**Spark with NLP processing**

**Author: Prashant**

**Index:**

**Chapter1: Cluster setup:**

**Chapter 2: Hadoop Installation Guide**

**Chapter 3: Spark Installation Guide**

**Chapter 4: HBASE Installation Guide**

**Chapter 5: SOLR Installation Guide**

**Chapter 6: LILY HBASE INDEXER Installation Guide**

**Chapter 7: Running the spark first time with NLP code**

**Chapter 8: Installing Tomcat**

**Chapter 9: Installing pdftable library**

**Chapter 10: Start the services and run the code**

**Chapter 11: Adding a new field in HBASE, SOLR, and Lily indexer:**

**Chapter 12: Clear Spark and HBASE**

**Chapter1: Cluster setup:**

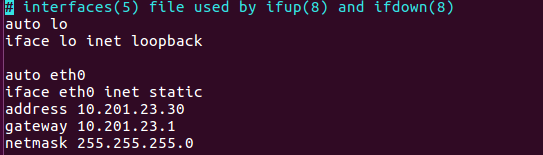
We have used 3 node cluster in our POC demo.

1. Total Number of nodes in the cluster: 3
2. IP address used:

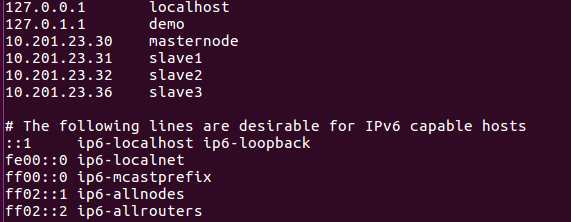
|  |  |
| --- | --- |
| IP address | Host Name |
| 10.201.23.30 | masternode |
| 10.201.23.31 | slave1 |
| 10.201.23.32 | slave1 |

**Following files need to be edit on 10.201.23.30 machine**

* 1. Edit: /etc/network/interfaces



* 1. Edit: /etc/hosts

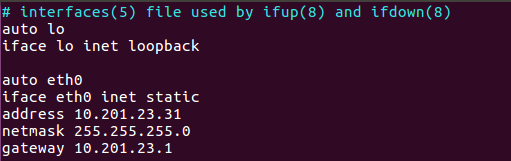


* 1. Edit: /etc/hostname

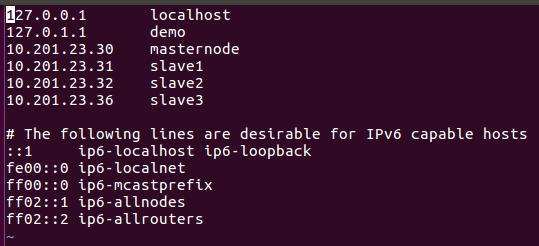


**Following files need to be edit on 10.201.23.31 machine**

1. Edit: /etc/network/interfaces



1. Edit /etc/hosts

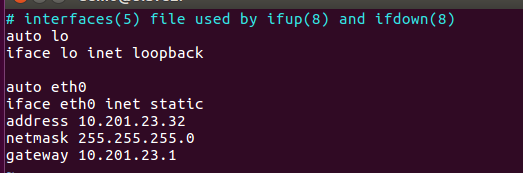


1. Edit: /etc/hostname

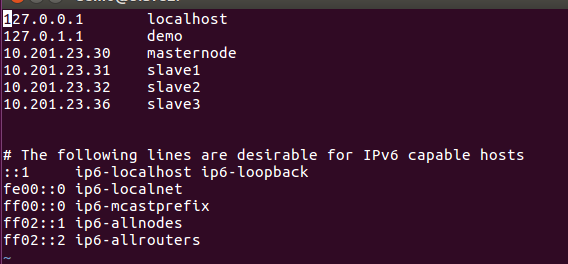


**Following files need to be edit on 10.201.23.32 machine**

1. Edit: /etc/network/interfaces



1. Edit /etc/hosts



1. Edit: /etc/hostname

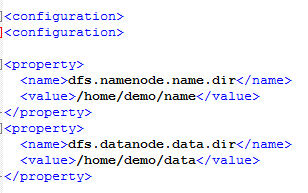


**Chapter 2: Hadoop Installation Guide**

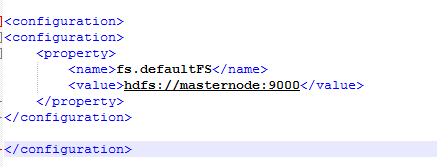
Note: Changes needs only on master

Hadoop Version: - 2.7.3

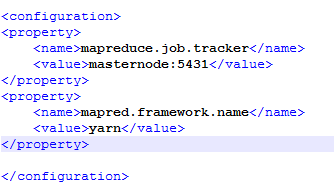
1. Download Hadoop 2.7.3
2. Copy to /opt/Hadoop
3. Create two folders in /home/demo/name and /home/demo/data
4. Edit: /usr/local/hadoop/etc/Hadoop/hdfs-site.xml



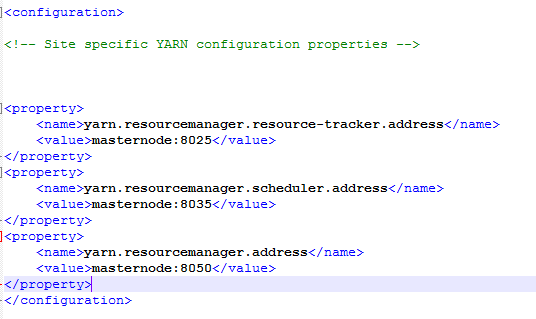
1. Edit: /usr/local/hadoop/etc/hadoop/core-site.xml



1. Edit: /usr/local/hadoop/etc/hadoop/mapred-site.xml



1. Edit: /usr/local/hadoop/etc/hadoop/yarn-site.xml



1. Edit : /usr/local/hadoop/etc/hadoop/masters

masternode

1. Edit: : /usr/local/hadoop/etc/hadoop/slaves

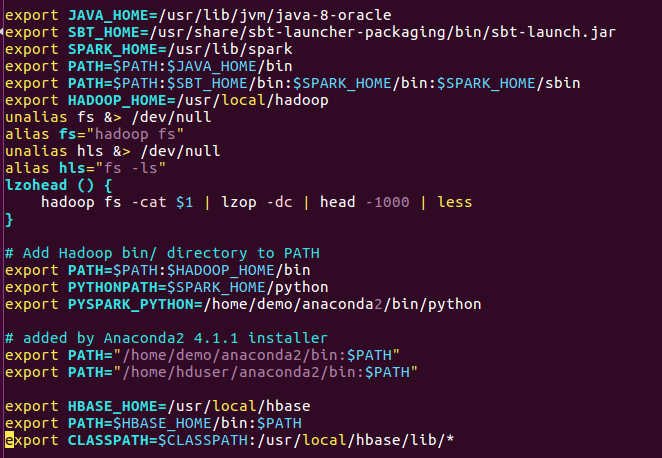
slave1

slave2

1. Edit: /usr/local/hadoop/etc/hadoop/hadoop-env.sh

export JAVA\_HOME=/usr/lib/jvm/java-8-oracle/jre

1. Edit: ~/.bashrc



1. Rsync to other nodes

sudo rsync -avxP /usr/local/hadoop/ hduser@slave1:/usr/local/hadoop/

sudo rsync -avxP /usr/local/hadoop/ hduser@slave2:/usr/local/hadoop/

Note: To see hadoop is running open web browser and check on 10.201.23.30:50070

**Chapter 3: Spark Installation Guide**

Note: Changes needs only on master

Spark Version: 2.1.0

1. Download spark 2.1.0
2. Copy it to /usr/lib/spark
   * 1. Edit: spark/conf/slaves

slave1

slave2

1. Edit: spark/conf/spark-env.sh

JAVA\_HOME=/usr/lib/jvm/java-8-oracle

SPARK\_WORKER\_MEMORY=3g

SPARK\_WORKER\_INSTANCES=1

SPARK\_MASTER\_IP=masternode

SPARK\_MASTER\_WEBUI\_PORT=8090

1. Edit: spark/conf/spark-defaults.conf

spark.master spark://masternode:7077

1. Rsync to other nodes

sudo rsync -avxP /usr/lib/spark/ hduser@slave1:/usr/local/spark/

sudo rsync -avxP /usr/lib/spark/ hduser@slave2:/usr/local/spark/

Commands:

1. To start
   1. cd /usr/lib/spark/sbin
   2. ./start-all.sh
2. To stop
   1. cd ./usr/lib/spark/sbin
   2. ./stop-all.sh

Note: To check spark is running properly, check it op 10.201.23.30:8090

**Chapter 4: HBASE Installation Guide**

Note: Changes needs only on master

HBASE version: **1.2.5**

1. Download hbase 1.2.5
2. copy hbase to /usr/local/hbase

sudo chown -R demo /usr/local/hbase

sudo chmod -R 755 /usr/local/hbase

sudo mkdir -p /var/hbase/pids

sudo chown -R demo /var/hbase/pids

sudo chmod -R 755 /var/hbase/pids

1. Edit hbase-env.sh in /usr/local/hbase/conf

export JAVA\_HOME=/usr/lib/jvm/java-8-oracle

export HBASE\_MANAGES\_ZK=true

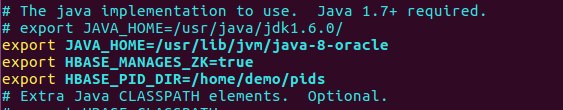
export HBASE\_PID\_DIR=/var/hbase/pids

1. Create the hbase and zookeeper folder in hbase

hdfs dfs -mkdir /hbase

hdfs dfs -mkdir /zookeeper

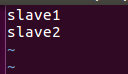
1. Edit hbase-env.sh



1. Edit hbase-site.xml



1. Edit regionservers



1. Rsync to other nodes:

scp -r /usr/local/hbase/\* demo@slave1:/usr/local/hbase

scp -r /usr/local/hbase/\* demo@slave2:/usr/local/hbase

Commands:

1. Start hbase:
   1. start-hbase.sh
2. Stop hbase:
   1. stop-hbase.sh
3. start hbase thrift server
   1. hbase thrift start

Note: To check hbase is running properly IP: 10.201.23.30:16010

**Chapter 5: SOLR Installation Guide**

Note: Changes needs only on master

1. Download solr version-6.5.1
2. Start the solr cloud
   1. ./bin/solr –e cloud
   2. It will ask the details for the configuration
   3. Default shard- 1
   4. Port – 8983
   5. Default replica- 1
   6. Collection Name- Spark\_NLP

To check solr is running properly:

1. Open in browser

http://localhost:8983/solr/Spark\_NLP/browse

**Chapter 6: LILY HBASE INDEXER Installation Guide**

Note: Changes needs only on master

Download the code: <https://github.com/NGDATA/hbase-indexer>

1. Go to hbase-indexer
2. Build the project:
   1. mvn clean install –DskipTests

Note: During build you may get proxy error:

hbase-indexer setup error:

Proxy setting in /etc/maven/settings.xml

<proxies>

<proxy>

<id>optional</id>

<active>true</active>

<protocol>http</protocol>

<username>pr349331</username>

<password>P\_9860195316</password>

<host>10.201.51.54</host>

<port>8080</port>

</proxy>

</proxies>

1. Start the lily hbase indexer
   1. ./bin/hbase-indexers server

NOTE: For lily hbase indexer: The hbase-site.xml must contains the following properties:

<property> <name>hbase.replication</name> <value>true</value> </property> <property> <name>replication.source.ratio</name> <value>1.0</value> </property> <property> <name>replication.source.nb.capacity</name> <value>1000</value> </property> <property> <name>replication.replicationsource.implementation</name> <value>com.ngdata.sep.impl.SepReplicationSource</value> </property>

**Chapter 7: Running the spark first time with NLP code**

Start NLP Program steps:

1.1 Changes needs only on master

1. start hadoop, spark and create data folder, zookeeper folder, and hbase folder in hdfs

* cd /usr/local/hadoop/sbin/
* hadoop namenode –format [1st time]
* ./start-dfs.sh
* ./start-yarn.sh
* cd /usr/lib/spark/sbin
* ./start-all.sh
* hadoop fs -mkdir /user
* hadoop fs -mkdir /user/demo
* hadoop fs -mkdir /user/demo/input
* hadoop fs -mkdir /user/demo/output
* hadoop fs -mkdir /hbase
* hadoop fs -mkdir /zookeeper

Note: Directly run the startSparkNlpServices.sh to start the services. But it will not create the folders in hdfs, do it manually.

1. start hbase
   1. start-hbase.sh
2. create table in hbase in hbase shell

Open: **hbase shell** in the terminal

create ‘spark\_nlp\_features’,

{NAME => 'file\_name', REPLICATION\_SCOPE => ‘1’},

{NAME => 'file\_type', REPLICATION\_SCOPE => ‘1’},

{NAME => 'header', REPLICATION\_SCOPE => ‘1’},

{NAME => 'nlp\_pos', REPLICATION\_SCOPE => ‘1’},

{NAME => 'nlp\_noun\_chunks', REPLICATION\_SCOPE => ‘1’},

{NAME => 'nlp\_ner', REPLICATION\_SCOPE => ‘1’},

{NAME => 'nlp\_triples', REPLICATION\_SCOPE => ‘1’},

{NAME => 'nlp\_keyterm', REPLICATION\_SCOPE => ‘1’},

{NAME => 'nlp\_keyterms\_grouping', REPLICATION\_SCOPE => ‘1’},

{NAME => 'nlp\_lexico\_semantics', REPLICATION\_SCOPE => ‘1’},

{NAME => 'nlp\_multiwords\_expression', REPLICATION\_SCOPE => ‘1’},

{NAME => 'raw\_data', REPLICATION\_SCOPE => ‘1’},

{NAME => ‘page\_number’, REPLICATION\_SCOPE => ‘1’},

{NAME => ‘table\_data’, REPLICATION\_SCOPE => ‘1’},

{NAME => ‘nlp\_semantic\_roles’, REPLICATION\_SCOPE => ‘1’}

1. Create core in solr Spark\_NLP
   1. Go to solr bin folder
   2. ./solr create –c Spark\_NLP
2. Create schema(by adding columns) in core Spark\_NLP
   1. nlp\_noun\_chunks -> datatype- text\_general
   2. nlp\_ner -> datatype – text\_general
   3. nlp\_triples -> datatype- text\_general
   4. nlp\_multiwords\_expression -> datatype- text\_general
   5. raw\_data -> datatype- - text\_general
   6. header -> datatype- - text\_general
   7. file\_name -> datatype- - text\_general
   8. page\_number -> datatype- - text\_general
   9. table\_data -> datatype- - text\_general
3. Create SPARK\_NLP.xml file

<?xml version="1.0"?>

<indexer table="spark\_nlp\_features">

<field name="file\_name" value="file\_name:"/>

<field name="header" value="header:"/>

<field name="raw\_data" value="raw\_data:"/>

<field name="page\_number" value="page\_number:"/>

<field name="table\_data" value="table\_data:"/>

<field name="nlp\_noun\_chunks" value="nlp\_noun\_chunks:"/>

<field name="nlp\_ner" value="nlp\_ner:"/>

<field name="nlp\_triples" value="nlp\_triples:"/>

<field name="nlp\_multiwords\_expression" value="nlp\_multiwords\_expression:"/>

</indexer>

1. add indexer

* Copy the SPARK\_NLP.xml file in bin.
* Go to hbase indexer bin folder (/usr/local/hbase-indexer-master/bin)
* Add the indexer using command:

./hbase-indexer add-indexer -n spark\_nlp\_indexer -c SPARK\_NLP.xml -cp solr.zk=localhost:9983 -cp solr.collection=Spark\_NLP

1. Start the indexer

./bin/hbase-indexer server

1. start the hbase server thrift

hbase thrift start

**Chapter 8: Installing Tomcat**

1. Download <http://apache.mirrors.ionfish.org/tomcat/tomcat-8/v8.5.16/bin/apache-tomcat-8.5.16.tar.gz>
2. Create /opt/tomcat folder

sudo mkdir /opt/tomcat

1. Untar tomcat

sudo tar xzvf apache-tomcat-8\*tar.gz -C /opt/tomcat --strip-components=1

1. Go to /opt/tomcat

cd /opt/tomcat

1. change chmod of conf and bin

sudo chmod 755 conf

sudo chmod 755 bin

1. go to bin folder and change file permission to 777

sudo chmod 777 startup.sh

1. Edit conf/tomcat-users.xml

<role rolename="tomcat"/>

<role rolename="manager-gui"/>

<user username="tomcat" password="tomcat123" roles="tomcat,manager-gui"/>

1. Copy the RegionExtractor.war file to /opt/tomcat/webapps and restart the tomcat.

**Chapter 9: Installing pdftable library**

**Installing pdftable library:**

Version: 0.0.4

Change pdftable.py file

IS\_TABLE\_COLUMN\_COUNT\_THRESHOLD = 1

IS\_TABLE\_ROW\_COUNT\_THRESHOLD = 1

**Chapter 10: Start the services and run the code**

**Starting Spark NLP**

1. **Starting Hadoop:**
   1. **cd /usr/local/hadoop/sbin/"**
   2. **./start-dfs.sh**
   3. **./start-yarn.sh**
2. **Starting Spark** 
   1. **cd /usr/lib/spark/sbin**
   2. **./start-all.sh**
3. **Starting HBASE**
   1. **start-hbase.sh**

**Note: To run 1, 2 and 3 services you can directly run startSparkNlpServices.sh file in the home directory /home/demo/startSparkNlpServices.sh**

1. **Staring hbase thrift server**
   1. **hbase thrift start**
2. **Starting hbase lily indexer**
   1. **cd /usr/local/hbase-indexer-master**
   2. **./bin/hbase-indexer server**
3. **Starting header text merger service**
   1. **cd /home/demo/SparkNLP\_Dev/src/HeaderTextMerger**
   2. **python HeaderTextMerger\_RES.py**
4. **Starting tomcat/RegionExtractor**
   1. **cd /opt/tomcat/bin**
   2. **sudo ./startup.sh**
5. **Starting solr service** 
   1. **cd /home/demo/solr/bin**
   2. **./solr -e cloud**
      1. **how many Solr nodes would you like to run in your local cluster? (specify 1-4 nodes): 1**
      2. **Please enter the port for node1 [8983]: ENTER**
      3. **Please provide a name for your new collection: [gettingstarted]: Spark\_NLP**
      4. **Do you want to re-use the existing collection or create a new one? Enter 1 to reuse, 2 to create new [1]: 1**

**Chapter 11: Adding a new field in HBASE, SOLR, and Lily indexer:**

* + 1. **Start solr, hbase, lily hbase indexer**
       1. **For solr**
          1. **Open localhost:8983**
          2. **Select the core**

**Add the new column name and type (Ex: column\_name: Page\_Number, type: text\_general )**

* + - 1. **For Hbase**
         1. **Delete the old table**

**E.x to update the new Table\_Name in SPARK\_NLP1**

**disable “SPARK\_NLP1”**

**drop “SPARK\_NLP1”**

**Create the table again:**

**create 'SPARK\_NLP1', { NAME => 'FILE\_NAME', REPLICATION\_SCOPE => '1' },{ NAME => 'FILE\_TYPE', REPLICATION\_SCOPE => '1' },{ NAME => 'HEADER', REPLICATION\_SCOPE => '1' },{ NAME => 'processed\_data\_pos', REPLICATION\_SCOPE => '1' },{ NAME =>** **'processed\_data\_noun\_chunks', REPLICATION\_SCOPE => '1' },{ NAME => 'processed\_data\_ner', REPLICATION\_SCOPE => '1' },{ NAME => 'processed\_data\_triples', REPLICATION\_SCOPE => '1' },{ NAME => 'processed\_data\_keyterm', REPLICATION\_SCOPE => '1' },{ NAME => 'processed\_data\_keyterms\_grouping', REPLICATION\_SCOPE => '1' },{ NAME => 'processed\_data\_lexico\_semantics', REPLICATION\_SCOPE => '1' },{ NAME => 'processed\_data\_multiwords\_expression',REPLICATION\_SCOPE => '1' },{ NAME => 'RAW\_DATA', REPLICATION\_SCOPE => '1' },{ NAME => 'Page\_Number', REPLICATION\_SCOPE => '1'}**

* + - 1. **For Lily Indexer**
         1. **Go to /usr/local/hbase-indexer-master/bin**
         2. **Edit the Spark\_NLP.xml file**

**Add the <field name="Page\_Number" value="Page\_Number:"/>**

* + - * 1. **Delete the old indexer**

**./hbase-indexer list-indexers**

**./hbase-indexer delete-indexer -n SPARK\_NLP\_INDEXER**

**Create the indexer again**

**./hbase-indexer add-indexer -n SPARK\_NLP\_INDEXER -c SPARK\_NLP.xml -cp solr.zk=localhost:9983 -cp solr.collection=Spark\_NLP**

**Chapter 12: Clear Spark and HBASE**

For Solr:

1. To delete the contents from the table Spark\_NLP

[http://localhost:8983/solr/Spark\_NLP/update?stream.body=%3Cdelete%3E%3Cquery%3E\*:\*%3C/query%3E%3C/delete%3E&commit=true](http://localhost:8983/solr/Spark_NLP/update?stream.body=%3Cdelete%3E%3Cquery%3E*:*%3C/query%3E%3C/delete%3E&commit=true)

1. To truncate the HBASE table spark\_nlp\_features
   1. Open hbase shell in the terminal
      1. hbase shell
   2. truncate spark\_nlp\_features