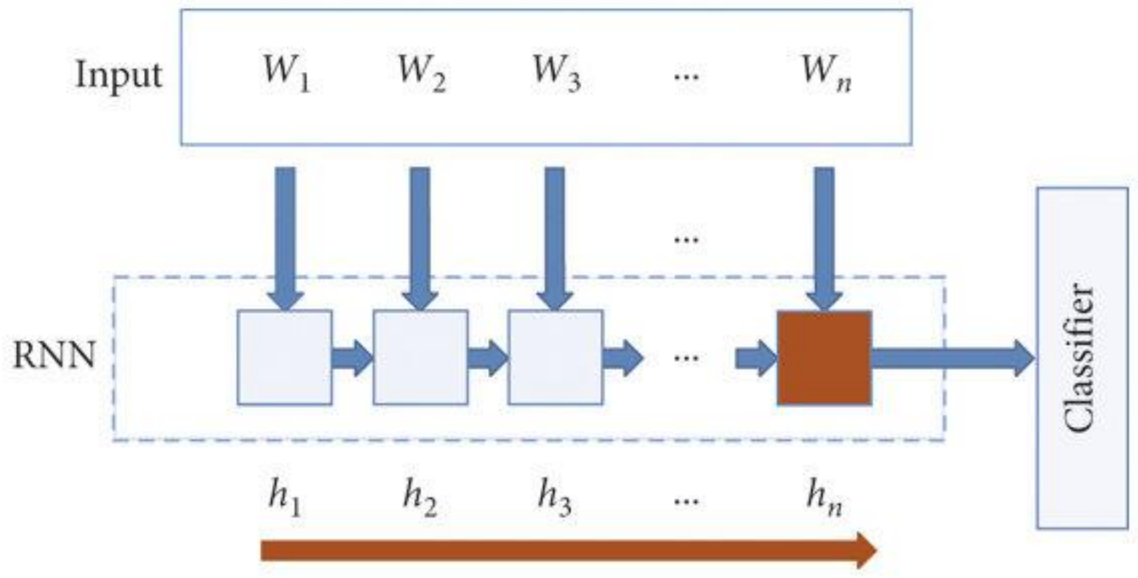
Assignment #3: RNN Implementation

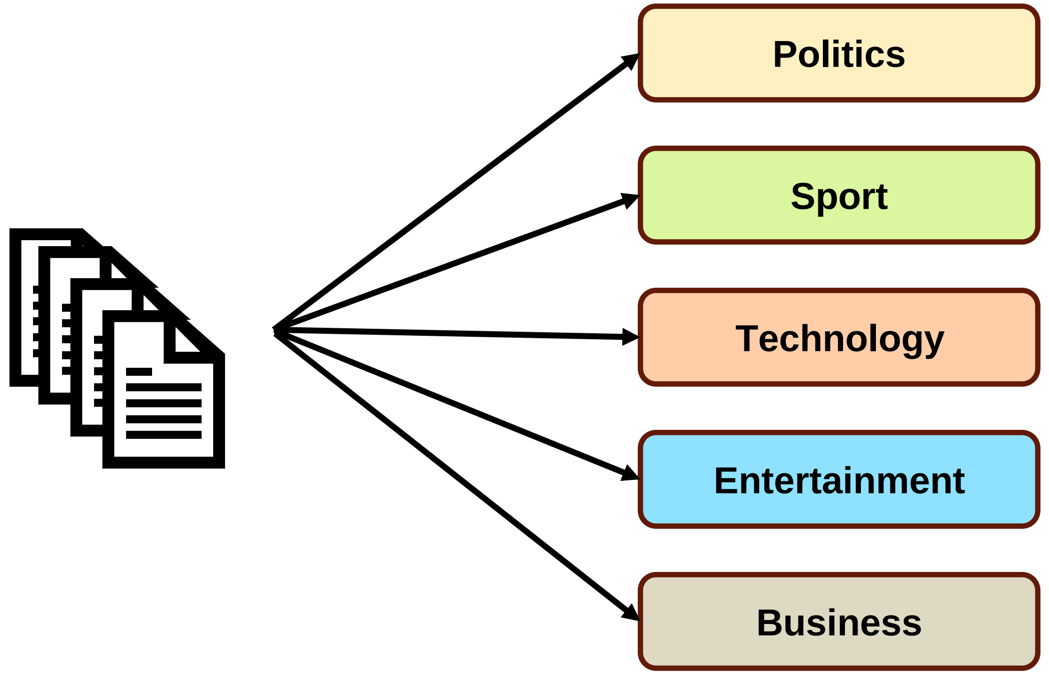
Paul Hongsuck Seo

Korea University

RNN Implementation

Implement RNN Classifier



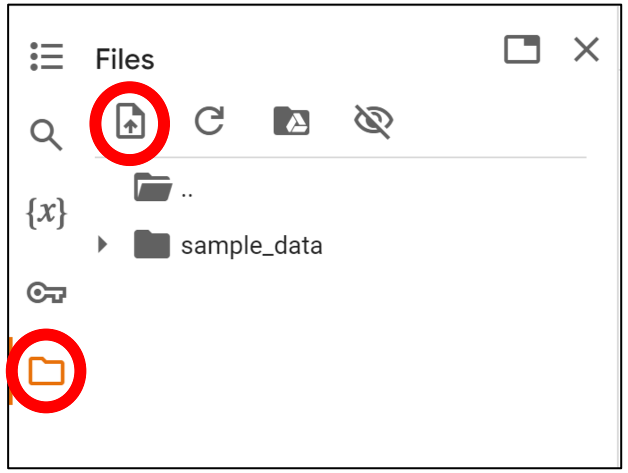


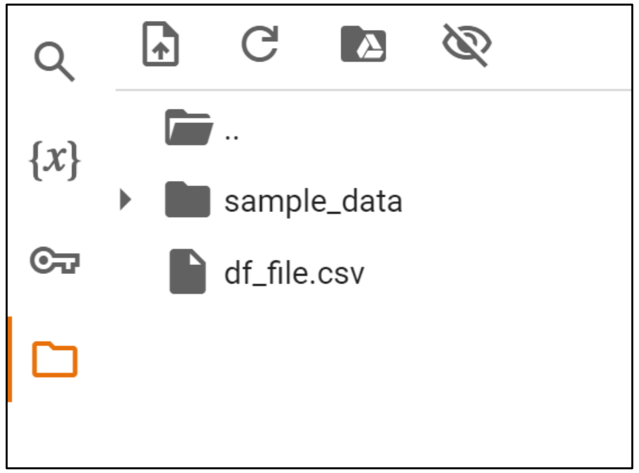
- Perform the text classification Recurrent Neural Network (vanilla RNN and GRU).
- The Ipython Notebook "RNN_Implementation.ipynb" will walk you through the implementation of RNN classifier.

RNN Implementation

Requirements

- Download the attached zip file
- Open the **RNN_Implementation.ipynb** with colab notebook
- Upload the other **df_file.csv** to session storage





RNN Implementation

Instructions

- Follow the instructions in the **RNN_Implementation.ipynb** notebook to complete the assignment.
  - Load the **text documentation** data **(No need for any modifications)**
  - Preprocessing the data **(No need for any modifications)**
  - Complete the vanilla RNN code and train the vanilla RNN model
  - Complete the GRU code and train the GRU model
  - Complete **GRU_skeleton.py** and **RNN_skeleton.py**

    → same as the cells in RNN_Implementation.ipynb

Text Documentation Classification Dataset

- Text Documentation Classification Dataset
  - Contains 2225 text data and five categories of documents
  - We can use this dataset for documents classification
  - https://www.kaggle.com/datasets/tanishqdublish/text-classification-documentation/data
- Dataset Composition
  - The dataset is provided in CSV format (2225 Rows and 2 Columns)
  - It consists different categories of text data and labels
  - Five different categories: Politics:0, Sport:1, Technology:2, Entertainment:3, Business:4

Budget to set scene for election

Gordon Brown will seek to put the economy at the centre of Labour's bid for a third term in power when he delivers his ninth Budget at 1230 GMT.
…

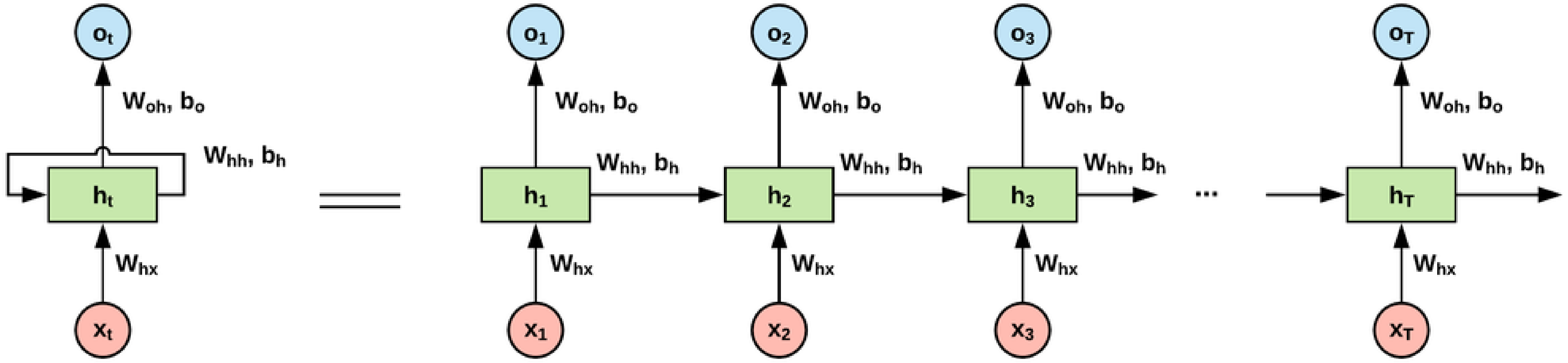
$\longrightarrow$ **Label: 0 (Politics)**

KOREA UNIVERSITY Multimodal Interactive Intelligence Laboratory

RNNs

Recurrent Neural Networks

- RNNs are neural networks for sequential data processing
- RNNs process data across multiple time steps, making them well adapted for modelling and processing text, speech and time series.
- We will develop a recurrent neural network with **vanilla RNN** and **GRU**

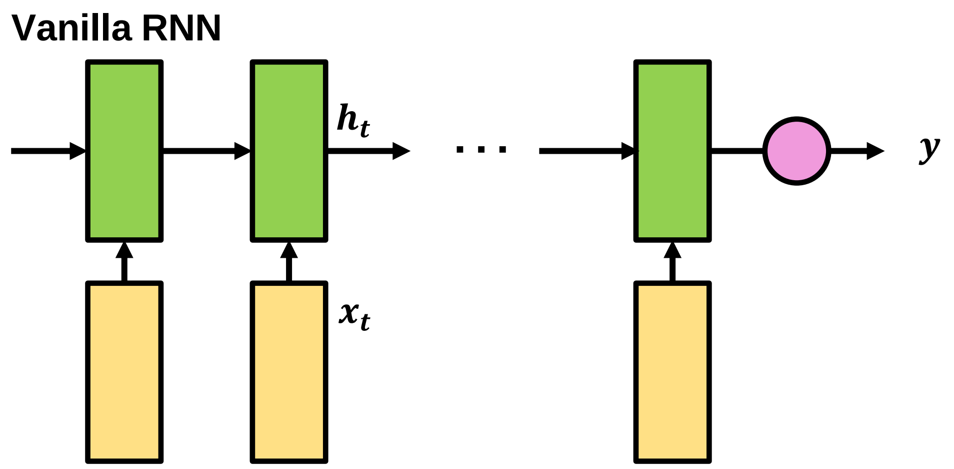


Vanilla RNN

Vanilla RNN

- Vanilla RNN is the most basic type of RNN architecture.
- Vanilla RNN takes the current input x_t and previous hidden state h_{t-1} to compute the current hidden state h_t
- After processing the entire sentence, the hidden state reflects the context of the full sequence.
- The final output passes through a fully connected (FC) layer.

$$h_t = \tanh(W_x x_t + W_h h_{t-1} + b_h)$$
$$y = \tanh(W_y h_t + b_y)$$



GRU

GRU (Gated Recurrent Unit)

- **GRU** is a type of RNN designed to handle sequential data and avoid the vanishing gradient problem.
- It uses two gates: **reset** and **update**, to control the flow of information.
- The **reset gate** decides how much past information to forget, and the **update gate** controls the new hidden state based on the current input and previous state.
- The final output is computed after processing the sequence, and it can pass through a fully connected (FC) layer.
- https://arxiv.org/abs/1406.1078

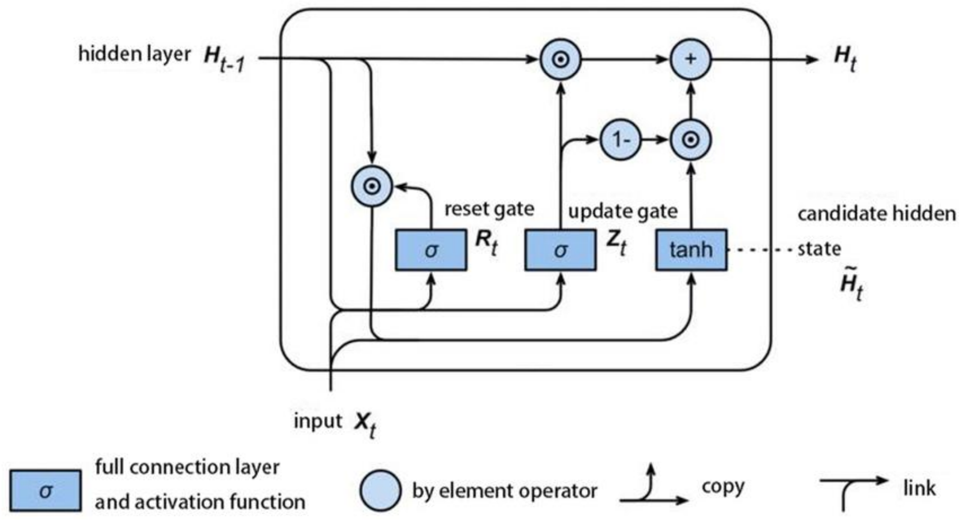
GRU

GRU (Gated Recurrent Unit)

- The structure of the GRU is as shown in the diagram and equations below.
- You need to understand the structure of the GRU and correctly fill in the blanks in the code.

$$
\begin{aligned}
z_t &= \sigma(W_z x_t + U_z h_{t-1} + b_z) \\
r_t &= \sigma(W_r x_t + U_r h_{t-1} + b_r) \\
\hat{h}_t &= \tanh(W_h x_t + U_h (r_t \odot h_{t-1}) + b_h) \\
h_t &= (1 - z_t) \odot h_{t-1} + z_t \odot \hat{h}_t
\end{aligned}
$$

σ: activation function (sigmoid)



RNN Implementation

- You must submit **“RNN_skeleton.py”** and **“GRU_skeleton.py”** along with the **report**. ***(Do not modify the name of the Python file.)***
- Include a **1 page** report in **CVPR** format that describes your code, results, and discussions.
- The report should be written in **English**.

CVPR format : https://cvpr.thecvf.com/Conferences/2025/AuthorGuidelines
→ Download CVPR 2025 Author Kit

RNN Implementation

Please do NOT copy your friends' and internet sources.

Please start your assignment EARLY.

"Late submissions will not be accepted"