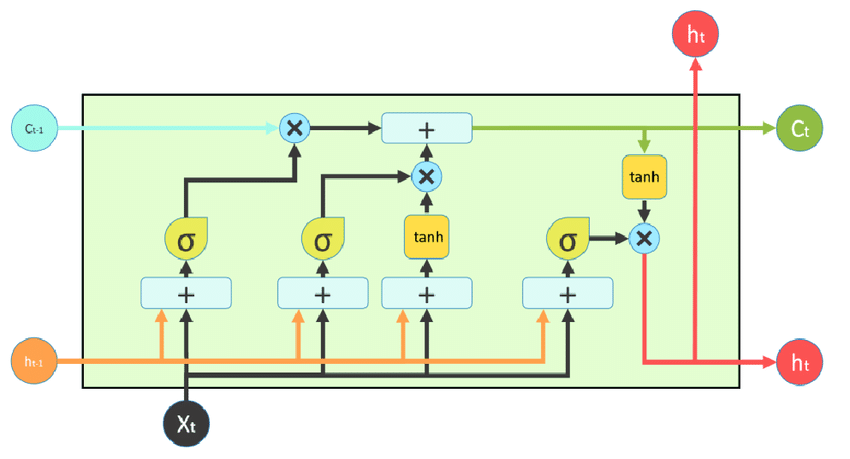
S. Hochreiter et al. came up with the idea by introducing a novel, efficient, gradient based method called long short-term memory (LSTM) to address the issue of vanishing gradients in traditional Recurrent Neural Networks. The LSTM network architecture consists of three parts; these three parts of an LSTM unit are known as gates. They control the flow of information in and out of the memory cell or LSTM cell. The first gate is called Forget gate, the second gate is known as the Input gate, and the last one is the Output gate. An LSTM unit that consists of these three gates and a memory cell or LSTM cell can be considered as a layer of neurons in a traditional feedforward neural network, with each neuron having a hidden layer and a current state.

**Forget Gate**: the first stage in architecture is Forget Gate. In this stage, the LSTM neural network will determine which elements of the cell state (long-term memory) are relevant based on the previous hidden state and the new input data.Calculate the forget gate according to the following formula:

\begin{equation} f\_{t}=\sigma(W\_{f}X\_{t} + U\_{f}h\_{t-1} + b\_{f}) \end{equation}

These output values are then multiplied element-wise with the previous cell state (C\_t−1).

**Input Gate**: the following stage involves the input gate and the new memory network. The objective of this step is to identify what new information should be incorporated into the network's long-term memory (cell state), based on the previous hidden state and the current input data. Calculate the input gate according to the following formula:

\begin{equation} i\_{t}=\sigma(W\_{i}X\_{t} + U\_{i}h\_{t-1} + b\_{i}) \end{equation}

The new memory network is a neural network that uses the tanh activation function and has been trained to create a "new memory update vector" by combining the previous hidden state and the current input data. Calculate the new memory network according to the following formula:

\begin{equation} \tilde{C}\_{t}=\tanh(W\_{c}X\_{t} + U\_{c}h\_{t-1} + b\_{c}) \end{equation}

The updated cell state is then passed through a tanh activation to limit its values to [-1,1] before being multiplied pointwise by the output of the output gate network to generate the final new hidden state.

\begin{equation} C\_{t} = f\_t \circ C\_{t-1} + i\_t \circ \tilde{C}\_t \end{equation}

**Output Gate**: In the final stage of an LSTM, the new hidden state is determined using the newly updated cell state, previous hidden state, and new input data. The output gate performs this decision-making process. Calculate the output gate according to the following formula:

\begin{equation} O\_{t} = \sigma(W\_{o}X\_{t} + U\_{o}h\_{t-1} + b\_{o}) \end{equation}

The updated cell state is then passed through a tanh activation to limit its values to [-1,1] before being multiplied pointwise by the output of the output gate network to generate the final new hidden state.

\begin{equation} h\_{t}=O\_{t} \circ \tanh(C\_{t}) \end{equation}

These output values are then multiplied element-wise with the previous cell state (Ct−1).

**Trích dẫn**

1. S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," Neural Comput., vol. 9, no. 8, pp. 1735-1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.
2. S. Saxena, "Learn About Long Short-Term Memory (LSTM) Algorithms," Analytics Vidhya, Mar. 16, 2021.<https://www.analyticsvidhya.com/blog/2021/03/introduction-to-long-short-term-memory-lstm/> (accessed Jun. 15, 2023).

\begin{document}

S. Hochreiter et al. came up with the idea by introducing a novel, efficient, gradient based method called long short-term memory (LSTM) to address the issue of vanishing gradients in traditional Recurrent Neural Networks. The LSTM network architecture consists of three parts; these three parts of an LSTM unit are known as gates. They control the flow of information in and out of the memory cell or LSTM cell. The first gate is called Forget gate, the second gate is known as the Input gate, and the last one is the Output gate. An LSTM unit that consists of these three gates and a memory cell or LSTM cell can be considered as a layer of neurons in a traditional feedforward neural network, with each neuron having a hidden layer and a current state.

\section\*{Forget Gate}

The first stage in architecture is Forget Gate. In this stage, the LSTM neural network will determine which elements of the cell state (long-term memory) are relevant based on the previous hidden state and the new input data. Calculate the forget gate according to the following formula:

\begin{equation}

f\_{t}=\sigma(W\_{f}X\_{t} + U\_{f}h\_{t-1} + b\_{f})

\end{equation}

The output values of the forget gate ($f\_t$) are then multiplied element-wise with the previous cell state ($C\_{t-1}$) as part of the cell state update (see Equation~\ref{eq:cell\_state\_update}).

\section\*{Input Gate}

The following stage involves the input gate and the new memory network. The objective of this step is to identify what new information should be incorporated into the network's long-term memory (cell state), based on the previous hidden state and the current input data. Calculate the input gate according to the following formula:

\begin{equation}

i\_{t}=\sigma(W\_{i}X\_{t} + U\_{i}h\_{t-1} + b\_{i})

\end{equation}

The new memory network is a neural network that uses the $\tanh$ activation function and has been trained to create a "new memory update vector" by combining the previous hidden state and the current input data. Calculate the new memory network according to the following formula:

\begin{equation}

\tilde{C}\_{t}=\tanh(W\_{c}X\_{t} + U\_{c}h\_{t-1} + b\_{c})

\end{equation}

\section\*{Cell State Update}

The cell state is updated by combining the influence of the forget gate on the previous cell state and the input gate on the new candidate values.

\begin{equation}

C\_{t} = f\_t \circ C\_{t-1} + i\_t \circ \tilde{C}\_t \label{eq:cell\_state\_update}

\end{equation}

\section\*{Output Gate}

In the final stage of an LSTM, the new hidden state is determined using the newly updated cell state, previous hidden state, and new input data. The output gate performs this decision-making process. Calculate the output gate according to the following formula:

\begin{equation}

O\_{t} = \sigma(W\_{o}X\_{t} + U\_{o}h\_{t-1} + b\_{o})

\end{equation}

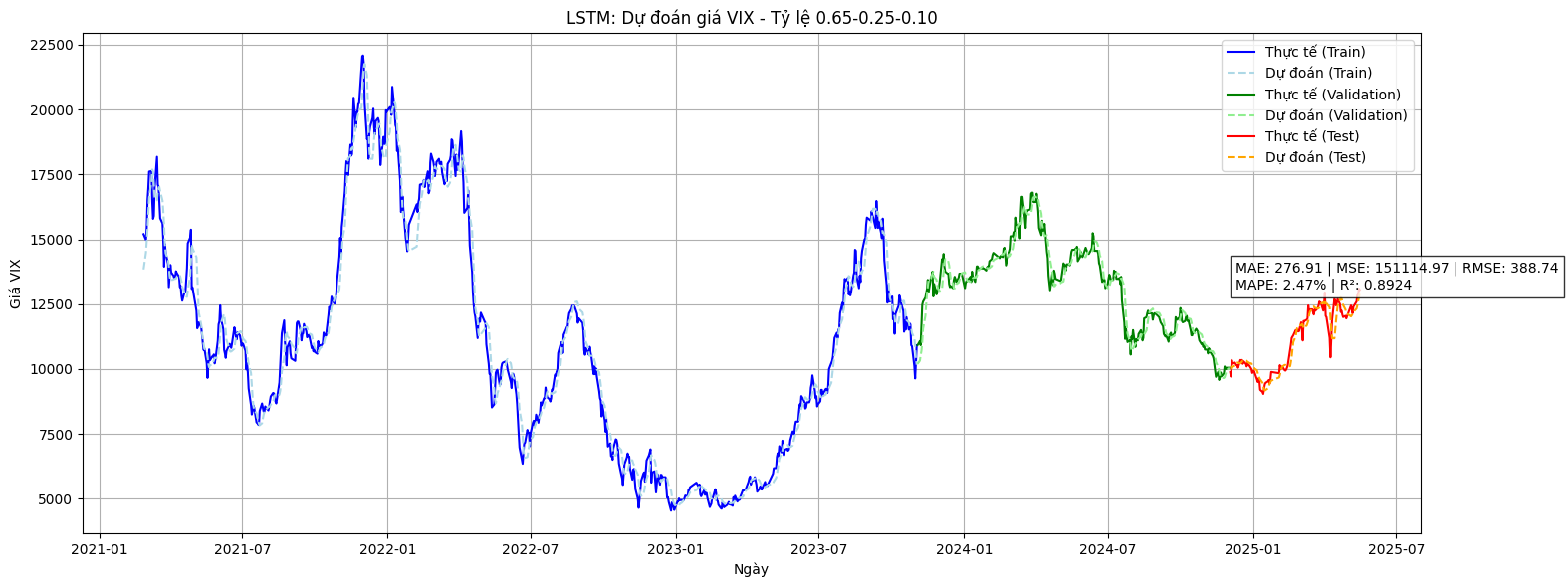
The updated cell state is then passed through a $\tanh$ activation to limit its values to [-1,1] before being multiplied pointwise by the output of the output gate network to generate the final new hidden state.

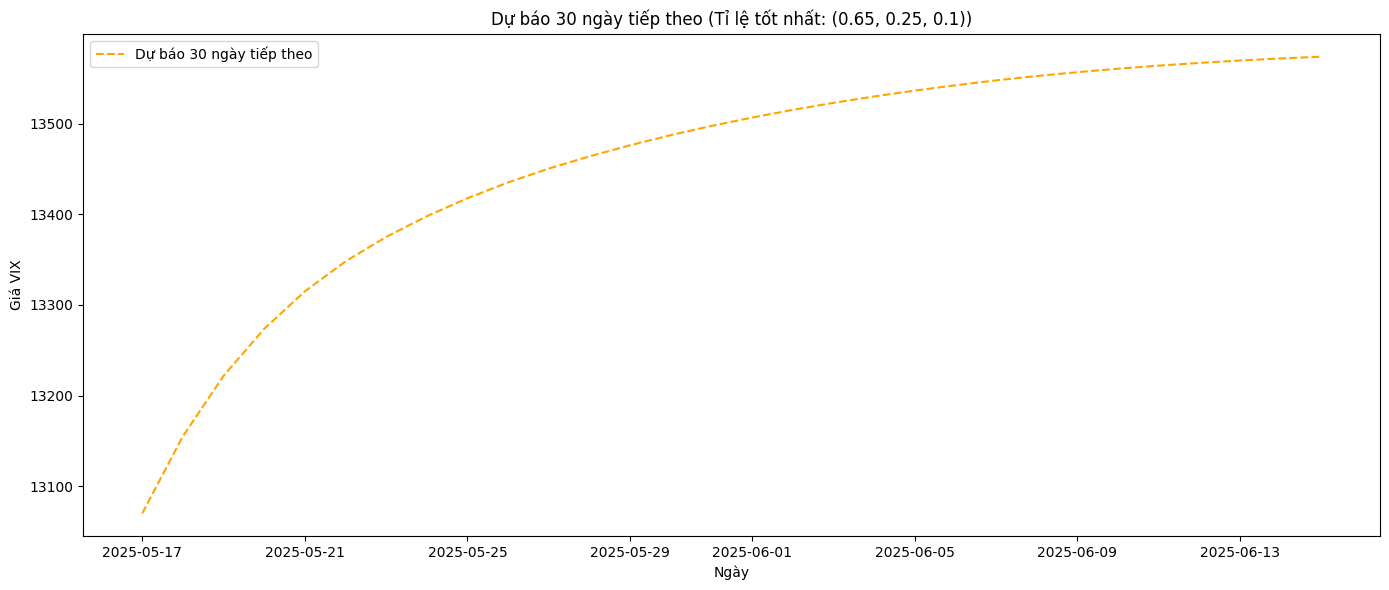
\begin{equation}

h\_{t}=O\_{t} \circ \tanh(C\_{t})

\end{equation}

\end{document}





|  | **MSE** | **RMSE** | **MAPE** | **MAE** | **R-SQUARE** |
| --- | --- | --- | --- | --- | --- |
| 65/25/10 | 311830.7117 | 558.4180 | 3.43% | 403.5856 | 0.8691 |
| 70/10/20 | 113906.9837 | 337.5011 | 2.11% | 237.0212 | 0.9092 |
| 75/10/15 | 130040.7985 | 360.6117 | 2.37% | 265.0320 | 0.8866 |

Batch\_size = 32

epochs = 50

Lookback = 30

Callback = 5