

**IF4074 Pembelajaran Mesin Lanjut**

**LAPORAN MILESTONE A**

**Tugas Long Short Term Memory Network**



Disusun oleh

Arung Agamani Budi Putera 13518005

Muh. Muslim Al Mujahid 13518054

Fritz Gerald Tjie 13518065

Faris Muhammad Kautsar 13518105

**Program Studi Teknik Informatika**

**Sekolah Teknik Elektro dan Informatika**

**Institut Teknologi Bandung**

2020/2021

## A. Kode Program

### 1. Kelas

Sequential
Kelas untuk membuat <i>sequential model</i>

Layer
Kelas dasar untuk kelas-kelas layer

Dense(Layer)
Kelas untuk membuat <i>dense layer</i>

Cell
Kelas untuk merepresentasikan cell pada sebuah layer LSTM

LSTM(Layer)
Kelas untuk membuat <i>layer</i> LSTM

TimeseriesDataset
Menerima sebuah dataset dan merubahnya ke dalam bentuk <i>time series</i>

### 2. Fungsi

linear(x:matrix)
Fungsi aktivasi linear, mengembalikan hasil fungsi aktivasi linear dari x

relu(x: matrix)
Fungsi aktivasi relu, mengembalikan hasil fungsi aktivasi relu dari x

sigmoid(x: matrix)
--------------------

Fungsi aktivasi sigmoid, mengembalikan hasil fungsi aktivasi sigmoid dari x

`softmax(x: matrix)`

Fungsi aktivasi softmax, mengembalikan hasil fungsi aktivasi softmax dari x

## B. Hasil Prediksi

### 1. Arsitektur Model

```
def main():
    # Load dataset
    dataset_train = pd.read_csv('./dataset/bitcoin_price_Training - Training.csv')
    # dataset = pd.read_csv('./dataset/bitcoin_price_1week_Test - Test.csv')

    # Preprocessing
    dataset_train = dataset_train.iloc[0:32, :]
    dataset_train = dataset_train.iloc[:, :-1].reset_index(drop=True)
    dataset_train = dataset_train.drop(columns=['Date'])
    dataset_train['Volume'] = dataset_train['Volume'].apply(
        lambda x: x.replace(',', ''))
    dataset_train['Market Cap'] = dataset_train['Market Cap'].apply(
        lambda x: x.replace(',', ''))
    dataset_train['Volume'] = pd.to_numeric(dataset_train['Volume'])
    dataset_train['Market Cap'] = pd.to_numeric(dataset_train['Market Cap'])
    # print(dataset_train)

    # Create model
    model = Sequential()
    model.add(layers.LSTM(64, input_shape=(1, 4, 6), name='lstm_1'))
    prediction = model.predict(dataset_train)
    print(prediction)
    model.summary()
```

### 2. Hasil Prediksi

```
[[-3.47425912 -3.00893095 -3.00904272 -3.00904276 -0.14820941 -3.47425912]]
```

Layer (type)	Output Shape	Params
lstm_1 (LSTM)	(1, 4, 64)	18176

```
Total params: 18176
Trainable params: 18176
Non-Trainable params: 0
```

## Pembagian Tugas

NAMA	NIM	TUGAS
Arung Agamani Budi Putera	13518005	<ul style="list-style-type: none"><li>• LSTM</li></ul>
Muh. Muslim Al Mujahid	13518054	<ul style="list-style-type: none"><li>• Cell</li><li>• TimeseriesDataset</li></ul>
Fritz Gerald Tjie	13518065	<ul style="list-style-type: none"><li>• Sequential</li></ul>
Faris Muhammad Kautsar	13518105	<ul style="list-style-type: none"><li>• Model summary</li><li>• Edit preprocessing dataset</li></ul>

## Link Video

<https://drive.google.com/file/d/1jqL8xIkRmkvFezue1jBa3BoblIEQ88N3/view?usp=sharing>