria, F. Lestari, N. Novriyanto and I. Marzuki, "Classification Academic Data using Machine Learning for Decision Making Process," *Journal of Applied Engineering and Technological Science*, vol. 4, 2023.
- [17] R. Oktafiani, A. Hermawan and D. Avianto, "Pengaruh Komposisi Split Data Terhadap Performa Klasifikasi Penyakit Kanker Payudara Menggunakan Algoritma Machine Learning," *Jurnal Sains dan Informatika*, vol. 9, 2023.
- [18] B. Raharjo, *Deep Learning dengan Python*, Semarang: Yayasan Prima Agus Teknik, 2022.
- [19] F. Chollet, *Deep Learning with Python (Second Edition)*, Shelter Island: Manning Publications, 2021.
- [20] J. W. G. Putra, "Pengenalan Konsep Pembelajaran Mesin dan Deep Learning (Edisi 1.4)," 17 8 2020. [Online]. Available: <https://wiragotama.github.io/resources/ebook/intro-to-ml-secured.pdf>. [Accessed 05 06 2024].