ML Methods for Solving Complex Sorting and Ranking Problems in Human Hiring

[1]Kavyashree M Bandekar, [2]Maddala Tejasree, [3]Misba Sultana S N, [4]Nayana G K, [5]Harshavardhana Doddamani
[1, 2, 3, 4]Engineering Student in Dept. of Computer Science Engineering, SJCIT, Chickballapur Dist, Karnataka, India
[5]Assistant Professor in Dept. of Computer Science Engineering, SJCIT, Chickballapur Dist, Karnataka, India

Abstract—*Every organization has its own job description in hiring the employees for his organization some concentrate on communication, technical skills, domain expertise, experience, flexibility. The job search engines takes input from the HR and provide the matching resumes of people who belong to that particular category and since the outcome result of search grows, the HR faces problem in selecting the best resume out of huge number. Understanding this hiring pattern here the role of Human Resource (HR) staff becomes important. The proposed method is to accommodate machine learning concepts to minimize the human intervention in hiring, understanding the intelligence behind the hiring pattern, offers the ranking system according to the hiring patterns predicts the ranking and sorting of resumes with high accuracy and simplifies the job of human resourcing efficiently.*

Keywords— *Human Resourcing, Hiring pattern, Data Cleaning algorithm, Machine Learning, Tokenization, IF-TDF, K-means, SVM.*

I. INTRODUCTION

New Technologies are playing their role in market and the human resource team are facing peculiar challenges in hiring the people in order to meet the requirements from client to client to survive in the market. As every organization carries a different point about a job description, reading through the resume also varies. Barely matching skills and experience is no more important alone for the serious organizations. For example, some companies consider the Domain expertise but some other gives more importance to the number of skills and total years of professional experience, flexibility. Human Resource (HR) agencies use various search methods and these search methods connected with the database which consist of millions of resumes.

In the previous approach the organization is to use the simple search engines that parses the resumes against the given keywords and offers the best match results. The list of the searching keywords is usually prepared by the HR after reading the job description several times. The HR downloads these searched resumes and does the manual work by opening and reading the resumes. By this ways, HR person tries to find the resumes which are best match to the JD. This is a cumbersome process and requires reasonable time and multiple discussions with the candidate before offering the resume to the client. Usually, due to the complexity of the database, many efficient resumes missed out from the search results or not considered due to stringent timelines of closure. Do manual analysis of the resume for various attributes like programming languages, domain, and years of experience. This process is cumbersome and if the HR has to short list the candidates within week/days then it becomes really difficult with the manual approach.

In our proposed approach the candidate uploads the resume. The various data mining algorithms are applied and then the attributes like years of experience, education, programming skills and domain are found out. The resumes are ranked based on the requirement from HR. The resumes are also classified into clusters of domains using Support Vector Machine and the resumes are ranked based on TF-IDF algorithm.

II. LITERATURE SURVEY

In the paper titled "*Software Engineering*" the authors describe that application of engineering methods and principles to the design, production and maintenance of software. While writing software began in the 1940s, the term software engineering stems from the 'Software Crisis' of the 1960s, 70s and 80s, with the objective of tackling the many software development projects that over ran budget and schedule

In the paper titled "*No Silver Bullet – Essence and Accidents of Software Engineering*" titled the authors describe that the objective of improving software development results has remained the same, the approaches have evolved over the past few decades. Various new approaches have appeared within Software Engineering literature, including; Formal methods, CASE tools, Object Oriented Programming, Structured Programming, Process analysis such as the Capability Maturity Model. As yet none of these themes have proved to be a 'silver bullet'

In the paper titled "*Natural Language Processing for Online Applications, Text Retrieval, Extraction and Categorization*" the authors describe that the emerging technologies of document retrieval, information extraction, and text categorization in a way which highlights commonalities in terms of both general principles and practical issues. It seeks to satisfy a need on the part of technology practitioners in the Internet space, faced with having to make difficult decisions as to what research has been done on what the best practices are. It is not intended as a vendor guide, or as a recipe for building applications.

In the paper titled "*A Survey of Text Clustering Algorithms*" the authors describe that Clustering is a widely studied data mining problem in the text domains. The problem finds numerous applications in customer segmentation, classification, collaborative filtering, visualization, document organization, and indexing. In this chapter, we will provide a detailed survey of the problem of text clustering. We will study the key challenges of the clustering problem, as it applies to the text domain. We will discuss the key methods used for text clustering, and their relative advantages. We will also discuss a number of recent advances in the area in the context of social network and linked data.

In the paper titled "*Similarity measures for text document were clustering*" the authors describe that Clustering is a useful technique that organizes a large quantity of unordered text documents into a small number of meaningful and coherent clusters, thereby providing a basis for intuitive and informative navigation and browsing mechanisms. Partitioned clustering algorithms have been recognized to be more suitable as opposed to the hierarchical clustering schemes for processing large datasets. A wide variety of distance functions and similarity measures have been used for clustering, such as squared Euclidean distance, cosine similarity, and relative entropy.

In the paper titled "*Bibliometrics to webometrics*" the authors Bibliometrics has changed out of all recognition since 1958; becoming established as a field, being taught widely in library and information science schools, and being at the core of a number of science evaluations research groups around the world. This was all made possible by the work of Eugene Garfield and his Science Citation Index. This resume reviews the distance that bibliometrics has travelled since 1958 by comparing early bibliometrics with current practice, and by giving an overview of a range of recent developments, such as patent analysis, national research evaluation exercises, visualization techniques, new applications, online citation indexes, and the creation of digital libraries. Web metrics, a modern, fast-growing offshoot of bibliometrics, is reviewed in detail. Finally, future prospects are discussed with regard to both bibliometrics and web metrics.

In the paper titled "*Revealing research themes and trends in knowledge management: From 1995 to 2010*" the authors describe that Visualizing the entire domain of knowledge and tracking the latest developments of an important discipline are challenging tasks for researchers. This study builds an intellectual structure by examining a total of 10,974 publications in the knowledge management (KM) field from 1995 to 2010. Document co-citation analysis, pathfinder network and strategic diagram techniques are applied to provide a dynamic view of the evolution of knowledge management research trends. This study provides a systematic and objective means in exploring the development of the KM discipline.

III. METHODOLOGY

Register: In this module the HR of the company as well as the candidates can register in the portal for performing various activities.

Login: This module performs the authentication of HR/Candidate and Admin.

Resume Upload: This module is responsible for allowing the candidate to upload the resume.

Data Cleaning of Resumes: The Data Cleaning algorithm is responsible for removal of stop words. Each of resumes are cleaned by removing the stop words from reviews. These are the set of words which do not have any specific meaning. The data mining forum has defined set of keywords which do not have any meaning like *a, able, about, across, after, all, almost, also, am, among, an etc*.

Tokenization of Resumes: Tokenization is a process of converting the clean data into a set of words known as tokens

Frequency Computation of Resumes: This is a process in which the frequency computation is performed. For each of the reviews the frequency is computed. Frequency is number of times a i^{th}

token appears in j^{th}. Resume.

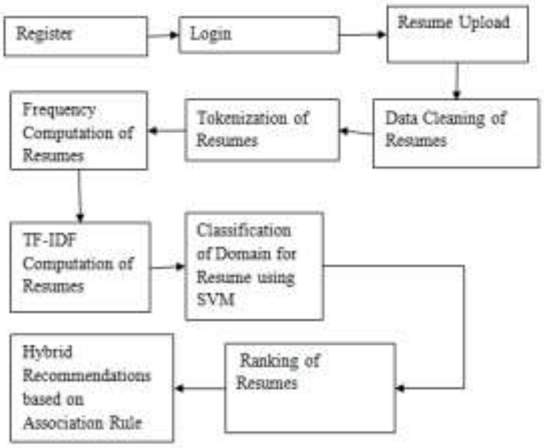

Register
Login
Resume Upload
Data Cleaning of Resumes
Tokenization of Resumes
Frequency Computation of Resumes
TF-IDF Computation of Resumes
Classification of Domain for Resume using SVM
Ranking of Resumes
Hybrid Recommendations based on Association Rule

Fig. 1. Dataflow diagram of resume sorting.

TF-IDF Computation of Resumes: This module is used to compute the Inverse document frequency based on the number of resumes and then frequency of the resume.

Classification of Domain for Resume using SVM: This module is responsible for training the support vector machine based on the test data set and then performs the attributes frequency. Find appropriate kernel and then classify the domain to which the resume mostly belongs to. The module also computes the probability and then classifies the domain to which the resumes belong to.

Ranking of Resumes: The entire query is divided into tokens and then frequency of those tokens across the various resumes is found and then finally the resumes are ranked based on descending order of the resume.

Hybrid Recommendations based on Association Rule Mining: This module is to combine multiple criteria of the resume and then rank the best resumes based on the requirements of multi attribute searches by doing intersection of the set of various algorithms.

IV. DESIGN

Resume Module

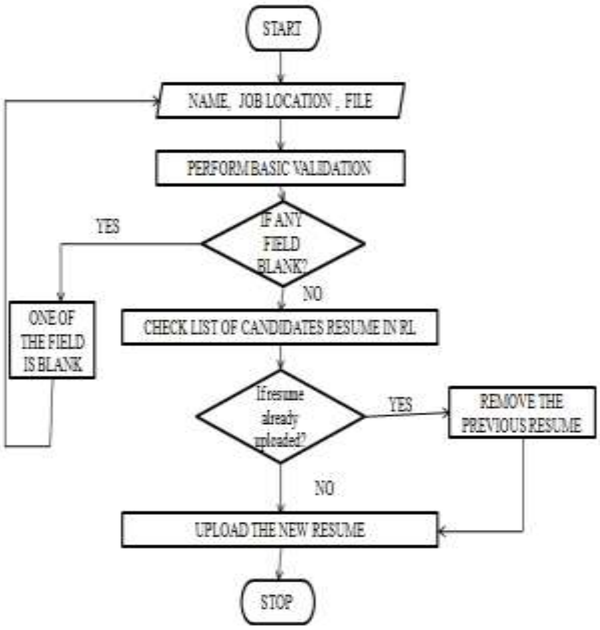

START
NAME, JOB LOCATION , FILE
PERFORM BASIC VALIDATION
IF ANY FIELD BLANK?
YES
NO
ONE OF THE FIELD IS BLANK
CHECK LIST OF CANDIDATES RESUME IN RL
If resume already uploaded?
YES
REMOVE THE PREVIOUS RESUME
NO
UPLOAD THE NEW RESUME
STOP

Fig. 2. Flow charts of resume upload.

The resume module is responsible for storage of resumes. Resume name and resume description acts as an input.

Data Cleaning Algorithm

The process of removing the stop words from the resumes is referred as data cleaning.

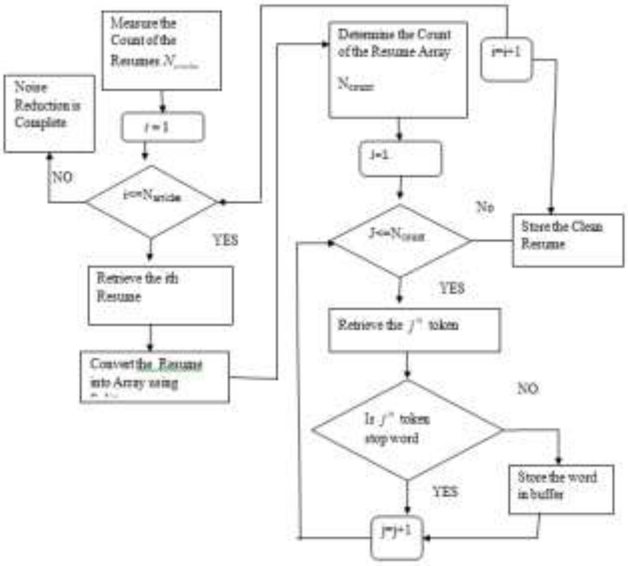

Measure the Count of the Resumes $N_{articles}$
Noise Reduction is Complete
i = 1
NO
i<=Narticles
YES
Retrieve the ith Resume
Convert the Resume into Array using Splitter
Determine the Count of the Resume Array
Ncount
j=1
J<=Ncount
No
YES
Store the Clean Resume
Retrieve the j^{th} token
Is j^{th} token stop word
NO
YES
Store the word in buffer
j=j+1
i=i+1

Fig. 3. Flow chart of data cleaning algorithm.

Tokenization

TABLE 1. Table for storing token words.

Token ID	Resume ID	Token Name

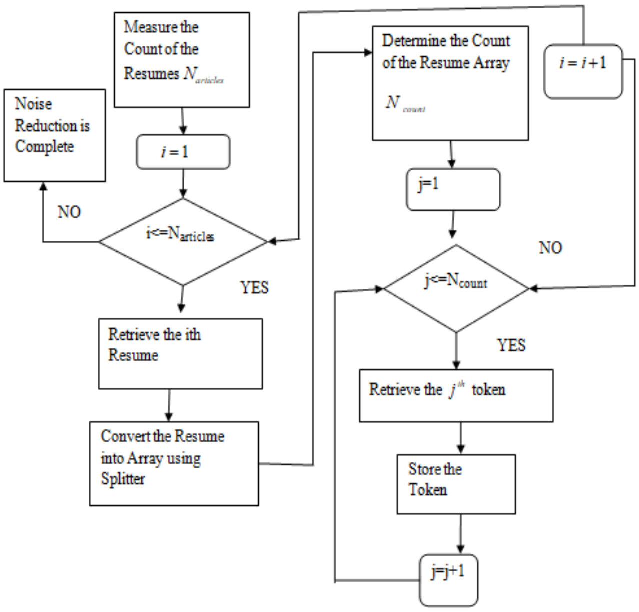

Measure the Count of the Resumes $N_{articles}$
Noise Reduction is Complete
$i = 1$
NO
i<=Narticles
YES
Retrieve the ith Resume
Convert the Resume into Array using Splitter
Determine the Count of the Resume Array
N_{count}
$i = i + 1$
j=1
NO
j<=Ncount
YES
Retrieve the j^{th} token
Store the Token
j=j+1

Fig. 4. Flow chart of Tokenization.

Frequency Computation

TABLE 2. Table for storing frequency of tokens.

Freq ID	Resume ID	Token Name	Frequency

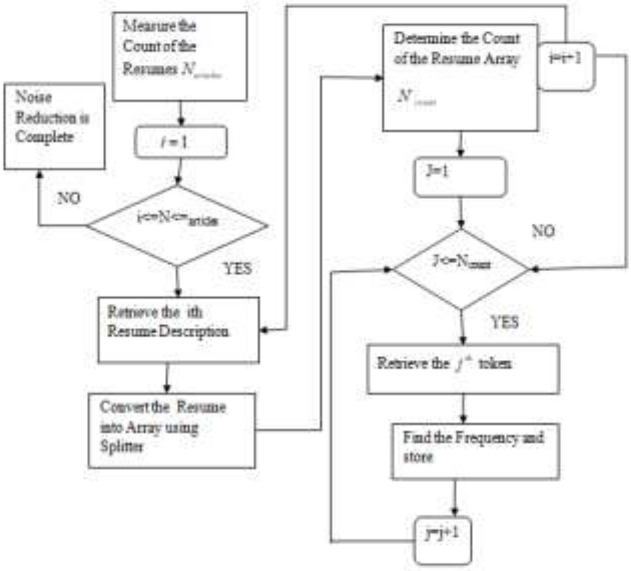

Measure the Count of the Resumes $N_{articles}$
Noise Reduction is Complete
i = 1
NO
i<=N<=articles
YES
Retrieve the ith Resume Description
Convert the Resume into Array using Splitter
Determine the Count of the Resume Array
N_{count}
i=i+1
J=1
NO
J<=Ncount
YES
Retrieve the j^{th} token
Find the Frequency and store
j=j+1

Fig. 5. Flow chart of frequency computation.

Feature Vector Computation

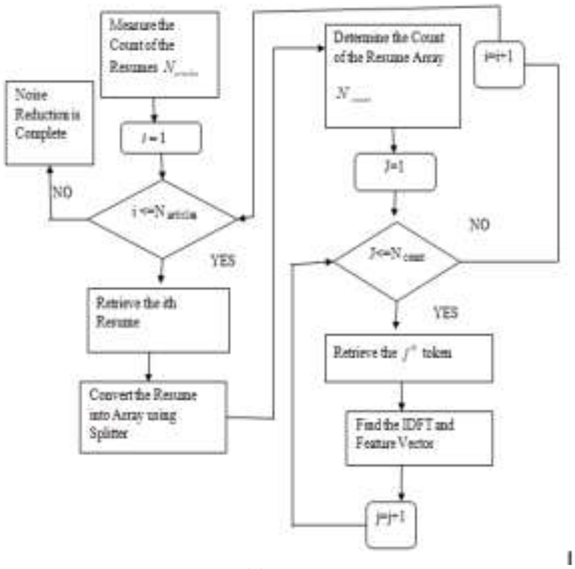

Measure the Count of the Resumes $N_{articles}$
Noise Reduction is Complete
$i = 1$
NO
$i <= N_{articles}$
YES
Retrieve the ith Resume
Convert the Resume into Array using Splitter
Determine the Count of the Resume Array N_{count}
i=i+1
J=1
NO
$J <= N_{count}$
YES
Retrieve the j^{th} token
Find the IDFT and Feature Vector
j=j+1

Fig. 6. Flow chart of feature vector computation.

The IDFT is computed using the following
IDFT= log (N/f)
Where,
N= Number of pages in which token is present
f= frequency of word

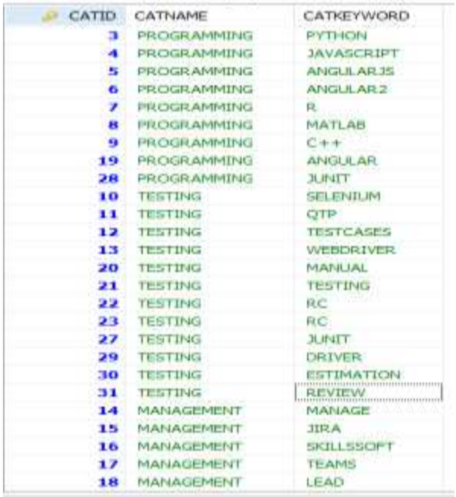
The Feature vector is computed using the following
$FV = f * IDFT$

Ranking of Resumes using TF-IDF

1. Divide the search string into words
2. For the list of unique resumes uploaded
3. Find the feature vector for each words of search and do a summation for the sequence of words for a resume
4. Repeat the process for all the resumes
5. Rank the resumes based on sorted order of the values

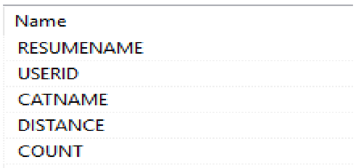
K Means Resume Classification

1. List of category along with training data set for each of the category is taken whose sample is as below



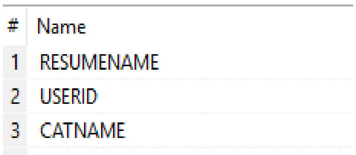
CATID	CATNAME	CATKEYWORD
3	PROGRAMMING	PYTHON
4	PROGRAMMING	JAVASCRIPT
5	PROGRAMMING	ANGULARJS
6	PROGRAMMING	ANGULAR2
7	PROGRAMMING	R
8	PROGRAMMING	MATLAB
9	PROGRAMMING	C++
19	PROGRAMMING	ANGULAR
28	PROGRAMMING	JUNIT
10	TESTING	SELENIUM
11	TESTING	QTP
12	TESTING	TESTCASES
13	TESTING	WEBDRIVER
20	TESTING	MANUAL
21	TESTING	TESTING
22	TESTING	RC
23	TESTING	RC
27	TESTING	JUNIT
29	TESTING	DRIVER
30	TESTING	ESTIMATION
31	TESTING	REVIEW
14	MANAGEMENT	MANAGE
15	MANAGEMENT	JIRA
16	MANAGEMENT	SKILLSSOFT
17	MANAGEMENT	TEAMS
18	MANAGEMENT	LEAD

2. For each of the category the word count is obtained for the resume
3. The distance is compute as maxValueCategory–resumecategory count
4. The following matrix is computed



Name
RESUMENAME
USERID
CATNAME
DISTANCE
COUNT

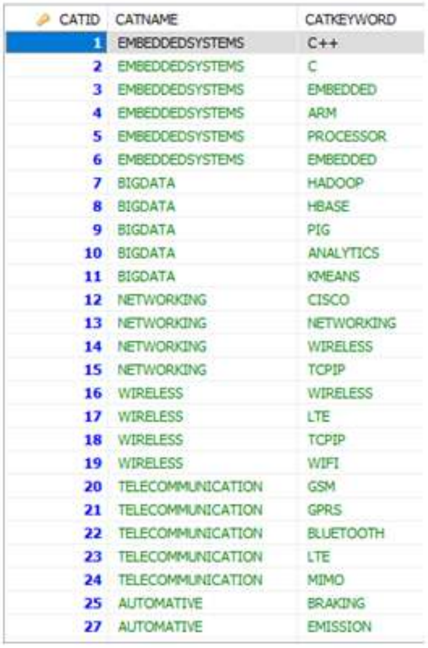
5. Finally the minimum distance is taken as the category for the resume and for each resume we compute the following



#	Name
1	RESUMENAME
2	USERID
3	CATNAME

SVM Classification based on probability

The classifier training vectors for the various domains are chosen

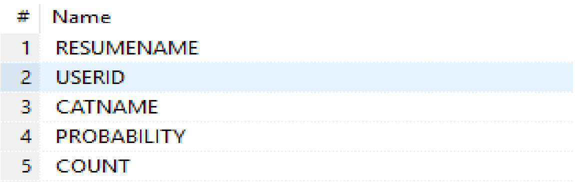


CATID	CATNAME	CATKEYWORD
1	EMBEDDEDSYSTEMS	C++
2	EMBEDDEDSYSTEMS	C
3	EMBEDDEDSYSTEMS	EMBEDDED
4	EMBEDDEDSYSTEMS	ARM
5	EMBEDDEDSYSTEMS	PROCESSOR
6	EMBEDDEDSYSTEMS	EMBEDDED
7	BIGDATA	HADOOP
8	BIGDATA	HBASE
9	BIGDATA	PIG
10	BIGDATA	ANALYTICS
11	BIGDATA	KMEANS
12	NETWORKING	CISCO
13	NETWORKING	NETWORKING
14	NETWORKING	WIRELESS
15	NETWORKING	TCPIP
16	WIRELESS	WIRELESS
17	WIRELESS	LTE
18	WIRELESS	TCPIP
19	WIRELESS	WIFI
20	TELECOMMUNICATION	GSM
21	TELECOMMUNICATION	GPRS
22	TELECOMMUNICATION	BLUETOOTH
23	TELECOMMUNICATION	LTE
24	TELECOMMUNICATION	MIMO
25	AUTOMATIVE	BRAKING
27	AUTOMATIVE	EMISSION

1. The probability is computed for each of the category using the following

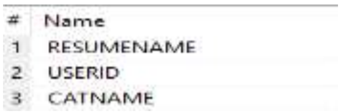
$$p(r \mid catname) = \frac{\text{Number of words of categ}}{\text{total words of resume}}$$

The following matrix is computed



#	Name
1	RESUMENAME
2	USERID
3	CATNAME
4	PROBABILITY
5	COUNT

2. Once the probability is computed for each of the resume
3. The highest probability is found
4. The class label is assigned based on the respective category which is highest



#	Name
1	RESUMENAME
2	USERID
3	CATNAME

Association Rule Mining

This is defined as intersection of criteria between skillset and domains along with ranking of feature vector

V. Conclusion and Future Scope

Conclusion

In this project 3 different actors have been used namely Candidate, HR and ADMIN. The candidate will be able to upload the resume. During resume upload sequence of datamining techniques datacleaning, tokenization, frequency computation, feature vector computation and also classification of skills using kmeans and domains using SVM is done. The candidate can even delete and upload new resume. The admin can view output of all the data mining techniques and classification output in the form of grid. The HR can register into the application and search based on association rule mining or based on the query. Once the search is performed the resumes are ranked based on the feature vector and domains and skills set related.

Future Scope

[1] The project can be future extended to include several more domains
[2] The project can be extended to support sentiment analysis if required.

References

[1] Junjie Wu, Advances in K-means Clustering, Springer-Verlag Berlin Heidelberg, 2012.

[2] Jure Leskovec, Anand Rajaraman, Jeffrey D. Ullman, Mining of Massive Datasets, Stanford Infolab, 2014.

[3] Michael Steinbach, Vipin Kumar, Pang-Ning Tan, Introduction to Data Mining, Pearson Publications, 2006.

[4] Yanchang Zhao, R and Data Mining: Examples and Case Studies, 2013.

Authors

1. Kavyashree M Bandekar presently associated with the Department of Computer Science Engineering at S.J.C. Institute of Technology (SJCIT) Chickballapur-562101, Chickballpur Dist., Karnataka, India as a student since 2014 to till date. kavyabandekar28@gmail.com

2. Maddala Tejasree presently associated with the Department of Computer Science Engineering at S.J.C. Institute of Technology (SJCIT) Chickballapur-562101, Chickballpur, Dist., Karnataka, India as a student since 2014 to till date. tejuprasad24@gmail.com

3. Misba Sultana. S.N presently associated with the Department of Computer Science Engineering at S.J.C. Institute of Technology (SJCIT) Chickballapur-562101, Chickballpur Dist., Karnataka, India as a student since 2014 to till date. misbasultanasn15@gmail.com

4. Nayana G K presently associated with the Department of Computer Science Engineering at S. J. C. Institute of Technology (SJCIT) Chickballapur-562101,Chickballpur Dist., Karnataka, India as a student since 2014 to till date. nayanagk28@gmail.com

5. Prof. Harshavardhana Doddamani presently working as Assistant Professor in the Department of Computer Science Engineering at S.J.C Institute of Technology (SJCIT) Chickballapur-562101, Chickballpur Dist., Karnataka, India 2009 to till date . hdoddamani@gmail.com