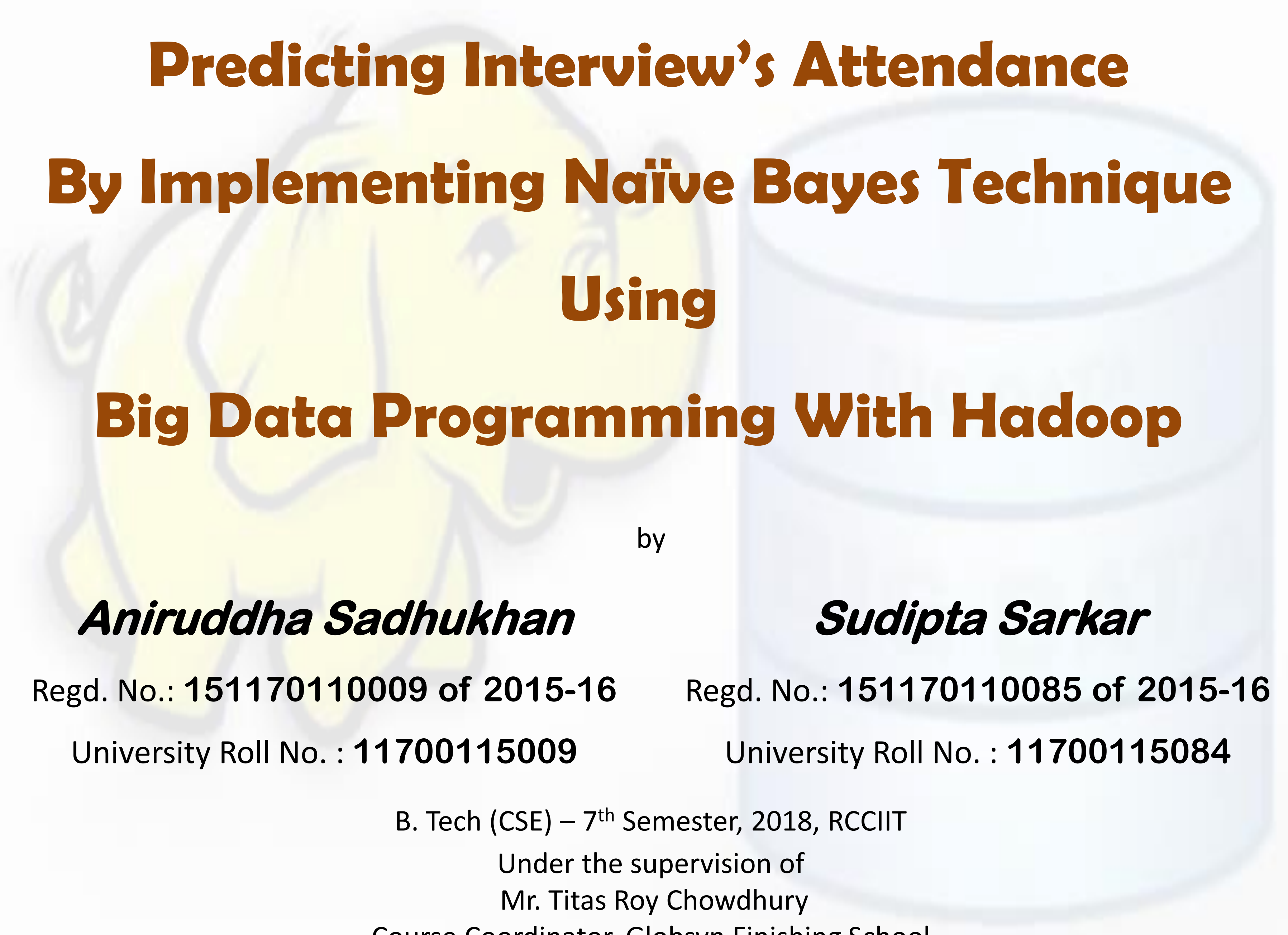


Predicting Interview's Attendance By Implementing Naïve Bayes Technique Using Big Data Programming With Hadoop

by

Aniruddha Sadhukhan

Regd. No.: **151170110009 of 2015-16**

University Roll No. : **11700115009**

Sudipta Sarkar

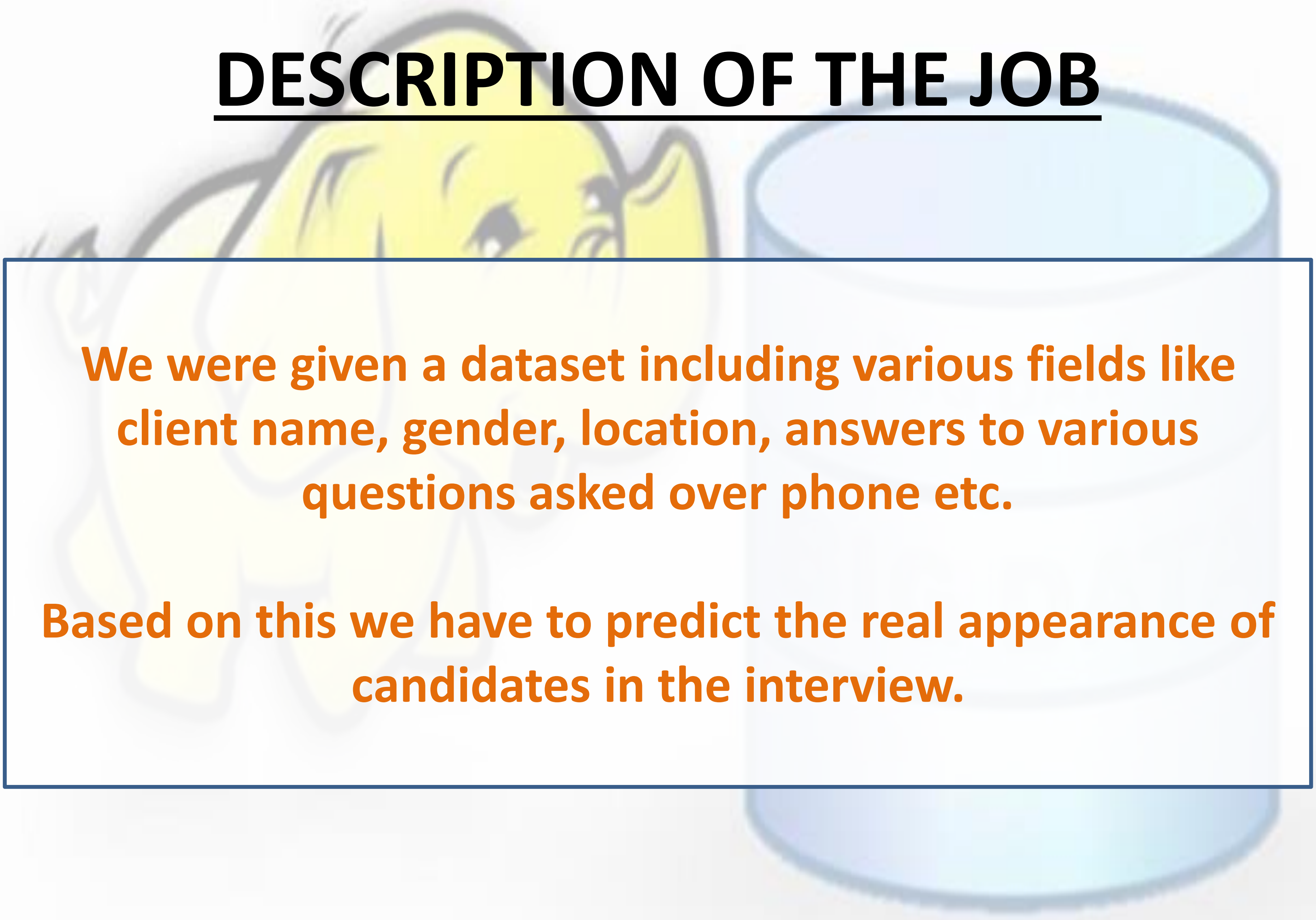
Regd. No.: **151170110085 of 2015-16**

University Roll No. : **11700115084**

B. Tech (CSE) – 7th Semester, 2018, RCCIIT

Under the supervision of
Mr. Titas Roy Chowdhury
Course Coordinator, Globsyn Finishing School

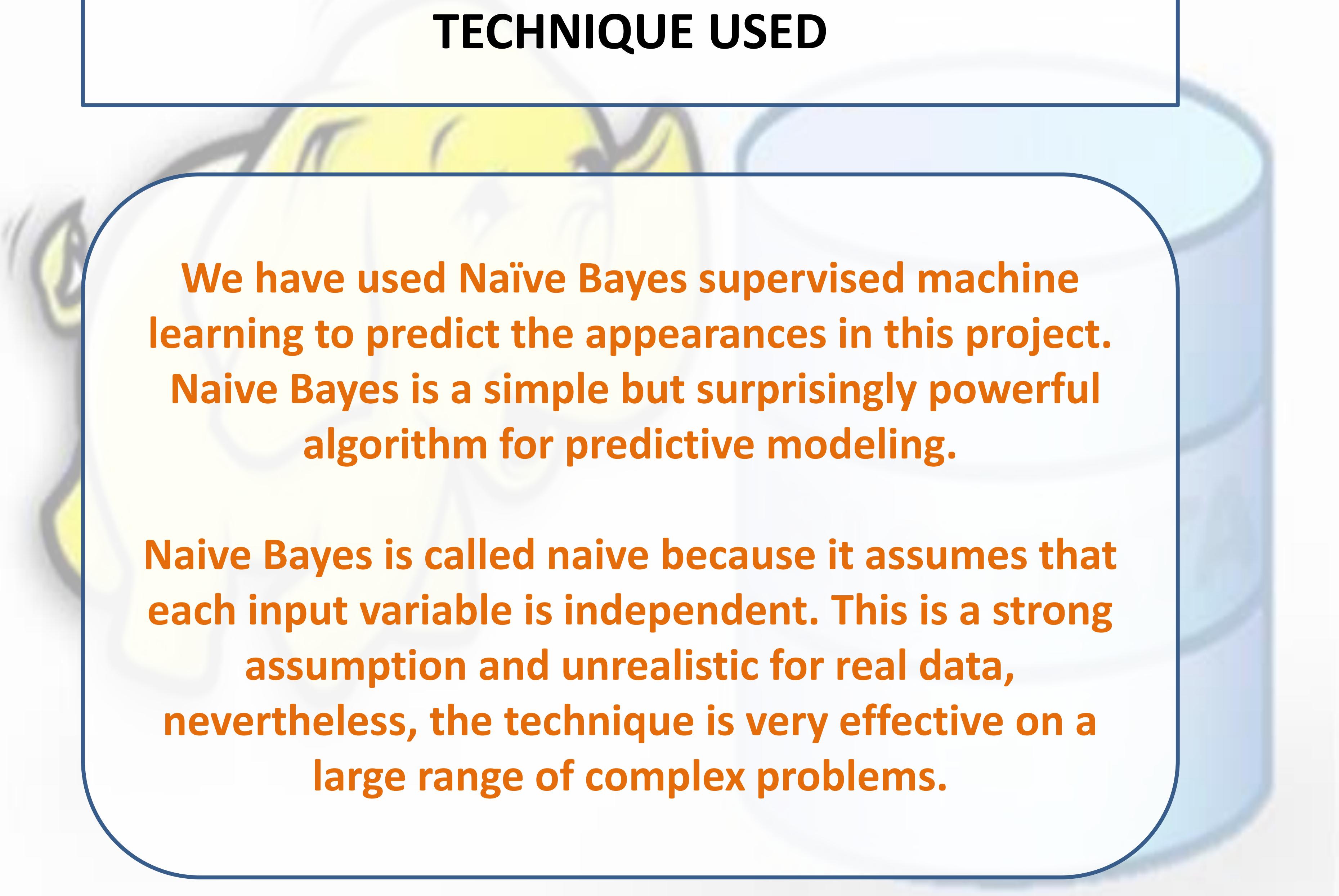
DESCRIPTION OF THE JOB



We were given a dataset including various fields like client name, gender, location, answers to various questions asked over phone etc.

Based on this we have to predict the real appearance of candidates in the interview.

TECHNIQUE USED



We have used Naïve Bayes supervised machine learning to predict the appearances in this project. Naive Bayes is a simple but surprisingly powerful algorithm for predictive modeling.

Naive Bayes is called naive because it assumes that each input variable is independent. This is a strong assumption and unrealistic for real data, nevertheless, the technique is very effective on a large range of complex problems.

$$\underbrace{P(c \mid x)}_{\text{Posterior Probability}} = \frac{\overbrace{P(x \mid c)}^{\text{Likelihood}}\,\overbrace{P(c)}^{\text{Class Prior Probability}}}{\underbrace{P(x)}_{\text{Predictor Prior Probability}}}$$

$$P(c \mid \mathrm{X}) = P(x_1 \mid c) \times P(x_2 \mid c) \times \cdots \times P(x_n \mid c) \times P(c)$$

Hardware and software requirements

HADOOP CLUSTER IS USED WHICH IS MADE OF COMMODITY HARDWARE

Master's recommendation

4–6 1TB hard disks
(1 for OS [RAID 1], 2 for the FS image [RAID 5/6],
1 for JournalNode)

2 CPUs(8-12 cores per CPU), running at least
2-2.5GHz

128-512GB of RAM

Bonded Gigabit Ethernet or 10Gigabit Ethernet

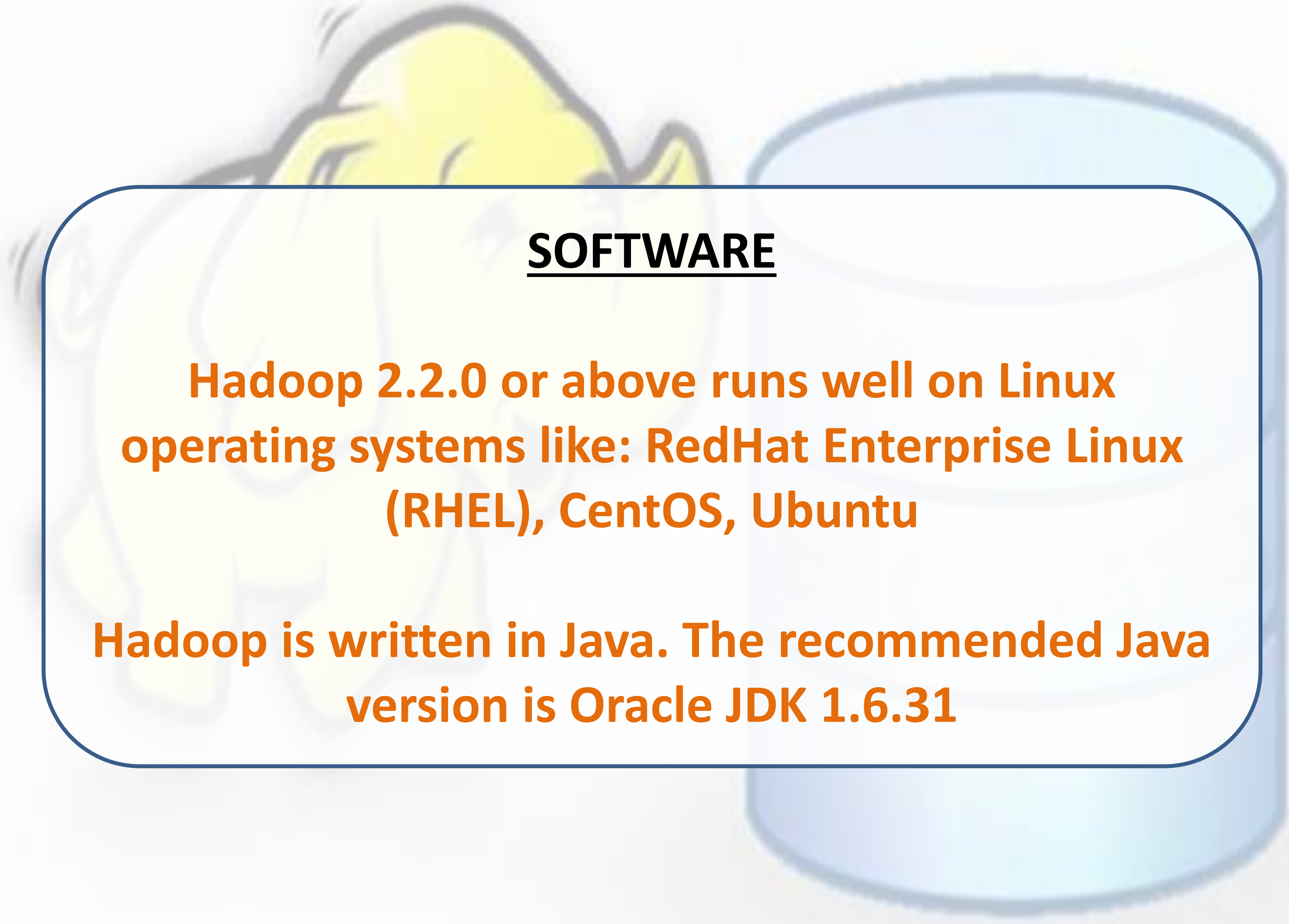
SLAVE’S RECOMMENDATION

12-24 1-4TB hard disks in a JBOD (Just a Bunch Of Disks) configuration

2 CPUs(8-12 cores per CPU), running at least 2-2.5GHz

64-128GB of RAM

Bonded Gigabit Ethernet or 10Gigabit Ethernet



SOFTWARE

Hadoop 2.2.0 or above runs well on Linux operating systems like: RedHat Enterprise Linux (RHEL), CentOS, Ubuntu

Hadoop is written in Java. The recommended Java version is Oracle JDK 1.6.31

SOURCE AND DESCRIPTION OF DATA

❑ THE DATA HAS BEEN PROVIDED BY THE INSTITUTE

❑ THE DATA CONSISTS OF 23 DISCRETE SPECIFICATIONS WHICH INCLUDES 1234 NUMBER OF RECORDS

❑ TYPES OF SPECIFICATIONS:

- DATE OF INTERVIEW-The date on which the interview is going to take place
- CLIENT NAME-Name of the company
- INDUSTRY-The type of industry concerned (Electronics' 'Telecom' 'IT')
- LOCATION-Location of the company ('Delhi' 'chennai‘)
- POSITION TO BE CLOSED-The post offered(Production- Sterile' 'Selenium testing‘)
- NATURE OF SKILL SET-skill required for the job('Routine' 'Oracle')
- INTERVIEW TYPE-Type of interview(Walk in, scheduled, scheduled walk in)
- NAME-Candidate Id
- GENDER
- CANDIDATE CURRENT LOCATION-Present address of the candidate
- CANDIDATE JOB LOCATION-Where the candidate will be working
- INTERVIEW VENUE
- CANDIDATE NATIVE LOCATION

- **HAVE YOU OBTAINED THE NECCESSARY PERMISSION TO START AT THE REQUIRED TIME-**
  - May be yes , no or not yet
- **HOPE THERE WILL BE NO UNSCHEDULED MEETINGS- TIME-**May be yes , no or not yet
- **CAN I CALL YOU THREE HOURS BEFORE THE INTERVIEW AND FOLLOW UP ON YOUR ATTENDANCE FOR THE INTERVIEW- TIME-**
  - May be yes , no
- **CAN I HAVE AN ALTERNATIVE NUMBER/DESK NUMBER-**
  - Yes or no
- **HAVE YOU TAKEN A PRINT OUT OF YOUR UPDATED RESUME-**
  - yes or no or NA
- **ARE YOU CLEAR WITH THE VENUE DETAILS AND THE LANDMARK-**

Yes or no or need to check or not sure

- **HAS THE CALL LETTER BEEN SHARED-**
  - Yes or no or haven't checked
- **EXPECTED ATTENDANCE-**
  - uncertain, yes or no
- **OBSERVED ATTENDANCE-**
  - yes or no
- **MARITAL STATUS-**
  - single or married

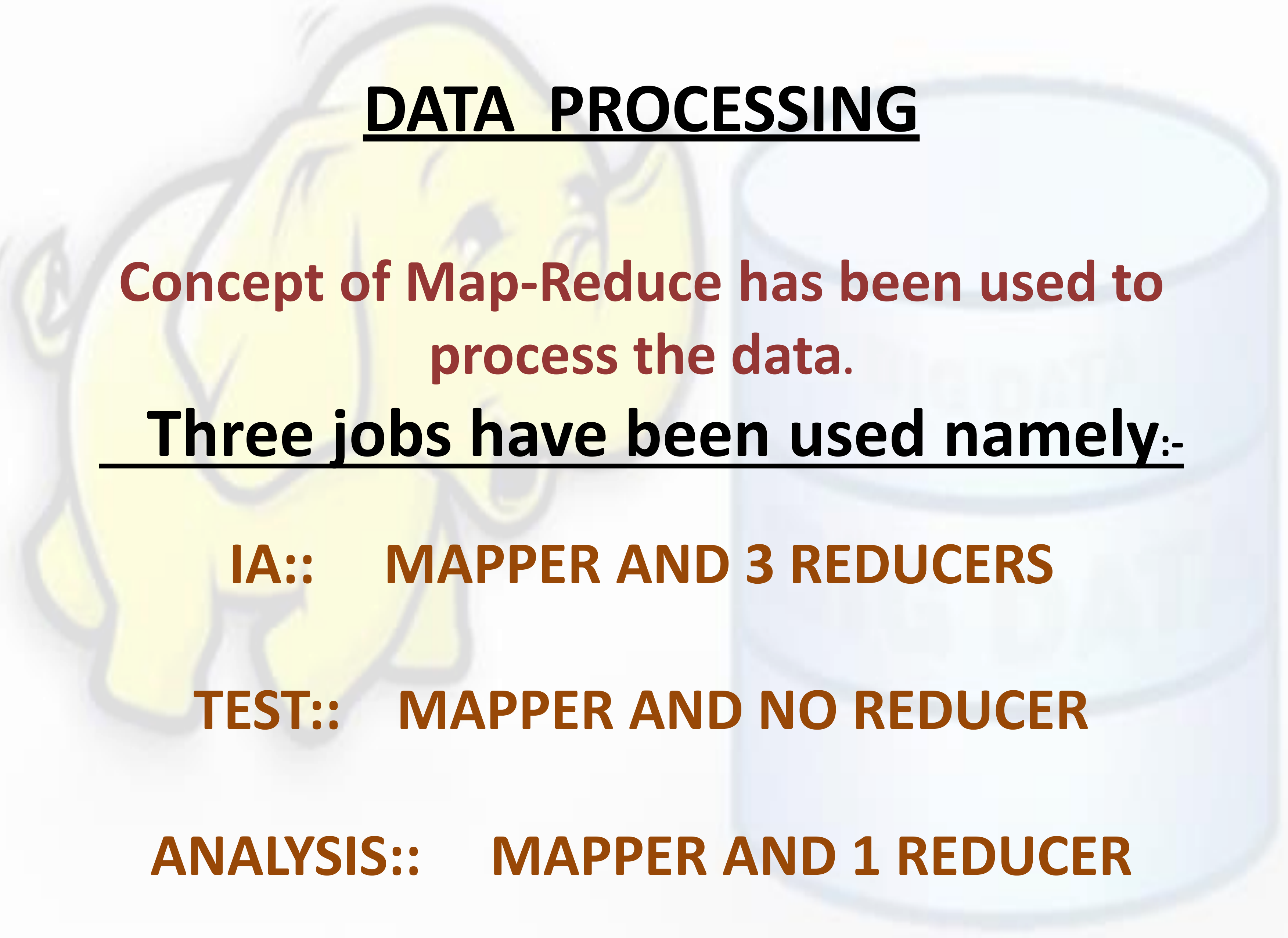
DATA CLEANSING

- Rows in which some data were missing or all the 23 data weren’t present are ignored

- Some data included incorrect punctuation marks or non word characters and are rectified

- Some values were nearly similar(e.g: liaison and liabilities, liaison & liabilities, liaison liabilities)and were changed to a single value

- Rows where some discrepancy in data was there were rectified



DATA PROCESSING

Concept of Map-Reduce has been used to process the data.

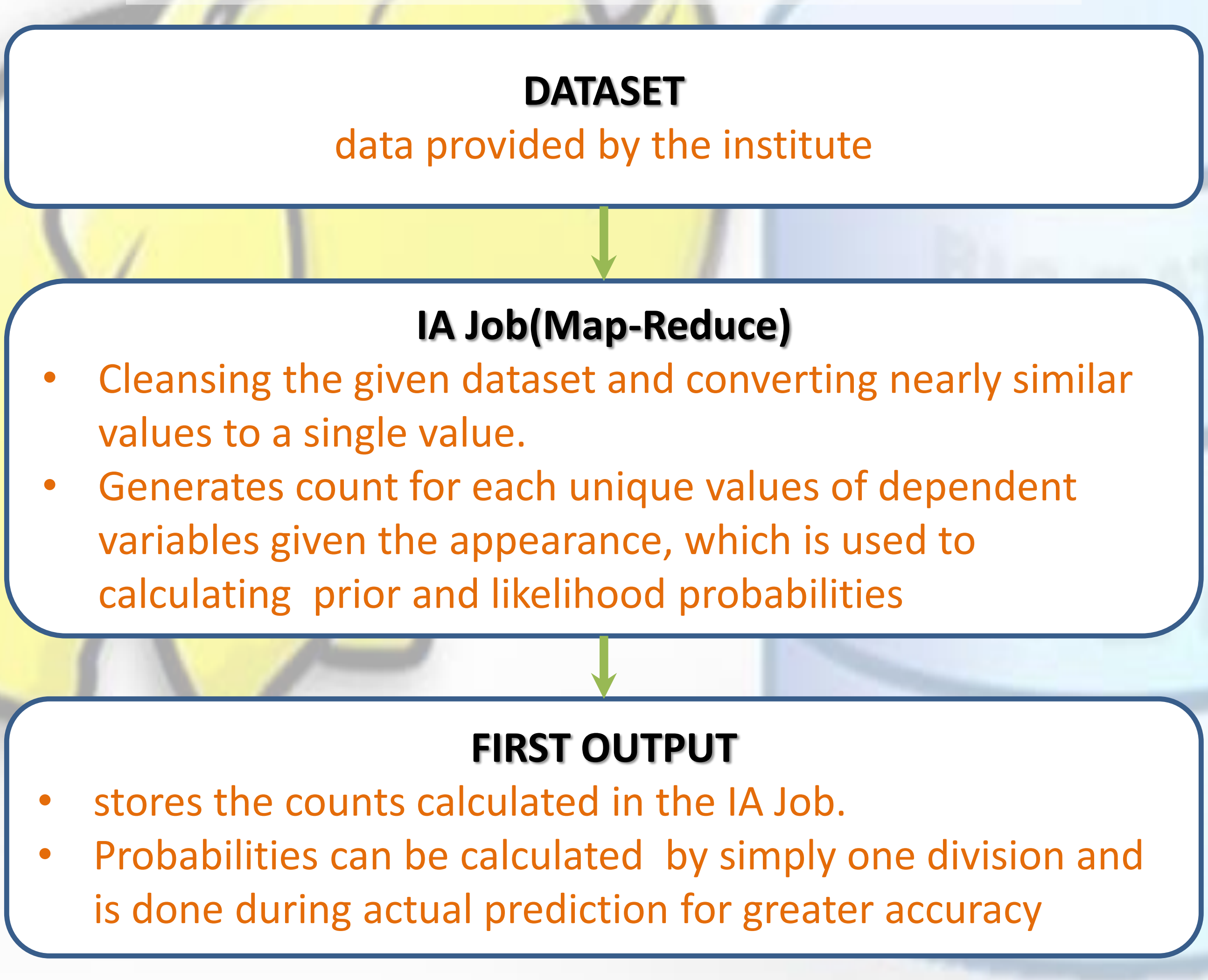
Three jobs have been used namely:-

IA:: MAPPER AND 3 REDUCERS

TEST:: MAPPER AND NO REDUCER

ANALYSIS:: MAPPER AND 1 REDUCER

Work Flow Diagrams



DATASET

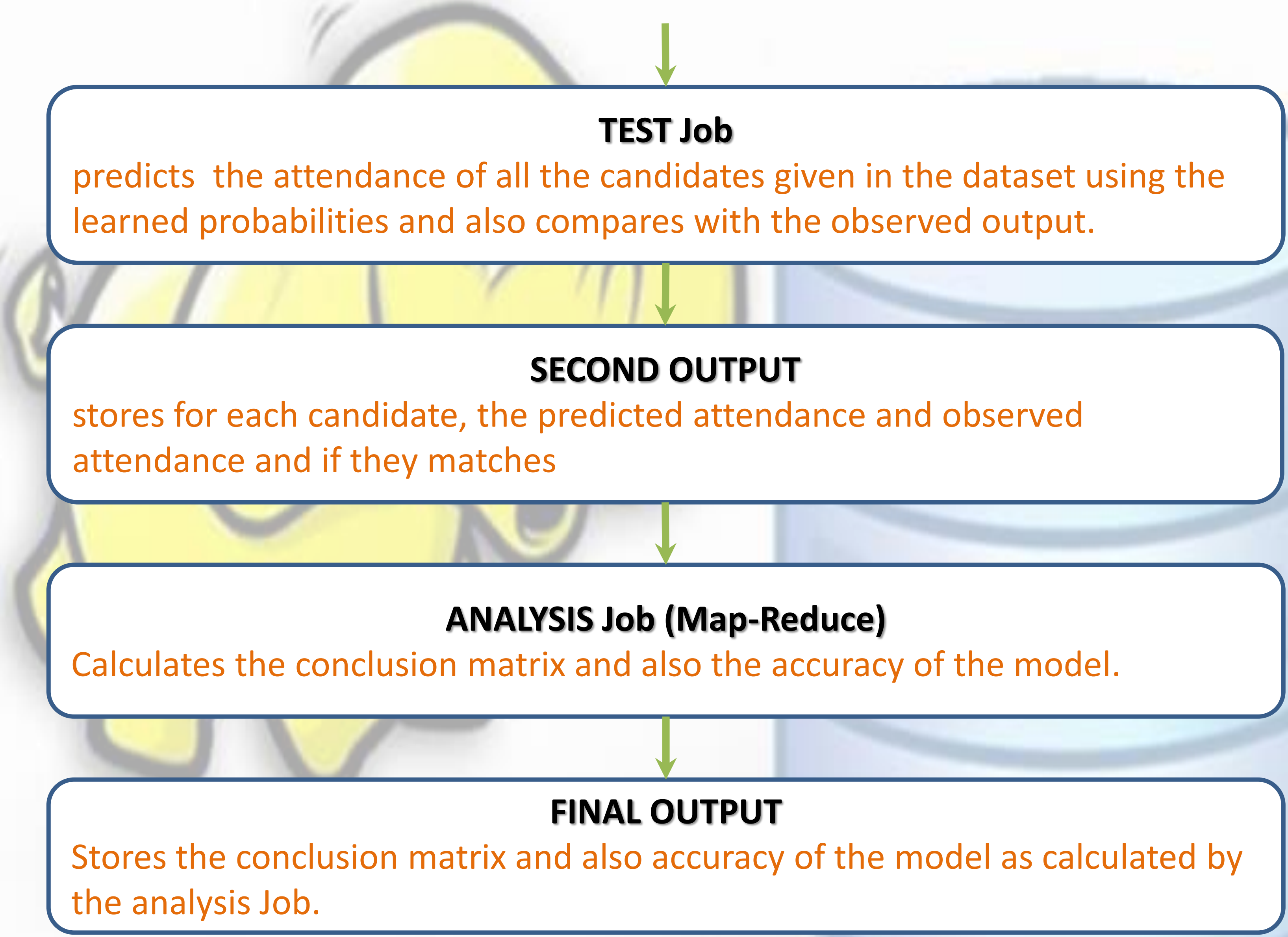
data provided by the institute

IA Job(Map-Reduce)

- Cleansing the given dataset and converting nearly similar values to a single value.
- Generates count for each unique values of dependent variables given the appearance, which is used to calculating prior and likelihood probabilities

FIRST OUTPUT

- stores the counts calculated in the IA Job.
- Probabilities can be calculated by simply one division and is done during actual prediction for greater accuracy



TEST Job

predicts the attendance of all the candidates given in the dataset using the learned probabilities and also compares with the observed output.

SECOND OUTPUT

stores for each candidate, the predicted attendance and observed attendance and if they matches

ANALYSIS Job (Map-Reduce)

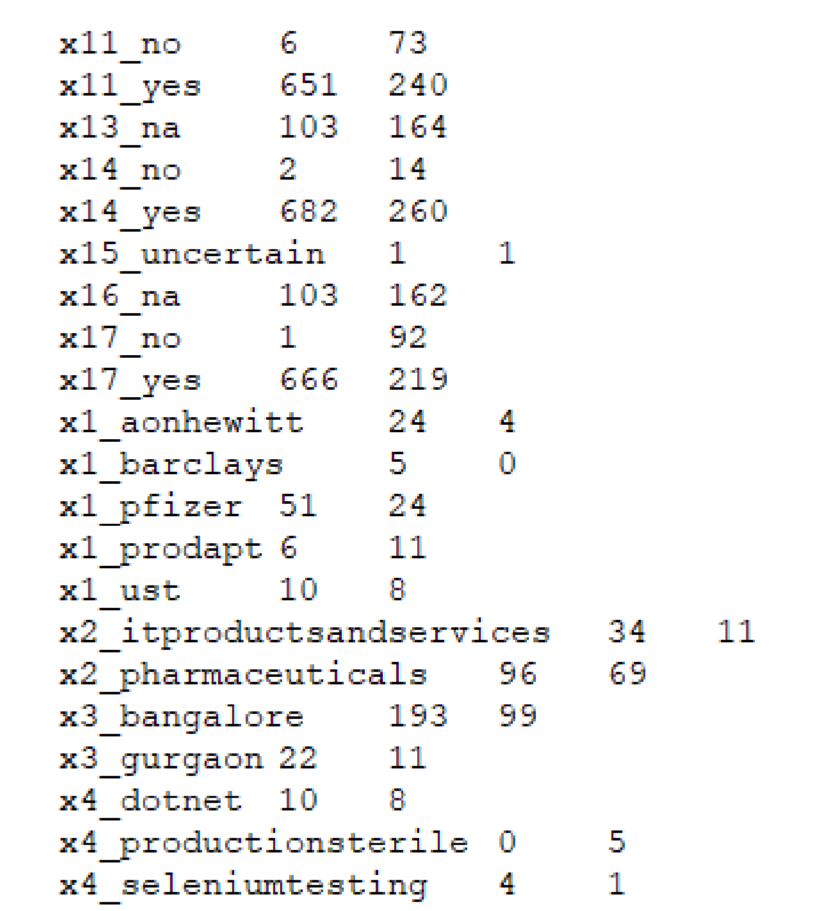
Calculates the conclusion matrix and also the accuracy of the model.

FINAL OUTPUT

Stores the conclusion matrix and also accuracy of the model as calculated by the analysis Job.

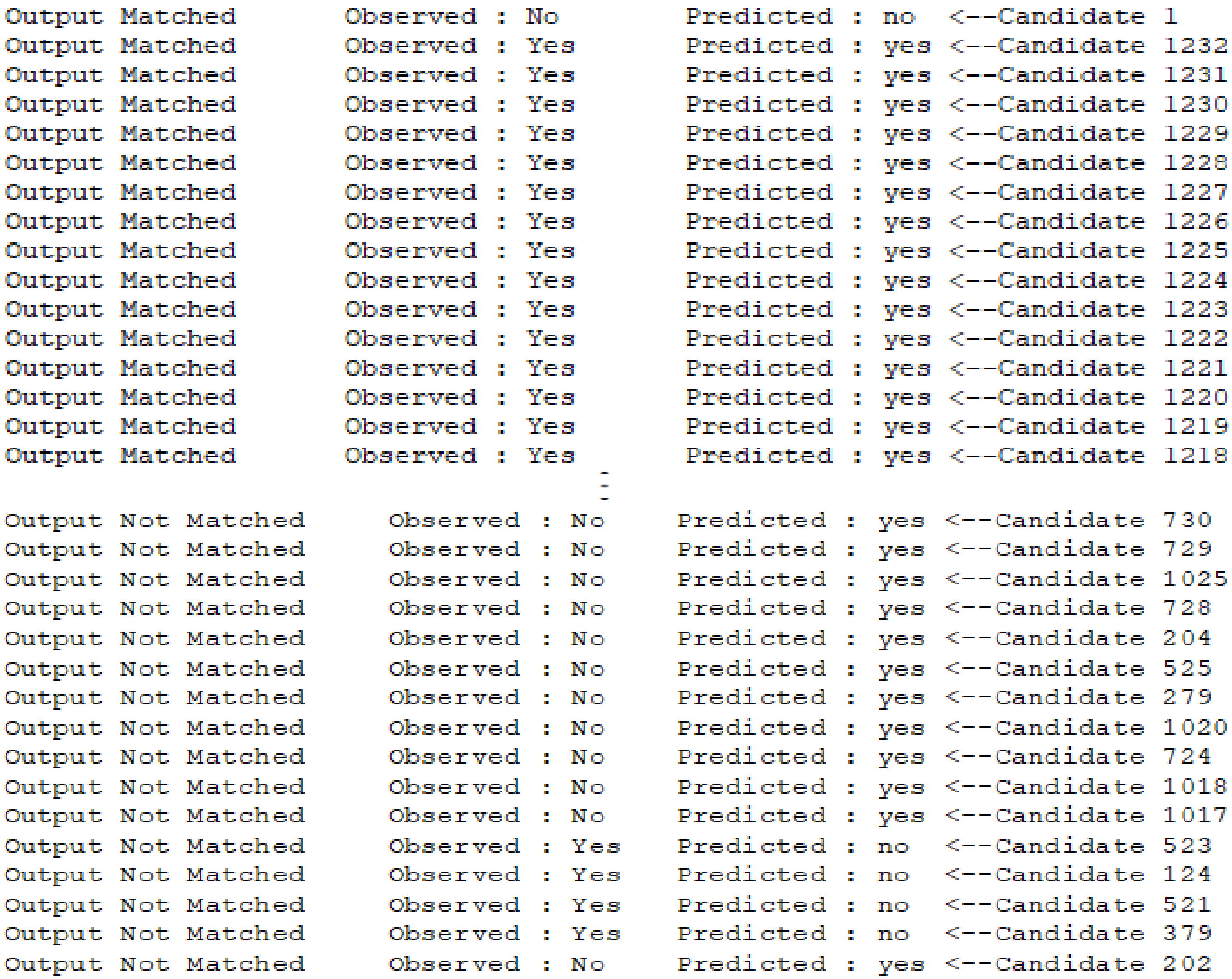
OUTPUT

Output of IA job



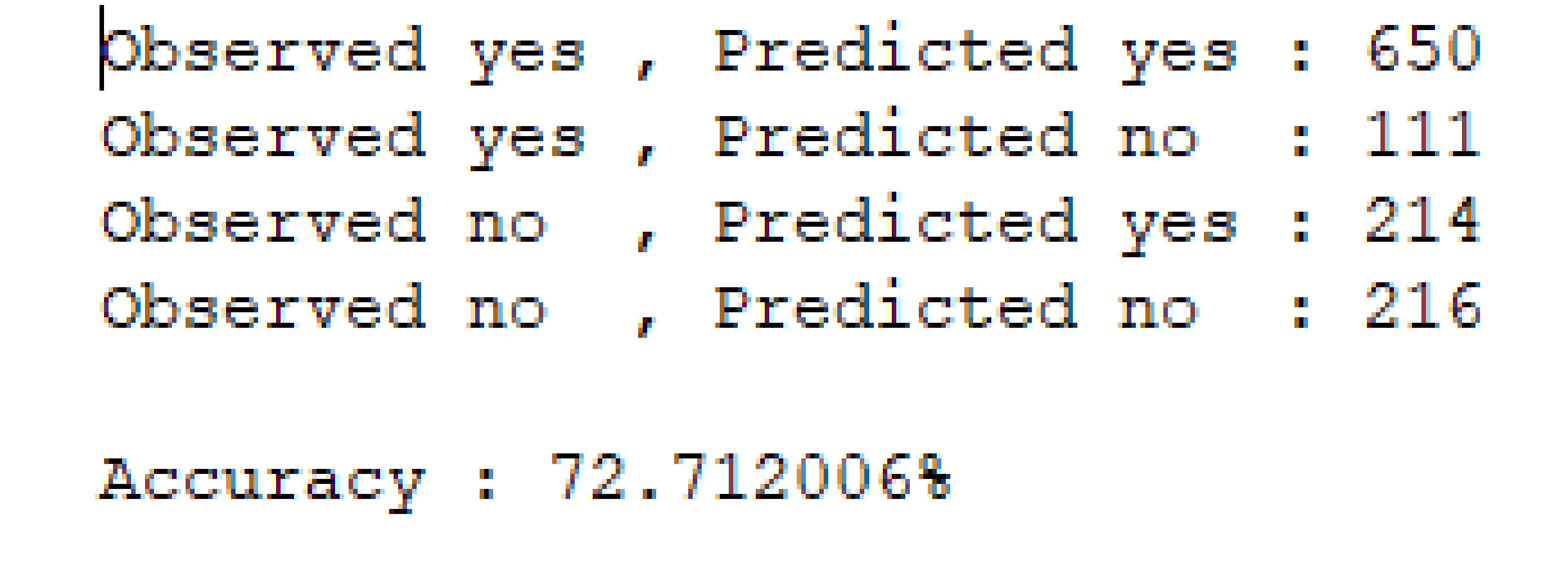
```
x11_no     6    73
x11_yes    651  240
x13_na     103  164
x14_no     2    14
x14_yes    682  260
x15_uncertain   1    1
x16_na     103  162
x17_no     1    92
x17_yes    666  219
x1_aonhewitt    24   4
x1_barclays     5    0
x1_pfizer  51   24
x1_prodapt 6    11
x1_ust     10   8
x2_itproductsandservices   34   11
x2_pharmaceuticals    96   69
x3_bangalore    193  99
x3_gurgaon 22   11
x4_dotnet  10   8
x4_productionsterile  0    5
x4_seleniumtesting    4    1
```

Output of Test Job



```
Output Matched        Observed : No        Predicted : no  <--Candidate 1
Output Matched        Observed : Yes       Predicted : yes <--Candidate 1232
Output Matched        Observed : Yes       Predicted : yes <--Candidate 1231
Output Matched        Observed : Yes       Predicted : yes <--Candidate 1230
Output Matched        Observed : Yes       Predicted : yes <--Candidate 1229
Output Matched        Observed : Yes       Predicted : yes <--Candidate 1228
Output Matched        Observed : Yes       Predicted : yes <--Candidate 1227
Output Matched        Observed : Yes       Predicted : yes <--Candidate 1226
Output Matched        Observed : Yes       Predicted : yes <--Candidate 1225
Output Matched        Observed : Yes       Predicted : yes <--Candidate 1224
Output Matched        Observed : Yes       Predicted : yes <--Candidate 1223
Output Matched        Observed : Yes       Predicted : yes <--Candidate 1222
Output Matched        Observed : Yes       Predicted : yes <--Candidate 1221
Output Matched        Observed : Yes       Predicted : yes <--Candidate 1220
Output Matched        Observed : Yes       Predicted : yes <--Candidate 1219
Output Matched        Observed : Yes       Predicted : yes <--Candidate 1218
                                      ⋮
Output Not Matched      Observed : No     Predicted : yes <--Candidate 730
Output Not Matched      Observed : No     Predicted : yes <--Candidate 729
Output Not Matched      Observed : No     Predicted : yes <--Candidate 1025
Output Not Matched      Observed : No     Predicted : yes <--Candidate 728
Output Not Matched      Observed : No     Predicted : yes <--Candidate 204
Output Not Matched      Observed : No     Predicted : yes <--Candidate 525
Output Not Matched      Observed : No     Predicted : yes <--Candidate 279
Output Not Matched      Observed : No     Predicted : yes <--Candidate 1020
Output Not Matched      Observed : No     Predicted : yes <--Candidate 724
Output Not Matched      Observed : No     Predicted : yes <--Candidate 1018
Output Not Matched      Observed : No     Predicted : yes <--Candidate 1017
Output Not Matched      Observed : Yes    Predicted : no  <--Candidate 523
Output Not Matched      Observed : Yes    Predicted : no  <--Candidate 124
Output Not Matched      Observed : Yes    Predicted : no  <--Candidate 521
Output Not Matched      Observed : Yes    Predicted : no  <--Candidate 379
Output Not Matched      Observed : No     Predicted : yes <--Candidate 202
```

Output of Analysis Job(Final Output)



```
Observed yes , Predicted yes : 650
Observed yes , Predicted no  : 111
Observed no  , Predicted yes : 214
Observed no  , Predicted no  : 216

Accuracy : 72.712006%
```

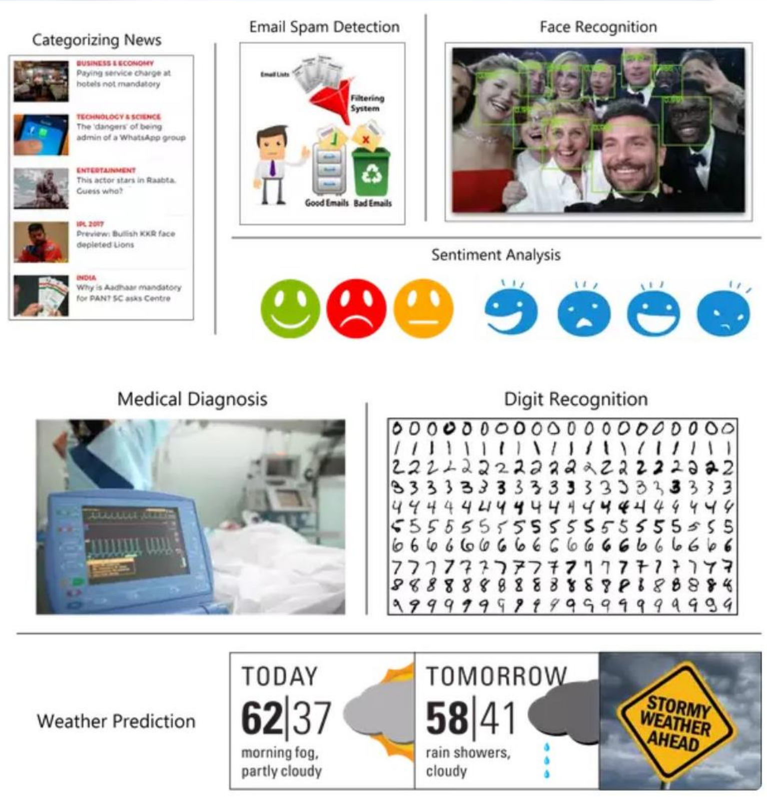
FUTURE IMPROVEMENT

The prediction analysis has been done based on a batch processing system instead of a real time system. For further improvement this has to be transformed into real time system using Apache Spark.

APPLICATION

This Naïve Bayes approach using MapReduce can be used similarly in many other applications:

- Churn Detection
- Vote Prediction
- Email Spam Detection
- News article categorization
- Sentiment Analysis
- Facial recognition
- Handwriting Recognition
- Weather Prediction



CONCLUSION

From a given data set, we have calculated the conditional probability for each value of the dependent attributes given the appearance of the candidates using Naïve Bayes supervised machine learning. After that we have predicted the candidate appearance and cross checked with the observed appearance and we have got a satisfactory *73% accuracy*. So we can say that this application is working as expected.

Thank you!!