IJITEB
Vol. 1
No. 2
2019

International Journal of Information Technology and Business

http://ejournal.uksw.edu/ijiteb

Integrating Real-Time Weather Forecasts Data Using OpenWeatherMap and Twitter

Christine Dewi[a], Rung-Ching Chen[b*]

[a] Departement of Information Management, Chaoyang University of Technology Taichung, Taiwan
[b] Departement of Information Management, Chaoyang University of Technology Taichung, Taiwan

Abstract : Weather forecasts are made by collecting as much data as possible about the current state of the atmosphere (particularly the temperature, humidity, and wind) and using an understanding of atmospheric processes (through meteorology) to determine how the atmosphere evolves in the future. There are several reasons why weather forecasts are important. It forewarns the people about future weather conditions so that people can plan their activities accordingly. It warns people about the impending severe weather conditions and other weather hazards such as thunderstorms, hurricanes, and heavy rainfalls. Thus far, accurate weather predictions have been able to save the lives of many. At its core, Twitter is a real-time public broadcast channel. These characteristics make Twitter a natural platform for public safety communication and early-warning systems. Furthermore, Twitter became an essential source for up-to-date meteorological data and agency announcements. OpenWeatherMap processes all data in a way that it attempts to provide accurate online weather forecast data and weather maps, such as those for clouds or precipitation. Besides, we will use Phyton programming language to get real-time weather data from OpenWeatherMap and post the information to our social media Twitter. Finally, OAuth and Tweepy are a very powerful library that enables the Python code to communicate with Twitter. Tweets about the weather could prove useful to anybody wanting to use it.

Keywords :
Tweepy, Phyton, Open Weather Map.

1. Introduction

Twitter can be identified as one of the most significant social networking sites. A large number of users have accepted Twitter as a universal platform for spreading news, sharing articles and socializing with other people globally. Twitter is a popular online social network and microblogging service for exchanging messages (also known as tweets) among people, supported by a huge ecosystem. Twitter announces that it has over 140 million active users creating more than 340 million messages every day [1] and over one million registered applications built by more than 750,000 developers [2]. In general, users tweet about events which are related to them, and there is a wealth of information that can be mined from this continuous stream of data. Though, at first glance, the twitter stream is unreliable, unorganized and uncontrolled, it is still possible to obtain near-real-time information about events and their locations given careful processing. One category of events that are often tweeted about by Twitter users in the category relating to their current weather conditions. Users can spread important weather information for others. Moreover, the content of these tweets ranges from simple tweets like ”its cloudy," to complete weather measurements such as temperature, humidity or wind speed. Given that some aspects of the weather are still better observed by human eyes and that social weather observations are decreasing as manned weather stations are replaced by automatic weather stations [3], tweets about the weather could prove useful. The

[*] Corresponding authors
e-mail addresses : s10714904@cyut.edu.tw, crching@cyut.edu.tw

weather data collected every day can have a range of new benefits if it is made open and interpreted in the right way. A surprisingly vast amount of weather data is freely available online to anybody wanting to use it. However, for the most part, it is not being utilized to its full potential. Accurate weather predictions are essential for planning our day-to-day activities. Farmers need information to help them plan for the planting and harvesting of their crops. Airlines need to know about local weather conditions to schedule flights. Weather forecasting helps us to make more informed daily decisions, and may even help keep us out of danger. For instance, weather forecasting helps people prepare for how to dress (i.e. warm weather, cold weather, windy weather, rainy weather), helps people plan outdoor activities (i.e., to see if rain/storms/cold weather will impact outdoor event) and helps businesses prepare for transportation hazards that can result from the weather (i.e., fog, snow, ice, storms, clouds as it relates to driving and flying for example). There are many online service website that provides weather data, including current weather data, forecasts, and historical data to the developers of web services and mobile applications. In this paper, we will use OpenWeatherMap to get real-time weather data. For data sources, it utilizes meteorological broadcast services, raw data from airport weather stations, raw data from radar stations, and raw data from other official weather stations. OpenWeatherMap processes all data in a way that it attempts to provide accurate online weather forecast data and weather maps, such as those for clouds or precipitation. In addition, we will use the Python programming language to get real-time weather data from OpenWeatherMap and post the information to our social media Twitter.

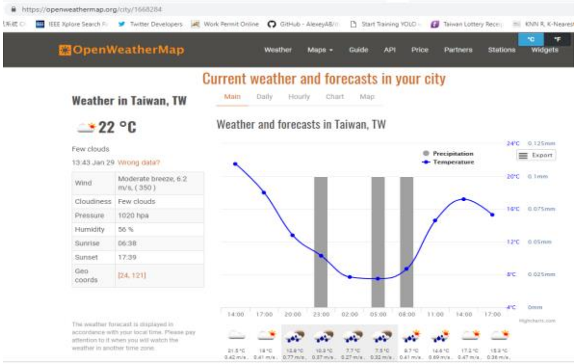
We begin with a brief discussion of related work in section 2. In section 3, we describe our proposed method for improving automated weather observations with human weather observations mined from the OpenWeatherMap and Twitter. Section 4 describes the experiment and software implementation used to evaluate the proposed method. Section 5 discusses the conclusion and brief consideration of future work.

2. Related Work

More and more researchers have begun to identify the significance of mining Twitter data and are increasingly dedicating their time and efforts to explore this field further. Experiments in [4] suggest that Twitter provides a suitable open publish-subscribe infrastructure for using sensors and smartphones. A crowd-sourcing system architecture over Twitter was designed, and two applications were developed (crowd-sourced weather radar and noise mapping application) as a means of evaluation. The architecture is comprised of two primary components, sense tweet and ask tweet. The first component is a mechanism by which to standardize the publishing of sensor observations over Twitter and the second a means to query ”crowds” for information. Twitter has also been used in the development of exciting and novel applications such as teaching English [5] and mobile learning [6]. Twitter has even been used to detect and notify registered users of earthquakes in Japan by applying Twitter to a probabilistic spatiotemporal model for events that can find the center and trajectory of event location [7]. In this paper, we will get real-time weather data from OpenWeatherMap and publish to our social media twitter using the Python programming language. This weather information could be used for other users who read this information and pay more attention to the weather in the specifics area.

OpenWeatherMap is an online service that provides weather data, including current weather data, forecasts, and historical data to the developers of web services and mobile applications. For data sources, it utilizes meteorological broadcast services, raw data from airport weather stations, raw data from radar stations, and raw data from other official weather stations. OpenWeatherMap processes all data in a way that it attempts to provide accurate online weather forecast data and weather maps, such as those for clouds or precipitation. Beyond that, the service is focused on the social aspect by involving weather station owners in connecting to the service and thereby increasing weather data accuracy. The ideology is inspired by OpenStreetMap and Wikipedia that make information free and available for everybody. It uses OpenStreetMap for display of weather maps.

OpenWeatherMap provides an API with JSON, XML and HTML endpoints and a limited free usage tier. Making more than 60 calls per minute requires a paid subscription starting at USD 40 per month. Access to historical data requires a subscription starting at US$150 per month. Users can request current weather information, extended forecasts and graphical maps (showing cloud cover, wind speed, pressure, and precipitation). We can get the API from https://openweathermap.org/city/1668284, which could be seen in figure 1.



Weather in Taiwan, TW

22 °C

Few clouds

13:43 Jan 29 Wrong data?

Wind	Moderate breeze, 6.2 m/s, (350)
Cloudiness	Few clouds
Pressure	1020 hpa
Humidity	56 %
Sunrise	06:38
Sunset	17:39
Geo coords	[24, 121]

The weather forecast is displayed in accordance with your local time. Please pay attention to it when you will watch the weather in another time zone.

Fig. 1. Open weather map website

3. Proposed Approach

To improvise upon the work of previous researchers in this field, Twitter makes use of a much simpler technology stack. Since the experimental set-up primarily involves an easy combination of Python programming and readily-available JavaScript libraries, any layman with merely little programming knowledge would be able to deploy and use it in his own space without any technical help. The architecture of this research could be seen in Figure 2.

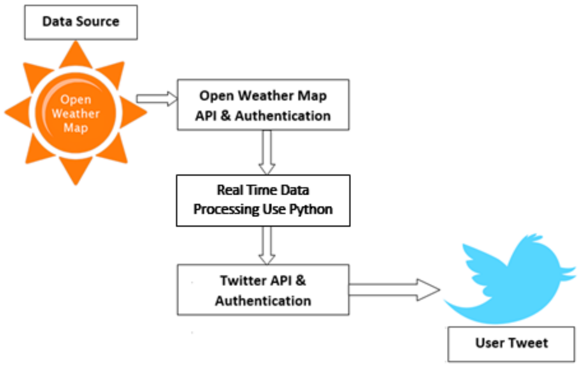
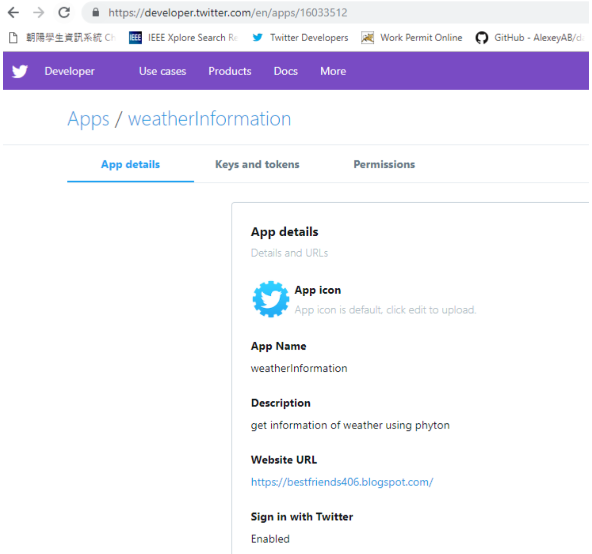

Data Source
Open Weather Map
Open Weather Map API & Authentication
Real Time Data Processing Use Python
Twitter API & Authentication
User Tweet

Fig. 2. Architecture system

Figure 2 describes the architecture of this research. First, the Data source gets from the open weather map website. With this website, we can get API and authentication to get real-time weather data from any country and city in the world. The next step, we use the weather data API and process the data with Python programming language. We can get Twitter API from https://developer.twitter.com/en/apps, which could be seen in Figure 3.



App details

Details and URLs

App icon

App icon is default, click edit to upload.

App Name

weatherInformation

Description

get information of weather using phyton

Website URL

https://bestfriends406.blogspot.com/

Sign in with Twitter

Enabled

Fig. 3. Twitter Developer API

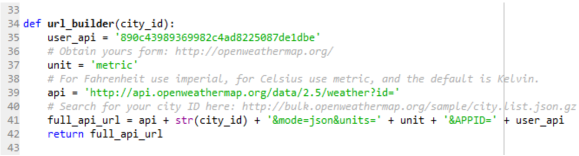
Furthermore, Twitter Authentication is the next step of the implementation process. It is done with the help of OAuth [8] and Tweepy [9]. OAuth is an open protocol that permits secure authorization by allowing the users to approve their application to act on their behalf without sharing their password. Tweepy, on the other hand, is an open-source library that enables the Python code to communicate with Twitter and use its API. It provides programmatic access to those Twitter Applications that are using OAuth and allows them to read Twitter data.

4. Experiment Result and Discussion

All implementations are carried out on Windows 10 having hardware configuration of Intel Core i5 processor with 8GB internal RAM and 1.60GHz of CPU speed.

4.1. OpenWeatherMap API & Authentication

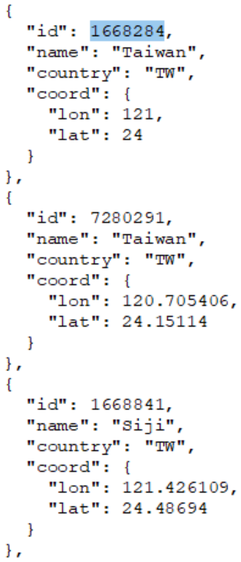
The first step of extracting datasets from OpenWeatherMap is to create API from https://home.openweathermap.org/. After the application is built, you can get user API. This API can put in the python source code to get the real-time weather data.



```
33
34 def url_builder(city_id):
35     user_api = '890c43989369982c4ad8225087de1dbe'
36     # Obtain yours form: http://openweathermap.org/
37     unit = 'metric'
38     # For Fahrenheit use imperial, for Celsius use metric, and the default is Kelvin.
39     api = 'http://api.openweathermap.org/data/2.5/weather?id='
40     # Search for your city ID here: http://bulk.openweathermap.org/sample/city.list.json.gz
41     full_api_url = api + str(city_id) + '&mode=json&units=' + unit + '&APPID=' + user_api
42     return full_api_url
43
```

Fig. 4. Snapshot of Python source code open weather map API

Figure 4 shows on line 35, we put the user API that we get from https://home.openweathermap.org/. Moreover, the list of the city in JSON format could be seen in Figure 5.

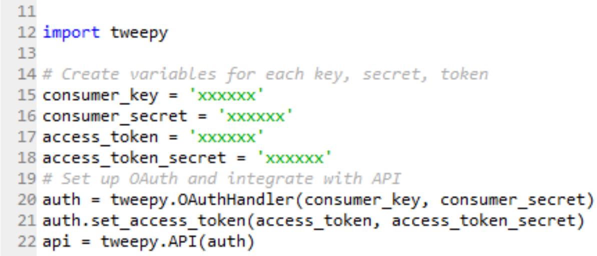


```
{
  "id": 1668284,
  "name": "Taiwan",
  "country": "TW",
  "coord": {
    "lon": 121,
    "lat": 24
  }
},
{
  "id": 7280291,
  "name": "Taiwan",
  "country": "TW",
  "coord": {
    "lon": 120.705406,
    "lat": 24.15114
  }
},
{
  "id": 1668841,
  "name": "Siji",
  "country": "TW",
  "coord": {
    "lon": 121.426109,
    "lat": 24.48694
  }
},
```

Fig. 5. Snapshot of city.list.json

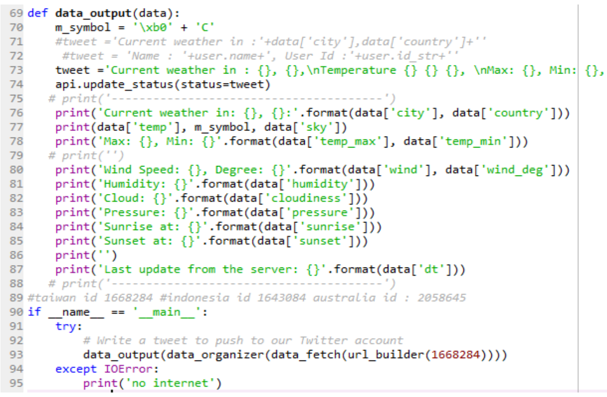
4.2. Twitter API & Authentication

The first step of extracting datasets from Twitter is to create a Twitter Application on developer.twitter.com. Once the application is created, four important tokens, namely, Consumer Key, Consumer Secret, Access Token and Access Token Secret are generated. These tokens are then passed for OAuth authentication. The implementation of the experiment could be seen in Figure 6. For example, we can put all the important tokens in Figure 6, line 15-18.



```
11
12 import tweepy
13
14 # Create variables for each key, secret, token
15 consumer_key = 'xxxxxx'
16 consumer_secret = 'xxxxxx'
17 access_token = 'xxxxxx'
18 access_token_secret = 'xxxxxx'
19 # Set up OAuth and integrate with API
20 auth = tweepy.OAuthHandler(consumer_key, consumer_secret)
21 auth.set_access_token(access_token, access_token_secret)
22 api = tweepy.API(auth)
```

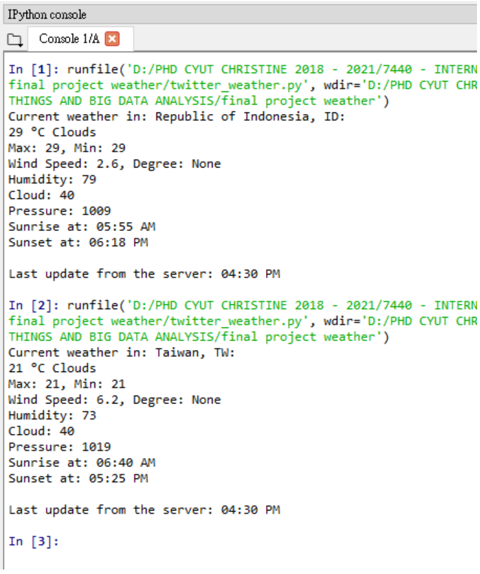
Fig. 6. Snapshot of Python source code Twitter API



```
69 def data_output(data):
70     m_symbol = '\xb0' + 'C'
71     #tweet ='Current weather in :'+data['city'],data['country']+''
72      #tweet = 'Name : '+user.name+', User Id :'+user.id_str+''
73     tweet ='Current weather in : {}, {},\nTemperature {} {} {}, \nMax: {}, Min: {},
74     api.update_status(status=tweet)
75    # print('-------------------------------------')
76     print('Current weather in: {}, {}:'.format(data['city'], data['country']))
77     print(data['temp'], m_symbol, data['sky'])
78     print('Max: {}, Min: {}'.format(data['temp_max'], data['temp_min']))
79    # print('')
80     print('Wind Speed: {}, Degree: {}'.format(data['wind'], data['wind_deg']))
81     print('Humidity: {}'.format(data['humidity']))
82     print('Cloud: {}'.format(data['cloudiness']))
83     print('Pressure: {}'.format(data['pressure']))
84     print('Sunrise at: {}'.format(data['sunrise']))
85     print('Sunset at: {}'.format(data['sunset']))
86     print('')
87     print('Last update from the server: {}'.format(data['dt']))
88    # print('-------------------------------------')
89 #taiwan id 1668284 #indonesia id 1643084 australia id : 2058645
90 if __name__ == '__main__':
91     try:
92         # Write a tweet to push to our Twitter account
93         data_output(data_organizer(data_fetch(url_builder(1668284))))
94     except IOError:
95         print('no internet')
```

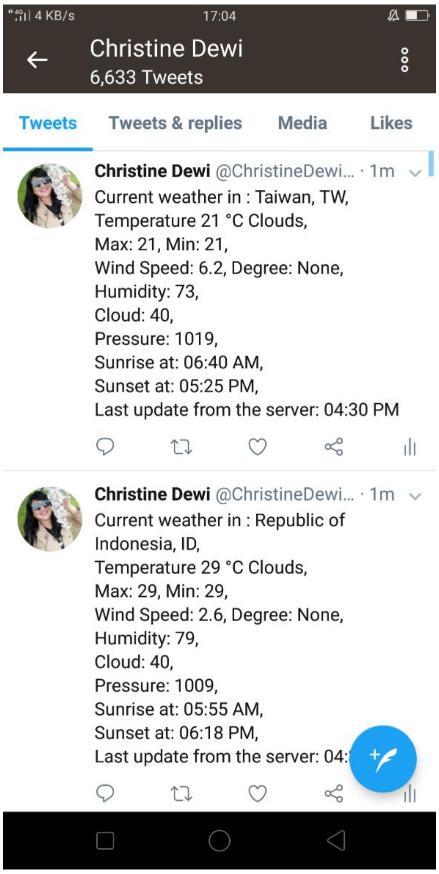
Fig. 7. Snapshot of Python output function

Figure 7 explains about data output function; this function used to get all weather data such as temperature, humidity, sunset time, pressure, sunrise time, cloud, and wind speed. Moreover, this data based on the id of the country that put in line 93. For instance, we can see on line 93 id 1668284 is the id of Taiwan country. The output of this complete source code could be seen in Figure 8 and Figure 9.



```
In [1]: runfile('D:/PHD CYUT CHRISTINE 2018 - 2021/7440 - INTERN
final project weather/twitter_weather.py', wdir='D:/PHD CYUT CHR
THINGS AND BIG DATA ANALYSIS/final project weather')
Current weather in: Republic of Indonesia, ID:
29 °C Clouds
Max: 29, Min: 29
Wind Speed: 2.6, Degree: None
Humidity: 79
Cloud: 40
Pressure: 1009
Sunrise at: 05:55 AM
Sunset at: 06:18 PM

Last update from the server: 04:30 PM

In [2]: runfile('D:/PHD CYUT CHRISTINE 2018 - 2021/7440 - INTERN
final project weather/twitter_weather.py', wdir='D:/PHD CYUT CHR
THINGS AND BIG DATA ANALYSIS/final project weather')
Current weather in: Taiwan, TW:
21 °C Clouds
Max: 21, Min: 21
Wind Speed: 6.2, Degree: None
Humidity: 73
Cloud: 40
Pressure: 1019
Sunrise at: 06:40 AM
Sunset at: 05:25 PM

Last update from the server: 04:30 PM

In [3]:
```

Fig. 8. Weather Data Output



Christine Dewi @ChristineDewi... · 1m

Current weather in : Taiwan, TW,
Temperature 21 °C Clouds,
Max: 21, Min: 21,
Wind Speed: 6.2, Degree: None,
Humidity: 73,
Cloud: 40,
Pressure: 1019,
Sunrise at: 06:40 AM,
Sunset at: 05:25 PM,
Last update from the server: 04:30 PM

Christine Dewi @ChristineDewi... · 1m

Current weather in : Republic of Indonesia, ID,
Temperature 29 °C Clouds,
Max: 29, Min: 29,
Wind Speed: 2.6, Degree: None,
Humidity: 79,
Cloud: 40,
Pressure: 1009,
Sunrise at: 05:55 AM,
Sunset at: 06:18 PM,
Last update from the server: 04:

Fig. 9. Weather Data Output on Twitter

5. Conclusion and Future Work

In this paper, complete real-time weather data get from the OpenWeatherMap website. This weather data will publish to twitter social media, which implemented using simple programming concepts of Python. Moreover, OAuth and Tweepy are a compelling library that enables the Python code to communicate with Twitter and use its API. In this experiment, we can get real-time weather data such as current weather in specific city and country, temperature (maximum and minimum), wind speed, humidity, cloud, pressure, sunrise time, and sunset time. Tweets about the weather could prove useful for anybody wanting to use it. Weather forecasting helps us to make more informed daily decisions, and may even help keep us out of danger. In the future, we will use an algorithm to predict and analyze the weather data and implement in the mobile application so people can use it easily.

6. References

[1] Twitter blog. (2011). One million registered twitter apps. [Online]. Available: http://blog.twitter.com/2011/07/ one-million-registered-twitter-app s.html

[2] Twitter blog. (2012). Shutting down spammers. [Online]. Available: http://blog.twitter.com/2012/04/ shutting-down-spammers.html

[3] C. A. Fiebrich, “History of surface weather observations in the united states,” Earth-Science Reviews, Jan 2009.

[4] M. Demirbas, M. A. Bayir, C. G. Akcora, Y. S. Yilmaz, and H. Ferhatosmanoglu, “Crowdsourced sensing and collaboration using twitter,” XIn 2010 IEEE International Symposium on ”A World of Wireless, Mobile and Multimedia Networks”, pp. 1–9, Jun 2010.

[5] G. Grosseck and C. Holotescu, "Microblogging multimedia-based teaching methods best practices with cirip. eu," Procedia-Social and Behavioral Sciences, no. 2, pp. 2151–2155, Jan 2010.

[6] M. Ebner and M. Schiefner, "Microblogging-more than fun," Proceedings of IADIS Mobile Learning Conference 2008, pp. 155–159, Jan 2008.

[7] T. Sakaki, M. Okazaki, and Y. Matsuo, “Earthquake shakes twitter users: real-time event detection by social sensors,” Proceedings of the 19th international conference on World Wide Web, Jan 2010.

[8] "OAuth Community Site," available at https://oauth.net/, visited in January 2019.

[9] "Tweepy," available at http://www.tweepy.org/, visited in January 2019.