National University of Singapore

TIC4302 – Email Bot [AI]

Name	**Matric Number**	**NUS Email**
Desmond Yew Wen Yang	A0177731Y	e0260237@u.nus.edu
Png Cheng Xiang (Lucas)	A0177741X	e0260247@u.nus.edu
Tanujah Swaminathan	A0177714X	e0260220@u.nus.edu

Contents

Introduction

This bot is an automated marketing tool that will be beneficial for companies which leverage on web activities to interact with their customers. It mimics that of a conversation AI assistant, which helps to provide automated response to customers queries through our pre-programmed and structured queries and responses. Ultimately automating the delivery process.

This email bot is intricately designed to work for telecommunication organisations to enhance and automate the correspondence between the organisation and their customers. Email bots are becoming increasingly common within multiple companies due to its multiple benefits that ends up saving cost as well as time productivity within its processes.

3 of the most common benefits for the usage of Email Bots include:

- Costs reduction within the organisation with reduced manpower allocated to answering email queries from customers
- Increased Time efficiency & Productivity with email automation as it ends up saving a great deal of time within the organisation as the bot will be able to handle repetitive queries which would have otherwise been time intensive for the organisation
- Increased Brand Awareness with clear concise email responses as email automation provides a consistent response to the customers, therefore constructing an increasingly organised appearance to its customers

Technical Design

The purpose of an email bot is to automate the process of responding to various customer enquiries using the existing organisations email service along with the brand, by using contextual analysis. In the case of a telecommunications company, the email bot would be able to handle queries in relation to the various services, contract plans as well as other telecommunication products that are provided by the company with customised email response.

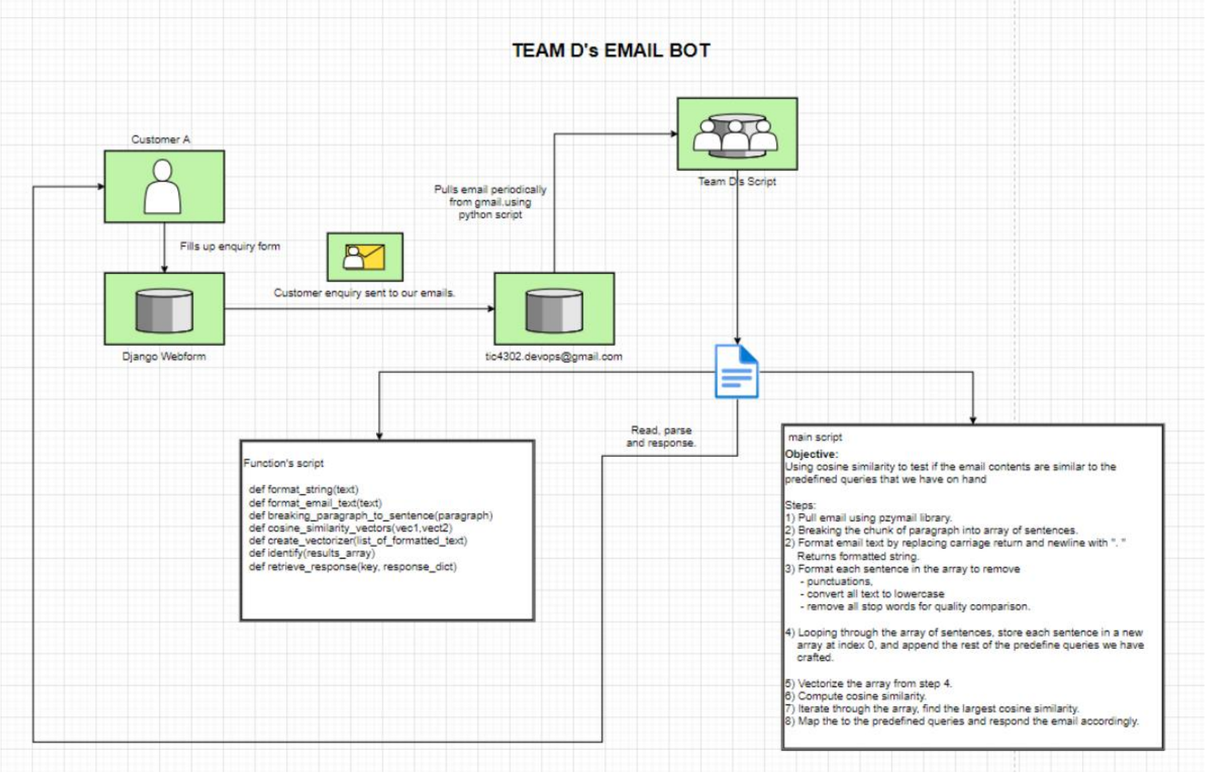
Prior to the communication with the email bot, a webform will be used as the platform to bridge the correspondence between the customers and the users. The webform would provide the customers the ability to input their enquiries in related to the contract plans, services as well as products provided by the telecommunication organisation.

Once submitting their enquiries, the email bot will be able to parse in the content based on the subject header as well as the body of the email for contextual analysis. Based on the contextual analysis done by the email bot, the response will be constructed in a customised fashion to respond to the queries posed by the customer.

The Email Bot makes use of the concept of cosine similarity, which is a metric used to measure the similarities between two documents irrespective of their size. In this context, cosine similarities are used to test if the contents within the email are similar to the predefined queries that have been pre-built within the system.

Once an email query has been received, it would first be pulled by the pzymail library. The pulled email's contents will then be broken down further into arrays of sentences instead of a single paragraph. The email text will then be formatted by replacing carriage returns and new lines with ".", which will return the formatted string. Each sentence of the array will be further formatted by removing punctuations, standardising texts to lower casing and removing all stop words for quality comparison. Upon standardisation, each sentence within the array will be stored at a new index 0, following that the pre-defined queries will be appended in accordingly.

The array will then be vectorized and the cosine similarity will be computed. The array will then be iterated through to identify the largest cosine similarity, mapping the pre-defined queries providing the bot the ability to respond accordingly.


TEAM D's EMAIL BOT
Customer A
Fills up enquiry form
Django Webform
Customer enquiry sent to our emails.
tic4302.devops@gmail.com
Pulls email periodically from gmail.using python script
Team D's Script
Read, parse and response.

Function's script

```
def format_string(text)
def format_email_text(text)
def breaking_paragraph_to_sentence(paragraph)
def cosine_similarity_vectors(vec1,vect2)
def create_vectorizer(list_of_formatted_text)
def identify(results_array)
def retrieve_response(key, response_dict)
```

main script

Objective:
Using cosine similarity to test if the email contents are similar to the predefined queries that we have on hand

Steps:
1) Pull email using pzymail library.
2) Breaking the chunk of paragraph into array of sentences.
2) Format email text by replacing carriage return and newline with ". "
Returns formatted string.
3) Format each sentence in the array to remove
- punctuations,
- convert all text to lowercase
- remove all stop words for quality comparison.

4) Looping through the array of sentences, store each sentence in a new array at index 0, and append the rest of the predefine queries we have crafted.

5) Vectorize the array from step 4.
6) Compute cosine similarity.
7) Iterate through the array, find the largest cosine similarity.
8) Map the to the predefined queries and respond the email accordingly.

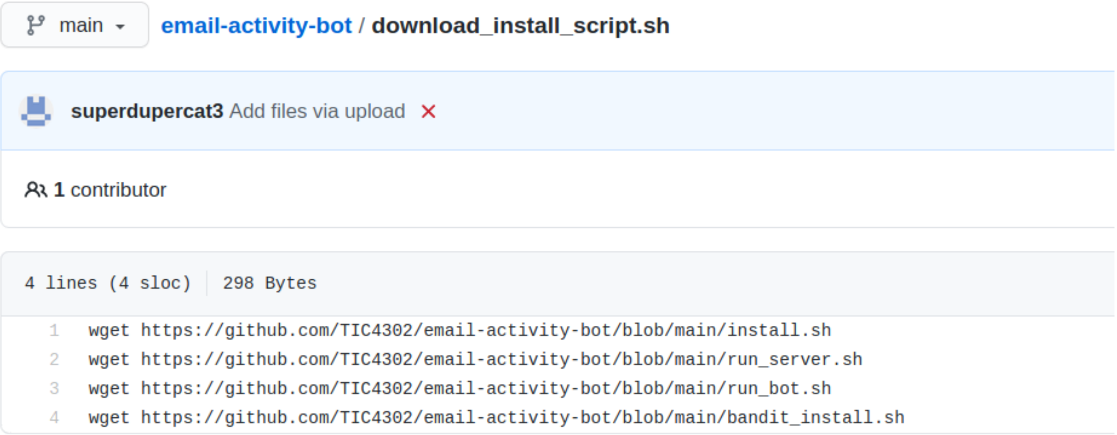
Pre-Requisite

- Python3.6.9
- Git (To clone the repository, alternatively you can download the zip folder and extract it out)
- Linux OS (optional) (For windows user, you may download the shell script and run the commands manually)

Setup Guide

Installation Script

The installation script would need to be downloaded using the link as prescribed: Installation Script



email-activity-bot / download_install_script.sh

superdupercat3 Add files via upload ✕

1 contributor

4 lines (4 sloc) | 298 Bytes

```
wget https://github.com/TIC4302/email-activity-bot/blob/main/install.sh
wget https://github.com/TIC4302/email-activity-bot/blob/main/run_server.sh
wget https://github.com/TIC4302/email-activity-bot/blob/main/run_bot.sh
wget https://github.com/TIC4302/email-activity-bot/blob/main/bandit_install.sh
```

Navigate to your desired directory in command and pull the script from git hub.

Command:

```
$ Wget https://github.com/TIC4302/email-activity-bot/blob/main/download_install_script.sh
```

Upon download, modify permission on the script to enable execution.

Command:

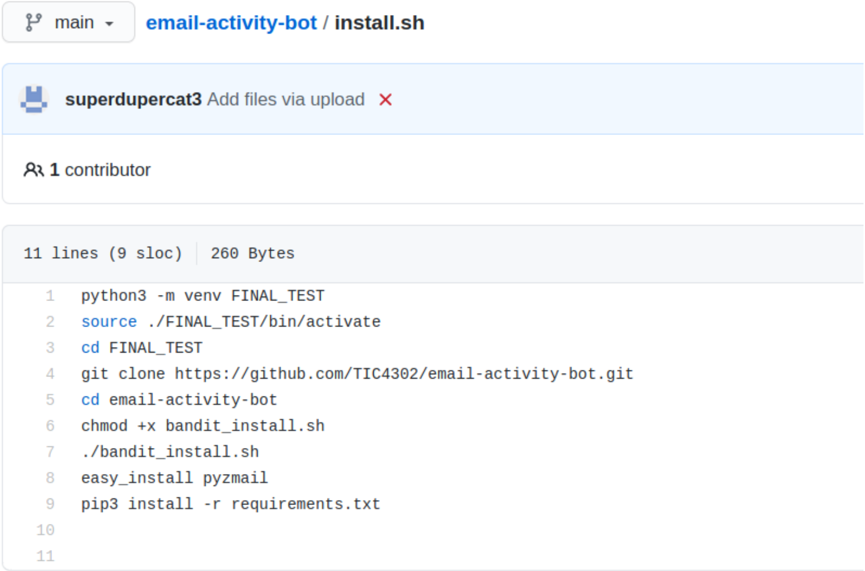
```
$ chmod +x ./download_install_script.sh
```

```
$ ./download_install_script.sh
```

Modify the permission and run the install script, which will create the virtual environment, run installation of bandit, python source code verifier, as well as all additional dependencies.

Command:

```
$ Chmod +x ./install.sh
$ ./install.sh
```

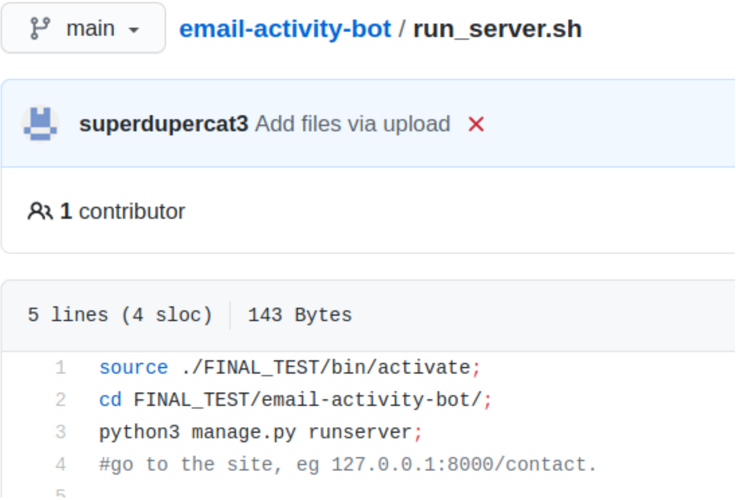

main
email-activity-bot / install.sh
superdupercat3 Add files via upload
1 contributor
11 lines (9 sloc) | 260 Bytes

```
python3 -m venv FINAL_TEST
source ./FINAL_TEST/bin/activate
cd FINAL_TEST
git clone https://github.com/TIC4302/email-activity-bot.git
cd email-activity-bot
chmod +x bandit_install.sh
./bandit_install.sh
easy_install pyzmail
pip3 install -r requirements.txt


```

Modify permission and run the server file to enable Django Webserver.

Command:

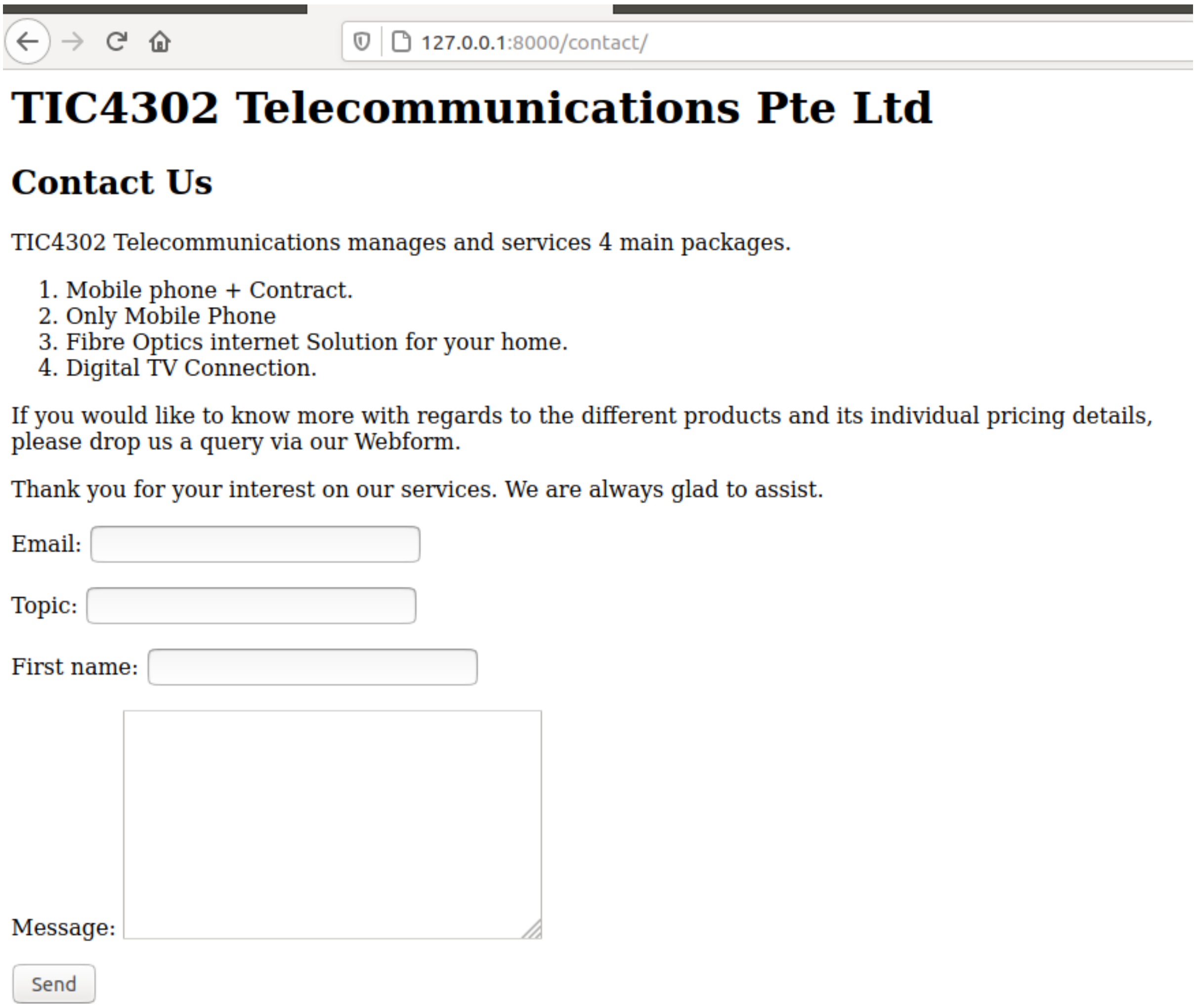
```
$ Chmod+x run_server.sh

$ ./run_server.sh
```


main
email-activity-bot / run_server.sh
superdupercat3 Add files via upload
1 contributor
5 lines (4 sloc) | 143 Bytes

```
source ./FINAL_TEST/bin/activate;
cd FINAL_TEST/email-activity-bot/;
python3 manage.py runserver;
#go to the site, eg 127.0.0.1:8000/contact.
```

Test Webform

To confirm if the installation was successful and that the webform is indeed up and running, navigate to: 127.0.0.1:8000/contact. Once successfully up, the webform should appear as below.



TIC4302 Telecommunications Pte Ltd

Contact Us

TIC4302 Telecommunications manages and services 4 main packages.

1. Mobile phone + Contract.
2. Only Mobile Phone
3. Fibre Optics internet Solution for your home.
4. Digital TV Connection.

If you would like to know more with regards to the different products and its individual pricing details, please drop us a query via our Webform.

Thank you for your interest on our services. We are always glad to assist.

Email:

Topic:

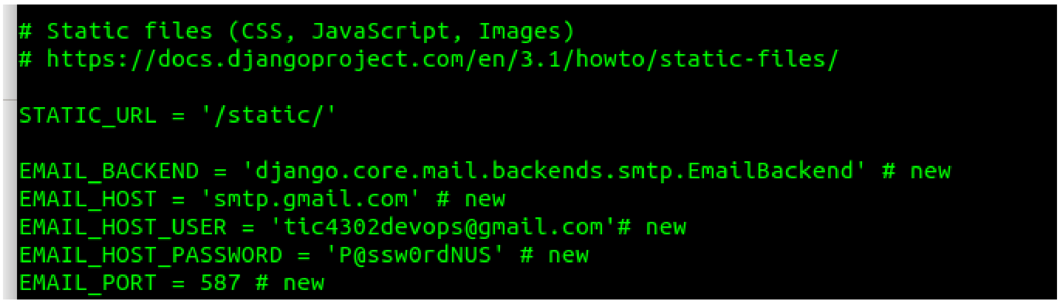
First name:

Message:

Modification of Scripts

1. Modify Email Credentials

To modify credentials of the email account, navigate to /FINAL_TEST/email-activity-bot/config/settings.py, and modify credentials under the variable, EMAIL_HOST_USER and EMAIL_HOST_PASSWORD.



```
# Static files (CSS, JavaScript, Images)
# https://docs.djangoproject.com/en/3.1/howto/static-files/

STATIC_URL = '/static/'

EMAIL_BACKEND = 'django.core.mail.backends.smtp.EmailBackend' # new
EMAIL_HOST = 'smtp.gmail.com' # new
EMAIL_HOST_USER = 'tic4302devops@gmail.com'# new
EMAIL_HOST_PASSWORD = 'P@ssw0rdNUS' # new
EMAIL_PORT = 587 # new
```

2. Modify Web Form

To modify the web form as required, navigate down to the following path, /FINAL_TEST/email-activity-bot/sendemail/view.py

3. Modify Form Structure

To modify the structure of the form, navigate down to the following path, /FINAL_TEST/email-activity-bot/sendemail/forms.py

Verify Source Code Vulnerability

To check for vulnerabilities within the source code, the bandit tool can be used. The bandit tool was downloaded along with the other dependencies within the installation script.

- Source (your saved directory)/FINAL_TEST/bin/activate
- Cd (your saved directory)/FINAL_TEST/email-activity-bot/bandit -r (file to test)
- (eg) :~/Desktop/TIC4302_Infosecurity_practicum/testing/FINAL_TEST/email-activity-bot$ bandit -r read_parse.py