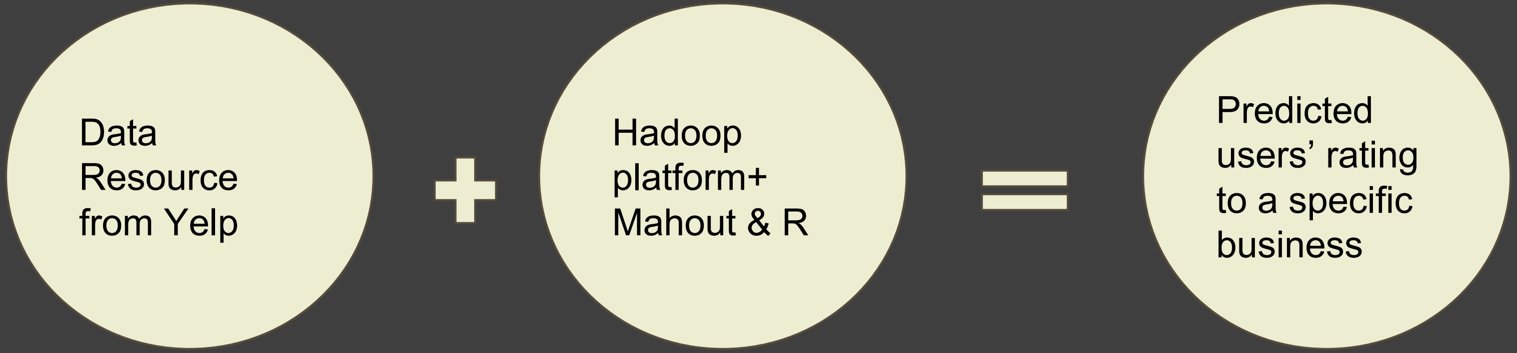
Agenda

- Project Background
- Project Steps
- System Architecture
- Performance Analysis

Project Background



Data Resource from Yelp

+

Hadoop platform+ Mahout & R

=

Predicted users’ rating to a specific business

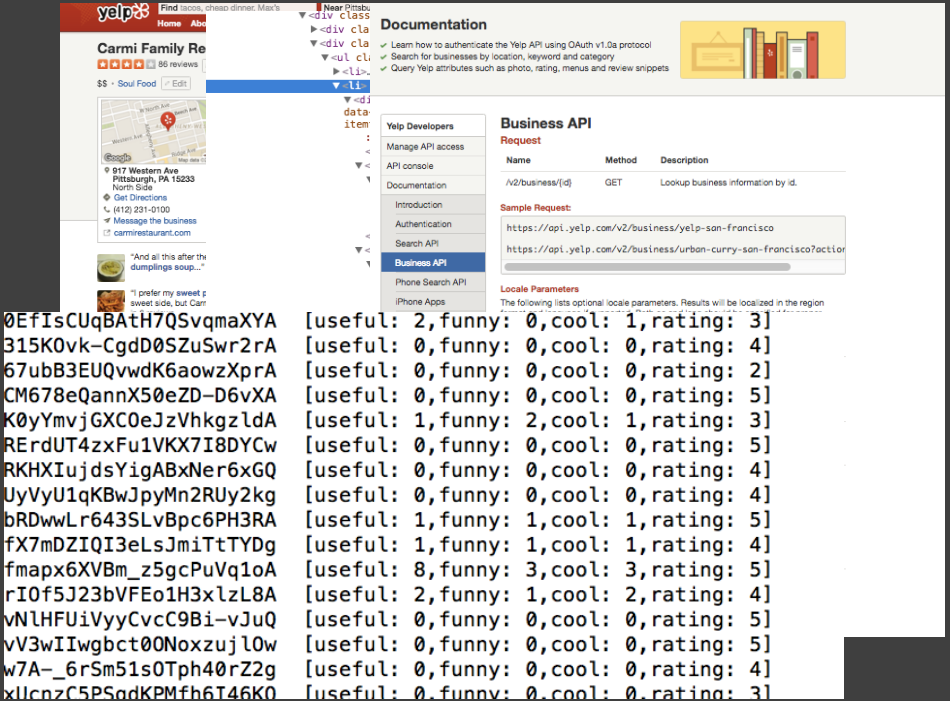
Project Steps| Data pre- process

Data Crawling(raw web data)

- API
- Web Spider

Data Pre-Processing

- User Information: MapReduce



Carmi Family Re
86 reviews
$$ · Soul Food Edit

917 Western Ave
Pittsburgh, PA 15233
North Side
Get Directions
(412) 231-0100
Message the business
carmirestaurant.com

"And all this after the dumplings soup..."

"I prefer my sweet p sweet side, but Carr

Documentation

- ✓ Learn how to authenticate the Yelp API using OAuth v1.0a protocol
- ✓ Search for businesses by location, keyword and category
- ✓ Query Yelp attributes such as photo, rating, menus and review snippets

Business API

Request

Name	Method	Description
/v2/business/{id}	GET	Lookup business information by id.

Sample Request:

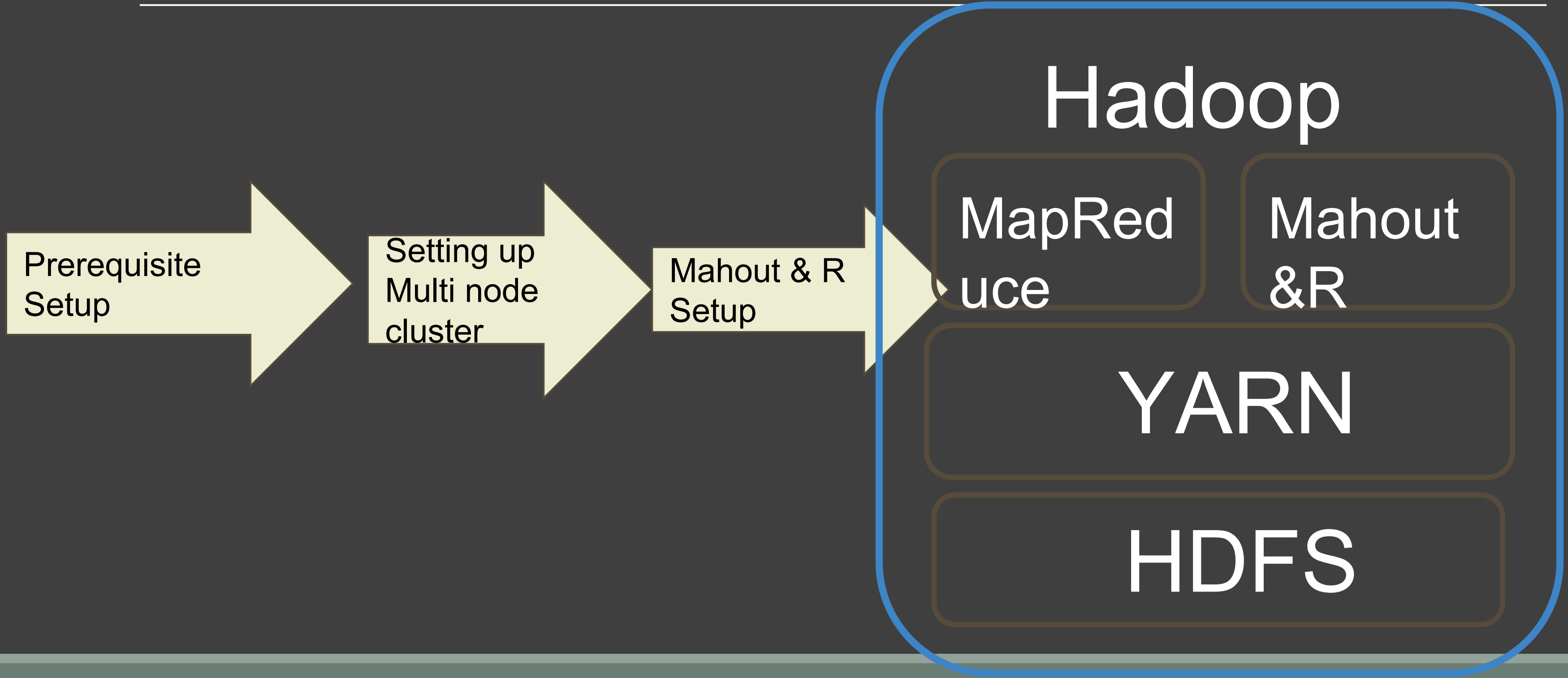
```
https://api.yelp.com/v2/business/yelp-san-francisco

https://api.yelp.com/v2/business/urban-curry-san-francisco?actior
```

Locale Parameters

The following lists optional locale parameters. Results will be localized in the region

```
0EfIsCUqBAtH7QSvqmaXYA  [useful: 2,funny: 0,cool: 1,rating: 3]
315KOvk-CgdD0SZuSwr2rA  [useful: 0,funny: 0,cool: 0,rating: 4]
67ubB3EUQvwdK6aowzXprA  [useful: 0,funny: 0,cool: 0,rating: 2]
CM678eQannX50eZD-D6vXA  [useful: 0,funny: 0,cool: 0,rating: 5]
K0yYmvjGXCOeJzVhkgzldA  [useful: 1,funny: 2,cool: 1,rating: 3]
RErdUT4zxFu1VKX7I8DYCw  [useful: 0,funny: 0,cool: 0,rating: 5]
RKHXIujdsYigABxNer6xGQ  [useful: 0,funny: 0,cool: 0,rating: 4]
UyVyU1qKBwJpyMn2RUy2kg  [useful: 0,funny: 0,cool: 0,rating: 4]
bRDwwLr643SLvBpc6PH3RA  [useful: 1,funny: 1,cool: 1,rating: 5]
fX7mDZIQI3eLsJmiTtTYDg  [useful: 1,funny: 1,cool: 1,rating: 4]
fmapx6XVBm_z5gcPuVq1oA  [useful: 8,funny: 3,cool: 3,rating: 5]
rIOf5J23bVFEo1H3xlzL8A  [useful: 2,funny: 1,cool: 2,rating: 4]
vNlHFUiVyyCvcC9Bi-vJuQ  [useful: 0,funny: 0,cool: 0,rating: 5]
vV3wIIwgbct0ONoxzujlOw  [useful: 0,funny: 0,cool: 0,rating: 5]
w7A-_6rSm51sOTph40rZ2g  [useful: 0,funny: 0,cool: 0,rating: 4]
xUcnzC5PSqdKPMfh6T46KO  [useful: 0,funny: 0,cool: 0,rating: 3]
```

Project Steps| Hadoop Framework Setup



Prerequisite Setup

Setting up Multi node cluster

Mahout & R Setup

Hadoop

MapReduce

Mahout &R

YARN

HDFS

Project Steps| Prediction Model Construction

Data Description

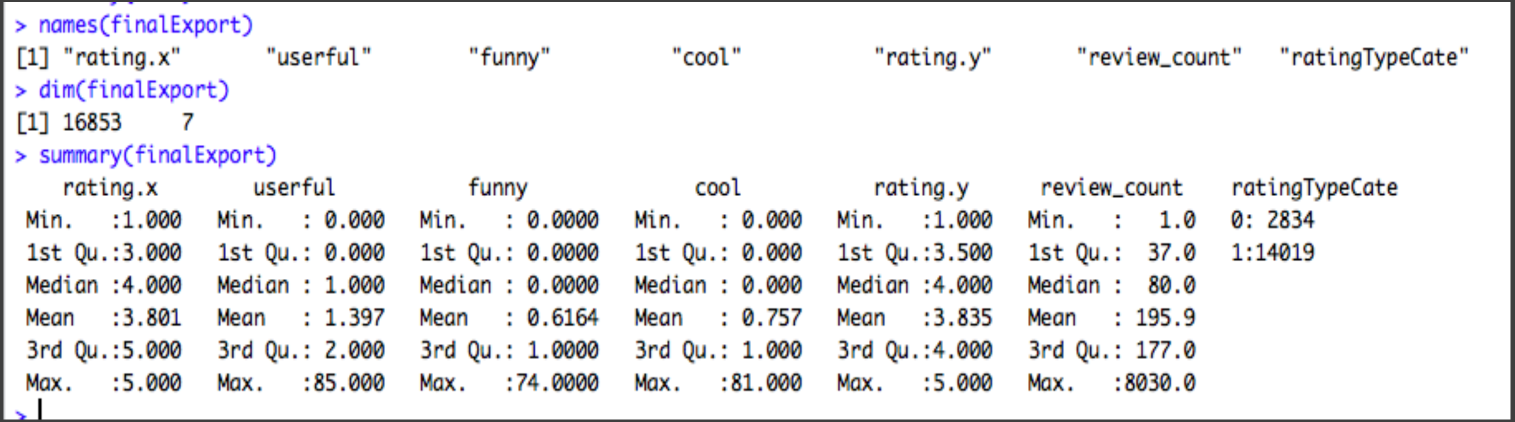
userID	Review Rating	Useful Count	Funny Count	Cool Count	Business ID	Review Count	Business Rating
							[Binary]

Logistic Regression

Business Rating ~ ReviewRating + UsefulCount + FunnyCount + CoolCount + ReviewCount

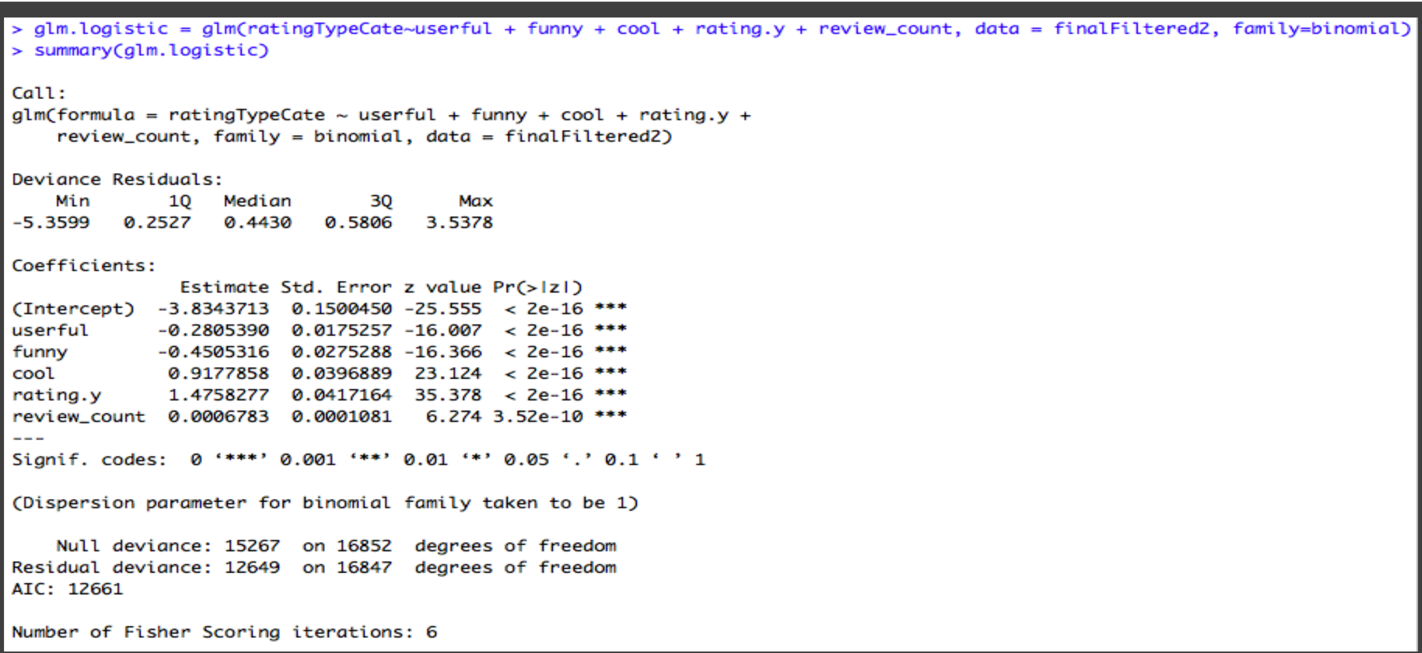
Predictive Model Construction

Data Statistics Summary



```
> names(finalExport)
[1] "rating.x"       "userful"         "funny"          "cool"           "rating.y"        "review_count"    "ratingTypeCate"
> dim(finalExport)
[1] 16853     7
> summary(finalExport)
    rating.x          userful            funny               cool             rating.y        review_count     ratingTypeCate
 Min.   :1.000   Min.   : 0.000   Min.   : 0.0000   Min.   : 0.000   Min.   :1.000   Min.   :   1.0   0: 2834
 1st Qu.:3.000   1st Qu.: 0.000   1st Qu.: 0.0000   1st Qu.: 0.000   1st Qu.:3.500   1st Qu.:  37.0   1:14019
 Median :4.000   Median : 1.000   Median : 0.0000   Median : 0.000   Median :4.000   Median :  80.0
 Mean   :3.801   Mean   : 1.397   Mean   : 0.6164   Mean   : 0.757   Mean   :3.835   Mean   : 195.9
 3rd Qu.:5.000   3rd Qu.: 2.000   3rd Qu.: 1.0000   3rd Qu.: 1.000   3rd Qu.:4.000   3rd Qu.: 177.0
 Max.   :5.000   Max.   :85.000   Max.   :74.0000   Max.   :81.000   Max.   :5.000   Max.   :8030.0
> |
```

Logistic Regression



```
> glm.logistic = glm(ratingTypeCate~userful + funny + cool + rating.y + review_count, data = finalFiltered2, family=binomial)
> summary(glm.logistic)

Call:
glm(formula = ratingTypeCate ~ userful + funny + cool + rating.y + 
    review_count, family = binomial, data = finalFiltered2)

Deviance Residuals: 
    Min       1Q   Median       3Q      Max  
-5.3599   0.2527   0.4430   0.5806   3.5378  

Coefficients:
               Estimate Std. Error z value Pr(>|z|)    
(Intercept)  -3.8343713  0.1500450 -25.555  < 2e-16 ***
userful      -0.2805390  0.0175257 -16.007  < 2e-16 ***
funny        -0.4505316  0.0275288 -16.366  < 2e-16 ***
cool          0.9177858  0.0396889  23.124  < 2e-16 ***
rating.y      1.4758277  0.0417164  35.378  < 2e-16 ***
review_count  0.0006783  0.0001081   6.274 3.52e-10 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 15267  on 16852  degrees of freedom
Residual deviance: 12649  on 16847  degrees of freedom
AIC: 12661

Number of Fisher Scoring iterations: 6
```

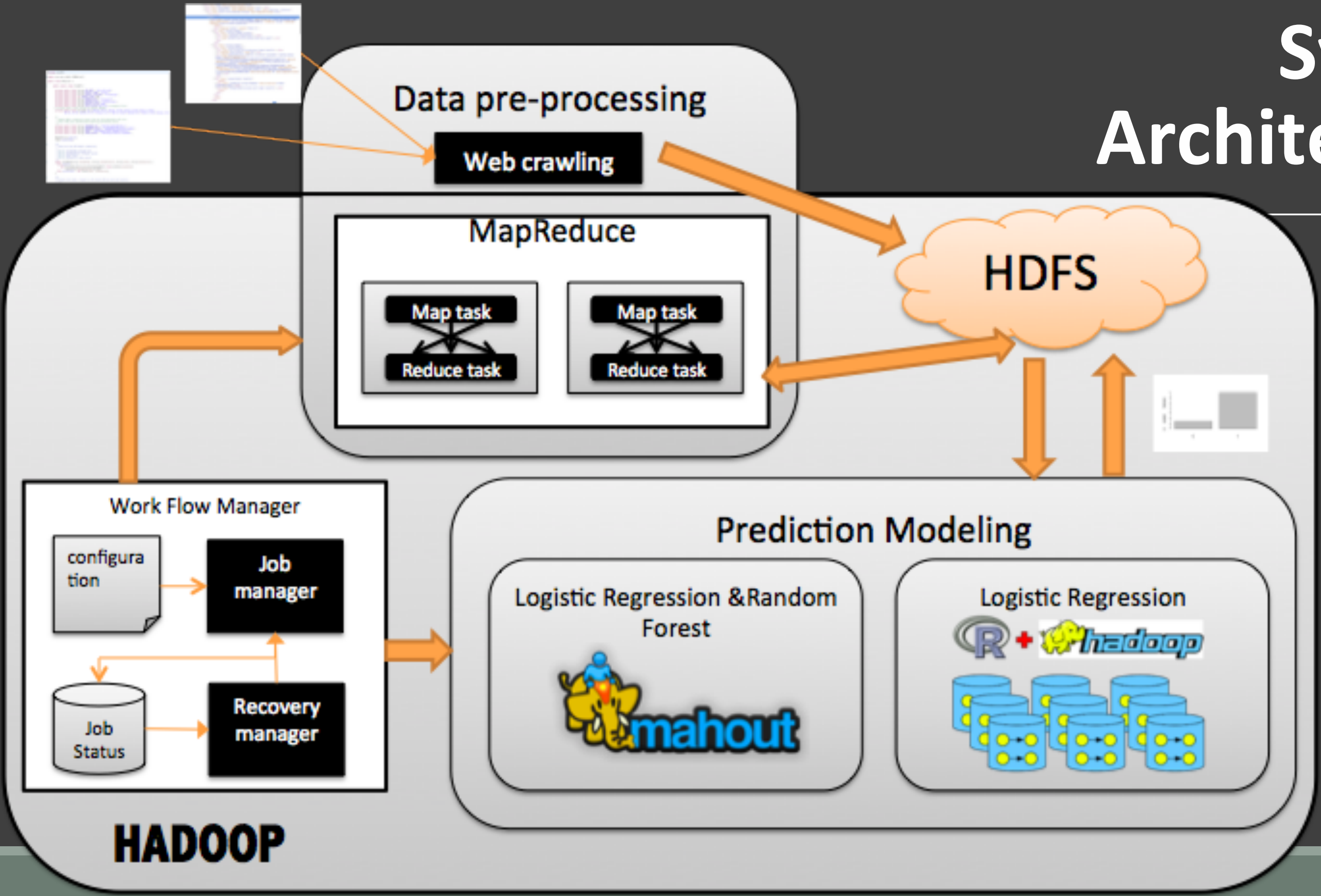
System Architecture|Mahout

Apache Software Foundation to produce free distributed implementation and scalable machine learning algorithm focus on recommendation, collaborative filtering, clustering and classification

System Architecture| RHadoop

RHadoop is a collection of five R packages that allow users to manage and analyze data with Hadoop. The packages have been tested (and always before a release) on recent releases of the Cloudera and Hortonworks Hadoop distributions and should have broad compatibility with open source Hadoop and mapR's distribution.

System Architecture



Data pre-processing

Web crawling

MapReduce

Map task

Reduce task

Map task

Reduce task

HDFS

Work Flow Manager

configura tion

Job manager

Job Status

Recovery manager

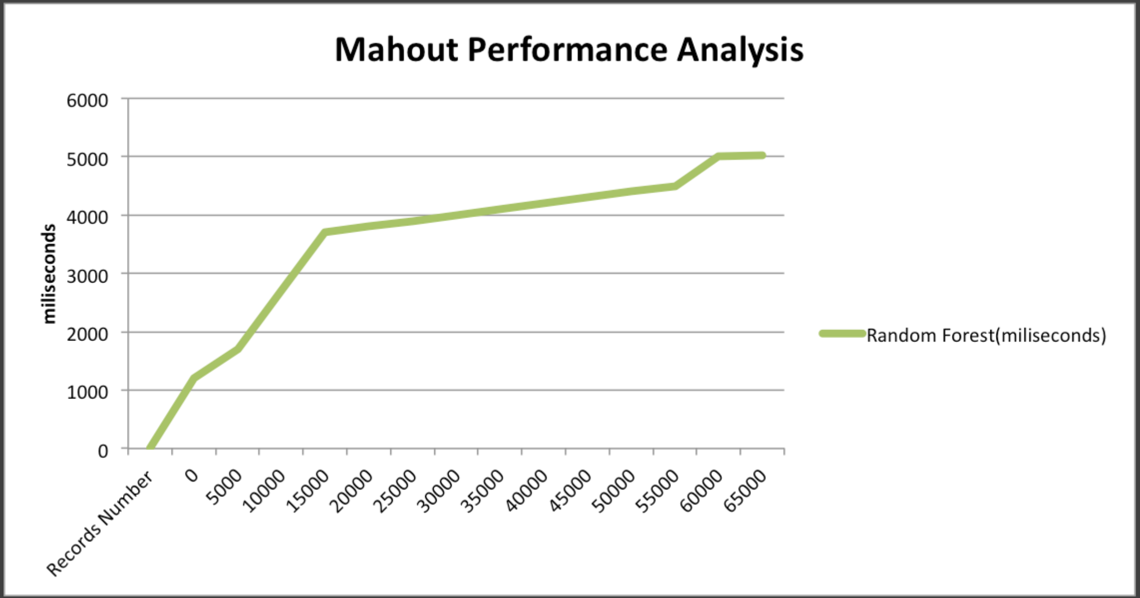
Prediction Modeling

Logistic Regression &Random Forest

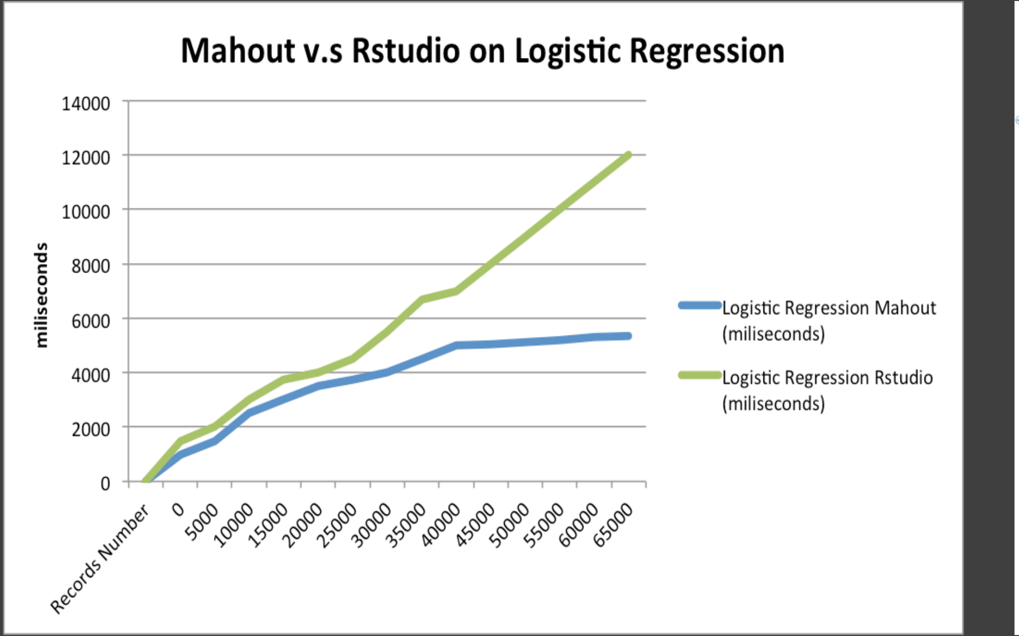
Logistic Regression

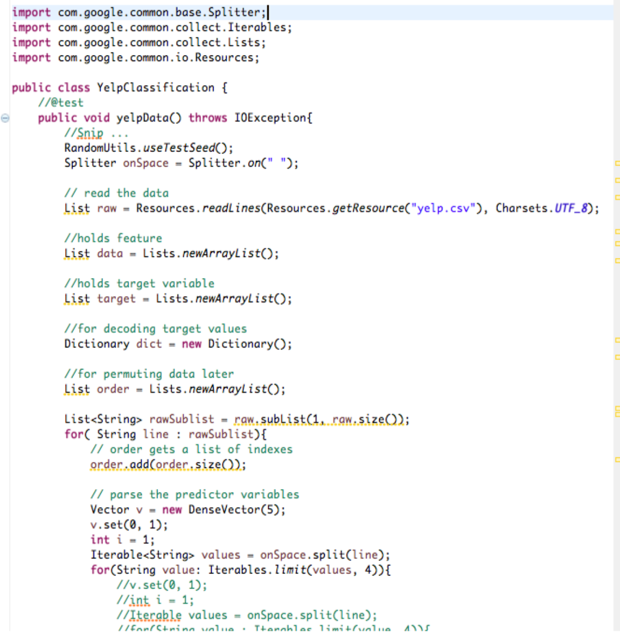
HADOOP

Performance Comparison



Performance Comparison





```
import com.google.common.base.Splitter;
import com.google.common.collect.Iterables;
import com.google.common.collect.Lists;
import com.google.common.io.Resources;

public class YelpClassification {
    //@test
    public void yelpData() throws IOException{
        //Snip ...
        RandomUtils.useTestSeed();
        Splitter onSpace = Splitter.on(" ");

        // read the data
        List raw = Resources.readLines(Resources.getResource("yelp.csv"), Charsets.UTF_8);

        //holds feature
        List data = Lists.newArrayList();

        //holds target variable
        List target = Lists.newArrayList();

        //for decoding target values
        Dictionary dict = new Dictionary();

        //for permuting data later
        List order = Lists.newArrayList();

        List<String> rawSublist = raw.subList(1, raw.size());
        for( String line : rawSublist){
            // order gets a list of indexes
            order.add(order.size());

            // parse the predictor variables
            Vector v = new DenseVector(5);
            v.set(0, 1);
            int i = 1;
            Iterable<String> values = onSpace.split(line);
            for(String value: Iterables.limit(values, 4)){
                //v.set(0, 1);
                //int i = 1;
                //Iterable values = onSpace.split(line);
                //for(String value : Iterables.limit(value, 4)){
```