Resume Analysis for Skill-Set Estimation using HDFS,MapReduce and R

Krishna Mohan Ankala, Sowmya Karra

Abstract— **A Resume is a document used to represent a person's background and skills. A typical resume contains a summary of past employment and education, technical skills, co curricular and extracurricular activities, personal details etc. It is usually one of the most important items submitted for an application for employment which is used to screen applicants based on the profile required.**

Hence analysis of resumes of students provides a good insight into how many people are proficient in various technologies. In this paper, we present a model that analyzes the current set of resumes using HDFS, MapReduce and R. While resumes from hundreds of students represent big data set, we use HDFS to store them. Map Reduce is the programming model that is used to extract patterns from the resumes. Finally the results are visualized using R.

Keywords—HDFS, MapReduce,R

I. Introduction

Organizations are always on hunt for good technical talent which is often expensive and short in supply. Resume is one such document which records a person's data that provides a useful window into some of the specific skills and areas of knowledge that are in greatest demand. The traditional approach is one where resumes are uploaded to the job portal or career section of an organization and then manually read by a person to identify if he/she has the primary skill(s) the job requires. It also helps determine a person's pay scale depending on the skill set. This old way has many disadvantages.

1.Every resume has to be manually checked by a person which is time consuming
2.The skill will have to be saved into a file along with the name and other details of the person
3.If different people analyze different resumes the overall result can be ambiguous as they may record details at varying levels (total no of people with the same skill set, some skills may not be appealing to some people etc..)

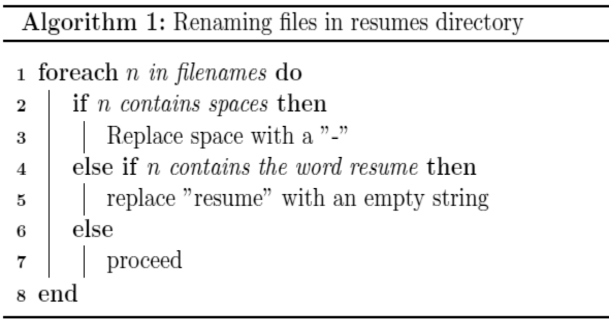
II. Proposed approach

In this paper ,an algorithm is proposed where resumes are collected from students and stored for further analysis.
Before proceeding with the entire analysis process two things are assumed.
1.Resumes are named after the student(If at all there are two people with the same name care should be taken to rename the files as the system will not take in duplicates)
2.A standardized format for all resumes should be followed for accurate results

The following are the steps :

Step-1: 'N' number of resumes are taken from the students and saved in a directory
(pdf, doc and docx files)

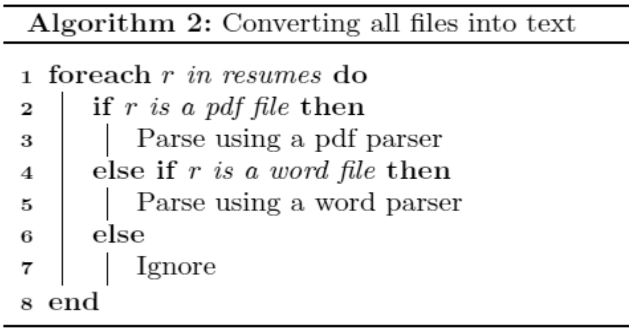
Step-2: (File Renaming) All files that have spaces in their file names are renamed by replacing the space with any special character of personal choice(" -" is used here)



```
Algorithm 1: Renaming files in resumes directory
foreach n in filenames do
    if n contains spaces then
        Replace space with a "-"
    else if n contains the word resume then
        replace "resume" with an empty string
    else
        proceed
end
```

Fig 1.File Renaming Algorithm

Step-3: **(File Conversion)** HDFS cannot store pdf and word formats . Hence all the files are converted into text files .

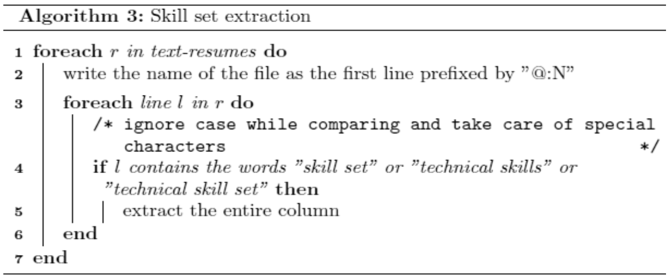


```
Algorithm 2: Converting all files into text
foreach r in resumes do
    if r is a pdf file then
        Parse using a pdf parser
    else if r is a word file then
        Parse using a word parser
    else
        Ignore
end
```

Fig 2.File Conversion Algorithm

Step-4: **(Skill Set Extraction)** The analysis procedure does not need the entire file. Only the skill set section along with the student's name is extracted.

Krishna Mohan Ankala is a professor and Head of the Department of Computer Science & Engineering ,JNTU(Jawaharlal Nehru Technological University),Kakinada-533003,AndhraPradesh,India
Phone:(+91)8008498555(e-mail:krishna.ankala@gmail.com)
Sowmya Karra is an M.Tech(Master of Technology) student in CSE department,JNTU,Kakinada-533003,AndhraPradesh,India
Phone:(+91)9966432969(e-mail:karrasowmya@gmail.com)

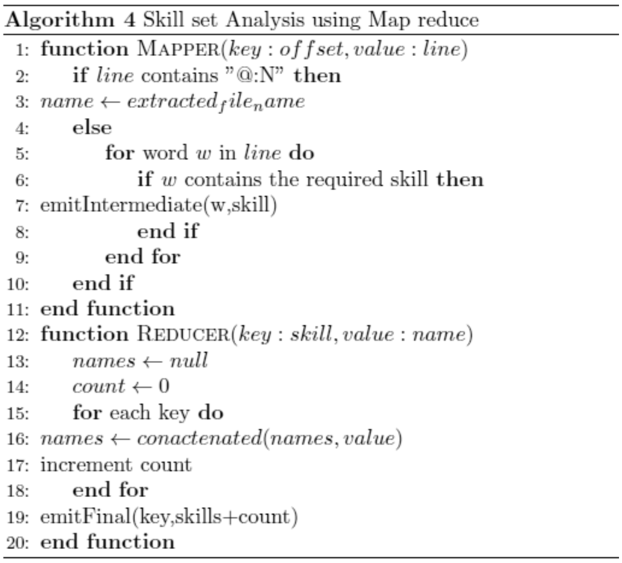


```
Algorithm 3: Skill set extraction
foreach r in text-resumes do
    write the name of the file as the first line prefixed by "@:N"
    foreach line l in r do
        /* ignore case while comparing and take care of special
           characters                                            */
        if l contains the words "skill set" or "technical skills" or
         "technical skill set" then
            extract the entire column
    end
end
```

Fig 3.Skill Set Extraction Algorithm

<u>Step-5</u>:(Storing on HDFS) The above file is now placed on HDFS.

<u>Step-6</u>:(Analyze) Using MapReduce , the required skills are analyzed. This gives the Skill - Names of students proficient - Count as the output. This file is also on HDFS



```
Algorithm 4 Skill set Analysis using Map reduce
function MAPPER($key : offset, value : line$)
    if line contains "@:N" then
$name \leftarrow extracted_{f}ile_{n}ame$
    else
        for word w in line do
            if w contains the required skill then
emitIntermediate(w,skill)
            end if
        end for
    end if
end function
function REDUCER($key : skill, value : name$)
    $names \leftarrow null$
    $count \leftarrow 0$
    for each key do
$names \leftarrow conactenated(names, value)$
increment count
    end for
emitFinal(key,skills+count)
end function
```

Fig 4.Skill Analyzing Algorithm(Map Reduce)

<u>Step-7</u>:(Visualization) Using RHadoop, write an R script that reads the analyzed skills file and visualize it as a bar plot.

<u>Step-8</u>:(Resume upload) Now when a new resume comes into the system, repeating all the above steps would be very expensive. Hence a slightly optimized procedure is followed

1.Upload the file into the system
2.Perfom Steps 2,3,4 on the given file only
3.Get the skills file from HDFS and append the current student's data to that
4.Place this file on HDFS again

<u>Step-9</u>: Along with the visualization part in R, the skills-students-count details are listed in the browser

III. LITERATURE SURVEY

This section presents a comprehensive literature review from different journals, academicians and other internet sources.

First is the process of extracting information from resumes. It is not an easy task as resumes come in variable formats and file types. In the paper " Resume information extraction with cascaded hybrid model "[1] the author used an HMM(Hidden Markov Model) based pipelined approach to extract text in multiple iterations.

In the first pass, a resume is segmented into a consecutive blocks attached with labels indicating the information types. Then in the second pass, the detailed information, such as Name and Address, are identified in certain blocks (e.g. blocks labeled with Personal Information), instead of searching globally in the entire resume. But this however cannot be used as we are not interested in detailed extraction of fields like address, phone , zip code etc.. Instead the concentration is only on skill set and we do not wish to shift the crux of the paper from that to extracting unnecessary information. Hence Apache Tika is used which is more appropriate for the problem.

Apache Tika is a library that provides a flexible and robust set of interfaces that can be used in any context where metadata analysis and structured text extraction is needed. The key component of Apache Tika is the Parser (org.apache.tika.parser.Parser) interface because it hides the complexity of different file formats while providing a simple and powerful mechanism to extract structured text content and metadata from all sorts of documents.

The next step is Skill Extraction from the resumes. In the paper " An Approach to Extract Special Skills to Improve the Performance of Resume Selection "[2] the author(s) proposed an approach to identify resumes with special skill information. The notion of special features have been applied to improve the process of product selection in E-commerce environment. However, extending the notion of special features for the development of approach to process resumes is a complex task as resumes contain unformatted text or semi-formatted text. In this paper, the author(s) have proposed an approach by considering only skills related information of the resumes. This , however , cannot be used completely as organizations receive and store thousands (or even lakhs) of resumes which constitutes Big Data ! Hence for extracting the person and skill information Hadoop would be a better framework as it analyzes data in a distributed manner.

In the Analysis step Map Reduce is used for finding out the results(skill-students-count).In the paper " Analysis of Bigdata using Apache Hadoop and Map Reduce "[3] the author talked about the problem of explosion of data and the size of the databases. Processing or analyzing the huge amount of data or extracting meaningful information is a challenging task. The term “Big data” is used for large data sets whose size is beyond the ability of commonly used software tools to capture, manage, and process the data within a tolerable elapsed time. Big data sizes are a constantly moving target currently ranging from a few dozen terabytes to many peta bytes of data in a single data set. Difficulties include capture, storage, search, sharing, analytics and visualizing. He also talked about Hadoop Framework and results with a word count(word and no of occurences) example. This alone is not sufficient for the proposed system but can be used as a base for calculating the final result.

In all the above papers, data were analyzed with output in a basic form(either word-count or details of a person etc...) But as known, a picture speaks a thousand words, Hence R(programming language) is used to visualize the results as a graph along with data in a table format(in a browser) so that any novice will be able to understand the output.

IV. HADOOP

We live in the data age. It's not easy to measure the total volume of data stored electronically. The problem is simple: while the storage capacities of hard drives have increased tremendously over the time, access speeds—the rate at which data can be read from drives—have not kept up.

It takes a long time to read all data on a single drive and writing is even slower. The most efficient way to reduce the time is to read from multiple disks at once.Hadoop works in the same way. The storage component of hadoop is called HDFS(Hadoop Distributed File System) which segments the data file into blocks and replicates these across multiple drives.

V. MAP-REDUCE

MapReduce is the programming model that Hadop uses for data processing. Hadoop can run MapReduce programs written in various languages like Java, Ruby, Python, and C++. Most important, these programs are run in parallel, thus reducing the time taken for large scale data analysis.

MapReduce works by splitting the processing into two steps(phases): the map phase and the reduce phase. The input and output to these phases are key value pairs with varying data type options as required by the programmer.

In our algorithm, the input to the Mapper is the file that has skills and names of the person. Every name of the person is appended before the skill set using "N:@" prefix to identify it as a name. The skill set then follows till we find another name. In the Mapper phase each line of the file is taken as an input where the key is the offset and value is the entire line. The output from the mapper is the skill as the key and name as the value. The Reducer works by collecting all the values with the same key(i.e skill) and counting the total and concatenating all the students names who are proficient in that skill. So here the output is the skill as the key and concatenated names and count as value.

VI. R

R is a programming language and a software suite used for data analysis, statistical computing and data visualization. It is highly extensible and has object oriented features and strong graphical capabilities. At its heart R is an interpreted language and comes with a command line interpreter – available for Linux, Windows and Mac machines – but there are IDEs as well to support development like RStudio

R and Hadoop can complement each other very well. One of the most well-known R packages to support Hadoop functionalities is RHadoop that was developed by Revolution Analytics. RHadoop is a collection of three R packages: rmr, rhdfs and rhbase. rmr package provides Hadoop MapReduce functionality in R, rhdfs provides HDFS file management in R and rhbase provides HBase database management from within R.

The following are some of the graphs that we generated using Rhadoop to analyze the "Technology and Count" using resumes

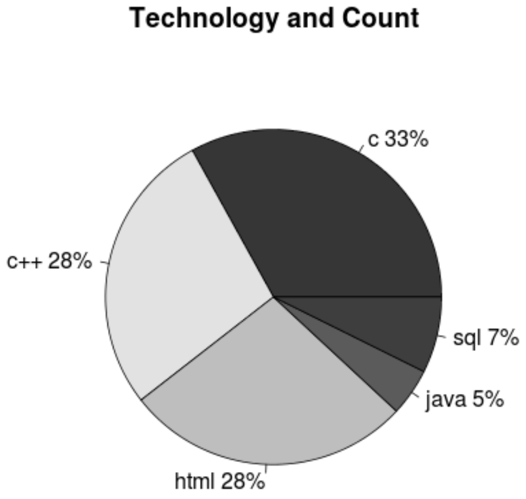


Fig 5. Pie Chart using R

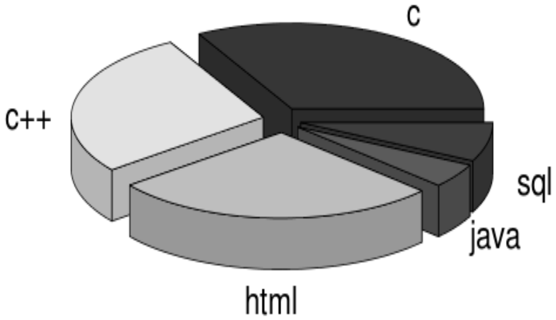


Fig 6. 3D Pie Chart Using R

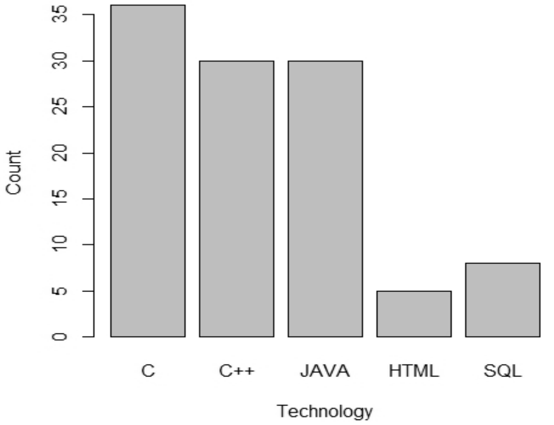


Fig 7. Bar Chart Using R

ISBN: 978-988-14047-1-8
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

VII. RESULTS

Table I. Upload File Input Vs Output

INPUT	OUTPUT
x.txt/x.png/x.jpg (Any extension other than pdf and word)	Only pdf and word files are allowed !
x.pdf (or) x.doc(x)	File Successfully Uploaded
x.pdf(same file name)	File Name exists ! Please choose a different name
y.pdf/doc(x)(Same File but Different name) (contents are compared)	File already exists !! Please choose a different file

Table II. Pseudo Distributed Mode

No of Resumes	Time for Map phase (ms)	Time for Reduce Phase(ms)	Total Time (ms)
56	1	8	14
100	1	15	24
200	5	27	38
320	6	45	59

Table III. 4 Node Cluster(1 Namenode & 3 Datanodes)

No of resumes	Block Size (MB)	No of blocks	Time Spent in sec (Mappers)	Time Spent in sec (Reducers)
107100	1	11	63.261	50.524
107100	2	6	38.82	53.957
107100	4	3	11.229	51.756
214200	1	23	172.541	193.759
214200	2	11	89.223	190.229
214200	4	6	39.986	194.206
321300	1	34	260.554	466.216
321300	2	17	123.78	463.012
321300	4	9	73.519	491.919
535500	1	56	720.693	1457.934
535500	2	28	214.672	1391.875
535500	4	14	129.519	1399.446

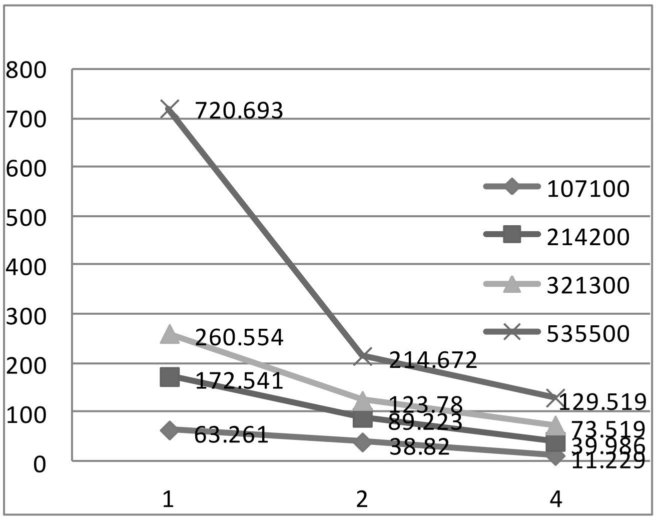


Fig 8. Block Size Vs Time Spent by Mappers

The above graph indicates that as the block size increases, there will be a decrease in the no of blocks which in turn reduces the time occupied by Mappers
E.g. No of Resumes = 107100
Total Size = 3181549 bytes = 3.03416MB
Block size = 1 MB
No of Blocks = 4
Since the no of blocks is 4 , 4 map tasks are launched which run in parallel and hence reduce the overall time.

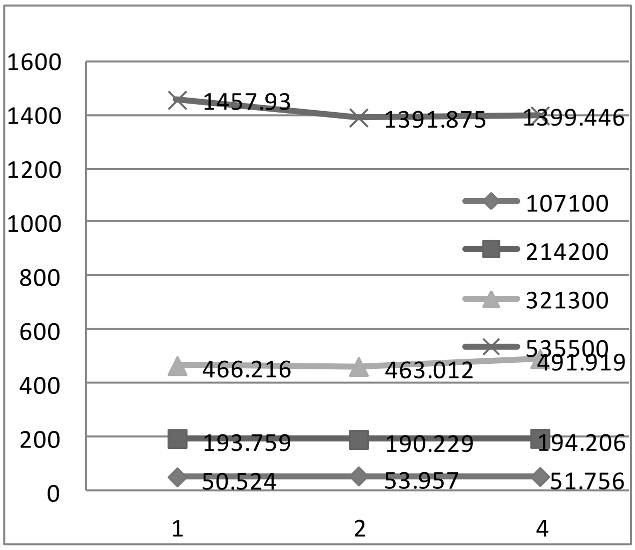


Fig 9. Block Size Vs Time Spent by Reducers

The above graph shows the Time spent by reducers after map phase. It is clear that despite the no of blocks , reduce phase takes almost the same time as it works on filtered data from the Mappers.

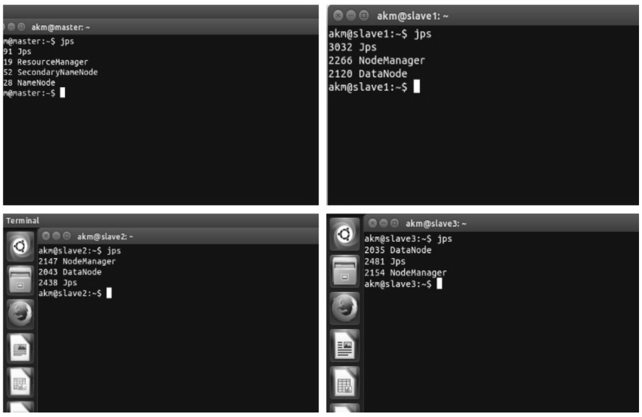
Table IV. Java Vs Hadoop

Size of the file(Approx)	Java(sec)	Block size HDFS	Hadoop (sec)
12 MB	145	1 MB	113.785
12 MB		2 MB	92.777
12 MB		4 MB	62.985
24 MB	292	1 MB	366.3
24 MB		2 MB	279.452
24 MB		4 MB	234.192
35 MB	572	1 MB	726.77
35 MB		2 MB	586.792
35 MB		4 MB	565.438
56 MB	1154	1 MB	2178.627
56 MB		2 MB	1606.547
56 MB		4 MB	1528.965
180 MB	1200	64 MB (Default)	360.867
400 MB	2700	64 MB	840.334
720 MB	6000	64 MB	2441.890

The above table shows that the real beauty of hadoop can be seen while working with very huge sets of data

VIII. SCREENSHOTS

The following were the screenshots of the full cluster installed in the Research laboratory



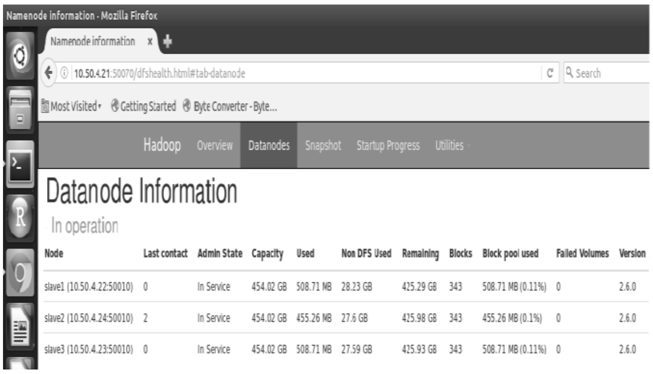
```
m@master:~$ jps
91 Jps
19 ResourceManager
52 SecondaryNameNode
28 NameNode
m@master:~$
```

```
akm@slave1:~$ jps
3032 Jps
2266 NodeManager
2120 DataNode
akm@slave1:~$
```

```
akm@slave2:~$ jps
2147 NodeManager
2043 DataNode
2438 Jps
akm@slave2:~$
```

```
akm@slave3:~$ jps
2035 DataNode
2481 Jps
2154 NodeManager
akm@slave3:~$
```

Fig 10. Hadoop Daemons (Master & Slaves)

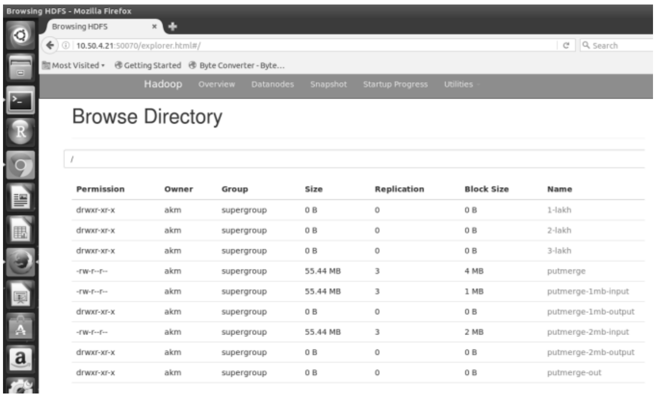


Datanode Information

In operation

Node	Last contact	Admin State	Capacity	Used	Non DFS Used	Remaining	Blocks	Block pool used	Failed Volumes	Version
slave1 (10.50.4.22:50010)	0	In Service	454.02 GB	508.71 MB	28.23 GB	425.29 GB	343	508.71 MB (0.11%)	0	2.6.0
slave2 (10.50.4.24:50010)	2	In Service	454.02 GB	455.26 MB	27.6 GB	425.98 GB	343	455.26 MB (0.1%)	0	2.6.0
slave3 (10.50.4.23:50010)	0	In Service	454.02 GB	508.71 MB	27.59 GB	425.93 GB	343	508.71 MB (0.11%)	0	2.6.0

Fig 11. Data nodes Health/Status



Browse Directory

/

Permission	Owner	Group	Size	Replication	Block Size	Name
drwxr-xr-x	akm	supergroup	0 B	0	0 B	1-lakh
drwxr-xr-x	akm	supergroup	0 B	0	0 B	2-lakh
drwxr-xr-x	akm	supergroup	0 B	0	0 B	3-lakh
-rw-r--r--	akm	supergroup	55.44 MB	3	4 MB	putmerge
-rw-r--r--	akm	supergroup	55.44 MB	3	1 MB	putmerge-1mb-input
drwxr-xr-x	akm	supergroup	0 B	0	0 B	putmerge-1mb-output
-rw-r--r--	akm	supergroup	55.44 MB	3	2 MB	putmerge-2mb-input
drwxr-xr-x	akm	supergroup	0 B	0	0 B	putmerge-2mb-output
drwxr-xr-x	akm	supergroup	0 B	0	0 B	putmerge-out

Fig 12. Files in HDFS

IX. CONCLUSION AND FUTURE WORK

In this paper, it has been discussed how resumes can be used to estimate the skill set of all the students. It has been shown how to preprocess the files and then extract relevant text for skill analysis. However the technique can be further extended to extract anything(any data like name, address etc..) from the resumes. It can be further optimized to extract text from image and video resumes. The practical results show that the proposed method of text classification gives better results as compared to the existing one and hence we further would like to carry on more investigation over the same

REFERENCES

[1] Kun Yu, Gang Guan, and Ming Zhou, "Resume Information Extraction with Cascaded Hybrid Model", In Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics, Stroudsburg, PA, USA, pp 499-506,2005

[2] Sumit Maheshwari, Abhishek Sainani, P.Krishna Reddy, "An Approach to Extract Special Skills to Improve the Performance of Resume Selection", In 6th International Workshop, DNIS 2010, Aizu-Wakamatsu, Japan, pp 256-273, 2010

[3] Mrigank Mridul, Akashdeep Khajuria, Snehasish Dutta, Kumar N, Prasad.M.R,"Analysis of Bidgata using Apache Hadoop and Map Reduce",In International Journal of Advanced Research in Computer Science and Software Engineering, Volume 4, Issue 5, pp 2277-128X,May 2014

[4] Katrina Sin, Loganathan Muthu, "Application Of Big Data In Education Data Mining And Learning Analytics – A Literature Review, In ICTACT journal on soft computing: special issue on soft computing models for big data, volume: 05, issue: 04 pp 2229-6956 ,July 2015

[5] Cristóbal Romero, Sebastián Ventura, "Educational Data Mining:A Review of the state of Art", In IEEE Transactions on Systems, Man and Cybernetics , Part C (Applications and Review) Volume 40,Issue 6, pp 601-618 ,July 2010

[6] Divna Krpan, Slavomir Stankov, "Educational Data Mining for grouping students in e-learning system",In Information Technology Interfaces (ITI), Proceedings of the ITI 2012 34th International Conference ,pp 207-212,June 2012

[7] Tom White "Hadoop: The Definitive Guide, Third Edition"

[8] Sunil Kumar Kopparapu, "Automatic Extraction Of Usable information from unstructured Resumes to aid search", In Progress in Informatics and Computing (PIC),IEEE International Conference on (Volume:1)", pp 99-103, Dec 2010