



Tweet Collector

Jaime L. Flores

jlopes@img.com.br

jaimelopesflores@gmail.com

Twitter: @_jaimelflores

Skype: jaimelopesflores

Cel: (12) 99114-3055

Sumário

Apresentação	2
Dependências	2
Configuração	2
config.xml	3
Twitter	4
Execução.....	6

Apresentação

Tweet Collector é um script que utiliza a [Twitter Streaming API](#) para espacializar os Tweets de acordo com um termo específico de busca.

Para cada termo um FILEGDB é criado com o nome do termo especificado.

Foi desenvolvido em Python 2.7 e utiliza recursos da biblioteca **ArcPy** fornecida pela suíte de aplicativos desktop do ArcGIS 10.2.

Utiliza a arquitetura multi-thread para uma otimização dos recursos de máquina.

O pacote do script Tweet Collector contém os seguintes itens listados abaixo. Todos os itens são obrigatórios para a execução do script.

- **oauth2 (pasta library)**
É uma biblioteca do Python necessária para a autenticação segura no Twitter via oauth.
- **twitter.gdb (file GDB)**
É o 'banco de dados' modelo do script. Dentro dele existe uma Feature Class Tweet que está completamente vazia. Cópias desse fileGDB serão realizadas para cada novo termo de pesquisa inserido.
Este banco nunca deve ser alterado, ou ter registros inseridos.
- **config.xml (arquivo.xml configuração)**
É o arquivo de configuração do script. Nele colocamos os termos que serão monitorados, e as chaves de autenticação de cada termo. Veremos ambos na seção Configuração.
- **tweet-collector (script)**
É o script propriamente dito, que contém toda a lógica de coleta e armazenamento dos Tweets. Após todas as configurações necessárias serem realizadas, é esse o arquivo que deve ser executado.

Dependências

Para execução correta é necessário ter instalado o Python 2.7, necessariamente o que vem com a instalação do ArcGIS Desktop 10.2.

Conter todos os arquivos descritos na apresentação na mesma pasta.

Configuração

O script é configurado através do arquivo **config.xml**, onde inserimos os termos de busca e as chaves de aplicação a token do Twitter. O token do Twitter é obtido através do site <http://dev.twitter.com> onde você precisa criar uma Twitter Application.

Tweet Collector

Para CADA tag **track** é necessário que seja criada uma Twitter Application, pois o script Tweet Collector funciona na arquitetura multi-thread, onde cada thread abre uma conexão de streaming com a API do Twitter. Por definição do Twitter não se pode abrir duas ou mais conexões de streaming utilizando as mesmas chaves de autenticação. Cada conexão deve ter suas chaves de autenticação, por isso a necessidade de se criar uma Twitter Application para cada termo de pesquisa.

config.xml

A tag **track** é a responsável por definir o termo de busca e as chaves para autenticação no Twitter.

A seguir vamos explicar o que cada atributo da tag **track** significa.

- **value**
É o termo de pesquisa. É aconselhado não começar com caracteres especiais e/ou números.
- **api-key e api-secret**
São as chaves de autenticação de API para cada Twitter Application
- **access-key e access-secret**
São as chaves de autenticação de Acesso para cada Twitter Application

Um exemplo do arquivo **config.xml** pode ser visto a seguir.

```
<tracks>
  <track
    value="iphone"
    api-key="VUA8dyEZ8dK..."
    api-secret="sontFQkxf..."
    access-key="551111126..."
    access-secret="GZfZqB..." />

  <track
    value="google"
    api-key="iAT2WwRUCIJf..."
    api-secret="khynvcBRL..."
    access-key="551111126..."
    access-secret="1WEs3T..." />

  <track
    value="dilma"
    api-key="W2J8MiN5yNfU..."
    api-secret="9p7j5Lmep..."
    access-key="551111126..."
    access-secret="KYEe9l..." />

</tracks>
```

As reticências não fazem parte das chaves

Twitter

As configurações necessárias no Twitter são, basicamente, criar uma *Twitter Application* onde a api-key e a api-secret são geradas na criação, e solicitar a criação imediata da access-key e access-secret para a plataforma do Twitter.

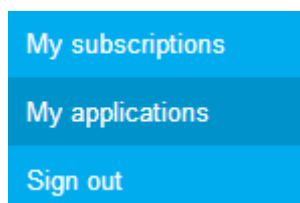
Para isso é necessário ter uma conta do Twitter.

1º passo

Acesse o site <http://dev.twitter.com> e realize o login com sua conta do Twitter clicando em **Sign In** no canto superior direito da tela.

2º passo

No mesmo local onde estava **Sign In** agora está a foto do seu perfil do Twitter. Posicione o mouse sobre sua foto sem clicar, vai aparecer um menu suspenso. Clique em **My Applications** para acessar a página **Application Management**.



Menu suspenso

3º passo

Nesta página você verá todas as suas Twitter Applications criadas anteriormente. Se for seu primeiro acesso sua página estará vazia. Clique no botão Create New App para criar sua nova Twitter Application.

Tweet Collector

Application Management

Twitter Apps

Create New App



collector-iphone

collector-iphone



collector-samsung

collector-samsung



collector-dilma

collector-dilma

Tela Application Management

4º passo

Preencha o campo Name com o nome que deseja dar para a sua Twitter Application, Description com a descrição e Website com o site da sua organização.

Após o preenchimento clique em **Create your Twitter application** no fim da página.

Obs: O campo Website tem por finalidade mapear de onde os Tweets criados por uma aplicação terceira estão vindo, porém no Tweet Collector não criamos Tweets, somente consumimos Tweets, por isso não necessitamos colocar o endereço da aplicação que fará uso das chaves de aplicação.

Create your Twitter application

Confirma a criação da Twitter Application

5º passo

Agora já podemos ver nossa aplicação criada na tela **Application Management**, basta somente verificar as chaves de autenticação.. Clique na sua aplicação, logo em seguida clique em **API Keys** que fica logo abaixo do nome da sua Twitter Application.

Details

Settings

API Keys

Permissions

Menus de gerenciamento da sua Twitter Application

Tweet Collector

Após aguardar o carregamento da tela, você poderá encontrar sua **api-key** e **api-secret** na seção **Application Settings**.

Logo abaixo existe um botão para solicitar a criação do Access Token. Clique no botão, aguarde alguns segundos e de refresh na página. A seção **Your Access Token** foi criada, e lá você pode obter a **access-key** e a **access-secret**.

Agora que já temos as chaves de uma Twitter Application, basta preencher corretamente o arquivo config.xml com o termo de busca, e as chaves correspondentes para o perfeito funcionamento do script.

Execução

Para execução do script, execute o arquivo **tweet-collector.py** com o comando "**python tweet-collector.py**" via prompt, ou clicando duas vezes caso sua instalação do Python esteja correta.

A seguinte tela será exibida após alguns segundos de inicialização do script

```

C:\Python27\ArcGIS10.2\python.exe
Processo: 12032
Buscando: iphone | Ok: 1229 | Nao OK: 238008 | Erro: 0

Processo: 2868
Buscando: google | Ok: 836 | Nao OK: 80905 | Erro: 0

Processo: 9992
Buscando: dilma | Ok: 230 | Nao OK: 11134 | Erro: 0
  
```

Console de execução do script

Vamos entender o que cada informação representa no console de execução.

Cada termo de pesquisa é separado por uma linha em branco, como podemos ver entre iphone e google, e google e dilma.

Para explicação pegaremos o processo 12032 – iphone.

```

Processo: 12032
Buscando: iphone | Ok: 1229 | Nao OK: 238008 | Erro: 0
  
```

- **Processo**

É o número do processo em que a thread foi iniciada. Note que cada bloco tem um número de processo.

- **Buscando**

É o termo de busca que foi definido no arquivo config.xml

- **Ok**

São os Tweets que tem coordenadas do Tweet ou o país/cidade de onde aquele Tweet foi postado. Somente Tweets que obedecem esses dois termos são armazenados.

No exemplo existem 1229 Tweets que satisfazem os requisitos de armazenamento, e eles podem ser encontrados no fileGDB **iphone.gdb/tweets**

- **Não OK**

São os Tweets que não obedecem os requisitos mínimos para armazenamento, ou seja, não tem coordenadas e nem país/cidade.

No exemplo existem 238008 Tweets que não satisfazem os requisitos de armazenamento, portanto, estes foram simplesmente desconsiderados pelo processo.

- **Erro**

São Tweets que deram algum problema na hora da gravação no fileGDB.

No exemplo felizmente não existem erros! ☺ Mas quando existem, estes não são armazenados de nenhuma forma, são simplesmente desconsiderados.

Obs:

O script “COMMITA” as inserções a cada duas inserções, visto que a frequência de Tweets que satisfazem os requisitos de inserção não é tão intensa.