

Indian Institute of Technology Madras

Twitter Data Streaming Using Kafka and Analysis Using Spark

Om Shri Prasath
EE17B113

1. Overview :

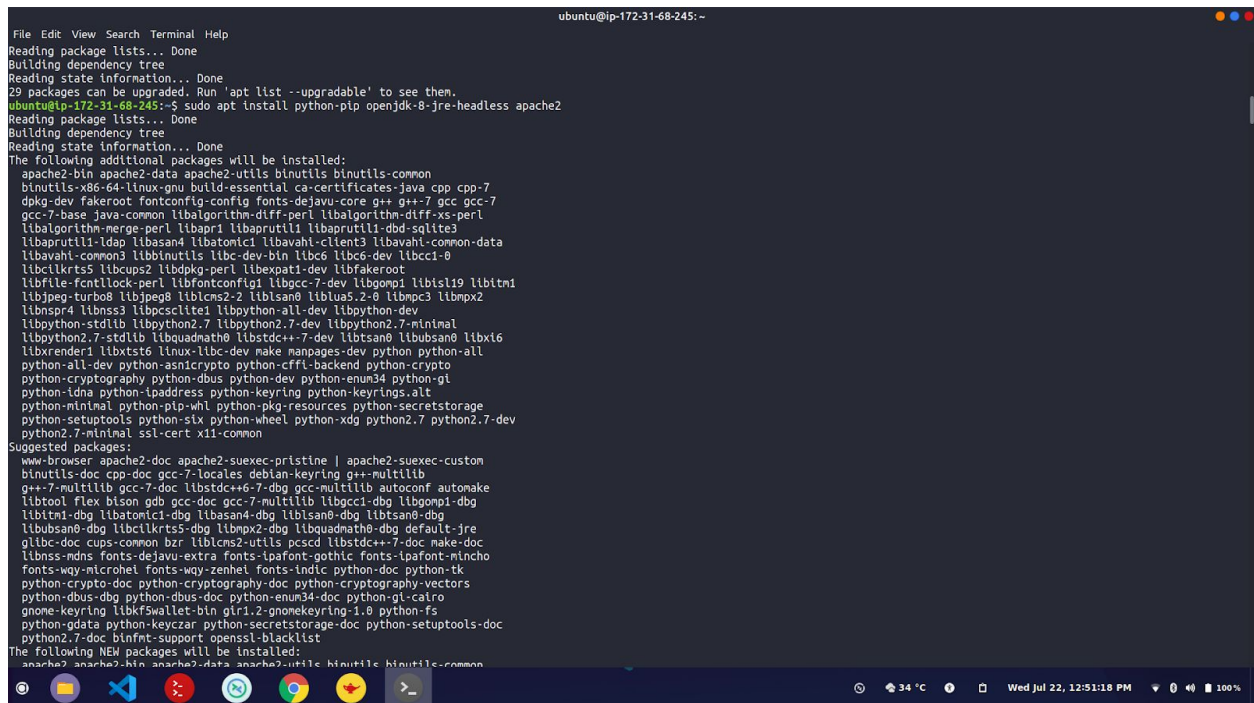
The assignment consists of analysing Twitter data by streaming the tweets using Kafka and analysing the tweets using Spark. The analysis consists of counting the number of tweets in a 10-minute interval and finding the frequently occurring word sets using FPGrowth Algorithm.

2. Approach :

The tweets are first read using the Twitter API credentials provided to us and using the python libraries `tweepy` (to download the tweets) and `kafka-python` (to publish the tweets to the respective Kafka topic). After this setup, the tweets are then consumed by a Spark application which then reads all the tweets in a 10-minute batch interval. These tweets are converted into an RDD and the count of the tweets is got by counting the number of rows in RDD and the FPGrowth algorithm is applied on the RDD to extract the frequent sets.

3. Implementation :

a) First we have to install java, pip and apache in the instance :



```
ubuntu@pip-172-31-68-245:~$ sudo apt install python-pip openjdk-8-jre-headless apache2
Reading package lists... Done
Building dependency tree
Reading state information... Done
29 packages can be upgraded. Run 'apt list --upgradable' to see them.
ubuntu@pip-172-31-68-245:~$ sudo apt install python-pip openjdk-8-jre-headless apache2
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils binutils binutils-common
  binutils-x86-64-linux-gnu build-essential ca-certificates-java cpp cpp-7
  dpkg-dev fakeroot fontconfig-config fonts-dejavu-core g++ g++-7 gcc gcc-7
  gcc-7-base java-common libalgorithm-diff-perl libalgorithm-diff-xs-perl
  libalgorithm-merge-perl libapr1 libaprutil1 libaprutil1-dbd-sqlite3
  libaprutil1-ldap libasan4 libatomic1 libavahi-client3 libavahi-common-data
  libavahi-common3 libbinutils libc-dev-bin libc6 libc6-dev libcc1-0
  libcilkrts5 libcups2 libdpkg-perl libexpat1-dev libfakeroot
  libfile-fcntllock-perl libfontconfig1 libgcc-7-dev libgomp1 libisl19 libitm1
  libjpeg-turbo8 libjpeg8 liblcms2-2 liblsan0 liblua5.2-0 libmpc3 libmpx2
  libnsspr4 libnss3 libpcsc-lite1 libpython-all-dev libpython-dev
  libpython-stdlib libpython2.7 libpython2.7-dev libpython2.7-minimal
  libpython2.7-stdlib libquadmath0 libstdc++-7-dev libstdc++6 libubsan0 libx16
  libxrender1 libxslt6 linux-libc-dev make manpages-dev python python-all
  python-all-dev python-asciicrypto python-cffi-backend python-crypto
  python-cryptography python-dbus python-dev python-enun34 python-gi
  python-idna python-ipaddress python-keyring python-keyrings.alt
  python-minimal python-pip-whl python-pkg-resources python-secretstorage
  python-setuptools python-six python-wheel python-xdg python2.7 python2.7-dev
  python2.7-minimal ssl-cert x11-common
Suggested packages:
  www-browser apache2-doc apache2-suexec-pristine | apache2-suexec-custom
  binutils-doc cpp-doc gcc-7-locales debian-keyring g++-multilib
  g++-7-multilib gcc-7-doc libstdc++6-7-dbg gcc-multilib autoconf automake
  libtool flex bison gdb gcc-doc gcc-7-multilib libgcc1-dbg libgomp1-dbg
  libitm1-dbg libatomic1-dbg libasan4-dbg liblsan0-dbg libstdc++6-dbg
  libubsan0-dbg libcilkrts5-dbg libmpx2-dbg libquadmath0-dbg default-jre
  glibc-doc cups-common bzip2 liblcms2-utils pcscd libstdc++-7-doc make-doc
  libnss-mdns fonts-dejavu-extra fonts-tpafont-gothic fonts-tpafont-mincho
  fonts-wqy-microhei fonts-wqy-zenhei fonts-indic python-doc python-tk
  python-crypto-doc python-cryptography-doc python-cryptography-vectors
  python-dbus-dbg python-dbus-doc python-enun34-doc python-gi-cairo
  gnome-keyring libkf5wallet-bin gir1.2-gnomekeyring-1.0 python-fs
  python-gdata python-keyczar python-secretstorage-doc python-setuptools-doc
  python2.7-doc binfmt-support openssl-blacklist
The following NEW packages will be installed:
  apache2 apache2-bin apache2-data apache2-utils binutils binutils-common
```

```
File Edit View Search Terminal Help
IDDD DS 2020
web.whatsapp.com +91 88843 09429: https://d...

Adding debian:Amazon_Root_CA_1.pem
Adding debian:Digicert_Assured_ID_Root_G3.pem
Adding debian:Cybertrust_Global_Root.pem
Adding debian:GlobalSign_Root_CA.pem
Adding debian:Starfield_Class_2_CA.pem
Adding debian:T-TeleSec_GlobalRoot_Class_3.pem
Adding debian:Go_Daddy_Root_Certificate_Authority_-_G2.pem
Adding debian:COMODO_Certification_Authority.pem
Adding debian:ACCVRAIZ1.pem
Adding debian:Entrust.net_Premium_2048_Secure_Server_CA.pem
Adding debian:Autoridad_de_Certificacion_Firmaprofesional_CIF_A62634868.pem
Adding debian:T-TeleSec_GlobalRoot_Class_2.pem
Adding debian:Go_Daddy_Class_2_CA.pem
Adding debian:TeliaSonera_Root_CA_v1.pem
Adding debian:Certum_Trusted_Network_CA.pem
Adding debian:TrustCor_ECA-1.pem
done.
Processing triggers for mime-support (3.60ubuntu1) ...
Processing triggers for ureadahead (0.100.0-21) ...
Processing triggers for libc-bin (2.27-3ubuntu1) ...
Processing triggers for systemd (237-3ubuntu0.41) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...
Processing triggers for ca-certificates (20191010-18.04.1) ...
Updating certificates in /etc/ssl/certs...
0 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d...
done.
done.
Setting up openjdk-8-jre-headless:amd64 (8u252-b09-1-18.04) ...
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/rmid to provide /usr/bin/rmid (rmid) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java to provide /usr/bin/java (java) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/keytool to provide /usr/bin/keytool (keytool) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/jjs to provide /usr/bin/jjs (jjs) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/pack200 to provide /usr/bin/pack200 (pack200) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/unpack200 to provide /usr/bin/unpack200 (unpack200) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/orbd to provide /usr/bin/orbd (orbd) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/servertool to provide /usr/bin/servertool (servertool) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/tnameserv to provide /usr/bin/tnameserv (tnameserv) in auto mode
Processing triggers for ufw (0.36-0ubuntu0.18.04.1) ...
ubuntu@ip-172-31-68-245:~$
```

b) Next, we download the latest build of Kafka and unzip it :

```
ubuntu@ip-172-31-68-245:~$
File Edit View Search Terminal Help
Adding debian:TeliaSonera_Root_CA_v1.pem
Adding debian:Certum_Trusted_Network_CA.pem
Adding debian:TrustCor_ECA-1.pem
done.
Processing triggers for mime-support (3.60ubuntu1) ...
Processing triggers for ureadahead (0.100.0-21) ...
Processing triggers for libc-bin (2.27-3ubuntu1) ...
Processing triggers for systemd (237-3ubuntu0.41) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...
Processing triggers for ca-certificates (20191010-18.04.1) ...
Updating certificates in /etc/ssl/certs...
0 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d...
done.
done.
Setting up openjdk-8-jre-headless:amd64 (8u252-b09-1-18.04) ...
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/rmid to provide /usr/bin/rmid (rmid) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java to provide /usr/bin/java (java) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/keytool to provide /usr/bin/keytool (keytool) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/jjs to provide /usr/bin/jjs (jjs) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/pack200 to provide /usr/bin/pack200 (pack200) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/unpack200 to provide /usr/bin/unpack200 (unpack200) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/orbd to provide /usr/bin/orbd (orbd) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/servertool to provide /usr/bin/servertool (servertool) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/tnameserv to provide /usr/bin/tnameserv (tnameserv) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/jexec to provide /usr/bin/jexec (jexec) in auto mode
Processing triggers for ufw (0.36-0ubuntu0.18.04.1) ...
ubuntu@ip-172-31-68-245:~$ wget https://downloads.apache.org/kafka/2.5.0/kafka_2.13-2.5.0.tgz
--2020-07-22 07:23:45-- https://downloads.apache.org/kafka/2.5.0/kafka_2.13-2.5.0.tgz
Resolving downloads.apache.org (downloads.apache.org)... 88.99.95.219, 2a01:4f8:10a:201a::2
Connecting to downloads.apache.org (downloads.apache.org)|88.99.95.219|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 61459093 (59M) [application/x-gzip]
Saving to: 'kafka_2.13-2.5.0.tgz'

kafka_2.13-2.5.0.tgz
100%[=====] 58.61M 15.8MB/s in 7.6s

2020-07-22 07:23:53 (7.69 MB/s) - 'kafka_2.13-2.5.0.tgz' saved [61459093/61459093]

ubuntu@ip-172-31-68-245:~$ tar -xzf kafka_2.13-2.5.0.tgz
ubuntu@ip-172-31-68-245:~$ rm kafka_2.13-2.5.0.tgz
ubuntu@ip-172-31-68-245:~$
```

c) Next, we start the Kafka and Zookeeper servers in the instance.

```
File Edit View Search Terminal Help
Processing triggers for systemd (237-3ubuntu0.41) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...
Processing triggers for ca-certificates (20190118-18.04.1) ...
Updating certificates in /etc/ssl/certs...
0 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d...
done.
done.
Setting up openjdk-8-jre-headless:amd64 (8u252-b09-1-18.04) ...
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/rmid to provide /usr/bin/rmid (rmid) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java to provide /usr/bin/java (java) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/keytool to provide /usr/bin/keytool (keytool) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/jjs to provide /usr/bin/jjs (jjs) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/pack200 to provide /usr/bin/pack200 (pack200) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/rmiregistry to provide /usr/bin/rmiregistry (rmiregistry) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/unpack200 to provide /usr/bin/unpack200 (unpack200) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/orbd to provide /usr/bin/orbd (orbd) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/servertool to provide /usr/bin/servertool (servertool) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/tnameserv to provide /usr/bin/tnameserv (tnameserv) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/lib/jexec to provide /usr/bin/jexec (jexec) in auto mode
Processing triggers for ufw (0.36-0ubuntu0.18.04.1) ...
ubuntu@ip-172-31-68-245:~$ wget https://downloads.apache.org/kafka/2.5.0/kafka_2.13-2.5.0.tgz
--2020-07-22 07:23:45-- https://downloads.apache.org/kafka/2.5.0/kafka_2.13-2.5.0.tgz
Resolving downloads.apache.org (downloads.apache.org)... 88.99.95.219, 2a01:4f8:10a::2
Connecting to downloads.apache.org (downloads.apache.org)|88.99.95.219|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 61459093 (59M) [application/x-gzip]
Saving to: 'kafka_2.13-2.5.0.tgz'

kafka_2.13-2.5.0.tgz      100%[=====] 58.61M  15.8MB/s   in 7.6s

2020-07-22 07:23:53 (7.69 MB/s) - 'kafka_2.13-2.5.0.tgz' saved [61459093/61459093]

ubuntu@ip-172-31-68-245:~$ tar -xzf kafka_2.13-2.5.0.tgz
ubuntu@ip-172-31-68-245:~$ rm kafka_2.13-2.5.0.tgz
ubuntu@ip-172-31-68-245:~$ nohup ~/kafka_2.13-2.5.0/bin/zookeeper-server-start.sh ~/kafka_2.13-2.5.0/config/zookeeper.properties & ~/zookeeper-logs &
[1] 10445
ubuntu@ip-172-31-68-245:~$ nohup ~/kafka_2.13-2.5.0/bin/kafka-server-start.sh ~/kafka_2.13-2.5.0/config/server.properties & ~/kafka-logs &
[2] 10446
ubuntu@ip-172-31-68-245:~$ nohup: Ignoring input and redirecting stderr to stdout
nohup: Ignoring input and redirecting stderr to stdout

ubuntu@ip-172-31-68-245:~$ |
```

d) We then create the required topic in Kafka.

```
File Edit View Search Terminal Help
0 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d...
done.
done.
Setting up openjdk-8-jre-headless:amd64 (8u252-b09-1-18.04) ...
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/rmid to provide /usr/bin/rmid (rmid) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java to provide /usr/bin/java (java) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/keytool to provide /usr/bin/keytool (keytool) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/jjs to provide /usr/bin/jjs (jjs) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/pack200 to provide /usr/bin/pack200 (pack200) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/rmiregistry to provide /usr/bin/rmiregistry (rmiregistry) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/unpack200 to provide /usr/bin/unpack200 (unpack200) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/orbd to provide /usr/bin/orbd (orbd) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/servertool to provide /usr/bin/servertool (servertool) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/tnameserv to provide /usr/bin/tnameserv (tnameserv) in auto mode
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/lib/jexec to provide /usr/bin/jexec (jexec) in auto mode
Processing triggers for ufw (0.36-0ubuntu0.18.04.1) ...
ubuntu@ip-172-31-68-245:~$ wget https://downloads.apache.org/kafka/2.5.0/kafka_2.13-2.5.0.tgz
--2020-07-22 07:23:45-- https://downloads.apache.org/kafka/2.5.0/kafka_2.13-2.5.0.tgz
Resolving downloads.apache.org (downloads.apache.org)... 88.99.95.219, 2a01:4f8:10a::2
Connecting to downloads.apache.org (downloads.apache.org)|88.99.95.219|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 61459093 (59M) [application/x-gzip]
Saving to: 'kafka_2.13-2.5.0.tgz'

kafka_2.13-2.5.0.tgz      100%[=====] 58.61M  15.8MB/s   in 7.6s

2020-07-22 07:23:53 (7.69 MB/s) - 'kafka_2.13-2.5.0.tgz' saved [61459093/61459093]

ubuntu@ip-172-31-68-245:~$ tar -xzf kafka_2.13-2.5.0.tgz
ubuntu@ip-172-31-68-245:~$ rm kafka_2.13-2.5.0.tgz
ubuntu@ip-172-31-68-245:~$ nohup ~/kafka_2.13-2.5.0/bin/zookeeper-server-start.sh ~/kafka_2.13-2.5.0/config/zookeeper.properties & ~/zookeeper-logs &
[1] 10445
ubuntu@ip-172-31-68-245:~$ nohup ~/kafka_2.13-2.5.0/bin/kafka-server-start.sh ~/kafka_2.13-2.5.0/config/server.properties & ~/kafka-logs &
[2] 10446
ubuntu@ip-172-31-68-245:~$ nohup: Ignoring input and redirecting stderr to stdout
nohup: Ignoring input and redirecting stderr to stdout

ubuntu@ip-172-31-68-245:~$ ~/kafka_2.13-2.5.0/bin/kafka-topics.sh --create --zookeeper localhost:2181 --replication-factor 1 --partitions 1 --topic newyork

Created topic newyork.
ubuntu@ip-172-31-68-245:~$
ubuntu@ip-172-31-68-245:~$ |
```


e) We then install the required python libraries :

```
ubuntu@ip-172-31-68-245: ~$ pip install requests-oauthlib python-twitter six tweepy numpy click Werkzeug itsdangerous MarkupSafe Jinja2 flask
Collecting requests-oauthlib (from python-twitter)
  Downloading https://files.pythonhosted.org/packages/a3/12/b9274d845a62e4edf04d2f4164d82532b5a0b03836d44d71c6f3d379/requests_oauthlib-1.3.0-py2.py3-none-any.whl
Collecting requests (from python-twitter)
  Downloading https://files.pythonhosted.org/packages/45/1e/0c169c6a5381e241ba7404532c16a21d08ab872c9bed8bdc4c423954103/requests-2.24.0-py2.py3-none-any.whl (61kB)
Collecting six>=1.10.0 (from tweepy)
  Downloading https://files.pythonhosted.org/packages/ee/ff/48bde5c0f013094d729fe4b0316ba2a24774b3ff1c52d924a8a4cb04078a/six-1.15.0-py2.py3-none-any.whl
Collecting click>=5.1 (from flask)
  Downloading https://files.pythonhosted.org/packages/d2/3d/fa76db83bf75c4f8d338c2fd15c8d33fd7ad23a9b5e57eb6c5de26b430e/click-7.1.2-py2.py3-none-any.whl (82kB)
Collecting Werkzeug>=0.15 (from flask)
  Downloading https://files.pythonhosted.org/packages/cc/94/5f7079a0e0b06863ef8f1da638721e9da21e5bace597595b318f71d62e/Werkzeug-1.0.1-py2.py3-none-any.whl (298kB)
Collecting itsdangerous>=0.24 (from flask)
  Downloading https://files.pythonhosted.org/packages/76/ae/44b03b253d6fde317f32c24d10b3b35c2239807046a4c953c7b89fa49e/itsdangerous-1.1.0-py2.py3-none-any.whl
Collecting Jinja2>=2.10.1 (from flask)
  Downloading https://files.pythonhosted.org/packages/38/9e/f663a2aa6a09d38042ae1a2c5659828bb9b41ea3a6efa20a20fd92b121/Jinja2-2.11.2-py2.py3-none-any.whl (125kB)
Collecting oauthlib>=3.0.0 (from requests-oauthlib->python-twitter)
  Downloading https://files.pythonhosted.org/packages/05/57/ce2e7a8fa7c0a9b54a0581b14a65b56e2b5759dbc98e80627142b8a3704/oauthlib-3.1.0-py2.py3-none-any.whl (147kB)
Collecting urllib3>=1.25.0,!=1.25.1,<1.26,!=1.21.1 (from requests->python-twitter)
  Downloading https://files.pythonhosted.org/packages/e1/e5/df302e8017440f11c1cc41a6b43283672f5a70aa29227bf58149dc72f/urllib3-1.25.9-py2.py3-none-any.whl (126kB)
Collecting chardet>=4.0.0 (from requests->python-twitter)
  Downloading https://files.pythonhosted.org/packages/bc/a9/01ffebfb562e4274b6487b4bb1dddec7ca55ec7510b22e4c51f14098443b8/chardet-3.0.4-py2.py3-none-any.whl (133kB)
Collecting certifi>=2017.4.17 (from requests->python-twitter)
  Downloading https://files.pythonhosted.org/packages/5e/c4/6c4fe72df5343c3226f0b4e0bb042e4dc1348328b4718baf286f86d87/certifi-2020.6.20-py2.py3-none-any.whl (156kB)
Collecting idna>=3.0,!=3.0.2 (from requests->python-twitter)
  Downloading https://files.pythonhosted.org/packages/5e/c4/6c4fe72df5343c3226f0b4e0bb042e4dc1348328b4718baf286f86d87/idna-2.10-py2.py3-none-any.whl (58kB)
Collecting MarkupSafe>=0.23 (from Jinja2>=2.10.1->flask)
  Downloading https://files.pythonhosted.org/packages/fb/40/f3adb7cf24a8012813c5ed20329eb22d5d2e2a0ecf73d21d6b85865da11/MarkupSafe-1.1.1-cp27mu-manylinux1_x86_64.whl
Building wheels for collected packages: future
  Running setup.py bdist_wheel for future ... done
  Stored in directory: /home/ubuntu/.cache/pip/wheels/8b/99/a0/81daf51dc359a9377b110a8a886b3895921802d2fc1b2397e
Successfully built future
Installing collected packages: kafka-python, future, oauthlib, urllib3, chardet, certifi, idna, requests, requests-oauthlib, python-twitter, six, tweepy, numpy, click, Werkzeug, itsdangerous, MarkupSafe, Jinja2, flask
Successfully installed Jinja2-2.11.2 MarkupSafe-1.1.1 Werkzeug-1.0.1 certifi-2020.6.20 chardet-3.0.4 click-7.1.2 flask-1.1.2 future-0.18.2 idna-2.10 itsdangerous-1.1.0 kafka-python-2.0.1 num
py-1.16.6 oauthlib-3.1.0 python-twitter-3.5 requests-2.24.0 requests-oauthlib-1.3.0 six-1.15.0 tweepy-3.9.0 urllib3-1.25.9
ubuntu@ip-172-31-68-245: ~$
```

f) We then upload the tweet-streaming.py code and run it. (The API keys are removed in image)

```
ubuntu@ip-172-31-68-245: ~$ nano python-streaming.py
GNU nano 2.9.3 python-streaming.py

'''
Tweet Streaming to Kafka Producer

Run the command using "python tweet-streaming.py"

Make sure you create a topic called "newyork" before running the code
'''

from tweepy.streaming import StreamListener
from tweepy import OAuthHandler
from tweepy import Stream
from kafka import KafkaProducer, KafkaClient

access_token = "<fill access_token>"
access_token_secret = "<fill access_token_secret>"
consumer_key = "<fill consumer_key>"
consumer_secret = "<fill consumer_secret>"

class StdOutListener(StreamListener):
    def on_data(self, data):
        # producer.send("newyork", data.encode('utf-8'))
        producer.send("california", data.encode('utf-8'))
        print(data)
        return True
    def on_error(self, status):
        print(status)

kafka = KafkaClient()
producer = KafkaProducer()
l = StdOutListener()
auth = OAuthHandler(consumer_key, consumer_secret)
auth.set_access_token(access_token, access_token_secret)
stream = Stream(auth, l)
# stream.filter(locations=[-79.762152,40.496103,-71.856214,45.01585]) # New York
stream.filter(locations=[-124.409591,32.534156,-114.131211,42.009518]) # California
producer.flush()
producer.close()
```



```
File Edit View Search Terminal Help
GNU nano 2.9.3 tweet-count-fp.py

import json
import os
from pyspark import SparkContext, SparkConf
from pyspark.streaming import StreamingContext
from pyspark.streaming.kafka import KafkaUtils
import string
import sys
import re
from pyspark.mllib.fpm import FPGrowth

stop_words = ["ourselves", "hers", "between", "yourself", "but", "again", "there", "about", "once", "during", "out", "very", "having", "with", "they", "own", "an", "be", "some", "for", "do"]
def writeToFile(rdd):
    with open("count.txt", "w") as f:
        f.write(str(rdd.count()))
    rdd_words = rdd.map(lambda line: list(filter(lambda a: a != "" and a not in stop_words, list(set(line.strip().split(' ')))))).filter(lambda x: x != [])
    model = FPGrowth.train(rdd_words, minSupport=0.02, numPartitions=20)
    result = model.freqItemsets().collect()
    with open("frequent_items.txt", "w") as g:
        for i in range(len(result)):
            g.write(json.dumps(result[i].items) + "\n")

os.environ['PYSPARK_SUBMIT_ARGS'] = '--jars ~/spark-2.3.4-bin-hadoop2.6/jars/spark-streaming-kafka-0-8-assembly_2.11-2.3.4.jar pyspark-shell'
# sc.stop()
conf = SparkConf().set('spark.io.compression.codec', 'snappy').set('spark.executor.memory', "4g").set('spark.driver.memory', "4g")
sc = SparkContext(conf = conf)
batch_interval = 600
ssc = StreamingContext(sc, batch_interval)

twitterKafkaStream = KafkaUtils.createStream(ssc, "localhost:2181", "spark-streaming", {"newyork": 1})
# twitterKafkaStream = KafkaUtils.createStream(ssc, "localhost:2181", "spark-streaming", {"california": 1})
tweets = twitterKafkaStream.map(lambda v: json.loads(v[1])).map(lambda x: re.sub(r'[^a-zA-Z0-9@/!\' ]+', '', x['text']).lower())
tweets.foreachRDD(writeToFile)

ssc.start()
ssc.awaitTermination()
```

```
File Edit View Search Terminal Help
ubuntu@ip-172-31-68-245:~$
--2020-07-22 07:53:35-- https://downloads.apache.org/spark/spark-2.4.6/spark-2.4.6-bin-hadoop2.7.tgz
Resolving downloads.apache.org (downloads.apache.org)... 88.99.95.219, 2a01:4f8:10a:201a::2
Connecting to downloads.apache.org (downloads.apache.org)|88.99.95.219|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 233215067 (222M) [application/x-gzip]
Saving to: 'spark-2.4.6-bin-hadoop2.7.tgz'

spark-2.4.6-bin-hadoop2.7.tgz
spark-2.4.6-bin-hadoop2.7.tgz
100%[=====] 222.41M 16.0MB/s in 15s

2020-07-22 07:53:50 (15.1 MB/s) - 'spark-2.4.6-bin-hadoop2.7.tgz' saved [233215067/233215067]

ubuntu@ip-172-31-68-245:~$
ubuntu@ip-172-31-68-245:~$ tar -xzf spark-2.4.6-bin-hadoop2.7.tgz
ubuntu@ip-172-31-68-245:~$ rm spark-2.4.6-bin-hadoop2.7.tgz
ubuntu@ip-172-31-68-245:~$
ubuntu@ip-172-31-68-245:~$
ubuntu@ip-172-31-68-245:~$
ubuntu@ip-172-31-68-245:~$
ubuntu@ip-172-31-68-245:~$
ubuntu@ip-172-31-68-245:~$ wget https://repo1.maven.org/maven2/org/apache/spark/spark-streaming-kafka-0-8-assembly_2.11/2.4.6/spark-streaming-kafka-0-8-assembly_2.11-2.4.6.jar -P ~/spark-2.4.6-bin-hadoop2.7/jars/
--2020-07-22 07:55:50-- https://repo1.maven.org/maven2/org/apache/spark/spark-streaming-kafka-0-8-assembly_2.11/2.4.6/spark-streaming-kafka-0-8-assembly_2.11-2.4.6.jar
Resolving repo1.maven.org (repo1.maven.org)... 199.232.64.209
Connecting to repo1.maven.org (repo1.maven.org)|199.232.64.209|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 11709363 (11M) [application/java-archive]
Saving to: '/home/ubuntu/spark-2.4.6-bin-hadoop2.7/jars/spark-streaming-kafka-0-8-assembly_2.11-2.4.6.jar'

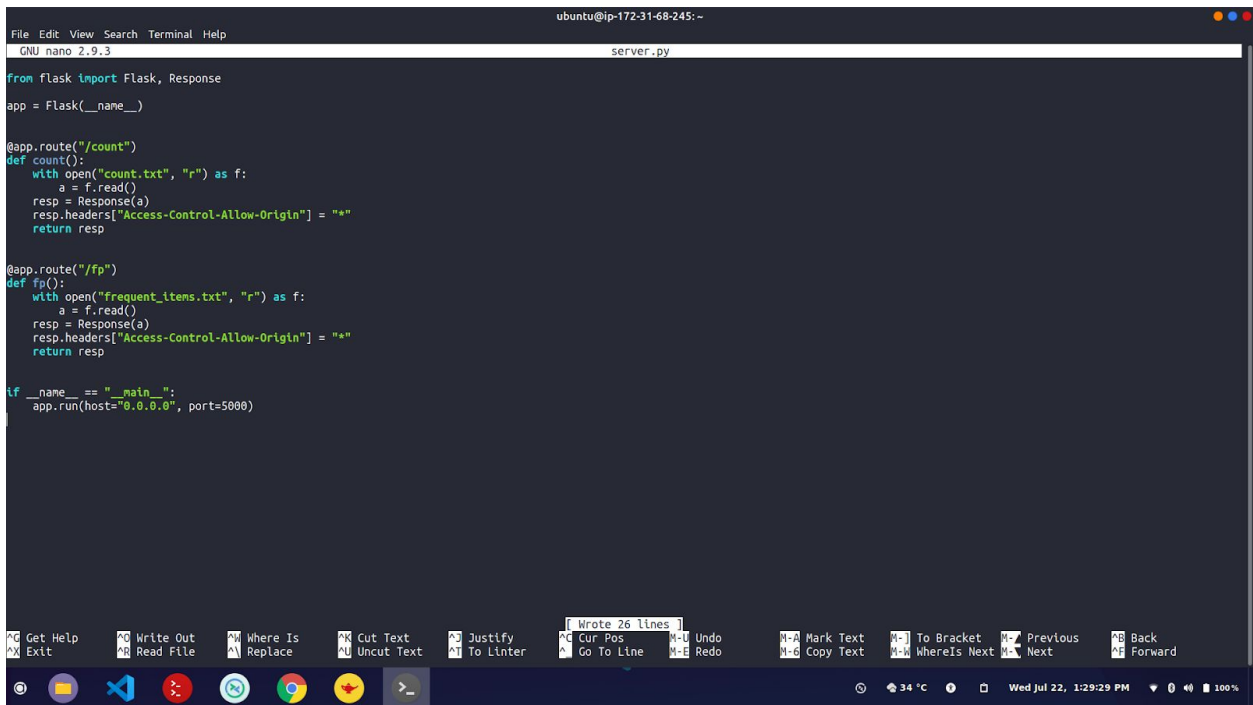
spark-streaming-kafka-0-8-assembly_2.11-2.4.6.jar 100%[=====] 11.17M 66.3MB/s in 0.2s

2020-07-22 07:55:51 (66.3 MB/s) - '/home/ubuntu/spark-2.4.6-bin-hadoop2.7/jars/spark-streaming-kafka-0-8-assembly_2.11-2.4.6.jar' saved [11709363/11709363]

ubuntu@ip-172-31-68-245:~$ nano tweet-count-fp.py
ubuntu@ip-172-31-68-245:~$ nohup spark-2.4.6-bin-hadoop2.7/bin/spark-submit tweet-count-fp.py > out.log &
[1] 13307
ubuntu@ip-172-31-68-245:~$ nohup: ignoring input and redirecting stderr to stdout
ubuntu@ip-172-31-68-245:~$
```

i) The data is stored in files `count.txt` and `frequent_items.txt`.

- j) Next, we create a server in the instance by using Flask. We upload the `server.py` and run it.



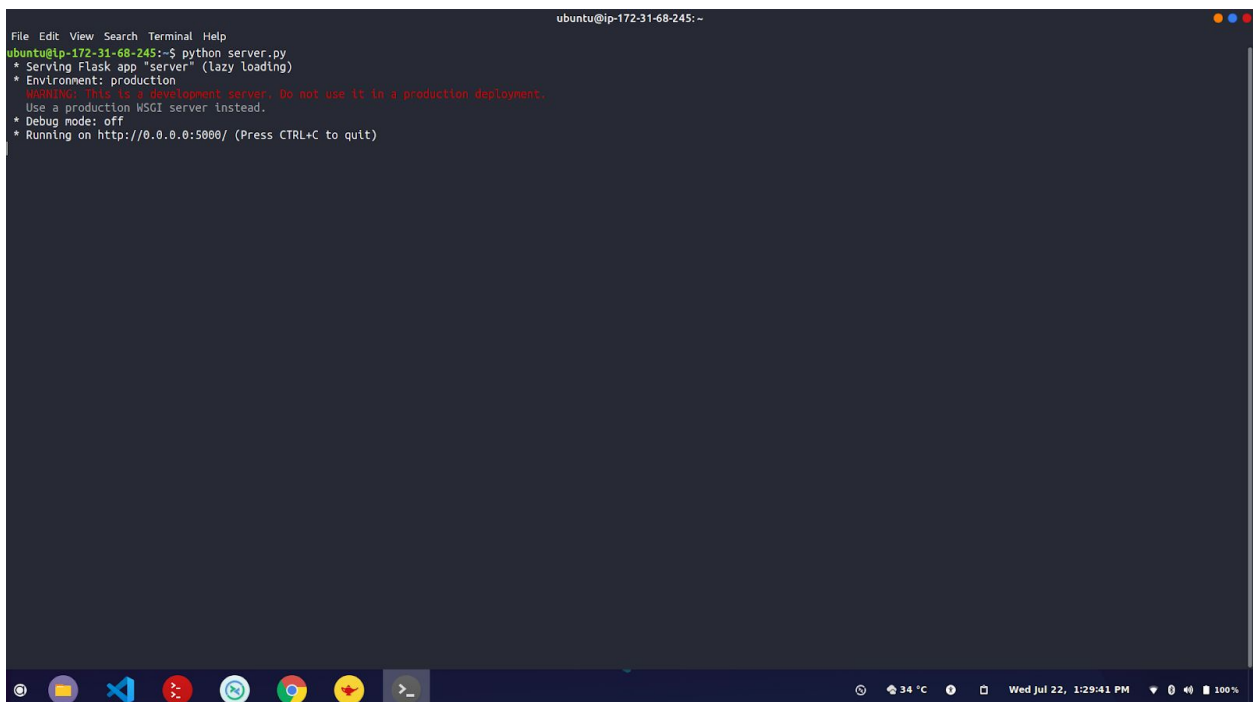
```
File Edit View Search Terminal Help
GNU nano 2.9.3 server.py

from flask import Flask, Response
app = Flask(__name__)

@app.route("/count")
def count():
    with open("count.txt", "r") as f:
        a = f.read()
        resp = Response(a)
        resp.headers["Access-Control-Allow-Origin"] = "*"
        return resp

@app.route("/fp")
def fp():
    with open("frequent_items.txt", "r") as f:
        a = f.read()
        resp = Response(a)
        resp.headers["Access-Control-Allow-Origin"] = "*"
        return resp

if __name__ == "__main__":
    app.run(host="0.0.0.0", port=5000)
```



```
File Edit View Search Terminal Help
ubuntu@ip-172-31-68-245:~$ python server.py
* Serving Flask app "server" (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: off
* Running on http://0.0.0.0:5000/ (Press CTRL+C to quit)
```


- k) We then do the same above for the other state, and then we upload the `test.html` in one of the instances and then we can see the output.

Number of tweets in New York : 241
Frequent Pairs in New York : ["get"] ["dont"] ["@"] ["directions"] ["im"] ["construction"] ["like"]
