

Práctica 3: Integración de tecnologías de Sistemas Distribuidos

Santiago Godoy Poce

José Correro Barquín

6 de junio de 2018



Índice

1. Introducción	3
2. Tecnologías utilizadas	3
2.1. RabbitMq	3
2.2. Redis	3
2.3. Celery	3
2.4. Twitter	3
2.5. Dropbox	3
2.6. Matplotlib	3
3. Flujo de trabajo	4
4. Ejecución	4
5. Capturas de pantalla	4
6. Referencias	6

1. Introducción

La aplicación consiste en recabar la información necesaria para realizar una comparativa gráfica entre el número de retweets de los últimos 100 tweets (o los que se le indique) de tantas cuentas como se indique. Esta información se registra en un fichero (que se sube a Dropbox) y a partir de éste, se genera una gráfica. Todo el proceso se ha programado para que se ejecute cada 10 segundos.

2. Tecnologías utilizadas

2.1. RabbitMq

Se usa junto a Celery, ya que éste necesita un bróker de mensajería para enviar y recibir mensajes.

2.2. Redis

Se utiliza como backend en Celery para guardar los estados de los procesos.

2.3. Celery

Necesitamos una cola asíncrona de paso de mensajes distribuidos a la que se le pasan los procesos y se comunica vía mensajes mediante un broker entre *workers* y clientes.

2.4. Twitter

Usamos el api de Twitter para python (*tweepy*) para obtener la información de los retweets de diferentes cuentas para almacenarla luego en un fichero.

2.5. Dropbox

El fichero que generamos gracias a la información que cogemos de Twitter, lo vamos a almacenar en la nube, en este caso, Dropbox, usando su api para python.

2.6. Matplotlib

El fichero, nos servirá como datos de entrada para generar una gráfica comparativa mediante el uso de la biblioteca Matplotlib para python.

3. Flujo de trabajo

El flujo de trabajo que hemos utilizado es el siguiente:

```
chord([getTweets.s(twitter) for twitter in twitters]) (chain(writeRetweetsInFile.s(), generarGrafica.s()))
```

En primer lugar, se extraen cada uno de los timelines de las cuentas de Twitter que le hemos indicado en paralelo con un **chord**. Una vez termina este proceso, se ejecuta **en cadena**, las tareas de escribir los datos en un fichero y subirlo a Dropbox con un delay (asincronía) y generar la gráfica correspondiente. Usamos las *signatures* para envolver los argumentos y pasarlos a lo largo del flujo.

4. Ejecución

Antes que nada, se inicia el servidor de rabbit y de redis con ***sudo rabbitmq-server*** y ***sudo redis-server***. Luego, para ejecutar las tareas de Celery, en el mismo directorio que el archivo tasks.py, se ha utilizado la siguiente orden: ***celery -A tasks worker -B -l=info***.

5. Capturas de pantalla

```
santi@santi-300E4A-300E5A-300E7A:~/Documentos/SD/TrendingWords_GRUPO_NoTeVayaMané/TrendingWords$ celery -A tasks worker -B -l=info

----- celery@santi-300E4A-300E5A-300E7A v4.1.1 (latentcall)
-----
***
* * * * * Linux-4.13.0-43-generic-x86_64-with-Ubuntu-17.10-artful 2018-06-06 21:35:21
* * * * *
** ----- [config]
** ----- .> app: tasks:0x7f1463658b70
** ----- .> transport: amqp://guest:**@localhost:5672//
** ----- .> results: redis://
** ----- .> concurrency: 4 (prefork)
** ----- .> task events: OFF (enable -E to monitor tasks in this worker)
-----
** ----- [queues]
** ----- .> celery exchange=celery(direct) key=celery

[tasks]
. tasks.generarGrafica
. tasks.getTweets
. tasks.process
. tasks.upload_dropbox
. tasks.writeRetweetsInFile


[2018-06-06 21:35:21,326: INFO/Beat] beat: Starting...
[2018-06-06 21:35:21,333: INFO/MainProcess] Connected to amqp://guest:**@127.0.0.1:5672//
[2018-06-06 21:35:21,348: INFO/MainProcess] mingle: searching for neighbors
[2018-06-06 21:35:21,538: INFO/Beat] Scheduler: Sending due task task3 (tasks.process)
[2018-06-06 21:35:22,388: INFO/MainProcess] mingle: all alone
[2018-06-06 21:35:22,428: INFO/MainProcess] celery@santi-300E4A-300E5A-300E7A ready.
[2018-06-06 21:35:22,429: INFO/MainProcess] Received task: tasks.process[853edadc-2d54-4ed9-bb34-b48aafdeab39]
[2018-06-06 21:35:22,568: INFO/MainProcess] Received task: tasks.getTweets[bdd6890c-fa79-453d-bfd5-c2fc2a7f13e9]
[2018-06-06 21:35:22,571: INFO/MainProcess] Received task: tasks.getTweets[627646c6-24e4-44ae-9209-f9616863cf39]
[2018-06-06 21:35:22,571: INFO/ForkPoolWorker-3] Task tasks.process[853edadc-2d54-4ed9-bb34-b48aafdeab39] succeeded in 0.037012747000005675s:
None
```

En esta primera captura se puede ver cómo Celery echa a correr con las tareas que

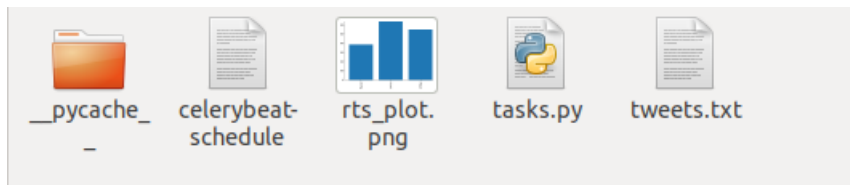
hemos definido en el flujo de trabajo, que se han programado cada 10 segundos en el proceso *task3*. Además se observa cómo se conecta a las direcciones que hemos definido de amqp y redis.

```
[2018-06-06 21:35:22,572: INFO/MainProcess] Received task: tasks.getTweets[183b47b3-487a-49a3-b086-c84cf2c957f2]
[2018-06-06 21:35:23,810: INFO/ForkPoolWorker-3] Task tasks.getTweets[627646c6-24e4-44ae-9209-f9616803cf39] succeeded in 1.237515712000004s: [
{'created_at': 'Wed Jun 06 19:00:21 +0000 2018', 'id': 1004437857619103745, 'id_str': '1004437857619103745', 'text': 'With 0a under siege by t
he Red Lantern Corps, it's up to the Titans, Booster Gold, Blue Beetle and Lobo to save the_ https://t.co/upEyoUrCGK', 'truncated': True, 'ent
itles': {'hashtags': [...], 'symbols': [...], 'user_mentions': [...], 'urls': [...], 'source': '<a href=http://www.spredfast.com rel=nofol
low>Spredfast app</a>', 'in_reply_to_status_id': None, 'in_reply_to_status_id_str': None, 'in_reply_to_user_id': None, 'in_reply_to_user_id_s
tr': None, 'in_reply_to_screen_name': None, 'user': {'id': 18173624, 'id_str': '18173624', 'name': 'DC', 'screen_name': 'DCComics', 'location'
: '', 'description': 'The official hone of Batman, Superman, Wonder Woman, Green Lantern, The Flash and the rest of The World's Greatest Super
Heroes!', 'url': 'https://t.co/BRS80ou04f', 'entities': {...}, 'protected': False, 'followers_count': 2803176, 'friends_count': 749, 'listed_
count': 12071, 'created_at': '...', ...}}]
[2018-06-06 21:35:24,095: INFO/ForkPoolWorker-4] Task tasks.getTweets[183b47b3-487a-49a3-b086-c84cf2c957f2] succeeded in 1.5214269790000117s:
[{'created_at': 'Wed Jun 06 17:27:08 +0000 2018', 'id': 1004414400571244544, 'id_str': '1004414400571244544', 'text': 'Empleza la SuperLiga Or
ange la mejor liga de España y del mundo con los dos mejores comentaristas de España y del m. https://t.co/5VEIH9Wk', 'truncated': True, 'en
titles': {'hashtags': [...], 'symbols': [...], 'user_mentions': [...], 'urls': [...], 'source': '<a href=http://twitter.com/download/iphone
rel=nofollow>Twitter for iPhone</a>', 'in_reply_to_status_id': None, 'in_reply_to_status_id_str': None, 'in_reply_to_user_id': None, 'in_re
ply_to_user_id_str': None, 'in_reply_to_screen_name': None, 'user': {'id': 2754746065, 'id_str': '2754746065', 'name': 'Ibañ', 'screen_name':
'LVPibai', 'location': 'Barcelona, España', 'description': 'Hablo durante muchas horas en la @LVPes y cuando termino me voy a mi casa a descan
sar con mi familia.', 'url': 'https://t.co/v2Pluo0Gf7', 'entities': {...}, 'protected': False, 'followers_count': 316764, 'friends_count': 413
, 'listed_count': 571, 'created_at': '...', ...}}]
[2018-06-06 21:35:24,345: INFO/ForkPoolWorker-2] Task tasks.getTweets[bdd6890c-fa79-453d-bfd5-c2fc2a7f13e9] succeeded in 1.7752075209999998s:
[{'created_at': 'Wed Jun 06 19:15:11 +0000 2018', 'id': 1004441591019270144, 'id_str': '1004441591019270144', 'text': 'Editor-in-chief @C8Cebu
tski and series editor @EDevinLewis take you inside #CloakAnd Dagger, a brand new digital exc. https://t.co/Ej8Ba6ZanJ', 'truncated': True, 'en
titles': {'hashtags': [...], 'symbols': [...], 'user_mentions': [...], 'urls': [...], 'source': '<a href=https://studio.twitter.com rel=no
follow>Media Studio</a>', 'in_reply_to_status_id': None, 'in_reply_to_status_id_str': None, 'in_reply_to_user_id': None, 'in_reply_to_user_id
_str': None, 'in_reply_to_screen_name': None, 'user': {'id': 15687962, 'id_str': '15687962', 'name': 'Marvel Entertainment', 'screen_name': 'M
arvel', 'location': 'New York, NY', 'description': 'The official Twitter for Marvel comics, movies, games, and more.', 'url': 'https://t.co/Id
c279fL7C', 'entities': {...}, 'protected': False, 'followers_count': 6539888, 'friends_count': 783, 'listed_count': 22340, 'created_at': 'Fri
Aug 01 11:26:10 +0000 2008', 'f...', ...}}]
[2018-06-06 21:35:24,348: INFO/MainProcess] Received task: tasks.writeRetweetsInFile[5a35cb54-c5a5-4f33-9691-0b71ac59f295]
[2018-06-06 21:35:24,378: INFO/ForkPoolWorker-3] Task tasks.writeRetweetsInFile[5a35cb54-c5a5-4f33-9691-0b71ac59f295] succeeded in 0.003326646
9999944093s: [19292, 32083, 27487]
[2018-06-06 21:35:24,377: INFO/MainProcess] Received task: tasks.upload_dropbox[89971e5e-c3ff-425b-a69a-d94adbeb65b0]
[2018-06-06 21:35:24,381: INFO/ForkPoolWorker-2] Request to files/upload
[2018-06-06 21:35:25,806: INFO/ForkPoolWorker-5] Task tasks.generarGrafica[89427eac-2d63-42ec-8d86-01b1829e2018] succeeded in 1.425764104999998
47s: None
[2018-06-06 21:35:25,867: INFO/ForkPoolWorker-2] Task tasks.upload_dropbox[89971e5e-c3ff-425b-a69a-d94adbeb65b0] succeeded in 1.486554976999999
```

En la segunda captura, se ve la ejecución de todas y cada una de las tareas defi-nidas en el proceso que se ha programado.

Dropbox > Aplicaciones > practica 3				 <input type="text" value="Buscar"/>
<input type="checkbox"/>	Nombre ↑	Última modificación ▼	Miembros ▼	⋮ ▼
	tweets.txt	hace 1 min	Solo tú	...

Aquí podemos observar el fichero ya subido a Dropbox en la carpeta reservada pa- ra la aplicación que hemos creado.



Así es como queda el directorio de trabajo tras ejecutar el proceso cada 10 segundos. Tenemos la gráfica creada y el fichero que se ha subido a Dropbox. La carpeta y el otro fichero pertenecen al propio Celery.

Para más detalle, aconsejamos que le eches un vistazo al código o que lo ejecutes con la orden que te hemos puesto.

6. Referencias

- Celery
<http://docs.celeryproject.org/en/master/index.html>
- Flujos de trabajo
<http://docs.celeryproject.org/en/latest/userguide/canvas.html>
- Tweepy
<http://docs.tweepy.org/en/v3.5.0/api.html>
- JSON de los Status de Tweepy
<https://gist.github.com/dev-techmoe/ef676cdd03ac47ac503e856282077bf2>
- Dropbox
<https://www.dropbox.com/developers-v1/core/start/python>
- Matplotlib
<https://matplotlib.org/api/index.html>