# LDSA Assignment 2

**Uvais Karni**

## Part-1

### Task 1.1:

### 4)

a) The files found in the output folder are '_SUCCESS' and 'part-r-00000'. The file '_SUCCESS' represents the successful execution of the WordCount.java and generation of Output file 'part-r-00000' which contains words counts generated from the Input file '20417.txt.utf-8'.

b) The word 'Discovery' appears 5 times in the text analysed.

c) In Local Mode everything is run on the local filesystem, running in local makes it faster. In this NameNode Or DataNode are not used. It is easier to fix error as everything is present in the local file system. Meanwhile in pseudo-distributed mode a single node acts as a distributed system of DataNodes and NameNodes. It is similar to running Hadoop on distributed nodes but in this case it is virtualised in a single node.

### Task 1.2:

### 3)

a) Both 'core-site.xml' and 'hdfs-site.xml' are configuration files. The config file 'core-site.xml' is used to configure input and output settings such as what localhost can make requests. The config file 'hdfs-site.xml' is used to configure settings foe name node and data nodes.

b) The roles listed after 'jps' are:

27557 Jps

27317 SecondaryNameNode

26861 NameNode

27070 DataNode

**NameNode:** It is also called master. Its responsible for managing all the slaves or DataNode. It keeps track of all data and also its manipulation. It is important component of HDFS but does not store actual data.

**DataNode:** It is also called slave. It where the data is stored and processed. Multiple DataNode are present, which are controlled by NameNode.

**SecondNameNode:** As the name denotes it's a secondary NameNode that helps NameNode by maintaining checkpoints. In case of failure it allows recovery of metadata.

**Jps:** It stands for Java Virtual Machine Process Status Tool. It is used to check what daemons are running.

**Task 1.3:**

**6)**

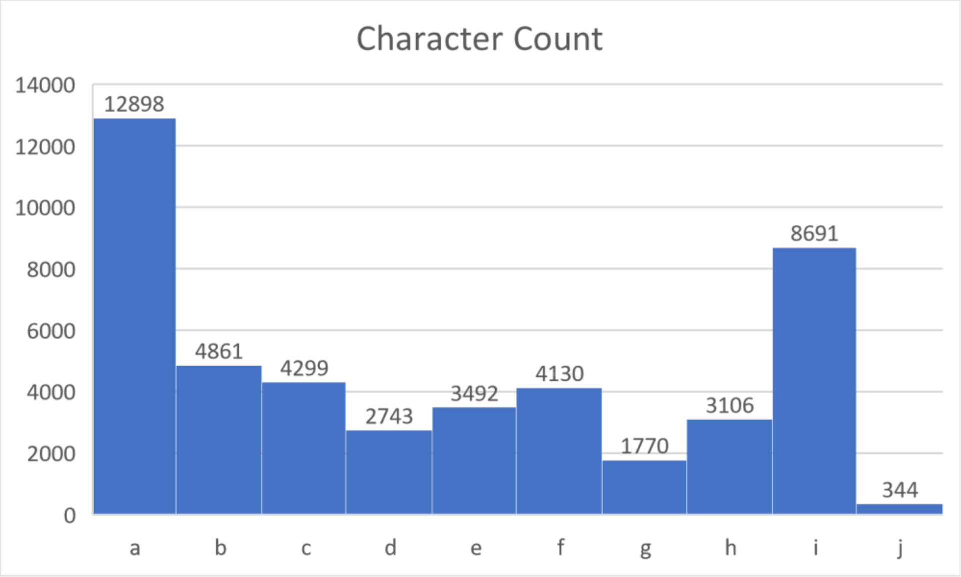
a) TokenizerMapper: It's allows tokenized words from string to be assigned to a key value pair. TokenizerMapper class is called for every word. The output is key value pair with key and value one.

IntSumReducer: It's allows key pair with same keys to be combined together to denote the reduce step. The IntSumReducer class is called for every Key. The output is key value pair with key and value of total occurrence in the list of inputs.

b) HDFS stand for Hadoop Distributed File System, it's a distributed file system that's a allows files to be stored and manipulated across different nodes. It allows storing of bulk data, can handle 1000 of nodes, computes data over distributed nodes parallelly which allows high throughput but also can work in system that is not optimised for high throughput. HDFS has a large block size of 64-128 MB which is large compared to 512 bytes in regular system. In HDFS the block size can be altered unlike in normal file system. In HDFS files are also replicated and stored in multiple nodes ,to prevent loss of data in case of failure.

**Task 1.4:**

**2)** Plot for first 10 characters only:



**Task 2.1:**

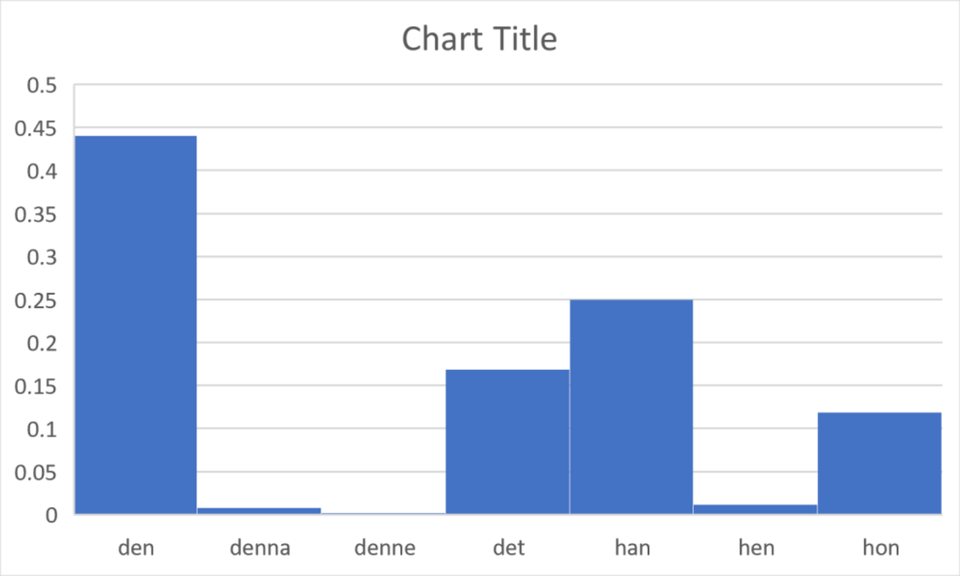
**1)**

a) JSON format tweets can be classified as semi-structured data. In order to be classified as structured it should follow a formal structure and it does not even though it has tags with values example tag text with the tweet texts.

b) RDBMS does not supports storing and processing of unstructured data while HDFS supports. Due to the data being stored over multiple nodes HDFS has a throughput and low latency than RDBMS.

Scaling out is supported by HDFS that is adding more machines to the cluster unlike Scaling up that is adding resources to the machine which is supported by RDMS (Scaling out is better than Scaling up).

**3)** Plot for Pronouns Count:



**Part-2**

**Task 1.1**

**1)** SQL are Relational database like RDBMS, Amazon DynamoDB etc. NoSQL are Non-Relational or Distributed database like MongoDB, Oracle etc.

Pros of SQL:

- It has secure implementation which govern by schema-based rules and thus preferred by banks.
- It has good support for handling queries.

Cons of SQL:

- SQL can only store and process structured and semi-structured data, not unstructured data.
- It prefers Normalised data (It adds another processing Overhead)
- It supports Scaling up, which is expensive.

Pros of NoSQL:

- NoSQL can store and process all types structured, semi-structured and even unstructured data.

- It supports Scaling out, which is less- expensive as more resources can be added when required.
- Supports hierarchical data storage.

Cons of NoSQL:

- It does have built-in features like ACID properties, which makes it less secure.
- They are less stable compared to SQL.

Scenario for SQL:

Banks use SQL Database because it provides features like ACID, it has roll back to revert mishaps in case in-complete transaction which is crucial. It uses table-based schema which allows for easy setup and debugging.

Scenario for NoSQL:

A good scenario for NoSQL is when there is a need to store and process data with high variety, velocity and volume. These are the properties of Big Data and companies like Facebook, Google use NoSQL. It allows for any type of data to be saved which makes it flexible. It is also cheap due to scaling out their fore resources can be added when required.

**3)** The MongoDB implementation is like the one in Map reduce. First, I read the files in local and moved it to the Mongo Database. In Mongo Db we read each tweet parsed from json to key and value like pair. The file of implementation is attached below.