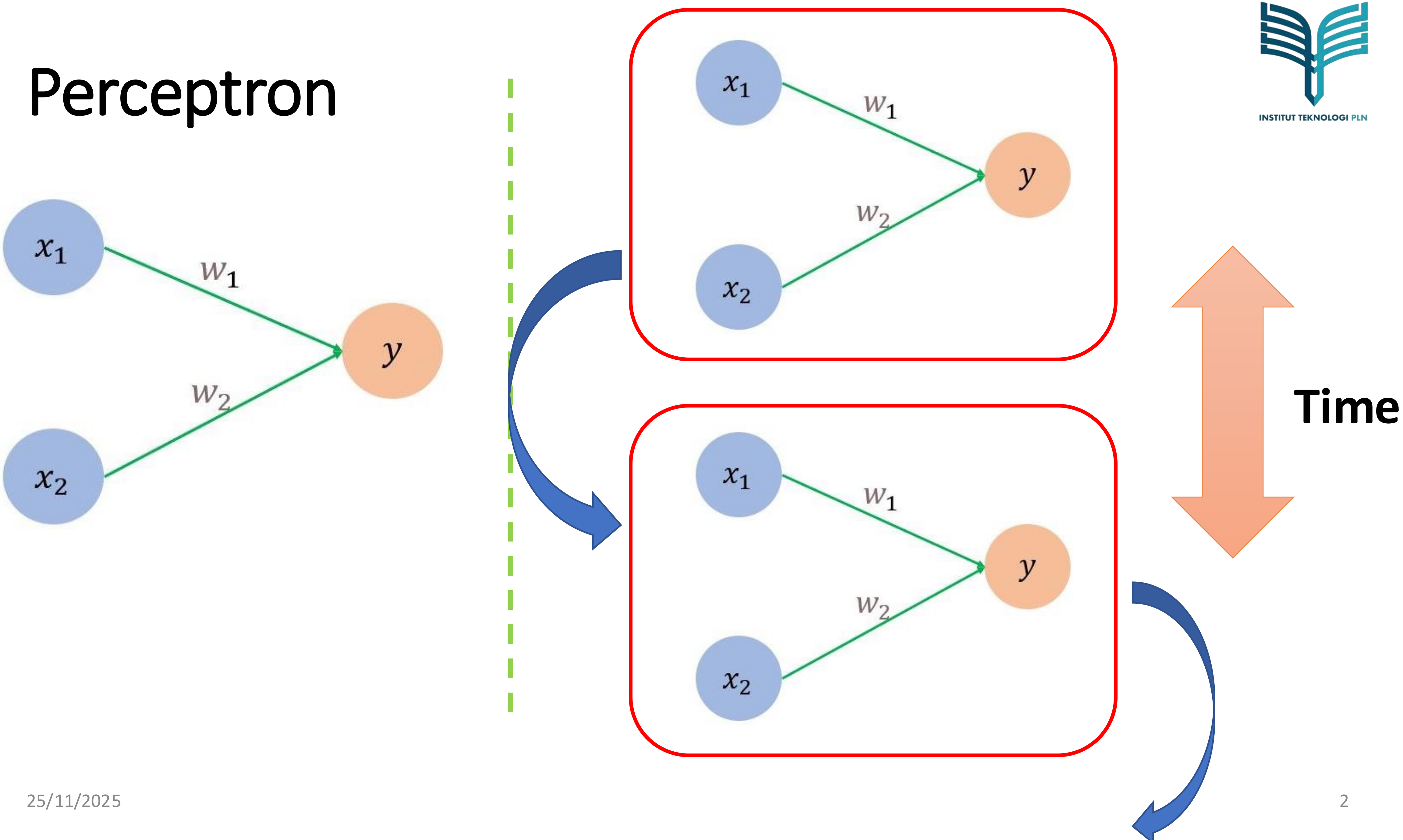
Deep Learning: Long Short-Term Memory (LSTM)

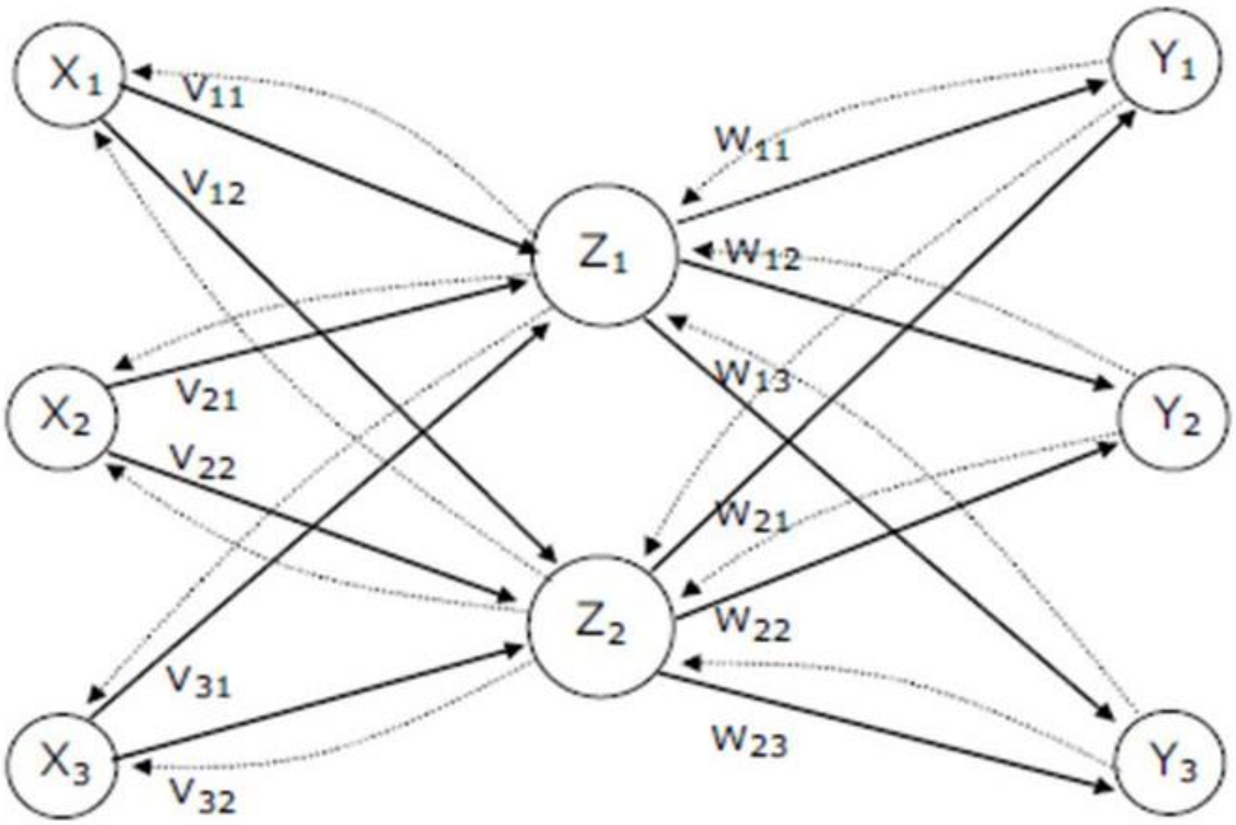
Dr. Ir. Abdul Haris, S.Kom., M.Kom

Institut Teknologi PLN
Jakarta

Perceptron



Backpropagation



Langkah Maju untuk mencari Nilai
Langkah mundur untuk melakukan evaluasi neuron (Error)

Arsitektur Deep Learning

Arsitektur Asal	Arsitektur Turunan => Metode Berbeda
CNN	RCNN (Regional CNN)
	Fast-RCNN, Mask RCNN
	YOLO, v1,v2,3,4,5 dst (mungkin akan terus berkembang)
RNN	LSTM (Long Short-Term Memory)
	GRU (Gated Recurrent Unit)

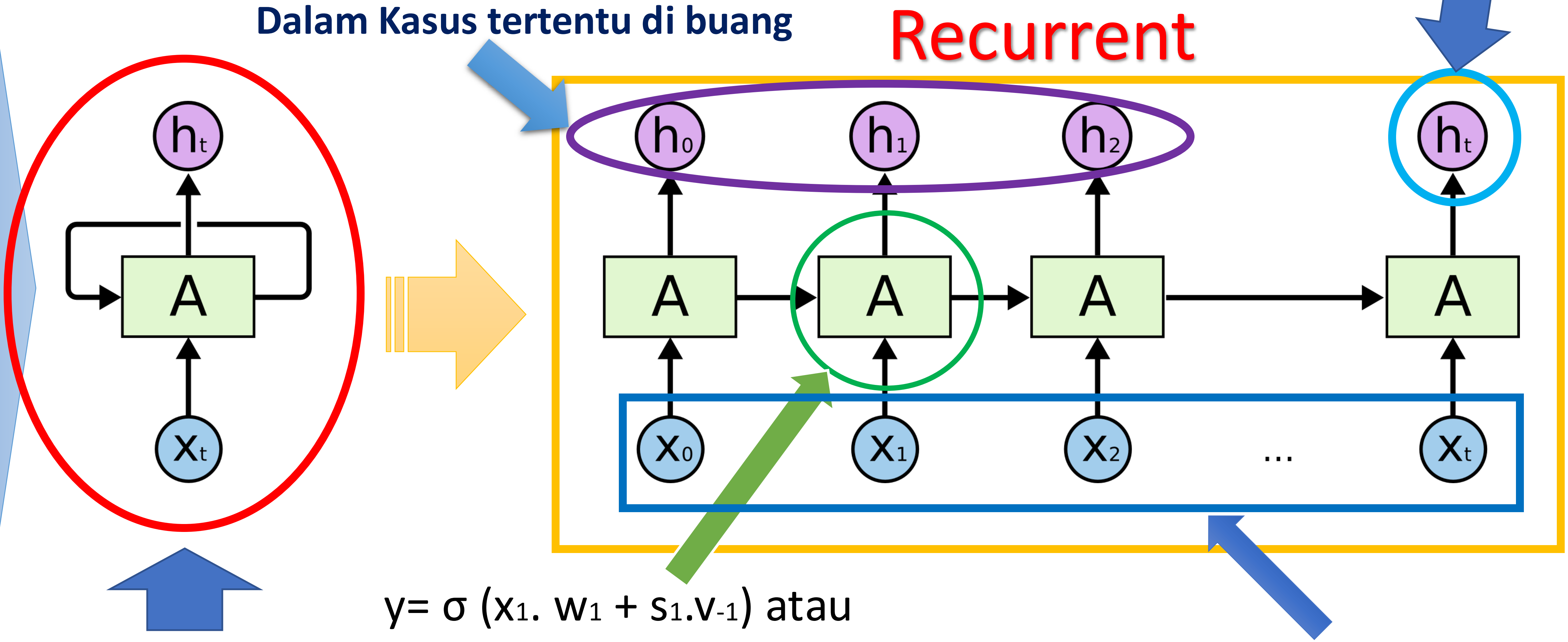
Arsitektur Turunan LSTM

Long Short-Term Memory (LSTM)

- Vanilla LSTM
- Stacked LSTM
- CNN LSTM
- Encoder-Decoder LSTM
- Bidirectional LSTM
- Generative LSTM

Dalam kasus tertentu Ambil

Arsitektur Recurrent Neural Network (RNN)



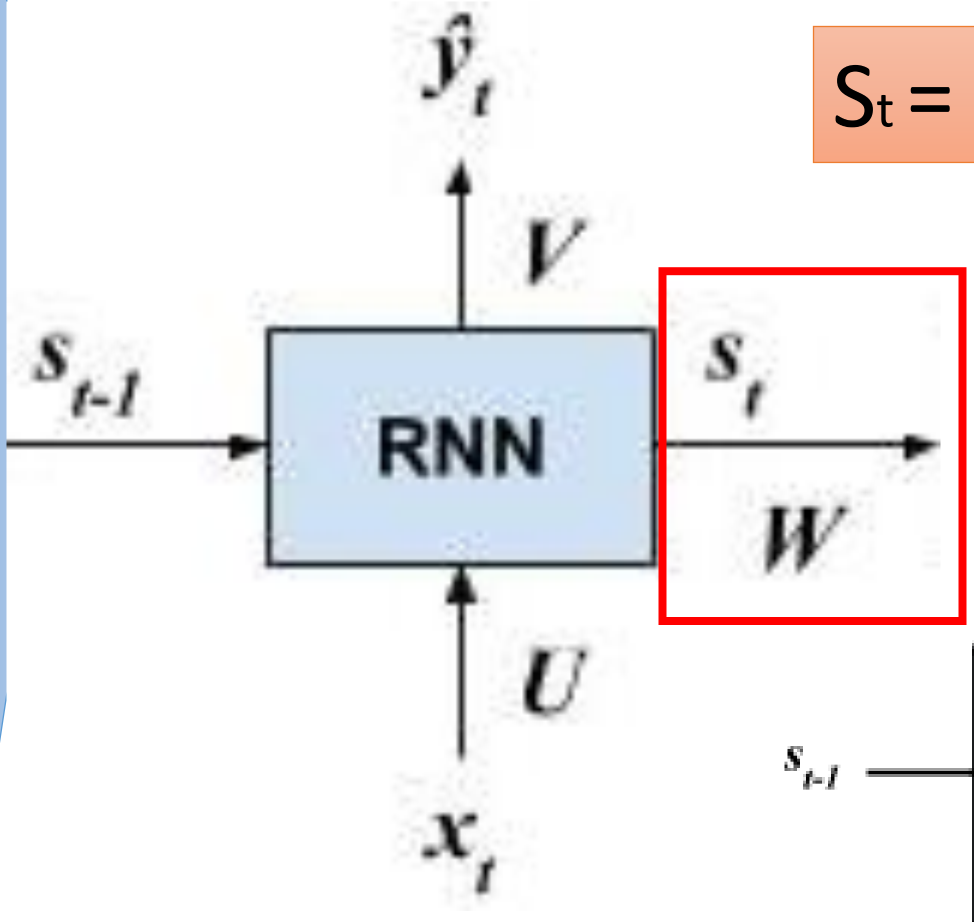
Xt = Tunggal

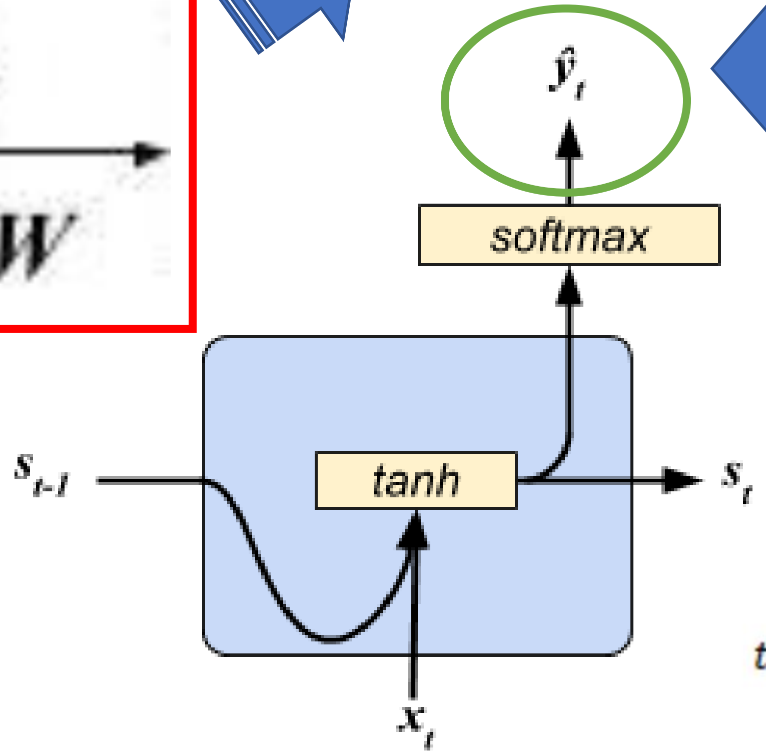
$y = \sigma\,(x_1 .\ w_1 + s_1 . v_{-1})$ atau

$y = \tanh\,(x_1 .\ w_1 + s_1 . v_{-1})$

Step RNN

$S_t = \tanh(\, U \cdot X_t + W \cdot S_{t-1}\,)$



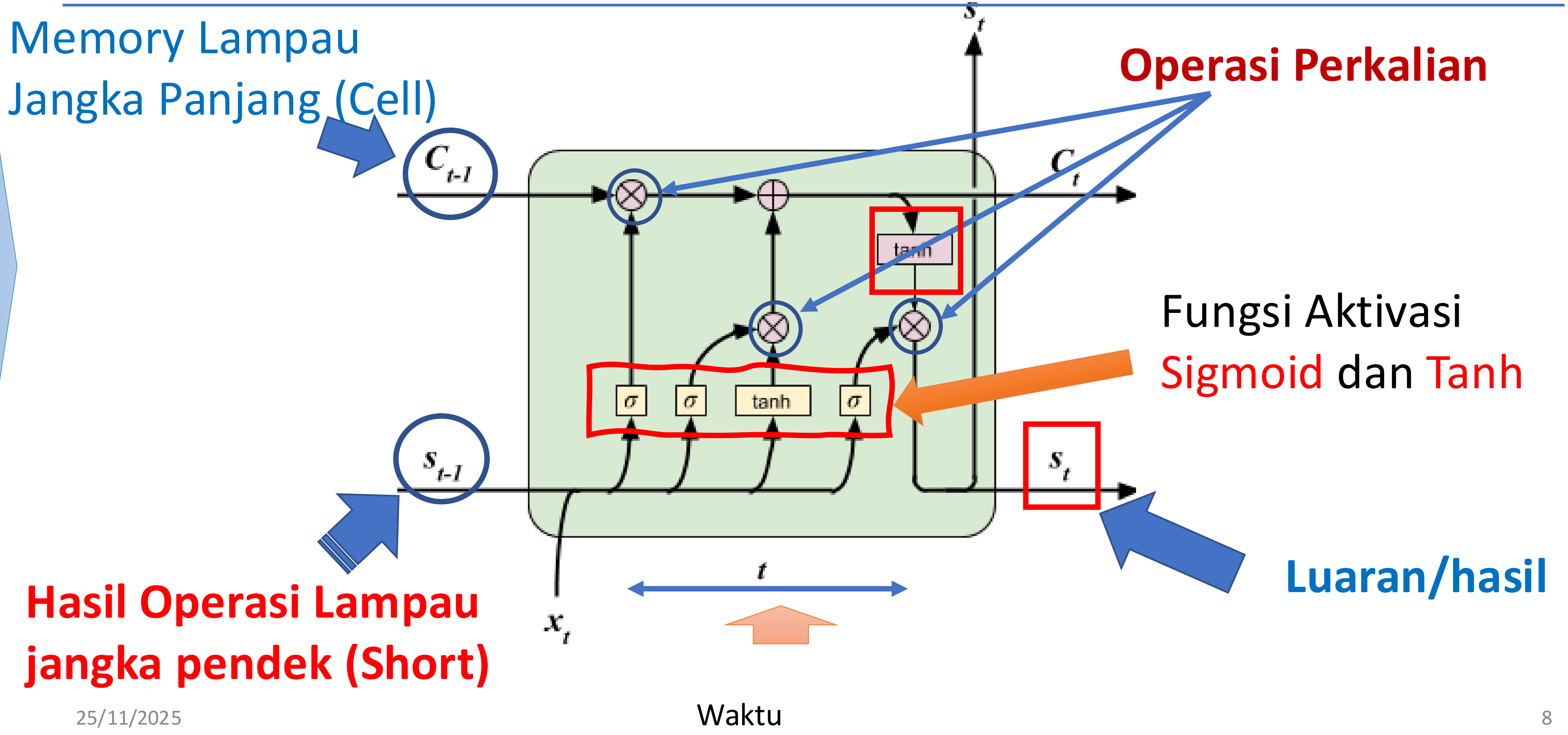


$\hat{y}_t = \text{softmax}(\, V \cdot S_t\,)$

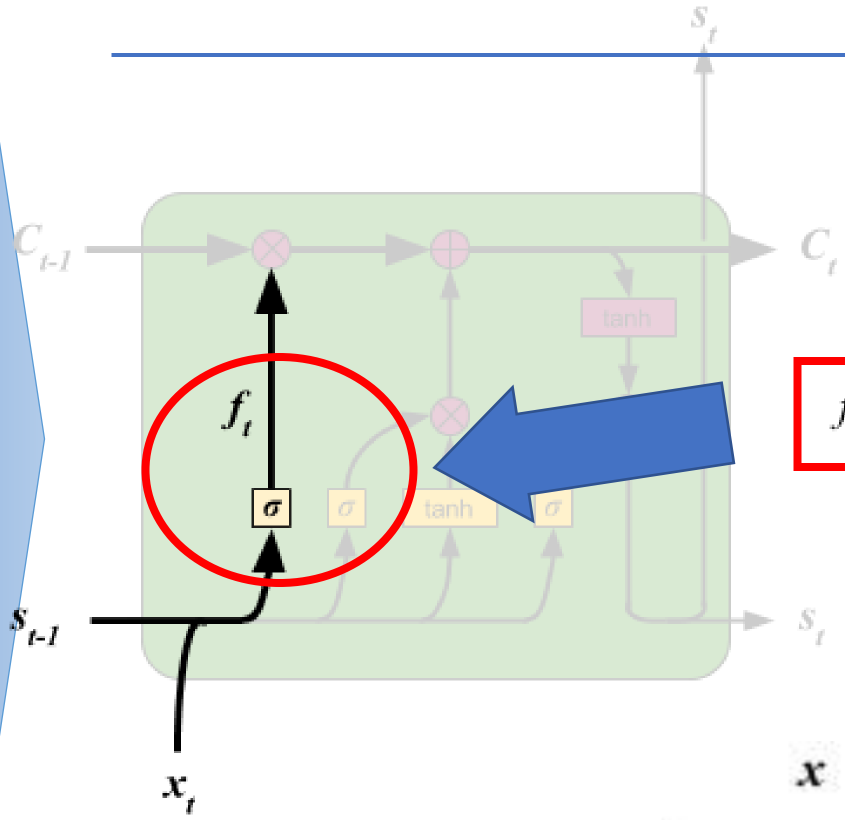
$$softmax(xi) = \frac{\exp(x_i)}{\sum_j \exp(x_j)}$$

$$tanh(x) = \frac{\sin(x)}{\cos(x)} = \left(\frac{e^x - e^{-x}}{e^x + e^{-x}}\right) = \left(\frac{e^{2x} - 1}{e^{2x} + 1}\right)$$

Arsitektur Long Short-Term Memory



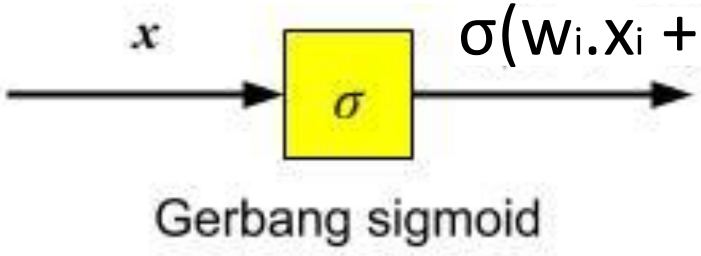
Step 1 Input Nilai X



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

Operasi ini adalah melakukan perkalian nilai Input X_t dan S_{t-1}

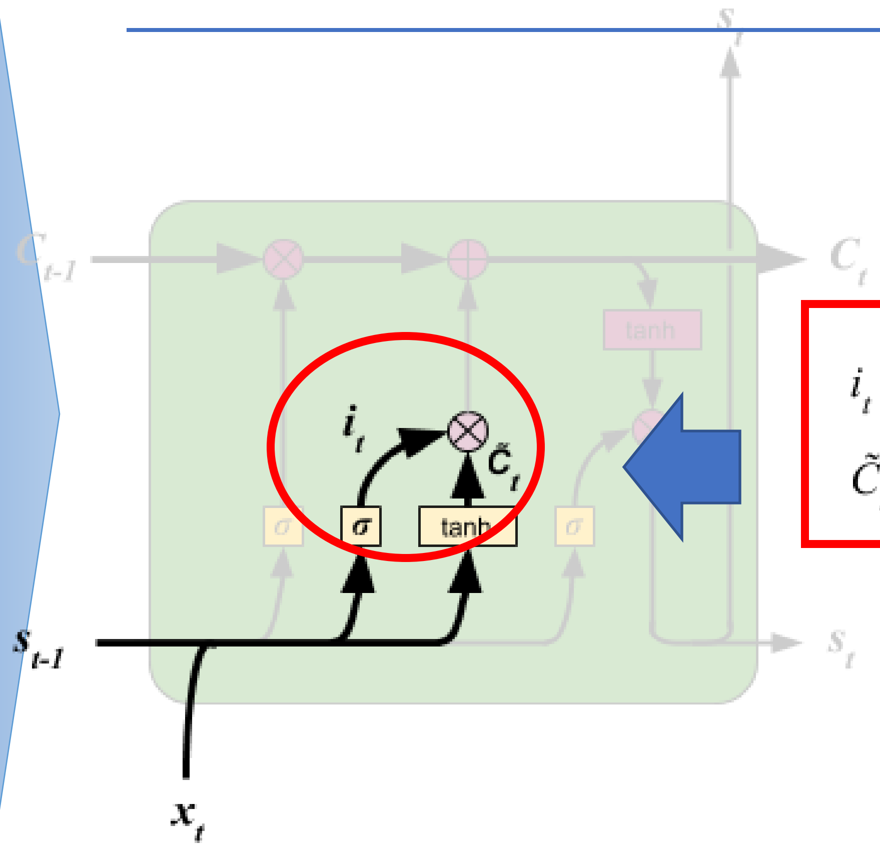
Jika Awal, maka S_{t-1} dinetralkan dengan nilai 0



$\sigma(w_i.x_i + b)$

Operasi pada gerbang

Step 2 Operasi Gerbang



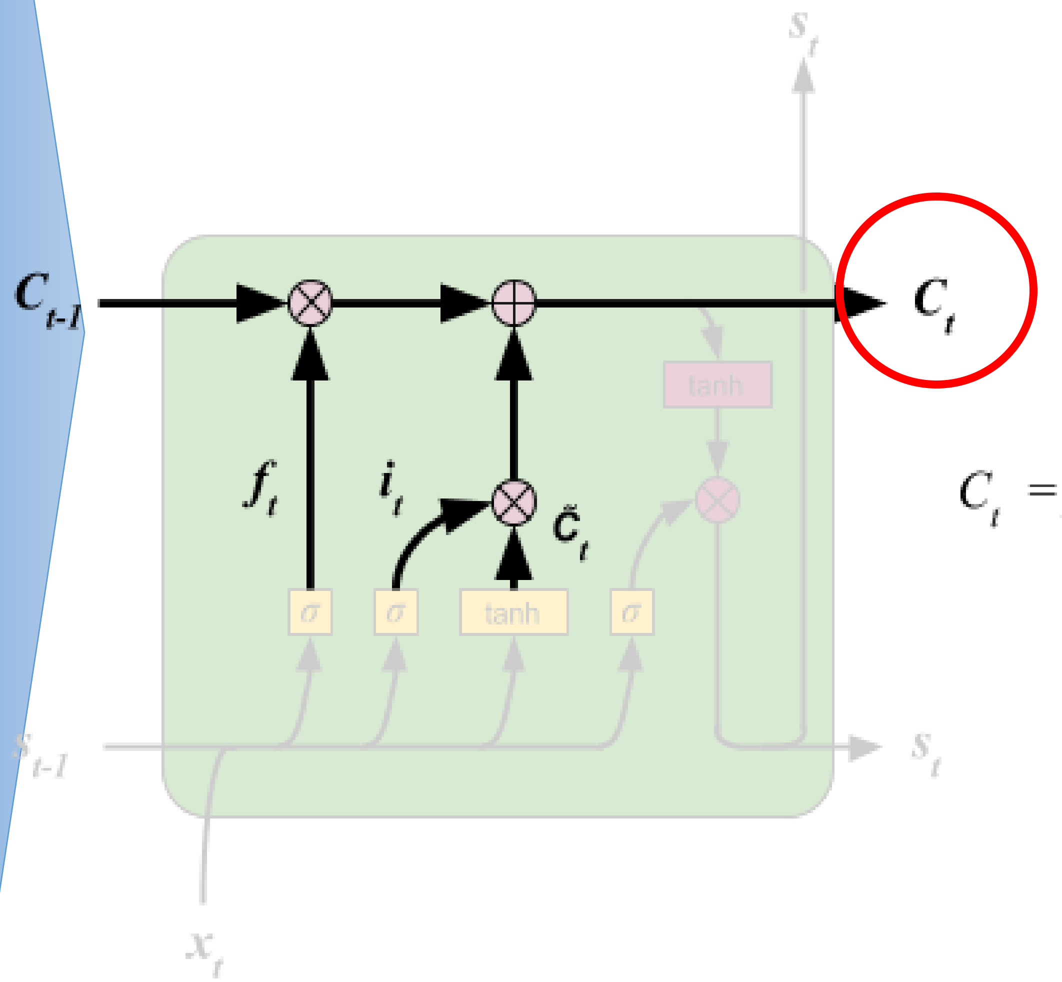
$$i_t = \sigma(W_i \cdot [s_{t-1}, x_t] + b_i)$$

$$\tilde{C}_t = tanh(W_C \cdot [s_{t-1}, x_t] + b_C)$$

Nilai i_t diperoleh dari hasil perkalian wi dengan Nilai matrik $[S_{-t}, X_t]$di tambahkan dengan b yg di sigmoidkan

Nilai $\hat{C}_t$ diperoleh dari hasil perkalian nilai Wc dikalikan dengan $[S_{t-1},X_t]$ ditambah dengan bias (b_c)

Step 3 Operasi Cell State

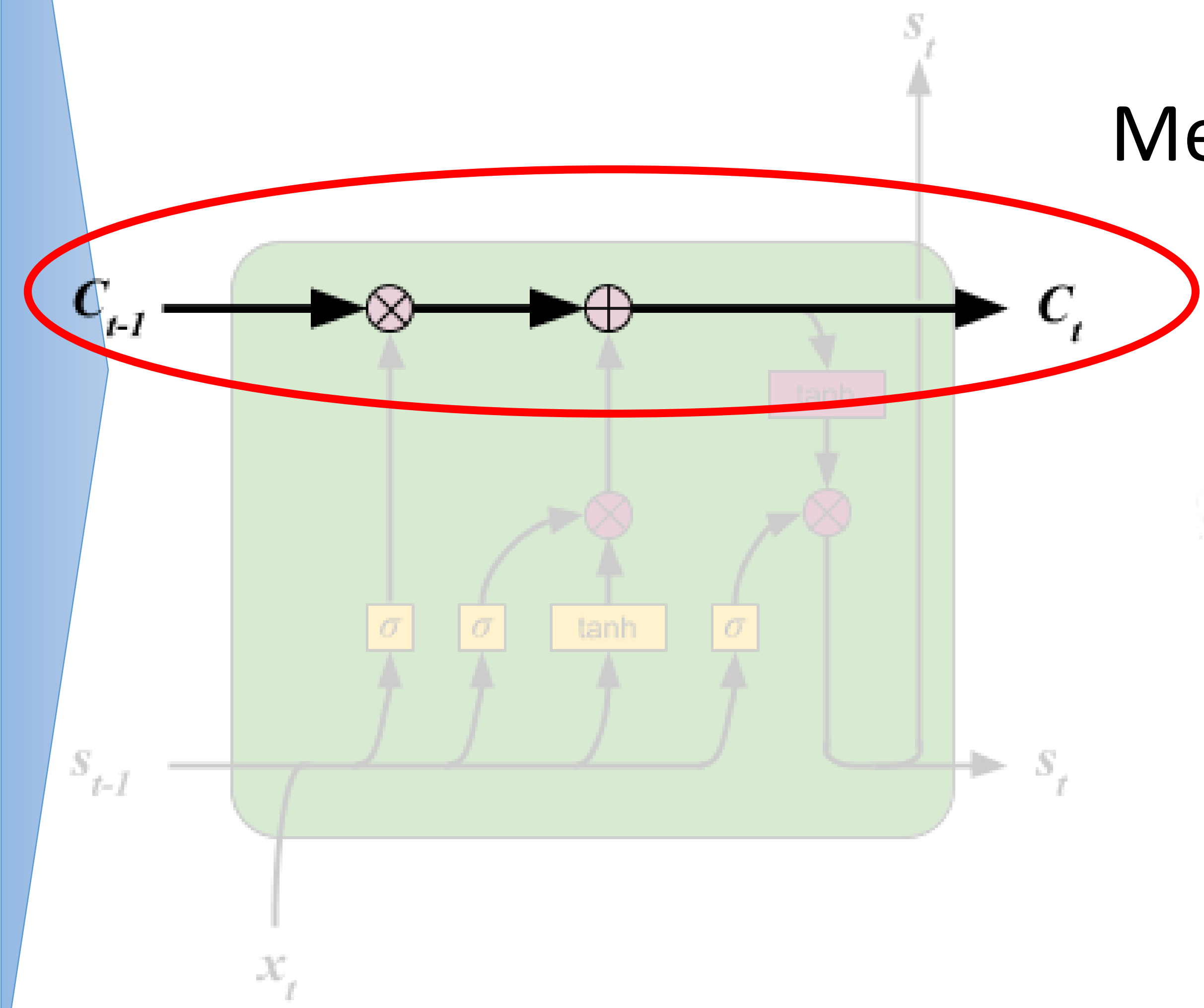


Ingat Nilai f_t diperoleh dari mana ??

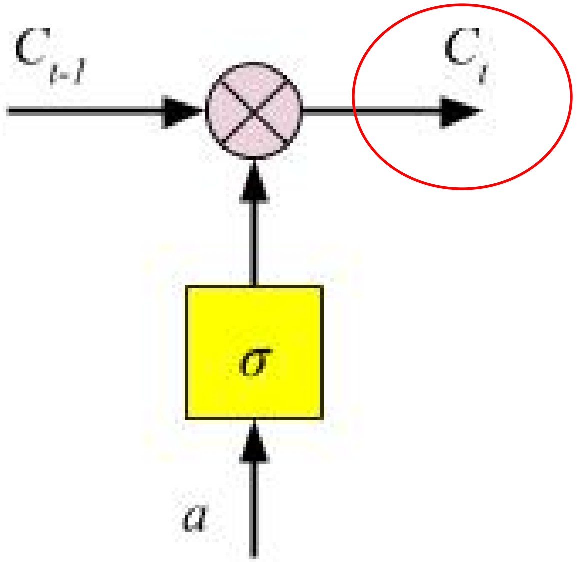
Ingat Nilai $\hat{C}_t$ diperoleh dari mana ??

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

Step 4 Cell State



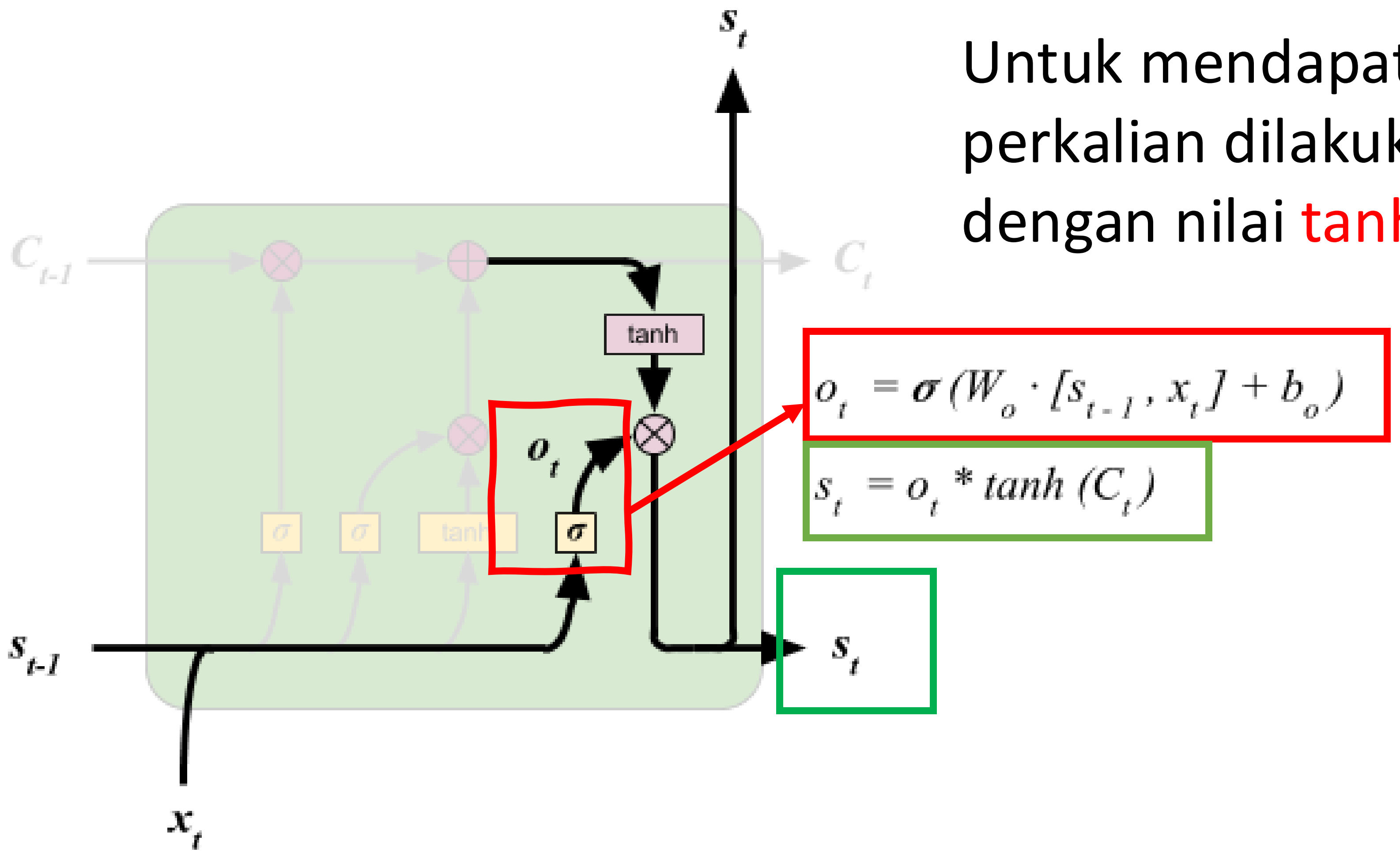
Memory Lampau (Long) atau Cell State



Luaran Cell State

Step 5 Output

Untuk mendapatkan nilai S_t dilakukan perkalian dilakukan perkalian nilai O_t dengan nilai $\tanh(C_t)$



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

$$s_t = o_t * \tanh(C_t)$$

Contoh Program Sederhana

INSTITUT TEKNOLOGI PLN

Matkul Deep Learning SI.ipynb

File Edit View Insert Runtime Tools Help

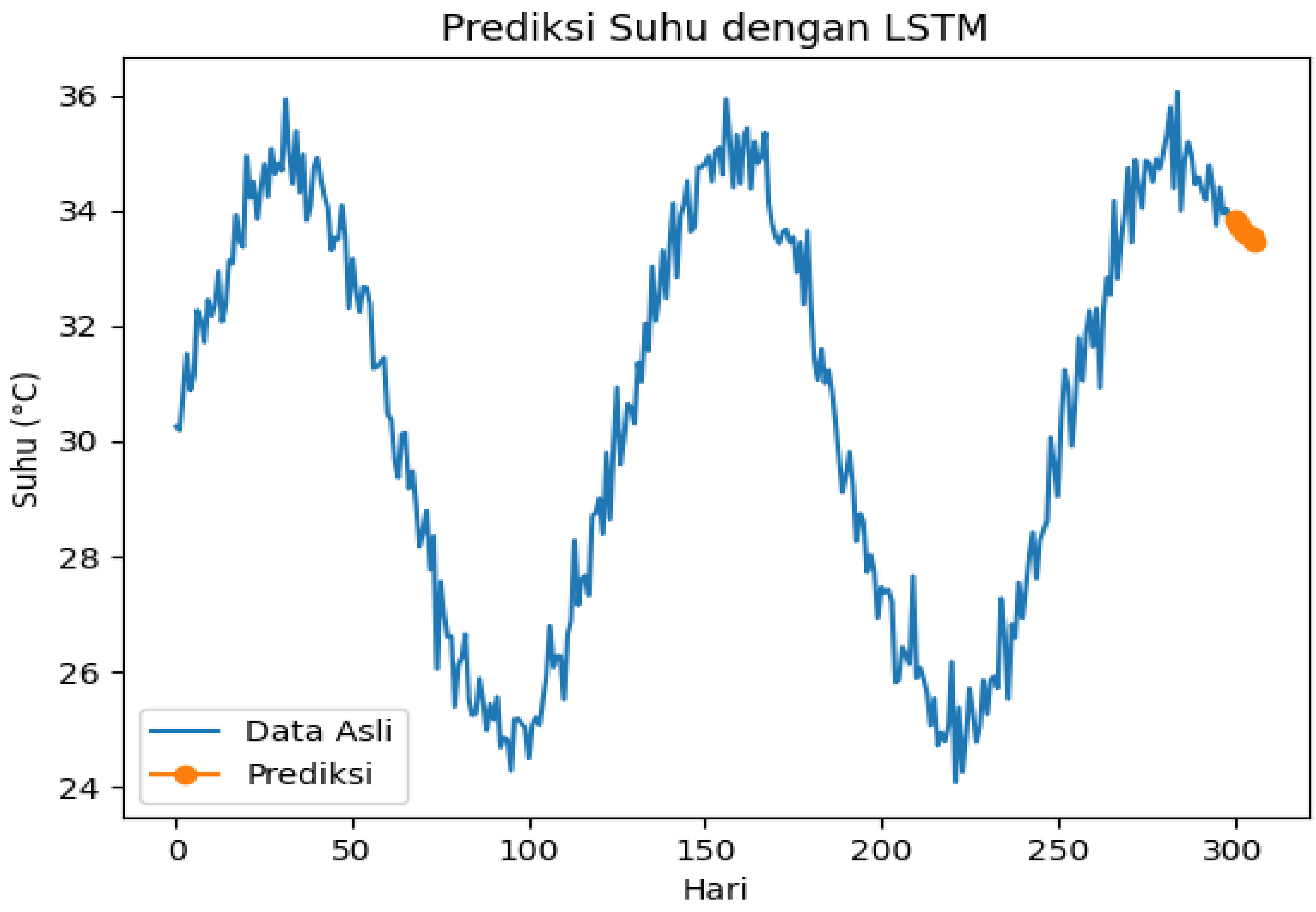
Commands + Code + Text Run all

[5]

```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.preprocessing import MinMaxScaler
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import LSTM, Dense

#Simulasi Dataset Suhu Misal suhu harian selama 300 hari
np.random.seed(42)
days = np.arange(0, 300)
temperature = 30 + 5 * np.sin(days * 0.05) + np.random.normal(0, 0.5, len(days))

df = pd.DataFrame({"day": days, "temperature": temperature})

# Normalisasi Data
scaler = MinMaxScaler()
scaled_temp = scaler.fit_transform(df[['temperature']])

# Fungsi untuk membuat dataset berbasis window time-series
def create_dataset(data, window_size=10):
    X, y = [], []
    for i in range(len(data) - window_size):
        X.append(data[i:i+window_size])
        y.append(data[i+window_size])
    return np.array(X), np.array(y)
window = 10
X, y = create_dataset(scaled_temp, window)
```

```python
# Ubah input ke bentuk LSTM: (samples, timesteps, features)
X = X.reshape((X.shape[0], X.shape[1], 1))
model = Sequential([
    LSTM(64, activation='tanh', return_sequences=True, input_shape=(window, 1)),
    LSTM(32, activation='tanh'),
    Dense(1)
])
model.compile(optimizer="adam", loss="mse")
print(model.summary())
history = model.fit(X, y, epochs=30, batch_size=16, verbose=1)

# Prediksi 7 Hari ke Depan
future_predictions = []
last_window = scaled_temp[-window:].reshape(1, window, 1)
for _ in range(7):
    next_pred = model.predict(last_window)[0][0]
    future_predictions.append(next_pred)

    # update window dengan prediksi baru
    last_window = np.append(last_window[:,1:,:], [[[next_pred]]], axis=1)

# Kembalikan ke nilai suhu asli
future_predictions = scaler.inverse_transform(np.array(future_predictions).reshape(-1,1))

print("\n Prediksi Suhu 7 Hari ke Depan:")
print(future_predictions.flatten())
# Visualisasi Hasil
plt.plot(df['temperature'], label="Data Asli")
plt.plot(range(len(df), len(df)+7), future_predictions, marker='o', label="Prediksi")
plt.legend()
plt.title("Prediksi Suhu dengan LSTM")
plt.xlabel("Hari")
plt.ylabel("Suhu (°C)")
plt.show()
```

Hasil Prediksi



Implementasi LSTM

Chaos, Solitons and Fractals
Nonlinear Science, and Nonequilibrium and Complex Phenomena
journal homepage: www.elsevier.com/locate/chaos

ELSEVIER

Predictions for COVID-19 with deep learning models of LSTM, GRU and Bi-LSTM

Farah Shahid, Aneela Zameer*, Muhammad Muneeb

Department of Computer & Information Sciences, Pakistan Institute of Engineering & Applied Sciences (PIEAS),

Prediksi Covid

ARTICLE INFO

Article history:
Received 10 July 2020
Accepted 16 August 2020
Available online 19 August 2020

Keywords:
Deep learning models
Bi-LSTM
GRU
Corona virus
COVID-19
epidemic prediction

ABSTRACT

COVID-19, responsible for infecting billions o study of the trend it follows to develop adeq ber of future cases. In this perspective, it is system to avoid deaths as well as managing ing autoregressive integrated moving average memory (LSTM), bidirectional long short term of confirmed cases, deaths and recoveries in mance of models is measured by mean abso the majority of cases, Bi-LSTM model outper good performance to the lowest in entire sc generates lowest MAE and RMSE values of 0.(r2_score value is 0.9997 for recovered cases i hanced prediction accuracy, Bi-LSTM can be management.

Algorithmic Trading using LSTM-Models for Intraday Stock Predictions

David Benjamin Lim
Department of Mathemtics
Stanford University
benlim@stanford.edu

Justin Lundgren
MS&E
Stanford University
julu1@stanford.edu

Prediksi Saham

Abstract

We investigate deep learning methods for return predictions on a portfolio of stocks in the information technology sector. We deploy standard time series models along-

...models work. Section 4 contains the results of our models on the test set, a simple trading strategy and an evaluation of this strategy on the data. We conclude in Section 5 with a discussion of future work.

ata

e use a data set available online [1] that has intraday series data at one minute intervals for all stocks in the 500 between 9/11/2017 and 2/16/2018. However, in to have a feasible strategy to act on, we only use times- s that are five minutes apart. Each timestamp reports price, high price, low price, open price and volume

International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878, Volume-8 Issue-4, November 2019

Sentence Wise Telugu to English Translation of Vemana Sathakam using LSTM

P.Sujatha, D. Lalitha Bhaskari

Translate Bahasa

Abstract: *Language translation is a power of humans where machines are lagging and need to acquire. Previous statistical machine translation is used for translation but is applicable for large and similar grammar structure dataset. In this paper neural machine translation ... for addressing the issue. ... translate Telugu literary ... exhibited satisfactory translation. The results are compared with existing and proposed methods. NMT with LSTM yields better in language translation.*

Keywords : *Machine translation, Neural machine translation, Long Short Term Memory.*

I. INTRODUCTION

From past 50 years machine translation was one of the initial task in research field by computer scientists. It can also referred as automatic translation. Machine translation system mainly of three types namely rule based system,

II. MACHINE TRANSLATION

A. Neural Machine Translation

Neural machine translation is a new methodology for ... artificial neural networks to increase ... accuracy. It is a simple encoder-decoder network model for machine translation. The ultimate goal of neural machine translation system is takes a sentence from one language as input and translate that sentence into other language as output. For any machine translation system the first task is to convert textual data into a numeric form. If we want to convert any textual data into numeric form, we have to transform each word into one hot encoding vector which is given as input to the model. Every word has given an index starting with 0. So that each word can have a corresponding hot encoding vector thus we can represent our dataset with numerical representation. This

“If you want something new, you have to stop doing something old”

Peter F. Drucker