SMART E-ADMISSION SYSTEM WITH PSO SWARM SEARCH AND FEATURE SELECTION TECHNIQUE USING HADOOP

Asmita R. Kamble[1], Ajinkya Paigude[2], Sameer Shelar[3], Shubham Shah[4], Sumit Rahate[5]

[1] Professor, Computer Engineering Department, SITS, Pune , Maharashtra, India
[2]Student, Computer Engineering Department, SITS, Pune , Maharashtra, India
[3] Student, Computer Engineering Department, SITS, Pune , Maharashtra, India
[4] Student, Computer Engineering Department, SITS, Pune, Maharashtra, India
[5] Student, Computer Engineering Department, SITS, Pune, Maharashtra, India

ABSTRACT

Hadoop is very much suitable for high volume data and it also provides high speed access to the data of the application which we want to use. Using PSO Swarm Search and Feature Selection (FS) technique, which reduces the number of features, noisy, removes irrelevant and redundant data. As data coming from multiple university is vast in this includes storing it into database with proper manner and processing appropriate data in less time, so both this technology can make system smarter than present system. The objective of this paper is to design and implement a smart and efficient online admission system with more advance feature.

Keywords- *Hadoop, Feature Selection, PSO Swarm Search, Online Admission.*

I. INTRODUCTION

Student admissions are essential part of any universities working because students are what keep a University alive. The student admission is one of the most prominent exercises within a university as no one can survive without students. A poor admissions system may lead to fewer students being admitted into a university because of mistakes or an overly slow response time.

The increasing numbers of students looking for admission in the Universities are causing terrible pressure on the administrative body of the institutes to arrange and manage the admission process manually. It is not efficient now to handle the process accurately and in time. Hence, the need for online admission is necessary.

In case of a manual system, it is a time consuming process and involves immense manpower wherein the online admission system ensures precise and very fast computerized information processing. Maintaining backing up of data is also very easy using 'Online Admission System'. The aim of 'Online Admission System' is to automate the Academic Institute's admission structure and its related operation and functionality.

The objective of the initiative is to provide support to the administration and admission seeking candidates by providing a transparent, faster and easy way of keeping records and use them for reference and further proceeding. Online admission system present till date is not capable of handling large amount of data. Those system were time consuming and less efficient. In our system we are trying to overcome these drawbacks.

Hadoop is based on distributed computing having Hadoop Distributed File System (HDFS). Hadoop can be deployed on low cost hardware and highly fault-tolerant. So using Hadoop we can overcome those drawbacks as this technology can handle large amount of data and provide high speed access. Hadoop architecture is cluster based, which is consist of nodes i.e. Data note, name node, physically separate to each other, in ideal condition. The performance of Hadoop can be increased by proper assignment of the tasks in the default scheduler. In Hadoop a program known as map-reduce is used to collect data according to query. As Hadoop is used for huge amount of data therefore scheduling can improve better performance in terms of storing and processing.

The roles of each individual in the system are shown in below figure. The system has three tier access models. The jobs are divided among these three tiers. The jobs like registration to the system, apply for admission and document upload are done by student/applicant. Administrator is supposed to do the administrative jobs like set up criteria for seats, verification of documents, merit list generation etc.

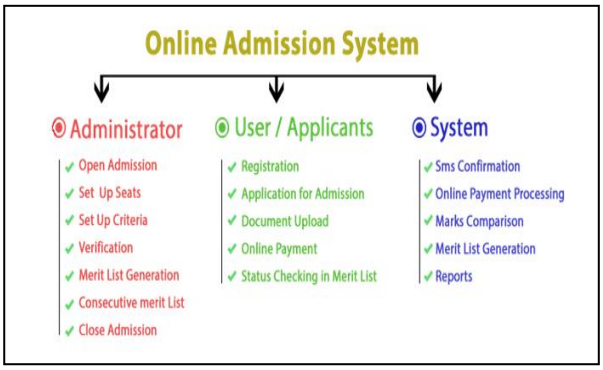
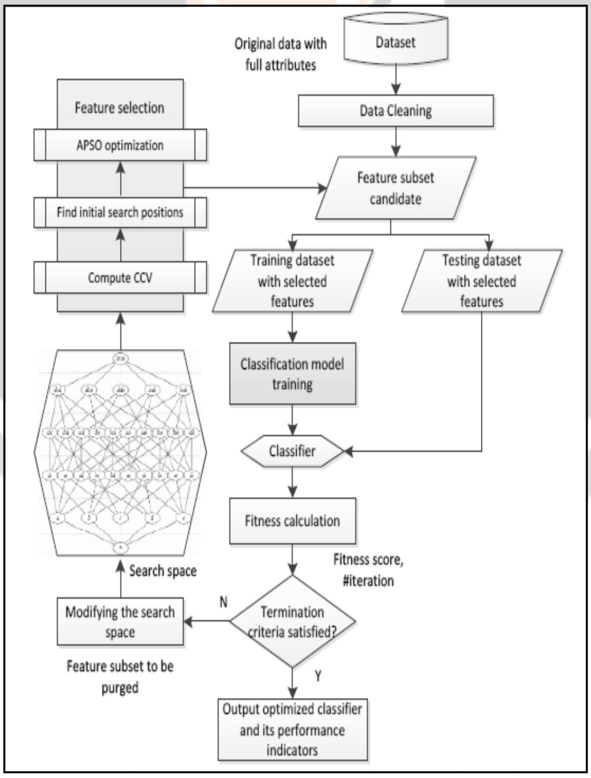


Fig -1 : Individual Roles

II. MATERIAL

1. PSO Swarm Search:

PSO (particle swarm optimization) first introduced by Kennedy and Eberhart. It was an optimization technique based on the intelligence and movement of the swarm. It inspired by social behaviour and dynamics of movement of fish and birds. PSO uses a number of particles that constitute a swarm moving around in the search space to find the best solution. Each particle is treated as a point in the search space which adjusts its flying according to its own flying experience and other particles flying experience. Initially, the PSO algorithm randomly selects candidate solutions (particles) within the search space. During each iteration of the algorithm, each particle is examined by the objective function being optimized, determining the fitness of the solution.


Original data with full attributes
Dataset
Data Cleaning
Feature subset candidate
Feature selection
APSO optimization
Find initial search positions
Compute CCV
Training dataset with selected features
Testing dataset with selected features
Classification model training
Classifier
Fitness calculation
Fitness score, #iteration
Search space
Modifying the search space
N
Termination criteria satisfied?
Feature subset to be purged
Y
Output optimized classifier and its performance indicators

.

Fig -2: Operation of Swarm Search Feature Selection with APSO

A contemporary type of feature selection algorithm, specially designed for choosing an optimal subset from a huge hyper-space is called Swarm Search-Feature Selection (SS-FS) Model. SS-FS is wrapper-based feature selection model which retains the accuracy of each trial classifier built from a candidate feature subset, chooses the highest possible fitness and considers the candidate feature subset as the choice output. The actual flow of the SS-FS Model is shown in Figure 2. It can be seen that the operation iterates beginning from a random selection of feature subset, remains to refine the accuracy of the classification model by looking for a preferable feature subset, in random manner. The flow makes capable the classification model and the chosen feature subset finally converges.

2. HADOOP:

As we know there is a lot of data surrounds us but we can't make that data beneficial to us. There is no capable tool that can make out analysis of information from this huge amount of data.

Hadoop is based on distributed computing having HDFS file system (Hadoop Distributed File System). Hadoop is very much appropriate for high volume of data and it also gives the high speed access to the data of the application. Cluster based architecture of Hadoop, which consist of nodes (data note, name node), physically separate to each other, in ideal situation. The performance of hadoop can be improved by proper allotment of the tasks in the default scheduler. In hadoop a program known as map-reduce is used to gather data according to query. As hadoop is used for huge amount of data therefore scheduling in hadoop must be efficient for desirable performance.

Hadoop Architecture is based on HDFS, which is hadoop distributed file system. In which data is equally (ideally) divided into each node in the hadoop system. When a client want to fetch, add, modify or delete some data from Hadoop, then hadoop system gather the data from each node of our interest and do the knowledgeable actions of our interest.

Name Node is a special kind of master node, which is having the self-information or we can say that meta-data about the all data node there is address, free space, data stored, active & passive data node, task tracker, job tracker and many other configuration such as data replication.

Data Node is a type of slave node in the Hadoop, which is used to store the data and there is task tracker in data node which keep an eye on the on-going job on the data node and the jobs which are coming from name node.

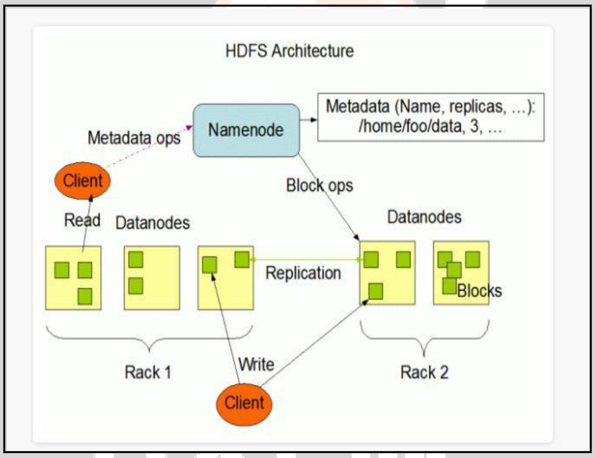


Fig -3: HDFS Architecture

III. METHODOLOGY

In our project we are developing a smart and efficient system for online admission process. Hadoop will be used for database handling. Using PSO swarm search & Feature selection redundancy & duplication in data is avoided which in turn increases speed and efficiency of system.

In following figure 4, the architecture of the system is given. Hadoop DB handles all the database related tasks. OLTP is used for securing the online transactions. OLTP itself is a secure protocol. Information analysis and processing takes care of processing of data. The data is submitted online via web page.

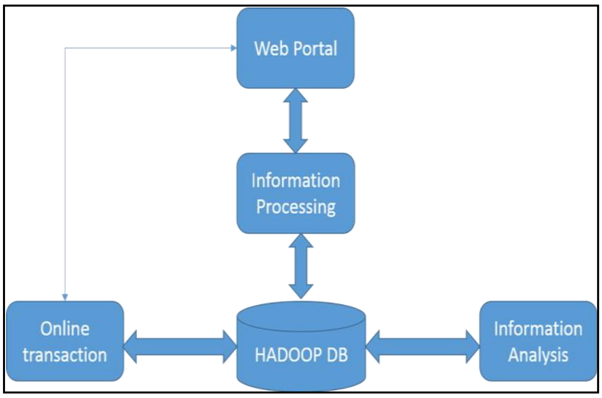


Fig -4: System Architecture

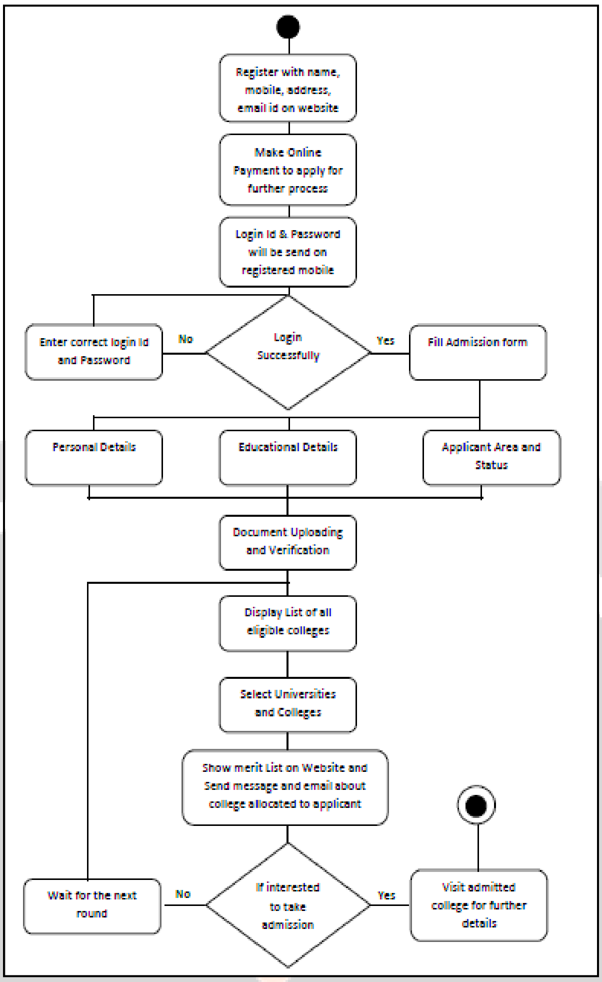

Register with name, mobile, address, email id on website
Make Online Payment to apply for further process
Login Id & Password will be send on registered mobile
Enter correct login id and Password
No
Login Successfully
Yes
Fill Admission form
Personal Details
Educational Details
Applicant Area and Status
Document Uploading and Verification
Display List of all eligible colleges
Select Universities and Colleges
Show merit List on Website and Send message and email about college allocated to applicant
Wait for the next round
No
If interested to take admission
Yes
Visit admitted college for further details

Fig -5: flowchart of the system

As shown in figure 5, the applicant is supposed to register him on a site. After successfully logging in to the system, they are supposed to fill the form. The personal and educational details are stored on Hadoop database. Based on their merit, list of colleges will be displayed for which they can apply. Applicant can select the colleges to which they want take admission. Next student should upload all documents. These documents are verified for measuring their truthfulness. After this entire process fee for forms will be paid by the applicants online. Secure transaction processing is used for making transaction secure. After some days merit list will be displayed. It might be the case that some of the students might not get admission in any colleges. So these students should apply for the second round. Rest students should go to their respective colleges and take admission. Hence admission process is completed successfully.

IV. RESULTS AND TABLES

The system is expected to give following results after successful execution:

1) Student Admission System gives the admission to the student.

2) Maintains student personal, academic and fee related data.

3) Marks details & verifications of filled forms for eligibility checking.

4) Generate Student’s Personal Detail Report.

5) Generate Student’s Fee Deposition Status Report.

6) It Stores Merit list provided by Universities.

7) The time required for whole admission process is reduced by huge extent.

8) Efficient data processing by removing redundant data.

9) Preparation of merit list for each category and each course.

10) Sending call letters to online users on e-mail/Mobile.

In the table given below present system and our system are compared. As given above, various drawbacks of present system are overcome in our proposed system.

Sr. No	Present System	Proposed System
1	Older system was not on Hadoop	The system will be in Hadoop
2	OLTP (Online Transactions were not present)	Online transaction were present
3	Manual Scheduling was taking place	Automatic Scheduling was taking place
4	Reduce efforts of the university	Reduces efforts of the university more than previous system.
5	Results will be displayed in more time.	Results will be displayed in less time.
6	Commission system is still present	Commission system will be eliminated

Table No.1 Comparison between Our system and Present system

V. CONCLUSION

Big Data grows continually with fresh data are being generated at all times; hence it requires an incremental computation approach which is capable of monitoring large scale of data dynamically. Since managing data from multiple universities is time consuming process. So, lightweight incremental algorithms should be used which are capable of achieving robustness, high accuracy and minimum pre-processing latency. In this paper, we designed a system which handles and process tremendous amount of data at the same time giving accurate and efficient results.

VI. ACKNOWLEDGEMENT

We would like to thank our project guide Prof. Asmita R. Kamble and all the staff members of our department for giving us valuable time and contributing their experience towards this project.

VII. REFERENCES

1) Simon Fong, Raymond Wong," Accelerated PSO Swarm Search Feature Selection for Data Stream Mining Big Data ", DOI 10.1109/TSC.2015.2439695, IEEE Transactions on Services Computing

2) Quinlan, J.R., C4.5: Programs for Machine Learning. Morgan Kauffmann Publishers, 1993

3) Ping-Feng Pai, Tai-Chi Chen, "Rough set theory with discriminant analysis in analysing electricity loads", Expert Systems with Applications 36 (2009), pp.8799–8806

4) Mohamed Medhat Gaber, Arkady Zaslavsky, Shonali Krishnaswamy, "Mining data streams: a review", ACM SIGMOD Record, Volume 34 Issue 2, June 2005, pp.18-26

5) Wei Fan, Albert Bifet, "Mining Big Data: Current Status, and Forecast to the Future", SIGKDD Explorations, Volume 14, Issue 2, pp.1-5

6) Suman Arora," Survey Paper on Scheduling in Hadoop", ISSN: 2277 128X.

7) Job Scheduling for Multi-User Map Reduce Clusters, Matei Zaharia, hruba Borthakur, Joydeep Sen Sarma, Khaled Elmeleegy, Scott Shenker, Ion Stoica; Electrical Engineering and Computer Sciences, University of California at Berkeley.