COMP7940 Cloud Computing

2023/24 S2 Lab 3 - Starting a Telegram chatbot

Role	Name	Email
Qichen Wang	Instructor	qcwang@hkbu.edu.hk
Yu Xu	Teaching Assistant	csyuxu@comp.hkbu.edu.hk
Hongduan Tian	Teaching Assistant	cshdtian@comp.hkbu.edu.hk

Intended Learning Outcomes

Throughout this lab you will be able to:

1. Experience in running a chatbot;
2. Customize your chatbot (greeting message, issue special command);

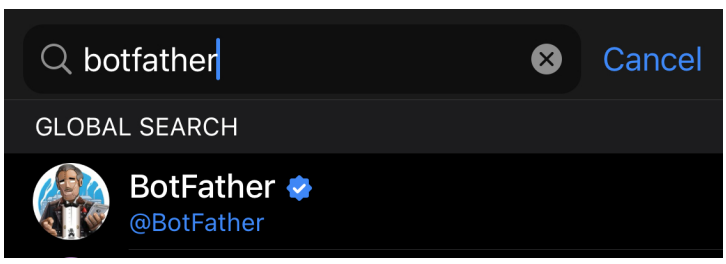
1. Installation

Install the following software on your phone: Telegram

> For iOS users, you may change your Apple ID to a Hong Kong ID (registered using your Hong Kong cellphone number) if you can not download the Telegram app from the Apple Store using a mainland ID.

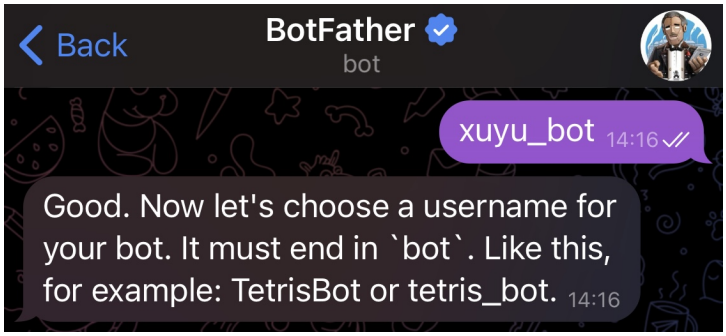
2. Creat a chatbot

Telegram provides an official Bots accounts that do not require additional phone number to set up. You can search and add @BotFather and just talk to the @BotFather follow some simple step to create your own chatbot.



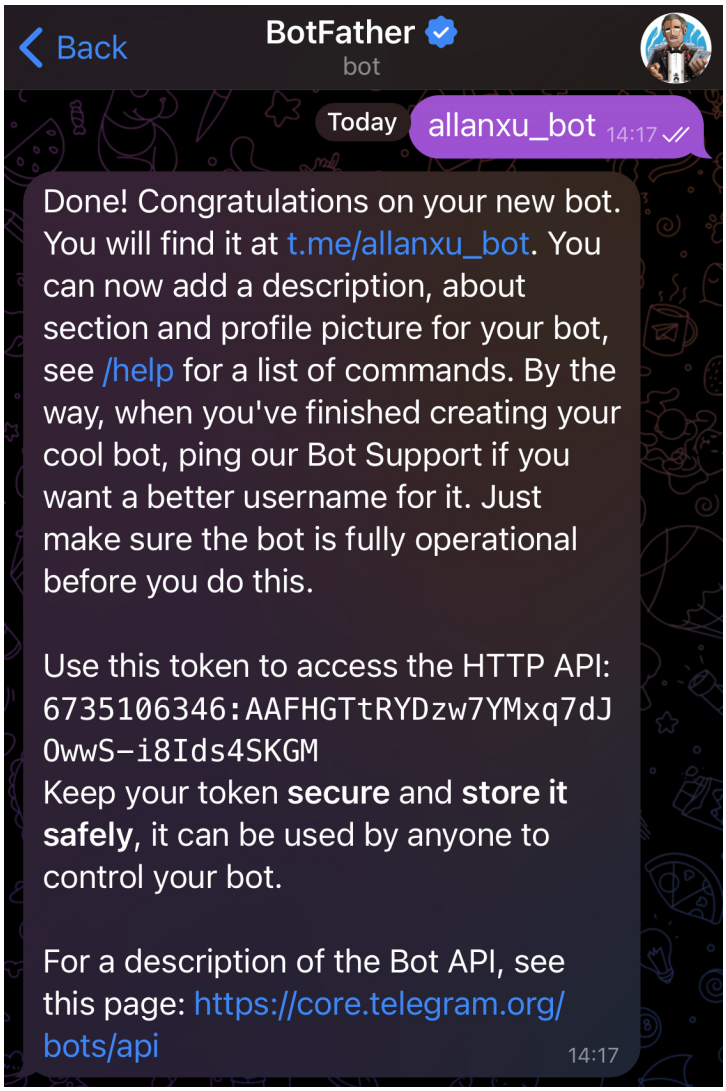
2.1 Send the the message `/newbot` to BotFather (in your phone) to create a new bot

2.2 Type a name for your bot (e.g., xuyu_bot)



2.3 Choose a user name for the bot

it should be end in 'bot' (e.g., allanxu_bot). The BotFather will provide you a link to find your bot and a token to manage your bot:

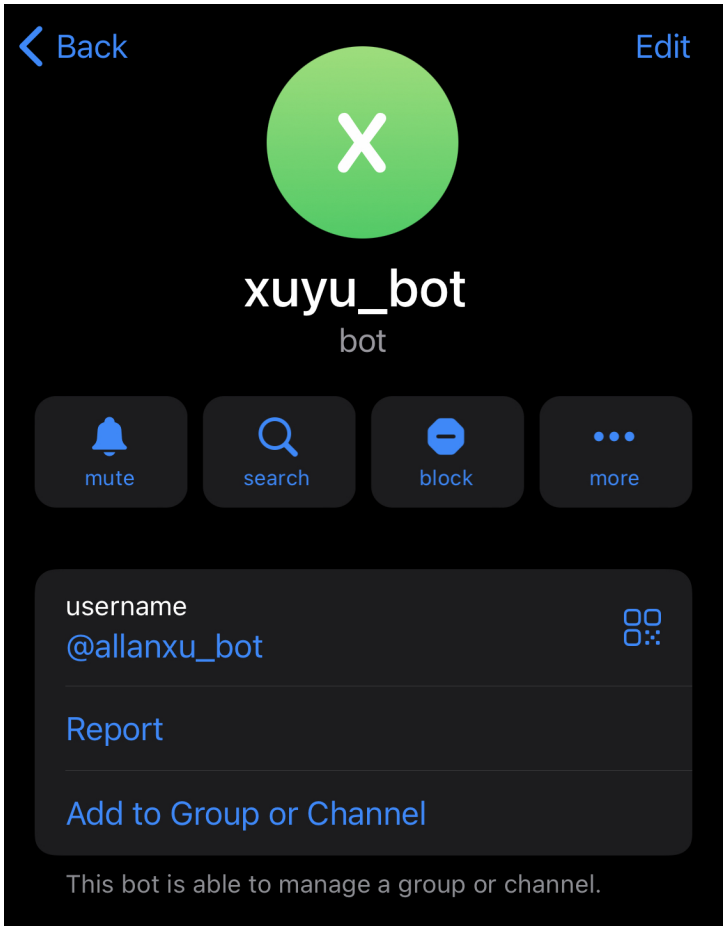


In the above example:

Link of my bot: `t.me/allanxu_bot`

Token of my bot: `6735106346:AAFHGTtRYDzw7YMxq7dJ0wwS-i8Ids4SKGM`

2.4 The information of your bot



2.5 Manage your chatbot using APIs

- Now the chatbot can receive your message but can not response. You can send message to the chatbot and the logs can be accessed at the following website: `https://api.telegram.org/bot6735106346:AAFHGTtRYDzw7YMxq7dJOwwS-i8Ids4SKGM/getUpdates`
- The general format of the Telegram APIs is presented as the following: `https://api.telegram.org/bot<token>/METHOD_NAME`
- You can find more about the Telegram bot API at: `https://core.telegram.org/bots/api`.

3. Developing your chatbot in Python

3.1 Install dependencies (Python packages)

Now we can try to customize our own bot to make some simple response. You need to install the following a few modules.
Clone an empty repository from Github, or reuse the repository created in previous labs. Add a new file `requirements.txt` into your local folder with the following content:

```
telegram
configparser
redis
```

Note that you should use Python with version=3.7 or > 3.7. Type the following in the terminal/command prompt:

```
python -m pip install --upgrade pip
pip install -r requirements.txt
pip install python-telegram-bot==13.7
pip install urllib3==1.26.18
```

3.2 Configure your profile

To enhance security, we utilize a config file to securely store the token and the webhook link:

- create a config file `config.ini`
- save your token in this file as follows:

  ```
  [TELEGRAM]
  ACCESS_TOKEN = bot6735106346:AAFHGTtRYDzw7YMxq7dJOwwS-i8Ids4SKGM
  ```

You can access this config file by calling the python function from the `configparser` module:

```
import configparser
config = configparser.ConfigParser()
config.read('config.ini')
print(config['TELEGRAM']['ACCESS_TOKEN'])
```

> Notice: Do not track and push this configuration file to public repositories.

3.3 Design your chatbot using Python

3.3.1 Python Telegram module

You can manage the chatbot to receive and echo message from Telegram by using the following APIs:

1. telegram.ext.Updater: Continuously fetches new updates from telegram and passes them on to the `Dispatcher` class
2. telegram.ext.Dispatcher: You can register different handlers in this class, it will sort the updates fetched by `Updater` according to the handlers you have registered.
3. telegram.ext.Handler: It contains subclass of handlers for different kind of updates (e.g. text,audio and so on)
4. telegram.ext.Fliters: It contain a number of filter to process the messages such as text, images and more.

> The document of various APIs (Python Telegram) can be found in the website.

3.3.2 A simple example of echo bot

You can add a python file `chatbot.py` with the following source code to start a echo chatbot. Make

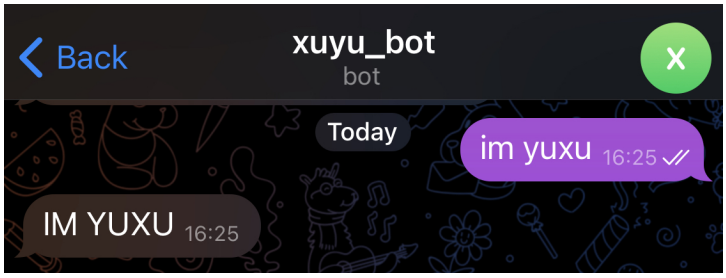
sure you place the file together with `config.ini` in the same directory.

```
## this file is based on version 13.7 of python telegram chatbot and version 1.26.18 of u
## chatbot.py
import telegram
from telegram.ext import Updater, MessageHandler, Filters
# The messageHandler is used for all message updates
import configparser
import logging

def main():
    # Load your token and create an Updater for your Bot
    config = configparser.ConfigParser()
    config.read('config.ini')
    updater = Updater(token=(config['TELEGRAM']['ACCESS_TOKEN']), use_context=True)
    dispatcher = updater.dispatcher
    # You can set this logging module, so you will know when and why things do not work a
    logging.basicConfig(format='%(asctime)s - %(name)s - %(levelname)s - %(message)s', le
    # register a dispatcher to handle message: here we register an echo dispatcher
    echo_handler = MessageHandler(Filters.text & (~Filters.command), echo)
    dispatcher.add_handler(echo_handler)
    # To start the bot:
    updater.start_polling()
    updater.idle()

def echo(update, context):
    reply_message = update.message.text.upper()
    logging.info("Update: " + str(update))
    logging.info("context: " + str(context))
    context.bot.send_message(chat_id=update.effective_chat.id, text= reply_message)

if __name__ == '__main__':
    main()
```

Run the following command on the terminal to start the chatbot:

```
python chatbot.py
```

Now you can send a message to your chatbot in Telegram in your phone, and receive the echo messages.



You can also check the log on your screen when you send messages the chatbot.

```
2024-01-28 16:25:33,847 - root - INFO - context: <telegram.ext.callbackcontext.CallbackContext object at 0x1020acb50>
2024-01-28 16:25:59,656 - root - INFO - Update: {'update_id': 502089893, 'message': {'message_id': 7, 'chat': {'id': 1810
356378, 'type': 'private', 'first_name': 'Allan'}, 'text': 'im yuxu', 'channel_chat_created': False, 'delete_chat_photo':
 False, 'caption_entities': [], 'new_chat_photo': [], 'new_chat_members': [], 'date': 1706430359, 'group_chat_created': F
alse, 'entities': [], 'photo': [], 'supergroup_chat_created': False, 'from': {'id': 1810356378, 'language_code': 'zh-hans
', 'first_name': 'Allan', 'is_bot': False}}}
2024-01-28 16:25:59,657 - root - INFO - context: <telegram.ext.callbackcontext.CallbackContext object at 0x1020ac950>
```

The source code of Python Telegram module: https://github.com/python-telegram-bot/python-telegram-bot;
The examples of Python Telegram module: https://github.com/python-telegram-bot/python-telegram-bot/tree/master/examples

Push your code to Github

This is the end of Lab3. Please push your code to Github. No write up for today's lab.