



LINE Chatbot

解決問題

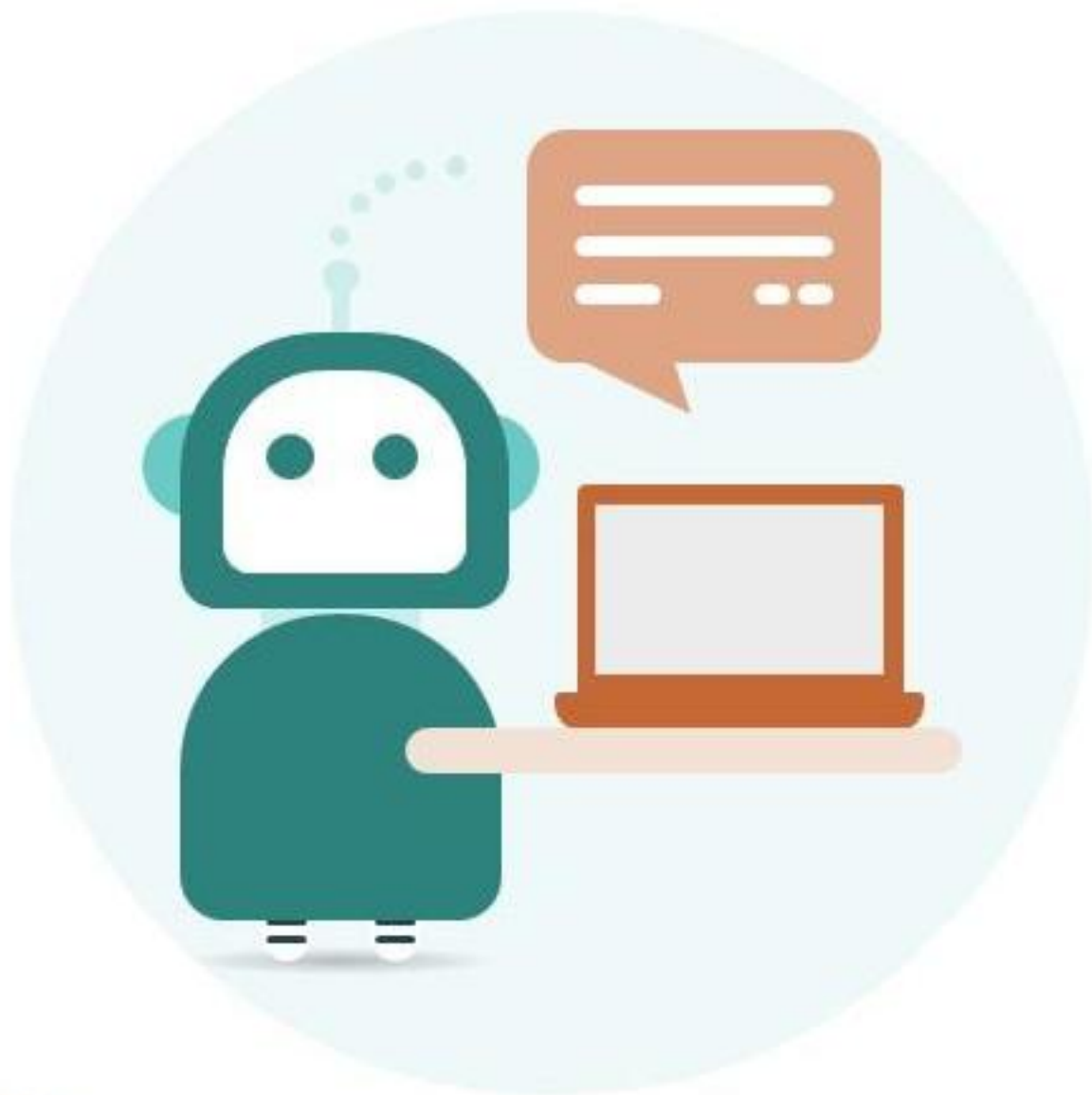
使用聊天機器人通常要另外打開瀏覽器或是安裝應用程式，這樣的過程相對來說比較繁瑣。透過LINE的平台則可以解決這個問題。



功能

能夠在LINE上執行簡單的問答。





實現方法

1. **模型選擇**：訓練一個自然語言處理模型。
2. **雲端部署**：將模型部署到Google Cloud上，通過網絡提供服務給用戶。。
3. **Line 整合**：將聊天機器人整合到 Line 平台，讓用戶可以直接通過 Line 與機器人對話。

資料準備流程圖

change.ipynb
資料準備

1. 載入簡體中文數據。
(train.txt)
2. 使用 OpenCC 將簡體中文轉換為繁體中文。
3. 將轉換後的內容保存到
新文件 (train2.txt)

資料集來源:

https://drive.google.com/file/d/1nEuww_KNpTMbyy7BO4c8bXMXN351RCp/view

sol_data.py
資料預處理

1. 讀取 train2.txt 文件。
2. 正規化標點符號。
3. 用 <EOS> 分隔多輪對話。
4. 將處理後的數據保存到新文件。
(dataset.txt)

word2seq.py
詞典生成

1. 定義 Word2Sequence 類：

fit 方法：構建詞彙表。
to_index 方法：將詞轉換為索引。
to_word 方法：將索引轉換為詞。
pad 方法：填充序列到固定長度。
2. 從 dataset.txt 中構建詞彙表。
3. 將詞彙表序列化保存 (ws.pkl)。

模型訓練流程圖

dataset.py
資料集載入

train.py
模型訓練

test.py
聊天機器人測試

定義 ChatDataset 類：

init方法：初始化數據集。

getitem 方法：獲取數據集中一個樣本。

len 方法：返回數據集的大小。

包含了對話的輸入和輸出，以及它們的長度。這些數據被組織成批次，每個批次中的樣本長度保持一致，便於模型訓練。

1. 加載詞彙表和數據集。
2. 初始化模型參數。定義損失函數（ LossWithLS ）。
3. 定義優化器（ Adam ）。
4. 訓練模型：前向傳播、計算損失、反向傳播和優化。
5. 保存訓練好的模型。

LossWithLS :帶有標籤平滑的損失函數

1. 加載訓練好的模型。
2. 定義生成回應的函數：
 - 輸入處理。
 - 模型推理。
 - 輸出處理。

檔案(F)編輯(E)選取項目(S)檢視(V)移至(G)PY4

train.py

config.py

__pycache__

.vscode

logs

change.ipynb

chat.py

config.py

dataset.py

dataset.txt

debug_word.txt

debug_word2.txt

gpt_model.py

model.pth.rar

requirements.txt

sol_data.py

test.py

token_and_secret...

train.py

train.txt

train2.txt

utils.py

word2seq.py

ws.pkl

train.py > ...

config.py > ...

問題2輸出偵錯主控台終端機連接埠

Epoch [172][16/58496]	Loss: 2.360	Acc: 48.173%
Epoch [172][17/58496]	Loss: 2.373	Acc: 47.851%
Epoch [172][18/58496]	Loss: 2.388	Acc: 47.862%
Epoch [172][19/58496]	Loss: 2.387	Acc: 47.698%
Epoch [172][20/58496]	Loss: 2.383	Acc: 47.698%
Epoch [172][21/58496]	Loss: 2.399	Acc: 47.274%
Epoch [172][22/58496]	Loss: 2.395	Acc: 47.212%
Epoch [172][23/58496]	Loss: 2.397	Acc: 47.225%
Epoch [172][24/58496]	Loss: 2.400	Acc: 47.138%
Epoch [172][25/58496]	Loss: 2.407	Acc: 47.098%
Epoch [172][26/58496]	Loss: 2.409	Acc: 47.021%
Epoch [172][27/58496]	Loss: 2.408	Acc: 46.886%
Epoch [172][28/58496]	Loss: 2.406	Acc: 46.905%
Epoch [172][29/58496]	Loss: 2.407	Acc: 46.844%
Epoch [172][30/58496]	Loss: 2.405	Acc: 46.920%
Epoch [172][31/58496]	Loss: 2.408	Acc: 46.870%
Epoch [172][32/58496]	Loss: 2.416	Acc: 46.814%
Epoch [172][33/58496]	Loss: 2.412	Acc: 46.988%
Epoch [172][34/58496]	Loss: 2.409	Acc: 47.029%
Epoch [172][35/58496]	Loss: 2.408	Acc: 47.011%

powershell

Python

第 13 行, 第 1 欄 空格: 4 UTF-8 CRLF Python 3.10.7 64-bit 2 Spell Prettier

經過了好幾天的訓練之後，模型訓練的準確度穩定在四成以上。

雲端佈署

Google Cloud

Google Cloud

Line Chat Bot

搜尋資源、文件、產品和其他項目 (請按 / 鍵)

搜尋



18



冠欣

Compute Engine



虛擬機器 (VM) 執行個體

建立執行個體

匯入 VM

重新整理

瞭解詳情

虛擬機器



VM 執行個體

執行個體範本

單一用戶群節點

機器映像檔

TPU

承諾使用折扣

保留項目

Migrate to Virtual Machin...

儲存空間



Marketplace

版本資訊

<|

執行個體

觀測能力

執行個體排程

虛擬機器 (VM) 執行個體

篩選 輸入屬性名稱或值

<input type="checkbox"/>	狀態	名稱 ↑	可用區	建議	使用者	內部 IP	外部 IP	連線
<input type="checkbox"/>		line4	asia-east1-c			10.140.0.5 (nic0)	104.199.197.147 (nic0)	SSH ▾

相關操作

HIDE

探索備份與災難復原 最新
備份 VM 並設定災難復原

查看帳單報表
查看及管理 Compute Engine 帳單

監控 VM
查看 CPU 和網路等不同指標中的離群 VM

探索 VM 記錄檔
查看、搜尋、分析及下載 VM 執行個體記錄檔

設定防火牆規則
控管 VM 執行個體的往來流量

管理修補程式
排定修補程式的更新時間，以及查看 VM 執行個體安裝修補程式的狀態

平衡 VM 之間的負載
透過設定，讓系統隨著使用者與流量的增加為應用程式進行負載平衡

開始使用 Compute Engine

部署網站或應用程式、備份及還原 VM 和磁碟、設定安全存取機制，以及設計可擴充的 VM

建立網站或應用程式

在 IIS 上建立「Hello World」網站

教學課程 25 分鐘

使用 Compute Engine 建立 IIS 網路伺服器 VM。

在 Apache 上建立「Hello World」網站

教學課程 10 分鐘

在 Linux VM 上建立 Apache 網路伺服器。

將檔案轉移至 Windows VM

教學課程 10 分鐘

上傳檔案到 Cloud Storage 值區，以及從該值區下載檔案到 Windows VM

將檔案傳輸到 Linux VM

教學課程 5 分鐘

瞭解如何將檔案轉入 Linux VM，或是轉出 Linux VM 中的檔案。

Virtual Machine

基本資訊


OS	Ubuntu 20.04.6 LTS
OS 版本	20.04
OS 設定代理程式版本	20240320.00-0ubuntu1~20.04.0
修補狀態	 有可用的更新

建置流程:

1. 安裝Python、pip....要所需工具
2. 上傳需要的檔案(model.pth.rar、chat.py等)
3. 安裝ngrok
4. 執行chat.py、ngrok

```
mh815219@line4:~$ ps aux | grep ngrok
mh815219  101044  0.2  0.3 1255892 29388 pts/0    Ssl  14:31   0:36 /snap/ngrok/159/ngrok http 5000 --log=stdout
mh815219  112682  0.0  0.0   8168   720 pts/0    S+   18:23   0:00 grep --color=auto ngrok
mh815219@line4:~$ ps aux | grep python
root      599  0.0  0.2  29856 18248 ?        Ss   Jun09   0:00 /usr/bin/python3 /usr/bin/networkd-dispatcher --run-startup-triggers
root      757  0.0  0.2 108124 20564 ?        Ssl  Jun09   0:00 /usr/bin/python3 /usr/share/unattended-upgrades/unattended-upgrade-shutdown --wait-for-signal
mh815219  100609  0.0 10.3 3587116 842268 pts/0    Ssl  14:30   0:13 python3 chat.py
mh815219  112685  0.0  0.0   8168   2568 pts/0    S+   18:23   0:00 grep --color=auto python
mh815219@line4:~$
```

Console home

Providers 

Admin

GPT

GX


Tools

Support


TOP

Recently visited channels

Admin



GPT

 Messaging API

← 設置了一個名為GPT的頻道

Providers (2) 

Console home

GPT

GX

Tools

Support

TOP > GPT > GPT > Messaging API

Basic settings Messaging API LIFF Security Statistics Roles

Messaging API settings

Bot information

Bot basic ID @833ymkon

QR code



Scan this QR code with LINE to add your LINE Official Account as a friend. You can share the code with others.

