**Software Development Report**

Anying Xiang

a1758639

1. **Organisation of Software Development Report**

This report is organised as follows. Section 2 presents the organisation of repository (i.e., the folder structure). Section 3 explains the artefacts in detail. Section 4 illustrates how to run the operational artefacts. Section 5 is a short conclusion.

1. **Organisation of** **Folder ‘Anying Xiang’ in Repository**

This folder contains what I have done in this project. Besides, the outcomes of experiments are included and analysed in my research report. The subfolders are as follows.

Anying Xiang

├─**Code**

│ ├─Adaptation Approach

│ ├─Automated Scripts

│ └─BDCA Systems

│ ├─Hadoop

│ │ └─NaiveBayes

│ └─Spark

│ ├─NaiveBayes

│ └─RandomForest

├─**Exploration and Reviews**

│ └─Exploration

│ └─Reviews

├─**Process**

└─**Runnable Jars**

1. **Artefacts Explanation**

**Exploration and Reviews Folder**

The *Exploration and Reviews* folder consists of *Exploration* and *Reviews*.

For *Exploration*, I have explored Hadoop, Spark, MapReduce, Mahout, SparkML, Naïve Bayes, Support Vector Machine and Cross Validation. I also tried to run some examples and did some practices in this phase.

For *Reviews*, I have reviewed papers about Big Data and Security, BDCA, KDD99 and Scalability.

**Process Folder**

The *Process* folder contains a part of files produced during the project process.

**Code Folder**

The *Code* folder consists of *Adaptation Approach*, *Automated Scripts* and *BDCA Systems*.

For *BDCA Systems*, I have implemented one with Naïve Bayes on Spark (i.e., *NaiveBayesSpark.scala*), one with Random Forest on Spark (i.e., *RandomForestSpark.scala*), and one with Naïve Bayes on Hadoop (i.e., *HadoopNB.java*).

For *Automated Scripts*, I have written some Shell Scripts (e.g., *TestHadoopKDD.sh, TestRFMainCIDDS.sh*) to automate some processes in experiments, where I need to run BDCA systems repetitively.

For *Adaptation Approach*, I have implemented the parameter-driven adaptation approach presented in the research report with Shell Script (i.e., *AdaptionBDCA.sh*).

**Runnable Jars Folder**

The *Runnable Jars* folder contains four Jar files (i.e., BDCA systems). Two Jar files can use KDD99, DARPA, CIDDS and CICIDS2017 with Naïve Bayes and Random Forest respectively on Spark (i.e., *BDCA\_Spark\_NaiveBayes.jar* and *BDCA\_Spark\_RandomForest.jar*). Two Jar files can use KDD99 and CIDDS respectively with Naïve Bayes on Hadoop (i.e., *BDCA\_Hadoop\_KDD99.jar* and *BDCA\_Hadoop\_CIDDS.jar*).

1. **Instructions on How to Run** **the Operational Artefacts**

The BDCA systems run in distributed mode on cloud. The files involved in the system (i.e., input data sets) should be put onto the HDFS.

The Jar files in *Runnable Jars* are produced by compiling source code files in *Code* folder. The *.scala* file is compiled to Jar file with *sbt*. The *.java* file can be exported to Jar file in Eclipse.

1. **The steps to run Spark BDCA systems (i.e.,** ***BDCA\_Spark\_NaiveBayes.jar* and *BDCA\_Spark\_RandomForest.jar*) are as follows.**
2. Start Hadoop, Yarn and Spark:

* $HADOOP\_HOME/sbin/start-dfs.sh
* $HADOOP\_HOME/sbin/start-yarn.sh
* $SPARK\_HOME/sbin/start-all.sh

1. Run cmd on Master node: **$SPARK\_HOME/bin/spark-submit --master yarn --deploy-mode cluster --num-executors *n path\_of\_Jar\_file path\_of\_train\_file path\_of\_test\_file dataset\_identifier***

* ***n*:** number of executors (worker nodes in Yarn)
* ***path\_of\_Jar\_file:*** path in master node
* ***path\_of\_train\_file****,* ***path\_of\_test\_file:*** path in HDFS
* ***dataset\_identifier:*** KDD99 (1), DARPA (2), CIDDS (3), CICIDS2017(4)

**Example**: $SPARK\_HOME/bin/spark-submit --master yarn --deploy-mode cluster --num-executors 5 /home/ubuntu/BDCA\_Spark\_NaiveBayes.jar /ubuntu/dataset/preprd-train-CIDIDS.txt /ubuntu/dataset/preprd-test-CIDIDS.txt 3

1. The **outcome** is like:

2019-05-05 02:37:03 INFO Client:54 -

client token: N/A

diagnostics: N/A

ApplicationMaster host: **cluster8-worker8-3.novalocal**

ApplicationMaster RPC port: 37485

queue: default

start time: 1557022824498

final status: SUCCEEDED

tracking URL: http://master:8088/proxy/application\_1556894988144\_0031/

user: ubuntu

The **result** is in ***yarn.nodemanager.log-dirs***/ application\_appId/container\_appId\_01\_000001/stdout of *ApplicationMaster host* (e.g., $HADOOP\_HOME/yarn-log/ application\_1556894988144\_0031/container\_1556894988144\_0031\_01\_000001/stdout in **cluster8-worker8-3.novalocal**)

1. **The steps to run Hadoop BDCA systems (i.e., *BDCA\_Hadoop\_KDD99.jar* and *BDCA\_Hadoop\_CIDDS.jar*) are as follows.**
2. Start Hadoop and Yarn:

* $HADOOP\_HOME/sbin/start-dfs.sh
* $HADOOP\_HOME/sbin/start-yarn.sh

1. Run cmd on Master node: **hadoop jar *path\_of\_Jar\_file***

* ***path\_of\_Jar\_file:*** path in master node

**Example**: hadoop jar /home/ubuntu/Cloud/OpenStack/nb\_hadoop/ BDCA\_Hadoop\_CIDDS.jar

**Note**: The path of train and test files are set in the source code for BDCA on Hadoop. If the paths need to be changed, then please modify them in the source code (i.e., *HadoopNB.java*).

1. **The steps to run Adaptation Implementation (i.e., *AdaptionBDCA.sh*) are as follows.**
2. Run cmd on local machine: ./AdaptionBDCA.sh ***path\_of\_Jar\_file*** ***path\_of\_train\_file path\_of\_test\_file dataset\_identifier*** ***responseTimeType***

* ***path\_of\_Jar\_file:*** path in master node
* ***path\_of\_train\_file****,* ***path\_of\_test\_file:*** path in HDFS
* ***dataset\_identifier:*** KDD99 (1), DARPA (2), CIDDS (3), CICIDS2017(4)
* ***responseTimeType:*** train or test

**Example**: ./AdaptionBDCA.sh BDCA\_Spark\_NaiveBayes.jar KDDTrain.txt KDDTest.txt 1 train

**Note**: The information of nodes is set in the Shell Script. If it needs to be changed (e.g., IPs of nodes), then please modify it in the Shell Script (i.e., *AdaptionBDCA.sh*).

1. **Conclusion**

This Software Development Report includes information about files and artefacts in folder ‘Anying Xiang’ in the Repository. It describes how to run the operational artefacts (i.e., BDCA systems and Adaptation Script) in detail.