**CS5590 BigData Programming - Project Fall 2018**

**Project Title:**

1. Web Traffic anolmolies and Web Traffic Prediction

2. Credit Loan risk analysis using Spark Machine learning.

**Objective**

The project has two part. In first part we will be using the server logs from renowned website defect and report web traffic anolmolies r using Spark, Graph and Cassandra.

For second part we will be using the dataset from UCI data library and perform Classification .Using Spark and Machine learning library.

**Motivation :**

Edmund.com Forecast Model

**Significance :**

It helps in managing the 24/7 availability of website and serve customer better and faster way.

**Feature :**

Spark , graph analysis and real time dashboard for good and bad http request.

**Reference:**

http://blog.cloudera.com/blog/2015/03/how-edmunds-com-used-spark-streaming-to-build-a-near-real-time-dashboard/

**Datasets:**

**Web Logs:**

http://ita.ee.lbl.gov/html/contrib/NASA-HTTP.html

**Credit Loan:**

<https://archive.ics.uci.edu/ml/datasets/statlog+(german+credit+data)>

**Increment 1: Part 1 (Web Traffic Anomalies )**

**Dataset:**

**Web Log Analysis:**

http://ita.ee.lbl.gov/html/contrib/NASA-HTTP.html

**Detail design of Features:**

In the First we have downloaded the dataset from NASA web server for using in our project.

**Analysis :**

We have done the analysis of dataset and found out that the dataset is in below format.

The logs are an ASCII file with one line per request, with the following columns:

1. **host** making the request. A hostname when possible, otherwise the Internet address if the name could not be looked up.
2. **timestamp** in the format "DAY MON DD HH:MM:SS YYYY", where DAY is the day of the week, MON is the name of the month, DD is the day of the month, HH:MM:SS is the time of day using a 24-hour clock, and YYYY is the year. The timezone is -0400.
3. **request** given in quotes.
4. **HTTP** reply code.
5. **bytes** in the reply.

**Implementation:**

At this stage no implementation was performed for Web Log Analysis

**Preliminary Results :**

Dataset need to be cleaned up properly for further analysis.

**Project Management**

**Implementation status report**

**Work completed:**

**Description:**

Web log Dataset has been loaded and format and feature has been identified/extraction.

**Responsibility (Task, Person)**

Downloading and analyzing Dataset: Raju Nekad**i**

**Contributions (members/percentage):**

Raju Nekadi 50%

Sushma Manne 50%

**Work to be completed :**

**Description**

Flume Setup

Reading Dataset From Flume and loading to Spark

Spark Analysis generating result

Loading dataset in Cassandra/Hive

Graph Generation

**Responsibility (Task, Person)**

Flume Setup/Raju Nekadi

Reading Dataset From Flume and loading to Spark/Raju Nekadi

Spark Analysis generating result /Raju Nekadi

Loading dataset in Cassandra/Hive Sushma Mane

Graph Generation/Sushma Mane

**Issues/Concerns:**

None

**References/Bibliography:**

https://blog.cloudera.com/blog/2016/06/how-to-detect-and-report-web-traffic-anomalies-in-near-real-time/

**Increment 2 : Part 1(Web Traffic Anomalies)**

**Dataset:**

**Web Log Analysis:**

http://ita.ee.lbl.gov/html/contrib/NASA-HTTP.html

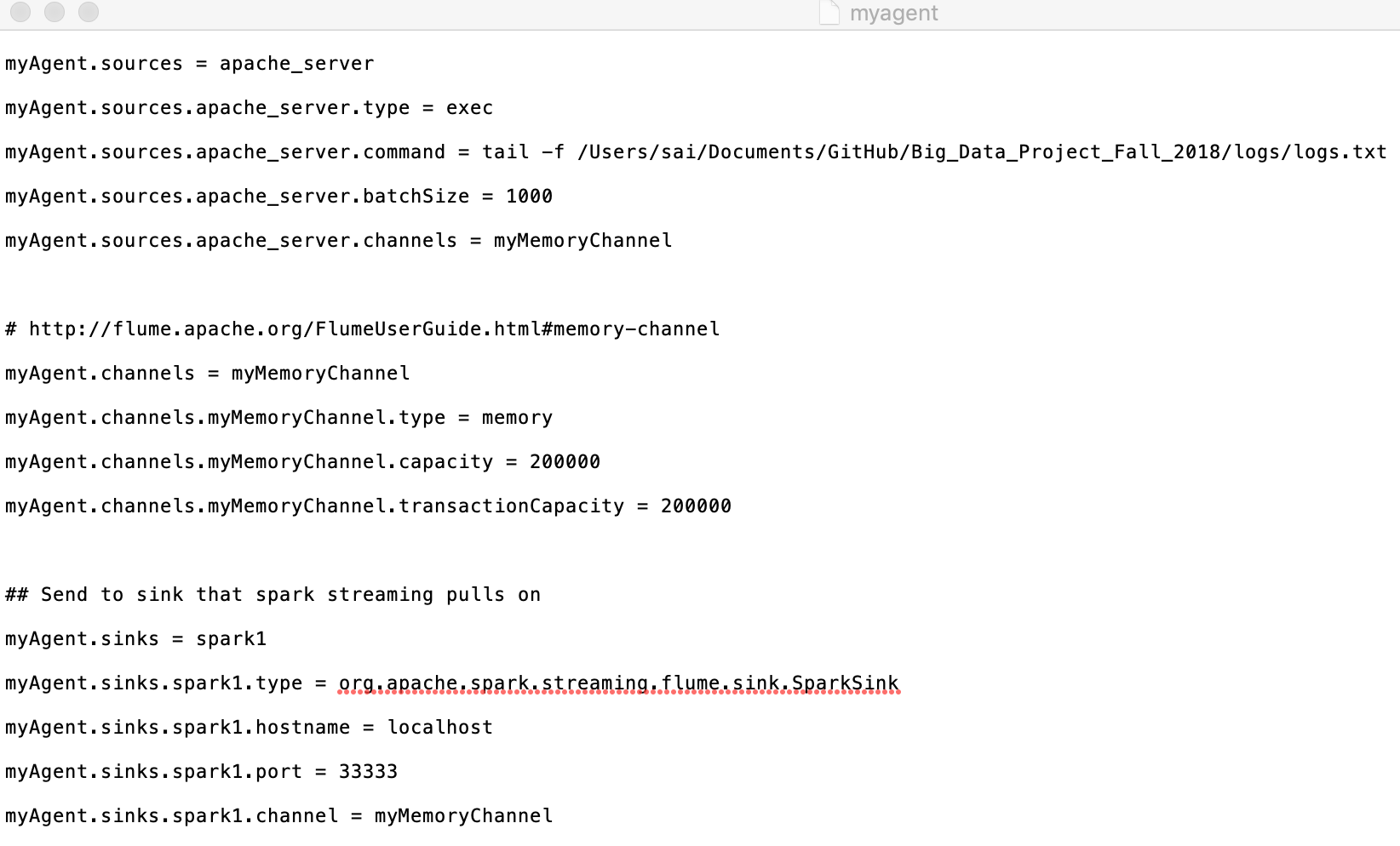
**Detail design of Features:**

In this increment we have have setup the flume agent in order to receive the log data at real time. The flume agent uses the conf file for running the source ,channel and sink.

Also we have setup the Spark Streaming which will pull the stream of data at run time. We have used the Pull based approach here

**Analysis :**

Flume agent setup we have used the exec source ,memory channel , and spark sink for polling as shown below.

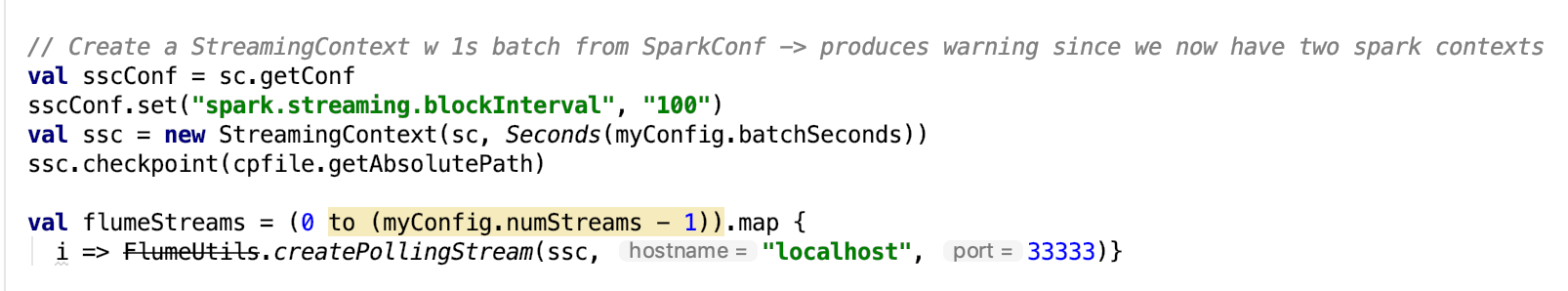


Spark Streaming pull based logic was developed to read the stream of data from Spark Sink.

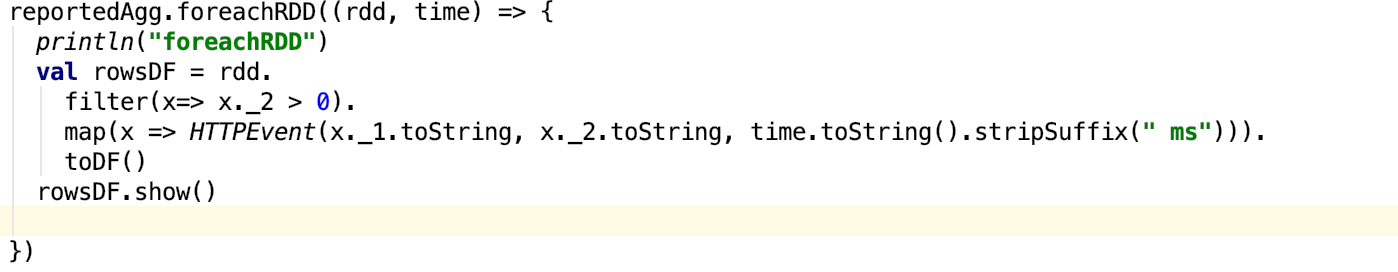
In this part we have developed Classes like HTTPcode,Config, and Interrogator.



We have used the Spark Streaming Context which will be reading the data from localhost port 33333.



And Finally we are saving the data in form of dataframe which shows the Status , Count and batch time in millisecond.



**Implementation:**

1. Implemented the Flume agent for generating spark sink.
2. Implemented Spark Stream which read the stream of data and write it dataframe.

**Preliminary Results :**

Flume Agent setup was successful and Spark Streaming Pooling was also successful.

**Steps to run flume with Spark Sink**

flume-ng agent -n myAgent --conf /usr/local/flume/conf -f /usr/local/flume/conf/myagent.conf -Dflume.root.logger=DEBUG,console

**Steps to run Spark Streaming Pull program**

run --aggfile /Users/sai/Documents/GitHub/data/agg --outfile /Users/sai/Documents/GitHub/data/out --cpfile /Users/sai/Documents/GitHub/data/checkpoint

**Project Management**

**Implementation status report**

**Work completed:**

**Description:**

**1.**Implemented the Flume agent for generating spark sink.

2. Implemented Spark Stream which read the stream of data and write it dataframe.

**Responsibility (Task, Person)**

Flume Agent and Spark Stream Pull based setup: Raju Nekad**i**

**Contributions (members/percentage):**

Raju Nekadi 80%

Sushma Manne 20%

**Work to be completed :**

**Description**

Loading data to Cassandra Table

**Responsibility (Task, Person)**

Loading dataset in Cassandra/Hive Sushma Mane(100%)

**Issues/Concerns:**

None

**References/Bibliography:**

https://blog.cloudera.com/blog/2016/06/how-to-detect-and-report-web-traffic-anomalies-in-near-real-time/