Automated Integration of Heterogenous databases

Author

Mamoona Safdar

FALL 2016-MS-16(CSE) 00000171508

MS-16 (CSE)

Supervisor

Dr. Urooj Fatima

DEPARTMENT OF COMPUTER & SOFTWARE ENGINEERING

COLLEGE OF ELECTRICAL & MECHANICAL ENGINEERING

NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY

ISLAMABAD

AUGUST, 2020

Automated Integration of Heterogeneous databases

Author

Mamoona Safdar

FALL 2016-MS-16(CSE) 00000171508

A thesis submitted in partial fulfillment of the requirements for the degree of

MS Software Engineering

Thesis Supervisor:

Dr. Urooj Fatima

Thesis Supervisor's Signature:____

DEPARTMENT OF COMPUTER & SOFTWARE ENGINEERING
COLLEGE OF ELECTRICAL & MECHANICAL ENGINEERING
NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY,
ISLAMABAD
AUGUST, 2020

DECLARATION

I certify that this research work titled "Automated Integration of Heterogeneous databases" is my own work under the supervision of **Dr. Urooj Fatima** &**Dr. Wasi Haider Butt**. This work has not been presented elsewhere for assessment. The material that has been used from other sources; it has been properly acknowledged/referred.

Signature of Student

Mamoona Safdar

FALL 2016-MS-16(CSE)00000171508

LANGUAGE CORRECTNESS CERTIFICATE

This thesis is free of typing, syntax, semantic, grammatical and spelling mistakes. Thesis is also according to the format given by the University for MS thesis work.

Signature of Student

Mamoona Safdar

FALL 2016-MS-16(CSE)00000171508

Signature of Supervisor

COPYRIGHT STATEMENT

ACKNOWLEDGEMENTS

I thank Almighty Allah (swt) my Creator Allah for his ultimate guidance throughout my research. Nothing would have been possible without his profound blessing. For all praise is due to God, the Sustainer of all the worlds. Also, my admirations be upon Prophet Muhammad (PBUH) and his Holy Household for being source of guidance for people.

I would like to my show gratitude to my supervisor **Dr. Urooj Fatima** and my **co-supervisor Dr. Wasi Haider Butt** for their tremendous support and cooperation whose constant motivation, persistent efforts and uninvolved words of wisdom ever proved a lighthouse for me. Despite they never-ending commitments, he did never mind giving them maximum whenever I requested for their time and support.

I would also like to thank my Guidance Committee Members **Dr. Muhammad Usman Akram** and **Dr. Arsalan Shaukat** for being on my thesis guidance and evaluation committee. Their recommendations are very valued for improvement of the work. I appreciate their guidance throughout the whole thesis.

Finally, I would like to express my gratitude to my family and all the individuals who have rendered valuable assistance to my study.

.

Dedicated to my exceptional parents and family whose tremendous support and cooperation led me to this wonderful accomplishment

ABSTRACT

In integration of heterogeneous databases, data integrate from different sources. It is very challenging task because data model and representation of data varies in different relational databases. It will be more complicated when we are talking about relational for example SQL (Structured Query Language) and non-relational for example NoSQL (not only SQL) databases. In past, researchers focused on integration of different relational databases. Now a days Integration of SQL and NoSQL become an important issue because of popularity of NoSQL. Until now, various techniques of *supervised machine learning algorithms* have been introduced to solve the problem of heterogeneous database integration. Every method perform integration in its own unique way. we are introducing *unsupervised machine learning algorithms* to perform integration. The main idea of this approach is to integrate relational and non-relational database for increasing the efficiency of data by using unsupervised machine learning algorithms. So, we don’t need to train and supervise our dataset. The proposed approach is to first get data from Mongo DB and apply clustering on that data by set centroid values. The algorithm is than represent clusters with different color. Each cluster represent specific table of SQL database. We would also explore best machine learning algorithm by comparing different algorithms based on accuracy. We only used K-means, spectral, agglomerative and mean shift algorithms. For validation of clustering of each algorithm we used confusion matrix. The proposed approach has been validated through multiple case studies.

Therefore, there is a gap between supervised machine learning algorithm techniques and unsupervised machine learning algorithm techniques. So, there is need to provide an unsupervised level solution to automatically integrate NoSQL to SQL databases to overcome research gap. We have proposed this solution for integration of relational and non-relational databases through unsupervised machine learning algorithm. This automatically predict similar data of non-relational database in the form of clusters so we can represent entities of relational database.

TABLE OF CONTENTS

DECLARATION .... **i**

LANGUAGECORRECTNESSCERTIFICATE .... **ii**

COPYRIGHTSTATEMENT .... **iii**

ACKNOWLEDGEMENTS .... **iv**

ABSTRACT .... **vi**

TABLEOF CONTENTS .... **vii**

LISTOF FIGURES .... **ix**

LISTOF TABLES .... **x**

CHAPTER1: INTRODUCTION **12**

1.1. Overview 12

1.1.1. Integration of Heterogenous Databases 13

1.1.3. Machine Learning 14

1.1.4. Unsupervised Learning and Clustering 15

1.2. Problem Statement 15

1.3. Proposed Methodology 16

1.4. Research Contribution 17

1.5. Thesis Organization 17

CHAPTER 2:LITERATUREREVIEW **20**

2.1. Research methodology 20

2.1.1. Platform to access SQL and NoSQL database systems uniformly 20

2.1.2. Research Questions 20

2.1.3. Selection and Rejection Criteria 21

2.1.4. Search Procedure 22

2.1.5. Quality Assessment 25

2.1.6. Data Extraction and Synthesis 27

2.2. Analysis of data 27

2.2.1. Classification of data 27

2.2.2. Approaches for Integration of heterogeneous databases 27

2.2.3. Tools Analysis for Heterogenous Database Integration 29

2.3. Relevant Study 30

CHAPTER 3:PROPOSEDMETHODOLOGY **41**

3.1. Clustering **41**

3.2. K-means **41**

3.2.1. K-means convergence(stopping) criterion 42

3.2.2. Proposed solution with example 43

3.3. Agglomerativeclustering **45**

3.3.1. Proposed solutionwith example 46

3.4. Meanshift 48

3.4.1. Proposed solutionwith example: 49
3.5. SpectralClustering 49
3.6. SelectionCriteria 50
CHAPTER4: IMPLEMENTATION 52
4.1. Databasecreation 52
4.1.1. School Management System 52
4.2. Implementation: 55
4.2.1. Implementationof K-means 55
4.2.2. Implementation of Agglomerative Clustering 56
4.2.3. Implementation of Spectral Clustering 56
4.2.4. Implementation of MeanShift: 57
CHAPTER 5: RESULTSAND DISCUSSION 59
5.1. Results 59
5.2. Comparison 60
5.2.1. K-meansalgorithm 60
5.2.1.1. Accuracy 61
5.2.1.2. Confusion Matrix for K-means 61
5.2.2. AgglomerativeClustering 62
5.2.2.1. Accuracy 62
5.2.2.2. Confusion Matrix 62
5.2.3. SpectralClustering 63
5.2.3.1. Accuracy 63
5.2.3.2 Confusion Matrix 63
5.2.4. MeanShift clustering 64
5.2.4.1. Accuracy 64
5.2.4.2. Confusion Matrix 64
CHAPTER 6: CONCLUSION ANDFUTUREWORK 66
6.1. FutureWork 66
6.2. Limitations 66

LIST OF FIGURES

Figure 1.1: Overview of research study..16
Figure1.2: Workflow 26
Figure 1.3: Thesis structure ....…26
Figure 2.2: Selected researchesperpublisher..26
Figure 3.1: Randomly initialize the cluster centers(1stiteration)..43
Figure 3.2: Determine distance from cluster centers 44
Figure 3.3: Results of 1st iteration...44
Figure 3.4: 2nd iteration...45
Figure 3.5: Result of 2nd iteration 45
Figure 3.6: Make each data pointacluster 46
Figure 3.7: From two closest clusters make themonecluster 47
Figure 3.8: Get one cluster 47
Figure 3.9: Dendrograms..48
Figure 3.10: Meanshift 49
Figure 4.1: ERD of School Management System 53
Figure 4.2: Clustering of K-means 60
Figure 4.3: Confusion Matrix for K-means 61
Figure 4.4: Agglomerative classifier..62
Figure 4.5: Spectral Clustering...63
Figure 4.6: MeanShift Clustering 64

LIST OF TABLES

Table 1: Details of research worksperdatabase..22
Table 2: Search terms and Search results details..24
Table 3: Details of extraction and synthesis of data..27
Table 4: Identification of integration of heterogenous databases approaches..28
Table 5: Identification of Tools for database integration..29
Table 6: Selected studies to perform Integration..30
Table 7: Terms used in this section..43
Table 8: Comparison of algorithms based on accuracy……………………………………44

Chapter 1

Introduction

CHAPTER 1: INTRODUCTION

This chapter gives comprehensive information of the thesis topic. It is divided into following portions. **Section 1.1** provides the overview of Integration of Heterogenous databases, **Section 1.2** explains the problem statement being addressed in dissertation**, Section 1.3** consists of research flow that is followed to complete the research, **Section 1.4** is refer to the research contribution and **Section 1.5** has the thesis organization**.**

1.1. Overview

Data is any kind of information which we need to store. A database is used to store data in way that we can easily manage data and retrieve data. we can store data in two ways **Structured** and **Document-like storing.** In structured, data organize into rows and columns. So, we have data that are stored or organized in one table or multiple tables which are interconnected. When all data stored in a single document without relations, this storing is called **Document-like storing**. This type of storing contains structured data that are stored in the form of an array. A relational database is generally implemented using SQL (Structured Query Language). **Relational databases** will be preferred if we are working on website which requires well-structured data [29]. For example, SQL is better choice if we are dealing with transactional data because it will be easily managed if data are stored in well managed relations, in the case of e- commerce platforms will also profit from the strict structure. The main disadvantage of an SQL database is,it needs well develop the architecture before adding data. Common examples of relational databases include PostgreSQL, MySQL, MS SQL Server, SQLite. A non-relational, or NoSQL database, works contrarily. It deals with semi-structured data. Each entry store in a JSON format. So, NoSQL database store data in the form of folders with files rather than a table. The main disadvantage is, it requires high cost for its flexibility with extra processing efforts, but it is very useful where we require various data types, the application has to handle like Social media, analytics software. Commonly used NoSQL databases are MongoDB, DynamoDB, Elasticsearch, HBase, Redis. When we need Non-relational databases for increasing performance and scalability along with relational database, there is a big challenge to integrate **SQL** and **NoSQL** databases because of many conflicts [9].

Our main objective is to create database in MongoDB. Then apply different algorithms for the separation of similar information into groups. These groups are basically representing tables of SQL.In last, would compare the accuracy of different algorithms and choose the best for integration. All details of this would be discussed in chapter 3 and 4.

1.1.1. Integration of Heterogenous Databases

Heterogeneous databases mean data store in multiple sources. integration means aggregation of information from multiple sources. It can be very lengthy and exhausting activity. It is evident that integration of databases has extensive need of today’s business, but it is complicated process. One of the significant issue is difference of model of source databases. This problem has given rise to series of researches to make integration at a different level. These different studies have tried to resolve the problem, using variety of techniques aiming to provide effective results. As discussed above, one way to perform integration through clustering algorithm and the other way is to devise some techniques that can integrate different databases [1][12][16]. Some of them are discussed in the next section.

1.1.2. Traditional Techniques

Usually relational database has been used to save, process and retrieve data. These databases are mostly SQL, ORACLE etc. as discussed in section 1.1. But before 1970, other types of databases were introduced like hierarchical, network, graph. These databases are called NoSQL or we can say Non-relational databases. These databases are used by world's largest organizations such as Google, Amazon and Facebook due to its advantages[31].

Database integration can be done at different levels [10]. The techniques can be grouped at different levels as follow:

1. Database integration done by manually
2. Providing the user interface for performing integration
3. Evolving an application for performing integration
4. Using a middleware GUI
5. How we can apply integration through uniform data access
6. Integration through shared data storing.

The above first four techniques provide simple solutions, but each approach has its own advantages and significant draw backs.It is very time consuming process if we done integration manually.It also very costly process. An integrated application can access large number of databases and return merged required as a result to the user. The main disadvantage of this solution is that if we increase number of source database, the system becomes more complex because it can manage only those databases which are already integrated into them. In the case of shared user interfaces, data is presented distinctly, and database integration manually done by user [31].There are some other techniques for database integration one of them is known as CORBA to solve the interoperability among different software and hardware products of relational databases [1].The other framework is implemented as a software system which we call DHResol. This system merge two different databases(relational) to recognize and resolved different levels of conflicts. Another technique that is still in practice involves hybrid approach. This technique-built relationship between relational database schema and ontology. This paper describes how to calculate the structure similarity. All these traditional techniques are only for relational databases integration.

1.1.3. Machine Learning

It is significant field of technology that gives learning power to systems. The idea behind machine learning is to create intelligent algorithms that can be trained from any given set of data. Based on which, it can predict useful results. Machine learning has emerged from pattern recognition and has some conceptual basis from artificial intelligence. It is also related to mathematics and statistics. Nowadays, it has wide applications in variety of tasks that involve complex calculations and programming [81][83]. Our day to day use applications such as voice recognition systems, social media facilities, video and audio surveillance, email filtering, finding online frauds, etc. largely revolve around machine learning techniques [32].

1.1.4. Unsupervised Learning and Clustering

Clustering is the process in which grouping of data take place based on certain features. Most unsupervised learning-based applications utilize the sub-field called clustering. The goal of clustering is to find homogeneous subgroups within the data; the grouping is based on the distance between observations.

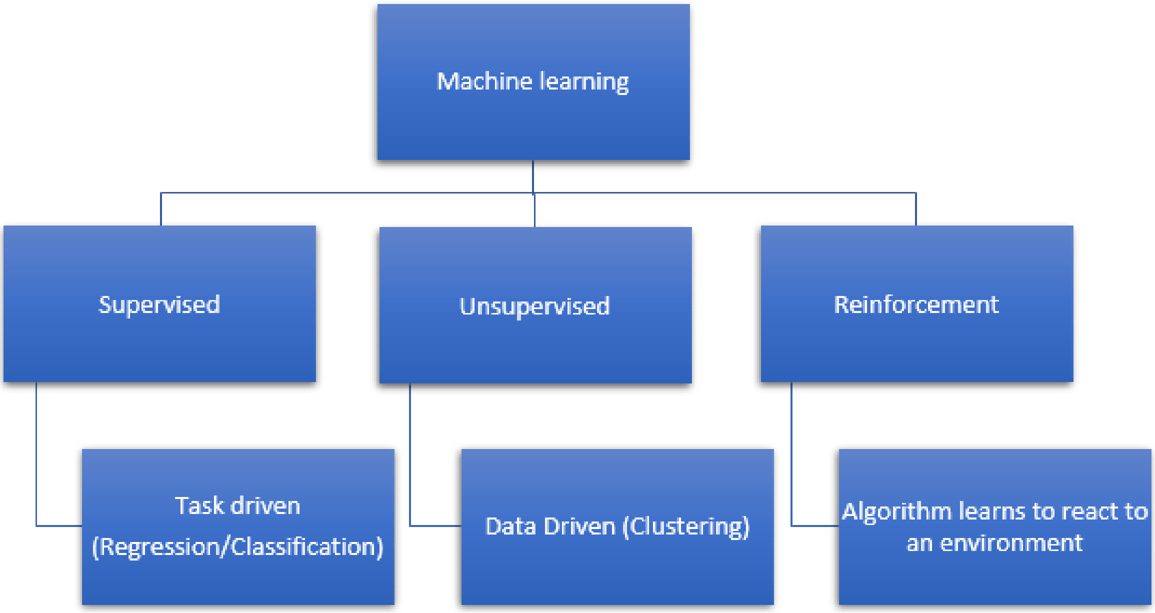

Machine learning
Supervised
Unsupervised
Reinforcement
Task driven (Regression/Classification)
Data Driven (Clustering)
Algorithm learns to react to an environment

Figure 1.1: Research study overview

1.2. Problem Statement

As discussed earlier in this chapter that integration of heterogeneous databases is very challenging task. It gained attention of researchers because of popularity of NoSQL increases day by day but we need SQL also because of its properties. So, integration of NoSQL with SQL hide numerous difficulties like model and structure of both type of databases is different. Different techniques have ben apply for their integration but either they are manual or supervised.

Therefore, there is a gap between supervised machine learning algorithm techniques and unsupervised machine learning algorithm techniques. So, there is need to provide an unsupervised level solution to automatically integrate NoSQL to SQL databases to overcome research gap of using unsupervised machine learning algorithms in integration.

1.3. Proposed Methodology

The research process is carried out in a systematic way as explained in **Figure 1.1**. For any research, the first step is problem identification. Once the problem is identified, we move to the problem-solving phase. Next to the proposed solution, a detailed study is carried out to develop a systematic literature review. The literature review covers the study of related work. The proposed methodology provides a clustering approach to integrate heterogenous databases.

For this purpose, unsupervised machine learning algorithms are used for clustering. Input database containing collection of NoSQL. Then implementation is performed, and results are validated through case studies. At next we find out the accuracy of different algorithms and compare their accuracy for choosing the best unsupervised clustering algorithm for integration. Finally, the conclusion is drawn from results and future work is given.

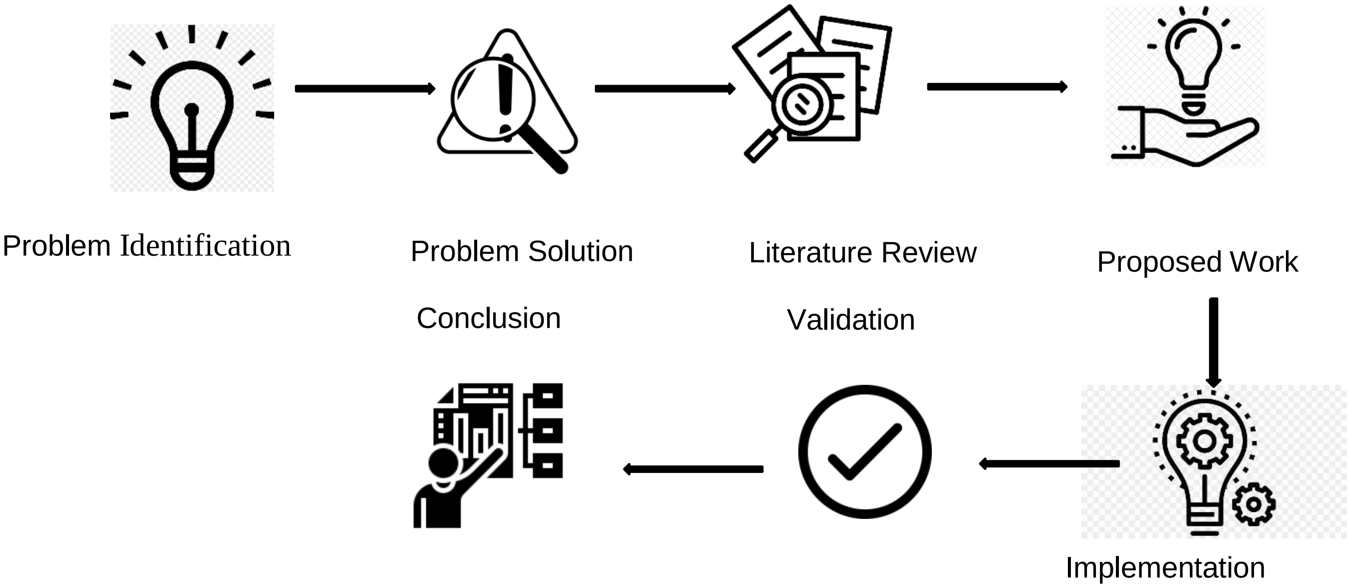


Figure 1.1: Workflow

1.4. Research Contribution

This research is performed to integrate database of SQL and MongoDB. A machine learning approach is followed to improve the process. Following are the main contributions of this work:

- Apply unsupervised machine learning algorithms on collection of NoSQL.
- Get clusters of similar data
- These clusters represent entities of SQL
- For finding whether we get true values in clusters or not we use confusion matrix
- Calculate accuracy of each algorithm.
- Compare all algorithms based on accuracy.
- Choose the best for integration.

There's wide range of literature available for integration of heterogenous databases, each using different techniques and methods while targeting the same goals. The work of this dissertation aims to provide improved, robust, and efficient results using probabilistic approach.

1.5. Thesis Organization

Thesis organization can be discussed using **FIGURE 3. CHAPTER 1** goes through brief introduction enlisting the overview and background knowledge of topic, problem statement, objectives and contributions and then thesis organization. **CHAPTER 2** has the detailed literature review. It specifies the work done in the field of integration of heterogeneous databases. **CHAPTER 3** presents the overview of methodology proposed. **CAHPTER 4** has the implementation details. MongoDB dataset formation is done by forming the coverage information and execution results. This information is given to develop clusters. As a result, we represent clusters as a table of SQL. **CHAPTER 5** gives results and comparison. It also discusses the limitations and drawbacks of a research. **CHAPTER 6** finally concludes the work and mentions future work for the research. **FIGURE 3** provides the structure of thesis.

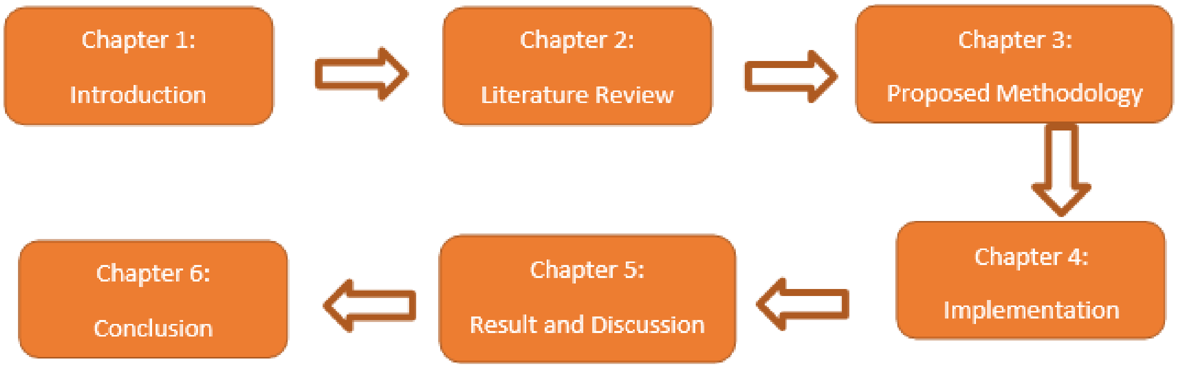


Figure 1. 2: Thesis Organization

Chapter 2

Literature Review

CHAPTER 2: LITERATURE REVIEW

This chapter presented literature review in detail. **Section 2.1** contains the research procedure, **Section 2.2** discusses the detailed analysis of selected papers on the basis of tools used, and approach followed, Section **2.3** presents the relative study.

2.1. Research methodology

The research for literature review is performed in an organized way following the series of steps. Few categories are broadly defined to help in searching and sorting of articles. Selection and rejection criteria are described to develop a screening process for the papers. Quality assessment, data synthesis and extraction are also explained in upcoming sections.

2.1.1. Platform to access SQL and NoSQL database systems uniformly:

To organize the research process, some categories are discussed. Identified papers fall under these categories. There are no hard and fast rules for category definition, and they are not mutually exclusive. There is fair chance of intersection between these categories.

This chapter presents the literature review conducted for the research. Literature review is performed in two steps. The main purpose of this research investigation is to simplify the integration between relational and non-relational databases. At first step, we analyzed various research articles to elicit the previous work performed in the integration of different databases. First step concluded that, no work has been presented for automatically integration of relational and non-relational databases. Moreover, in second step a comparison of all techniques mentioned in the literature in tabular form along with limitations of every technique.

2.1.2. Research Questions:

Research Questions that are considered in this section are as follows:

RQ1: Why we need integration in databases?

RQ2: What are the latest trends in database integration reported in 2010-2020?

RQ3: What are the challenges of integration of heterogeneous databases reported in 2010-2020?
RQ4: What are the techniques used for integration of heterogeneous databases?

To answer above questions, we have performed Systematic Literature Review. 33 research papers [1-33] are selected from 4 scientific libraries IEEE, SPRINGER, ACM AND ELSEVIER published from 2010 to 2020.

2.1.3. Selection and Rejection Criteria

To establish the literature review in systematic way, some conditions are imposed to bring quality in work. These conditions are criteria for including certain papers and excluding others. So, any research paper that is considered for literature study must fulfill these parameters in order to get resourceful information. They are discussed below in detail.

1. ***Topic relevancy:*** First and foremost, principal for article selection is its relevance to the subject under study. Only papers that are dealing with integration of heterogenous databases are added in the literature. Moreover, matters that go out of the scope of integration of heterogeneous databases are eliminated from study.
2. ***Year of publication:*** To make sure that the literature of this dissertation consists of state-of-the-art techniques, papers only from the year 2010 – 2020 are inclusive. Whereas researches that are not from this time period are rejected.
3. ***Publishers:*** Besides taking care of topic relevancy and year of publication, publishers are also added to selection and rejection criteria. Papers from four famous scientific databases including IEEE, ACM, ELSEVIER, and SPRINGER are extracted. Purpose of adding this rule is to bring the authenticity and validity in the literature. **Table 1** shows the details of different papers against their publishers.
4. ***Research effects:*** Another factor to keep in mind while selecting papers is to check the impact of work done in the relative field. Work with significant research and immense influence is the part of literature. Rest of the study is ignored.
5. ***Experimentation and Facts:*** Researches supported by strong experimentations and proven facts are part of this literature. Results based on some hypothetical claims are not considered justified and truthful. Special caution is taken while choosing the paper.
6. ***Repetition:*** Any finding that is based on some redundant studies is rejected to be a part of review. Only significant, non-redundant work can be added here.

7. ***Automation or tool usage:*** Techniques that involve some sort of automation or tools are added for pursuing the research. Manual ways of performing integration are not encouraged.

Table 1: Research Works Per Database Details

Sr. #	**Scientific Database**	**Type**	**Selected Research Works**	**No. of Researches**
1.	IEEE	Journal	[17][26][10][32]	4
		Conference	[1][2][3][12][13][15][18][21] [25]	9
2.	ACM	Journal	[4][7][30]	3
		Conference	[14][17][6][16]	4
3.	ELSEVIER	Journal	[23][5][19][31]	4
4.	SPRINGER	Conference	[8][9][24]	3
5.		Journal	[11][20]	2
6.	Others	Conference	[27]	1
7.		Journal	[22][28][29]	3
Total				33

2.1.4. Search Procedure

As it is already recognized that search process followed in this literature is systematic. Some rules are defined to make customize the process and improve its quality. Research is confined to the time period of 2010 to 2020. In between these years, papers from the four publishers (namely IEEE, ACM, Springer, and Elsevier) are selected to maintain the tone of literature. Some of the search terms are declared (such as SQL and NoSQL integration, Machine Learning unsupervised algorithms) to further enhance the results. They are given in detail in **Table 2**. The use of operators (AND, OR) helps refine the search process.

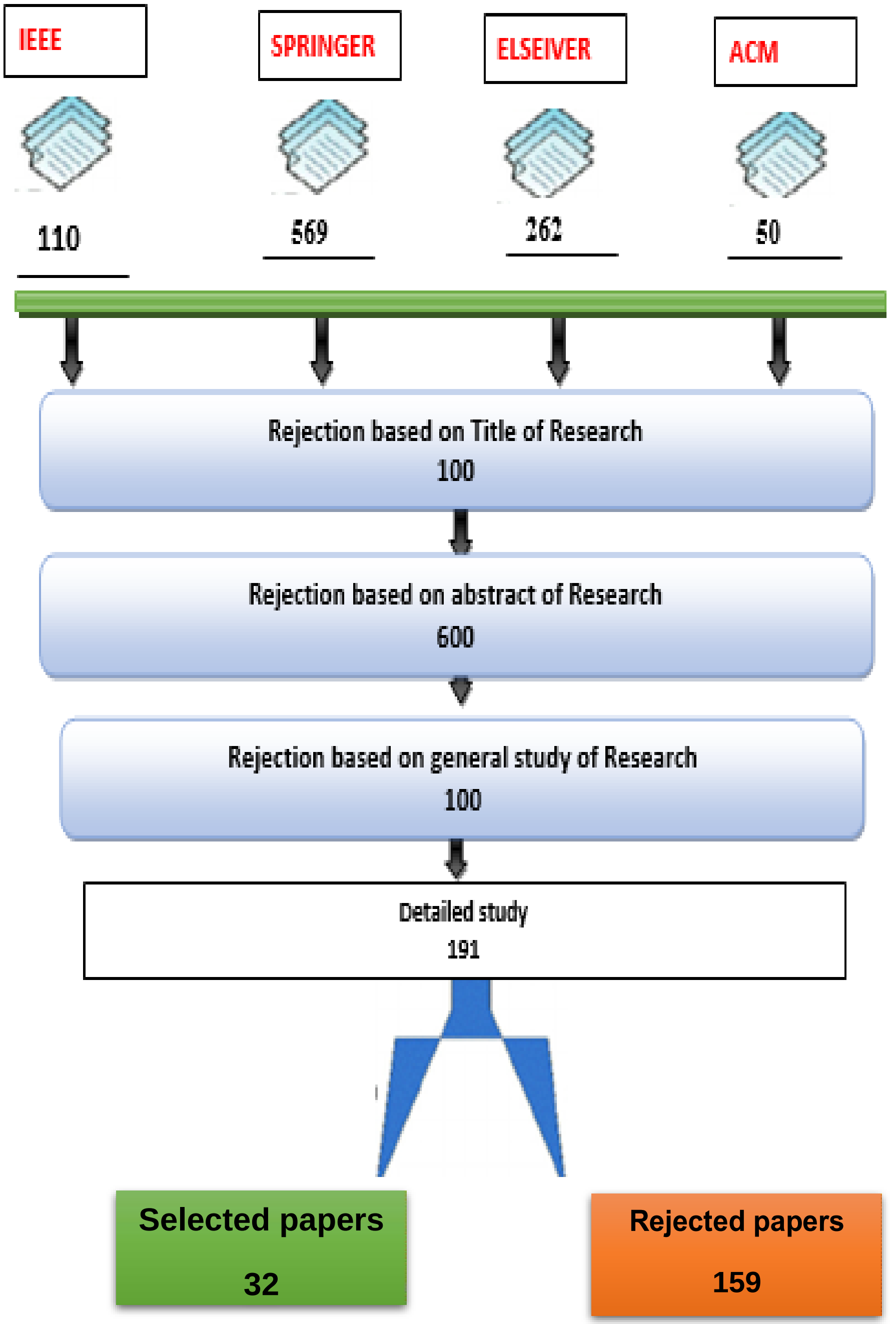

IEEE
SPRINGER
ELSEIVER
ACM
110
569
262
50
Rejection based on Title of Research
100
Rejection based on abstract of Research
600
Rejection based on general study of Research
100
Detailed study
191
Selected papers
32
Rejected papers
159

Figure 2. 1: Search Process

Following points depict step by step process of search flow.

1. **1.** As an initial step, 4 selected scientific databases are searched for specific terms and in return, total of 991 results are observed.
2. **2.** Checking the topic relevancy of abovementioned results, 891 papers are excluded as they do not match the title of our research.
3. **3.** The leftover papers are measured against another criterion of abstract relevancy. According to the results, 600 papers does not meet the research requirement.
4. **4.** To examine if the actual work done in residual papers, thorough general study of papers is completed on 191 articles. Only 191 researches made it to the next step.
5. **5.** A comprehensive study to 49 papers was performed to understand the experimentation performed. Through is investigation, data is processed, and facts and figures are measured. This in-depth study led to the rejection of 161 more papers.
6. **6.** The last 32 papers are used in literature because they strictly comply with the rules of selection and rejection.

Table 2: Search terms and Search results details

Search Terms	**Operators**	**Number of Search Results**			
		IEEE	**SPRINGER**	**ELSEVIER**	**ACM**
SQL NoSQL integration	AND	15	3	5	10
	OR	169	46	1	2
Heterogenous databases	AND	250	222	100	150
	OR	7960	248	11	58
Machine Learning	AND	400	321	0	10
	OR	5740	2482	3	172
	AND	1059	272	200	342

Conflicts in integration of databases	OR	54300	2352	256	1130
Unsupervised algorithms	N/A	524	69	5	17
K-means algorithm	N/A	1750	59	8	20

2.1.5. Quality Assessment

In this section, quality of paper is determined. Some check points established against which the literature is compared. Primarily, these points are developed to make sure that the work done meets our targeted standards. Whether the search process that we have opted is producing valuable information and if the inclusion and exclusion criteria is improving the standard of literature review. Following are the quality assessment measures.

1. Detailed study is made to make sure that data collected is not based on some imaginary or hypothetical information. Papers containing proper experimentation and validation of results are included as a part of systematic literature review.
2. To provide state of art information, old and traditional techniques are not considered to be part of study. Advance techniques comprising of integration of heterogeneous databases are included.
3. The aim of this research is focused on the automation and advanced techniques. For this purpose, this study only includes latest and state of art data. Work that was done before 2010 is not included in this dissertation. **Figure** shows the number of research papers in each year.
4. While selecting the paper, special care is taken to choose them based on their work. Papers are not searched or given priority based on author.
5. To improve the quality and remove the redundancy in papers, paper acquisition is restricted to 4 scientific databases. Thus, the studies include papers that are issued at least in one of these authentic and well-known scientific databases: ACM, SPRINGER, IEEE, and ELSEVIER. There are two more papers added in others category that are included because of quality of work. One is the dissertation from Cornel university library and the other one is from Taylor and Francis. **Figure 1** shows graphically the papers used in this section.

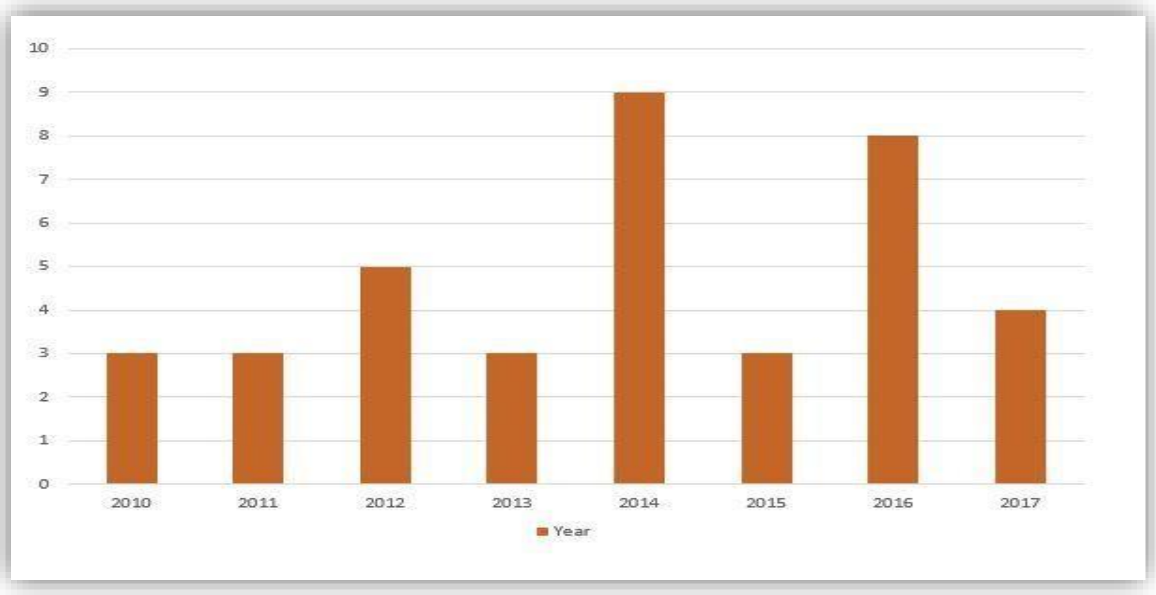


Figure 2.2: Selected researches per year

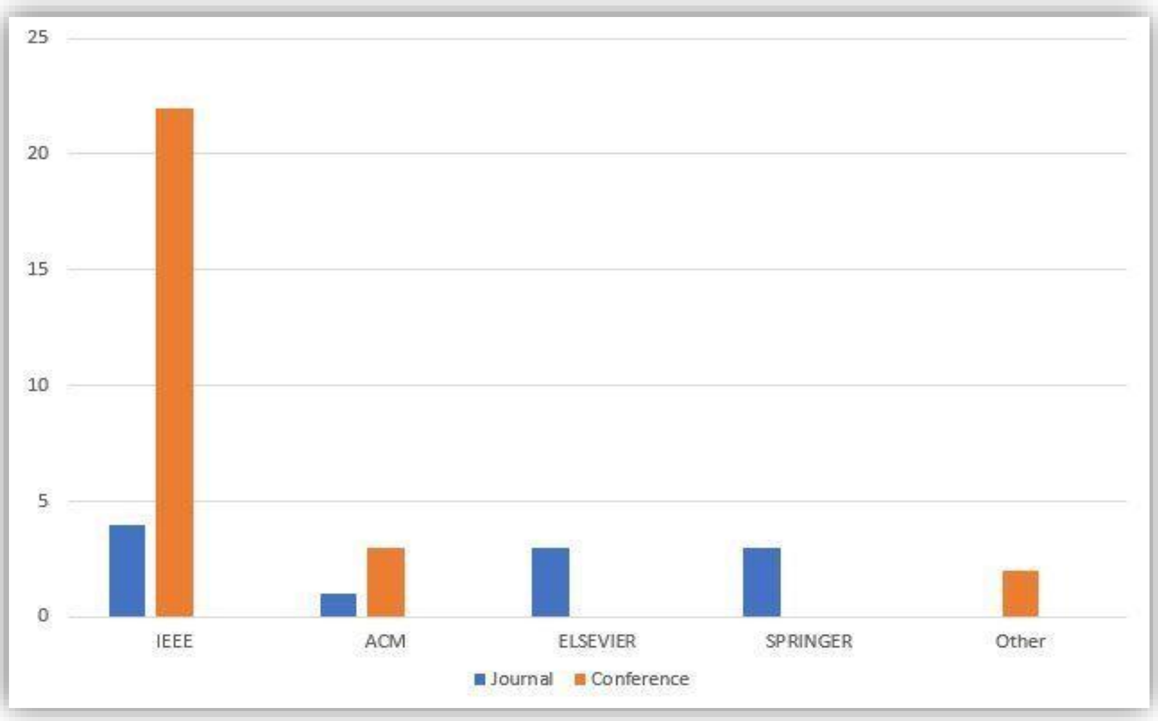


Figure 1.3: Selected researches per publisher

2.1.6. Data Extraction and Synthesis

Table 3 here, shows the data synthesis and extraction is performed. In data extraction, all the papers are studied to gather information regarding results produced, assumptions taken, and validation of those results while the next part, data synthesis, is performed to analyze the approaches used and put the papers in already specified categories.

Table 3: Extraction and data synthesis

Sr. #	Descriptions	Details
1.	Bibliographic information	Topic of research, author name, year of publication, detail of publisher and the type of publication (conference or journal)
Data Extraction		
2.	Overview	Gist of paper and research targets
3.	Outcomes	Results from the research
4.	Group of Data	Qualitative or Quantitative
5.	Assumption(s)	For validating the results
6.	Validation	Validation approaches to prove the results.
Data Synthesis		
7.	Classification	Applicability to pre-defined category
8.	Tools identification	Tools used for integration of Heterogeneous databases

2.2. Analysis of data

2.2.1. Classification of data

2.2.2. Approaches for Integration of heterogeneous databases

38 selected papers present multiple approaches for database integration of heterogeneous sources. This section provides detailed analysis of all these approaches. As mentioned before, we will only be looking at approaches that are devised after the year 2010**. Table 4** enlists them in concise way.

Table 4: Identification of integration of heterogenous databases approaches

Sr. #	Integration Approaches	No. of Researches	Research Identification
1	Hybrid database approach	1	[1]
2	Orthographic Software Modeling (OSM)	1	[17]
3	HSFRH-IoT HSFRH-IoT	1	[3]
4	chemogenomic data mining and proteomic data mining	1	[5]
5	PathSelClus	1	[30]
6	Ontology matching:	1	[26]
7	clustering algorithm	7	[1][6] [7][10] [12] [15][16]
8	WHIRL	1	[16]
9	entity matching	2	[19][23]
10	CORBA	1	[21]
11	CUOSP	1	[23]
12	UHDIS	1	[24]
13	Pathselclus	1	[30]
14	HSFRH-IoT	1	[3]

The analysis and study of 32 research papers gives 17 different approaches for implementing and improving the process of integration. They all are mentioned in **Table 4** along with the number of researches and papers that employ relevant technique. It has seen that there are some papers that use amalgam of some of these techniques to bring better results. Like, in paper [19], It proposes frameworks for entity matching. It also focusses on categorizing of entity framework based on evaluation. And some papers like [21] This paper use CORBA framework which solve the problems of heterogeneous data. These papers tend to apply various approaches while targeting the same goals.

There are 3 some papers that use mathematical models to have statistical based approach for database integration. If we go to the section 2.1 of literature review in category definition, statistical based category is defined as part of our classification measure. So, there are some approaches that also practice the classification category that we have defined. There can be overlapping between them. Probabilistic approach is followed in 5 papers of our study. There are some papers in it that uses machine learning constructs. While some of them uses other approaches. In [35], presents a Universal Heterogeneous Data Integration Standard which used parsing algorithm in real time. In [10] K-means and Self-Organizing Maps used for processing textual and numerical data types simultaneously. It can be seen from table, that there are some approaches like PathSelClus, Ontology matching, Orthographic Software Modeling (OSM), Hybrid database approach are commonly used. K-means algorithm for unsupervised learning are used in 3 papers [10][12][15]

2.2.3. Tools Analysis for Heterogenous Database Integration

After identifying the approaches used in different papers, tool identification is performed for further analysis of selected research papers. This section provides analysis of tools that are used in studies performed from 2010. It does not imply that tools before this time period are useless. It is simply not included in this research to develop the focused and state of art literature**. Table 5** here is developed after investigation of tools used for integrations and give their purpose according to their usage.

Table 5: Identification of Tools for data base integration

Sr. #	Tools	Purpose	Research Article	Author
1	IBM Enterprise Architect and Pa28 pyrus.	This is appropriate is based on metamodels of EMF/MOF, it deals with tools of heterogeneous modeling. It is not good for the integration of information from heterogeneous metamodels.	[17]	Erik Burger
2	Jupyter Notebook	That is Used for implementation of NetClus	[1][6][7] [16]	William Cohen, AK Jain , Yintao Yu, Yizhou Sun

3	Anaconda	Implementation of K-means and Self-Organizing Maps	[10]	Farid Bourennani
4	MongoDB compass community	Implementation of MongoDB (NoSQL)	[1]	Murtadha Arif Bin Sahbudin
5	SQL	Implementation of Relational database	[8]	K Fraczek

Table 6 demonstrates the tools used for integration of heterogeneous database. The table contains the name of the tool, its purpose and working, paper in which they are discussed/proposed, and the name of author for the specified study. There are total 5 tools that are identified from the review. There are some tools that are used by multiple researchers. For example, Jupyter Notebook at number 2, is used in this researches by William Cohen,AK Jain ,Yintao Yu, Yizhou that is used to execute program statements and perform clustering.

2.3. Relevant Study

After analyzing all the research data from different aspects, this section presents the methodologies and procedures used in study of database integration. Again, it comprises of papers that we have selected through process of screening. Techniques that are proposed from 2010 to 2020 are discussed here. Anything that is not properly validated through experimentations and integration through clustering is not added in the review. Special caution is taken to avoid redundancy and duplication. **Table 8** shows the relevant study. It contains title of publication. Its type (journal or conference) is also mention in a separate column, then there is name of author along with the year of publication. Then there is brief description of actual work done in that article.

Table 6: Selected studies to perform Integration

Sr.#	Title of Paper	Work done
1.	"MongoDB Clustering using K-means for Real-Time Song Recognition." Conference	This paper represents an used K-means clustering for MongoDB. It describes strategies through which we can improve accuracy and speed. This paper uses 2.4 billion fingerprint data of a company to evaluate the efficiency of proposed algorithm.

	Murtadha Arif Bin Sahbudin [1]	
2.	"A hybrid database approach using graph and relational database." Conference HR Vyawahar [2]	This approach didn’t unify the two databases, but two databases functioned independently in a single system. This hybrid model had a drawback that different natured databases were not integrated because of the concept that both databases have different data types and yield different access patterns so developed system had the power to facilitate hybrid functionality of the systems but it was not capable of facilitating integrated functionality of the system.
3.	"A Method for Building Shared Massive Heterogeneous IoT Data Environment." Conference Shanshan Wu [3]	This paper presents a framework called HSFRH-IoT, it is used to build shared massive heterogeneous IoT data environment, which solves the problem related to data retrieval of huge heterogeneous IOT and efficient storage. Finally, in the insertion and query performance a detailed testing was done, and the results shows that it has better performance than other RDBMS based solutions.
4.	"Mining heterogeneous information networks: a structural analysis approach." Journal Yizhou Sun [4]	This is review based paper that represent multitype data which are interconnected, data stored in relational database, as heterogeneous information networks. It also presents strategy for semantic meaning of structural types of objects and links in the networks. Based on this technique they develop a structural analysis approach on mining semi-structured, multi-typed heterogeneous information networks. In this article, they review a set of methodologies that can efficiently extract useful knowledge from information networks and point out some promising research directions.
5.	"Target discovery from data mining approaches." Journal Yongliang Yang [5]	This study briefly introduces two emerging data mining approaches, first one is chemogenomic data mining and other is proteomic data mining. Also discussed the various data mining approaches with their limitations which are found in the level of

		database integration, the data annotation quality, heterogeneity sample and performance of mining tools. For integration of different data sources different strategies were designed such as integrated text mining with high-throughput data analysis and direction of integration in mining data.
6.	"Ranking-based clustering of heterogeneous information networks with star network schema." Conference Yintao Yu [6]	This paper presents a novel algorithm approach called NetClus which is based on clustering of multi-typed heterogeneous networks. They used star network schema, that utilizes links across multitype objects to produce high-quality net-clusters. For development of effective ranking-based clustering iterative improvement method is used. As a result of experiments on DBLP data shows that NetClus generates more precise clustering results as compared to other baseline topic model algorithm such as PLSA and RankClu algorithms.
7.	"Relation strength-aware clustering of heterogeneous information networks with incomplete attributes Journal Yizhou Sun. [7]	In this paper, they focus challenges of heterogeneous integration by proposing a model- based clustering algorithm. They design a probabilistic model which is used user defined attributes and links from different relations as input and perform clustering. It also solves clustering problems by the help of iterative algorithm. The experimental results are based on real and synthetic data which demonstrate the effectiveness and efficiency of the algorithm.
8.	"Relational & Non-Relational Databases in Web Applications" Conference, K Fraczek [8].	The author was created simple application-based system, it was used to compare the relational and non-relational databases. The developed application supported three different types of databases. MongoDB, SQL which was tested with PostgreSQL and the third one was Apache Cassandra. The main purpose of the comparative analysis was to measure the performance of each database in terms of reading and writing capabilities.

9.	**“Classification of Databases”** Conference, N. Ahmed [9]	The author’s objective was to provide classes, features and development of available databases which may be used in analysis and prediction. Another objective of this paper was to provide teaching perspective from transferring relational database to big data approach. [9]. On the other hand, this paper was also focused different types of data which include structured, semi structured and unstructured data as well as their security issues. It also focuses on availability and performance of these database.
10.	"Clustering relational database entities using k-means." Conference Farid Bourennani, [10]	This paper present K-means and Self-Organizing Maps which are used in processing of textual and numerical data types by UV. They assess how the HDM-UV improves the clustering results of these two algorithms (SOM, K- means).Results were also compared with traditional homogeneous data processing.
11.	. "Research on semantic integration across heterogeneous data sources in grid." Journal Guofeng Liu [11]	In recent years grid technology which is a kind of network information technology evolves, which is used to resolve the problems related to fully sharing and interactive kinds of resources (such as computing resources, storage resources etc.) used in distributing wide area. This paper focuses on the conflicts of semantic integration across heterogeneous data which faced in grid database. With the help of automatic/semi-automatic schema matching algorithm, it analyzes the advantages and disadvantages of algorithm and presents a generic schema matching model that deals with schema and instance information.
12.	"Comparative study of k-means variants for mono-view clustering."	This research is review based it represents clustering methods which are K-means, IRP-K-means and FKM. Comparison was shown between these algorithms based on

	Conference . Safa Bettoumi [12]	accuracy and running time These three main approaches are implemented by using text data as input. Testing was also done with the help of five different descriptors with different sizes.
13.	"Integration and virtualization of relational SQL and NoSQL systems including MySQL and MongoDB." Conference Ramon Lawrence [13]	In this work a generic standards-based architecture was developed that allows NoSQL systems (Mongo DB) to be queried by using SQL. A system is built in NoSQL database that able to translates SQL queries into the source definite APIs virtually. The virtualization architecture allows users to query and join data from both NoSQL and SQL systems in a single SQL query.
14.	"Relational versus non-relational database systems for data warehousing." Conference Carlos Ordonez [14]	This paper present MapReduce which is a highly used for analyses of large dataset.Moreover, MapReduce needs advancement in evaluating relational queries. This paper compare advantages and disadvantages of each technology for data warehousing and also identify research issues.They also focused practical aspects like ease usage, flexibility of programming and its cost.They also presented some aspects like data modeling, data storage,hardware,scalability of data,query processing, fault tolerance and data mining.
15.	"Research on k-means clustering algorithm: An improved k-means clustering algorithm" Conference Shi Na [15]	This paper discusses k-means clustering algorithm and analyzes limitation of standard k-means algorithm,how efficiency of k means becomes low.It also proposes improved version of k-means algorithm in order to improve speed of clustering and accuracy. It also reduces the computational difficulty of the k-means.
16.	"Integration of heterogeneous databases without common	This paper proposes a logic which is called WHIRL. It find out resemblance of local names. The similarity is measured through vector-space model. They describe an efficient implementation of

	domains using queries based on textual similarity." Conference William Cohen [16].	WHIRL and evaluate World Wide Web experimentally on the base of data extraction. This paper shows that WHIRL is much faster than naive inference methods, even for short queries.
17.	"A methodology for integration of heterogeneous databases Journal M.P. Reddy [17].	The proposed methodology resolves many conflicts like schema level naming conflicts, scaling conflicts, type conflicts, level of abstraction and other types of conflicts it also covers data inconsistencies during data integration. A four-layered schema architecture (local schemes, local object schemes, global scheme, and global view schemes) is used for resolving schemes integration and database integration.
18.	"Analysis and design of heterogeneous bioinformatics database integration system based on middleware" Conference Yuelan Liu [18]	It proposes framework which is used for integration of heterogeneous data sources. Data is based on middleware technology. It resolves conflicts related to data format and semantic heterogeneity.
19.	"Frameworks for entity matching: A comparison" Journal Hanna Köpcke [19]	It proposes frameworks for entity matching. The proposed framework enable entity matching frameworks and their evaluations. It also focusses on categorizing of entity framework based on evaluation.
20.	"Large-scale data integration framework provides comprehensive view on glioblastoma multiforme." Journal Kristian Ovaska [20]	It present a novel data integration framework, **Anduril** which automatically generates thorough summary reports and a website that shows the most relevant features of each gene at a glance ,it also perform sorting which is based on different parameters, and provides direct links to more detailed data on genes, transcripts or genomic regions.
21.	"The design and research of the integration for heterogeneous	This paper use **CORBA** framework which solve the problems of heterogeneous data. Currently, the system has been applied in the campus card system.

	database in campus card system." 2011 Conference Lu Ping. [21]	
22.	"Data clustering: 50 years beyond K-means." Conference Anil Jain [22]	This paper provides a brief overview of clustering through k-means. It also summarize well known clustering methods, and also discuss the major challenges and key issues in designing clustering algorithms. some of the emerging and useful research direction are also focused. In this paper semi-supervised clustering, ensemble clustering, simultaneous feature selection during data clustering, and large scale data clustering was also focused.
23.	"Cloud-based ubiquitous object sharing platform for heterogeneous logistics system integration." Journal Ming Li [23]	This paper presents a cloud-based global object sharing platform (CUOSP) to provide integration across SME which is based on the concept of sharing economy. It acts as a middleware system to make heterogeneous logistics systems universal plug-and-play (UPnP) for enterprise information systems. System was designed called kernel-based agent (KBA) which is designed as the sharing entity of physical systems. It maintains the features of physical systems and it is scalable for different application scenarios. A prototype system is designed and implemented which is based on the framework of CUOSP.
24.	"A universal heterogeneous data integration standard and parse algorithm in real-time database." Conference Fei Chang [24]	This paper proposes a Universal Heterogeneous Data Integration Standard (UHDIS which is) based on parsing.It proposes technique for integration of data of UHDIS to the desired tables in database.It is universal system because parsing algorithm is used for different kinds of integrated data by adding new templates instead of changing the algorithm.
25.	"Online application of science and technology program oriented distributed	This paper proposes heterogeneous data integration method which is based on object mapping. It is used to solve the problem regarding mass of heterogeneous data that exist in online application of science and technology programs. This is based on

	heterogeneous data integration." Conference Hao Tang[25]	specific data standards; this method contains tree models for aggregation of logic and organizes data. It also describes the file structures of different science and technology programs through redefining the pruning of the models and takes the path of a data item for screening the heterogeneity of data
26.	"Rule-based multi-dialect infrastructure for conceptual problem solving over heterogeneous distributed information resources." Conference Leonid Kalinichenko[26]	This approach is applying on combination of the semantically different rule-based languages for interoperable conceptual programming over various rule-based systems (RS) and depend on the logic program transformation technique which is recommended by the W3C Rule Interchange Format (RIF). It is logically combined with the heterogeneous data base integration and apply semantic rule mediation. The basic functions of the infrastructure implementing are multi-dialect conceptual specifications by the interoperable RS and mediator programs. The results show the usability of the approach and independency of resources and re-usable data analysis in various application.
27.	"Ontology matching: state of the art and future challenges." Journal Pavel Shvaiko. [27]	This paper reviews the state of the art of ontology matching and analyze evaluations of ontology matching. These results show the speed of which is albeit slowing down and improvement in the field of matching.
28.	"Matching attributes across overlapping heterogeneous data sources using mutual information." Journal Huimin Zhao[28]	Matching attributes identification across heterogeneous data sources is a critical and time-consuming step in database integration. In this paper, the author proposes a novel method for overlapping heterogeneous data sources and matching the most frequently encountered types of attributes. They use mutual information for measurement of dependency of various types of

		Attributes which is used to demonstrate the utility of the proposed method, which is useful in developing real-world attribute matching tools.
29.	"Comparative study of relational and non-relations database performances using Oracle and MongoDB systems." Journal Azhi Faraj [29]	This research compares the performance of relational and non-relational databases.The main purpose of this paper is integration of document oriented database to relational database. The results show that data retrieval is faster in MongoDB but some queries and functions like sum, count, AVG are better in Oracle RDBMS
30.	"Pathselclus: Integrating meta-path selection with user-guided object clustering in heterogeneous information networks." Sun, Yizhou, et al. [30]	This paper uses meta-path selection with user-guided clustering for making cluster objects in networks, where a first small set of objects get from users which is used seeds for each cluster as guidance. Then the system measure weight for each meta-path that is consistent with guidance and generates clusters under the learned weights of meta-paths. For model learning PathSelClus algorithm is proposed, where the clustering quality and the meta-path weights mutually enhance each other.
31.	"Uniform data access platform for SQL and NoSQL database systems."	Integration of heterogeneous database systems is a very challenging task and it may hide several difficulties because data is stored in different sources. As NoSQL databases are growing in

	Journal Vathy Fogarassy [31]	popularity, integration of different NoSQL systems and interoperability of NoSQL systems with SQL databases become an increasingly important issue. In this paper, we propose a novel data integration methodology to query data individually from different relational and NoSQL database systems. The limitation of this paper is it does not support joins and aggregates across data sources; it only collects data from different separated database management systems according to the filtering options and migrates them. Metamodel approach is used in this paper and it covers the structural, semantic and syntactic heterogeneities of source systems. For applicability of the proposed methodology, they developed a web-based application, which convincingly confirms the usefulness of the novel method.
32.	"Unsupervised machine learning for networking: Techniques, applications and research challenges." Journal Muhammad Usama [32]	This survey paper is providing an overview of applications of unsupervised learning in the domain of networking. They provide a comprehensive survey on unsupervised learning techniques, along with their applications in various learning tasks. Also provide future directions and open research issues, it also recognizes potential pitfalls in unsupervised machine leaning implementation.

2.4. Research Gap

From literature review which is mentioned in this chapter we found that there is no such technique was proposed in which integration is performed through unsupervised machine learning algorithms. So, there is a gap between supervised machine learning algorithm and unsupervised machine learning algorithm.in this thesis we would bridge this gap with proper implementation of unsupervised machine learning algorithms to perform integration in the form of clustering.

We proposed a solution which is based on unsupervised machine learning algorithms. In which we don’t need to train our dataset and supervise our dataset which needs extra cost and effort. In this approach we have used some unsupervised learning algorithms for performing clustering. With the help of these clustering we will be able to integrate relational database with non-relational databases.

Chapter 3

Proposed Methodology

CHAPTER 3: PROPOSED METHODOLOGY

This chapter contains detailed explanation of concepts involved in our proposed methodology. The recommended solution is based on unsupervised machine learning algorithm for integration of heterogeneous databases. First, we get an idea of clustering technique in **Section 3.1, Section 3.2** contain detailed information about our proposed algorithm K-means clustering and working of it, **Section 3.3** represent all details of other machine learning algorithms such as Agglomerative Clustering**, Section 3.4** is about Spectral Clustering, **Section 3.5** contain detailed information of Mean Shift clustering. **Section 3.6** Selection criteria for best machine learning unsupervised algorithm for clustering

3.1. Clustering

Data analysis technique which is used to get a perception about the structure of the data is called *Clustering* which is one of the most common tentative technique of unsupervised machine learning. It can be used to identifying similar datapoints in the same subgroup (cluster) that are very similar while data points in different clusters are very different. In other words, it is used to find homogeneous subgroups within the dataset such that data points in each cluster are as similar as possible according to a similarity measure. Euclidean-based distance or correlation-based distance is used to measure similarity of data point from centroids.

3.2. K Means

K Means is clustering algorithm, it is an iterative algorithm that is used to divide the dataset into *K* pre-defined separate non-overlapping subgroups (clusters) where each data point belongs to only one group. It assigns centroid which is data points to a cluster for measuring the squared distance between the data points and the cluster's centroid.so whole data is divided into groups according to the distance from centroid.

The way K-means algorithm works is as follows:

Step-01:

- Choose the number of clusters *K*.

Step-02:

- Select any *K* data points randomly as cluster centres.
- Select cluster centres in such a way that they are as farther as possible from each other.

Step-03:

- Calculate the distance between from cluster centre to each data point.
- The distance may be calculated either by using given distance function or by using Euclidean distance formula.

Step-04:

- Assign each data point to some cluster.
- A data point is assigned to that cluster whose centre is nearest to that data point.

Step-05:

- Re-compute the centre of newly formed clusters.
- The centre of a cluster is computed by taking mean of data points of cluster.

Step-06:

- Keep repeating the procedure from Step-03 to Step-05 until stopping criteria is met.
- Centre of newly formed clusters do not change
- Data points remain in the same cluster

Maximum number of iterations are reached

3.2.1. K-means convergence (stopping) criterion

- Position of data points to different clusters will remain same
- No change observe in centroids
- Minimum decrease in the sum of squared error

$$E = \sum_{i=1}^{k} \sum_{x \in C_i} \left| x - m_i \right|^{2} \qquad \text{………… (3.1)}$$

Table 7: Terms used in this section

Term	Purpose
m_i	It represents centroid of cluster
C_i	This shows mean vector of all the data points in C_i
x-m_i	It shows (Euclidian) distance between data point x and centroid m_i

3.2.2. Proposed solution with example

Step-01:

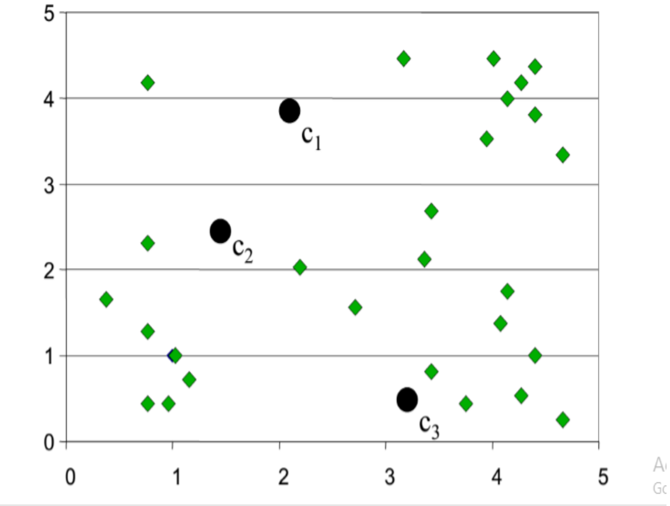


Figure 2.1: Randomly initialize the cluster centers (1^{st} iteration)

In step 1we have to choose random number of clusters like in this diagram there are 3 clusters c1, c2 and c3. Basically, these center values will help to calculate distance of each point. Those points which are nearest to centroid put it in cluster of respective centroids.

Step-02:

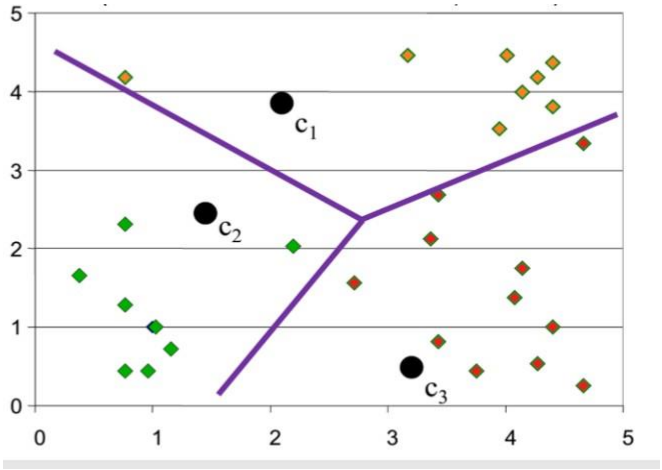


Figure 3.2 : Determine distance from cluster centers

In this step measures distance of each point from cluster centers

Step-03:

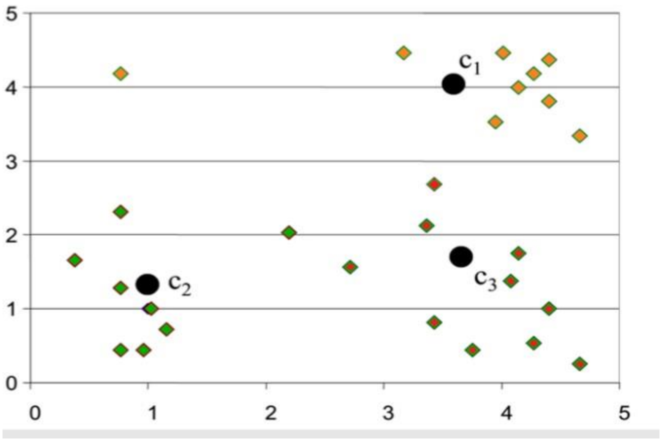


Figure 3.3: Results of 1st iteration

This is the last step of 1st iteration in which we get closest points of each clusters, but it needs more iteration for getting better clusters

Step-04:

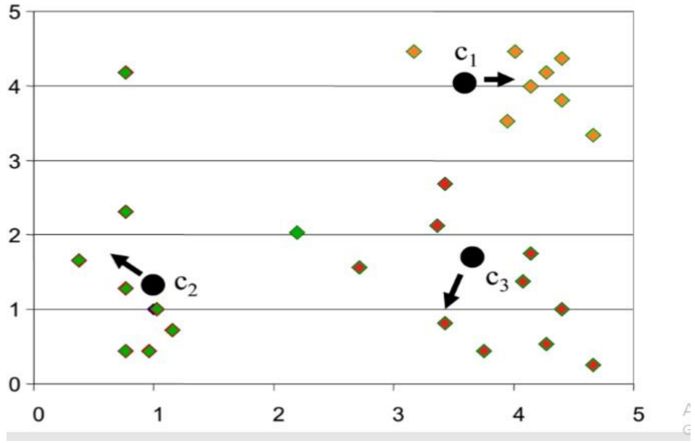


Figure 3.4: 2nd iteration

In this step again calculate the distance from center so we will get similar points in one cluster

Step-05:

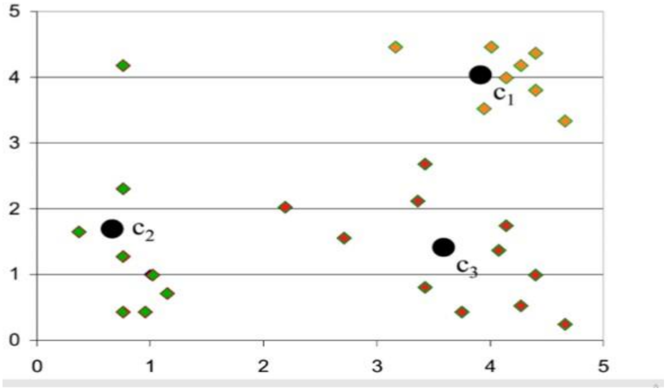


Figure 3.5: Result of 2nd iteration

This is the final step of K-means clustering in which we can get clusters of similar data.

3.3. Agglomerative clustering

In agglomerative clustering, initially we consider each data point as a separate cluster. As a result of each iteration we get similar clusters.There clusters then merge with other clusters until one cluster or K clusters are formed.

The basic algorithm of Agglomerative is straight forward.

Step-01:

In first step we have to compute the proximity matrix.

Step-02:

In this step we consider each data point as one separate cluster.

Step-03:

closest clusters mergerd until we get only a single cluster remains.This step will repeat until we will get all similar points in one cluster.

Step-04:

Dendrogram used to visualize groups and optimal number of clusters.

3.3.1. Proposed solution with example

Step-01:

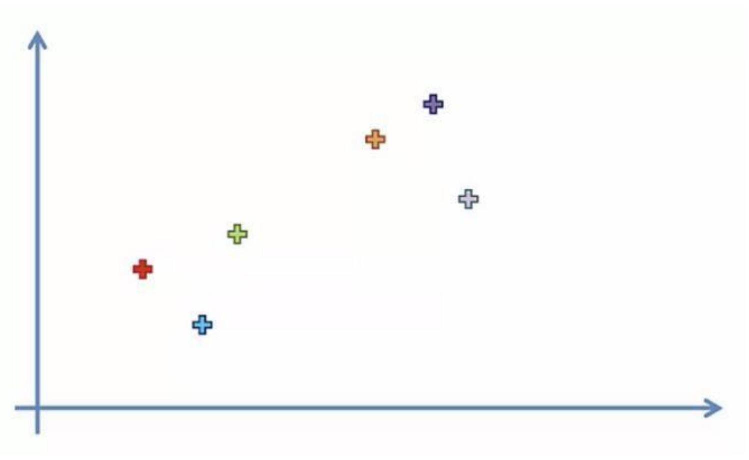


Figure 3.6: Make each data point a cluster

In agglomerative clustering first we develop proximity matrix. Then represent each data point each cluster.

Step-02:

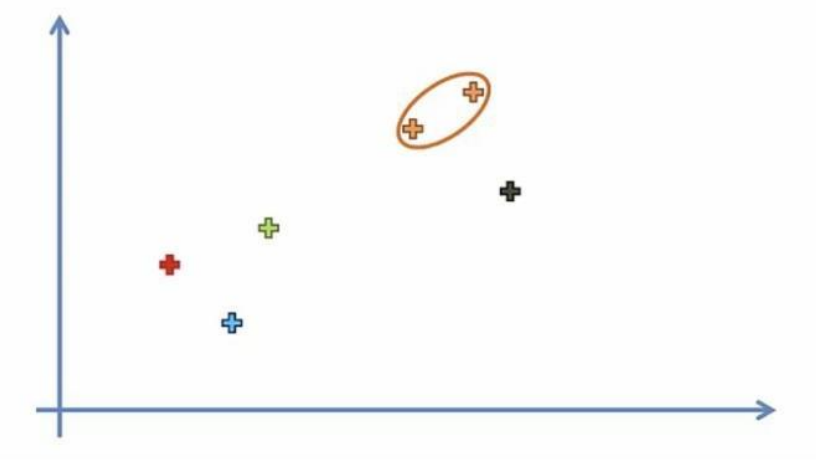


Figure 3.7: Make one cluster from two closest clusters

In this step those clusters which are closest make them one cluster.so all points that are similar come under one cluster.

Step-03:

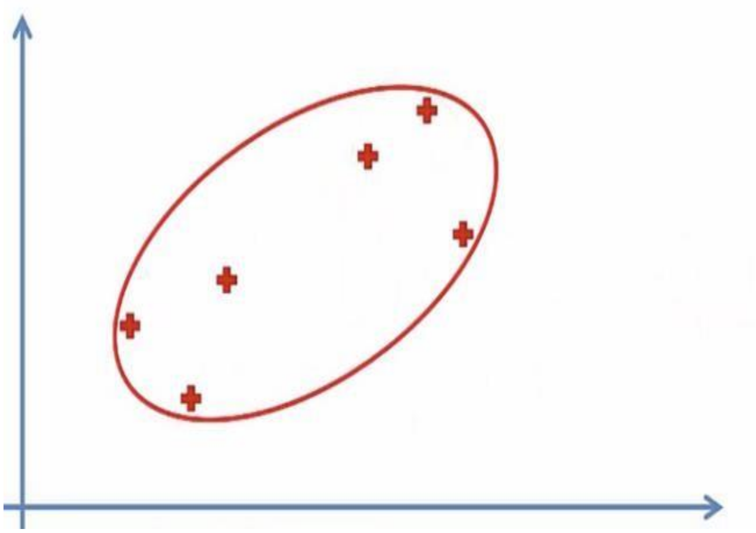


Figure 3.8: Formation **of** one cluster

Repeat this step until we will get one cluster from closest

points.

Step-04:

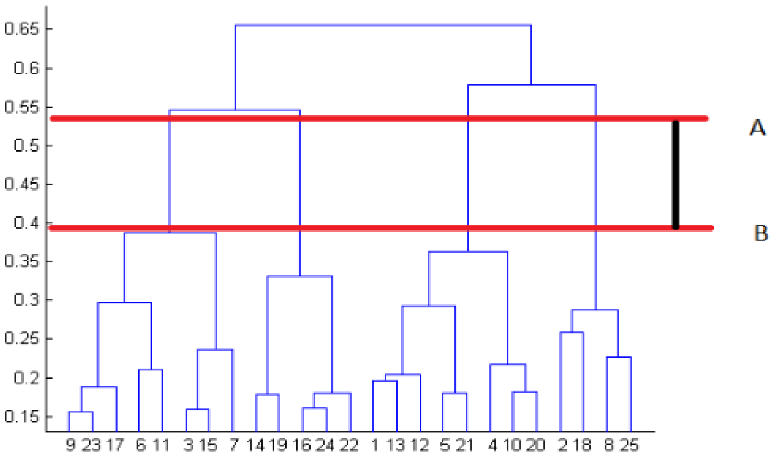


Figure 3.9: Dendrograms

Dendrogram is a visual representation of groupings and this can also be used to find out the optimal number of clusters.

.

3.4. Mean shift

Mean shift clustering is used to discover "blobs" in an even density of samples. This algorithm is centroid based, which works by applicants updating for centroids to be the mean of the points within a given area. In a post-processing stage these candidates are filtered in order to eliminate duplicates.

Step 1:

In step 1 we assign data points to cluster

Step 2:

Next, mean shift algorithm will compute the number of centroids.

Step 3:

In this step, updating location of new centroids.

Step 4:

Now, the process will be iterated and moved to the higher density region.

Step 5:

At last, it will be stopped once the centroids reach at position from where it cannot move further.

3.4.1. Proposed solution with example:

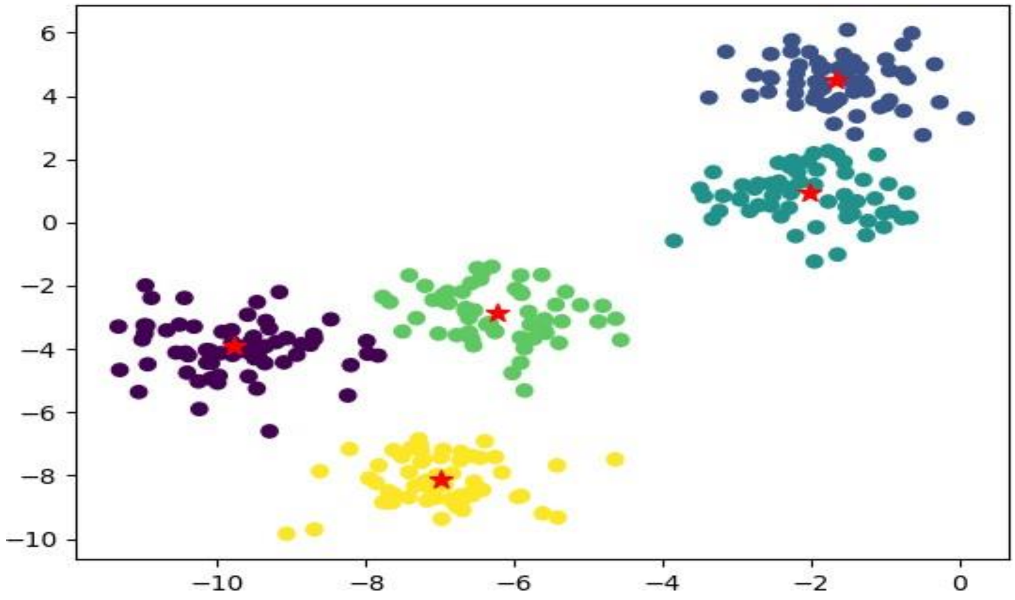


Figure 3.10 : Mean shift

First, we make all datapoints centroids. Then take mean of all feature sets within centroid's radius and set this mean as new centroid. Repeat step #2 until we will get full convergence.in this example we have 5 clusters.

3.5. Spectral Clustering

Spectral clustering is a technique which is used in graph theory, this approach is used to identify groups of nodes in a graph based on the edges connecting them. The method is basically used in image processing.

Step 1: ε-neighborhood graph

In first step we have to identify a threshold value, ε, we measure edges if the affinity between two points is greater than ε we select these edges.

Step 2: k-nearest neighbors

- In this step we insert edges between node and its k-nearest neighbors
- The criteria for nodes are each node will be connected to (at least) k nodes

Step 3:

- In this step we took pair of nodes and Insert an edge between every pair of nodes
- Then calculate weight of edge. This represents similarity
- So, we can identify and separate similar data through spectral clustering

3.6. Selection Criteria

Selection of algorithm depends on accuracy. Based on accuracy, we would choose the best algorithm which will use in grouping. This grouping represents table of SQL. So, with the help of this technique we would identify tables of SQL from collection of NoSQL (MongoDB). we would use confusion matrix. A confusion matrix technique for Machine learning classification which is used to measure performance. It is kind of matrix which helps to the know the performance of classification model on a set of test data for that the true values are known. The term confusion matrix itself is very simple, but its related terminology can be a little confusing.

Chapter 4

Implementation

CHAPTER 4: IMPLEMENTATION

This chapter presents the implementation details for this research. Case studies have been carried out to prove the dissertation. Major part of implementation is pre-processing of data that is discussed in detail in **Section 4.1.** First, create database on MongoDB. Then implement different algorithms for separation of data into groups. **Section 4.2** has the details of implementing naïve Bayesian algorithm.

4.1. Database creation

Data preprocessing involves many tasks that are discussed in detail in this section. The processed data is input into the classifier for getting results. This data obtained is used as an input for classifier to be unsupervised. For use as an input this data export from MongoDB in json file. The purpose here is to get collection of data from NoSQL and indicates tables of SQL with the help of classifiers. For this study, the all the systems and case studies used are implemented in C++ language. To execute these programs, we use MongoDB Compass Community, which in turn helped develop NOSQL database. Dataset of each case study is implemented in machine learning algorithms. Different algorithms are carried out on same case study. Their results and the comparison are performed in next chapter of the dissertation. All executions are performed on a core i3 laptop with 1.70GH processor and with installed RAM of 4GB. It has operating system of 64-bit windows 10. Implementation details of each case study used is discussed here.

4.1.1. School Management System

This case study is related to school management system we create a database of this case study. It is developed in MongoDB Compass community. Main purpose of picking this dataset is to check the efficiency of our proposed methodology. It contains 480 rows and 17 columns.it is collection of student teacher and course data.in next step collection is exported in the form of json (). **Figure 4.1** shows the ERD (entity relationship diagram) of school management system

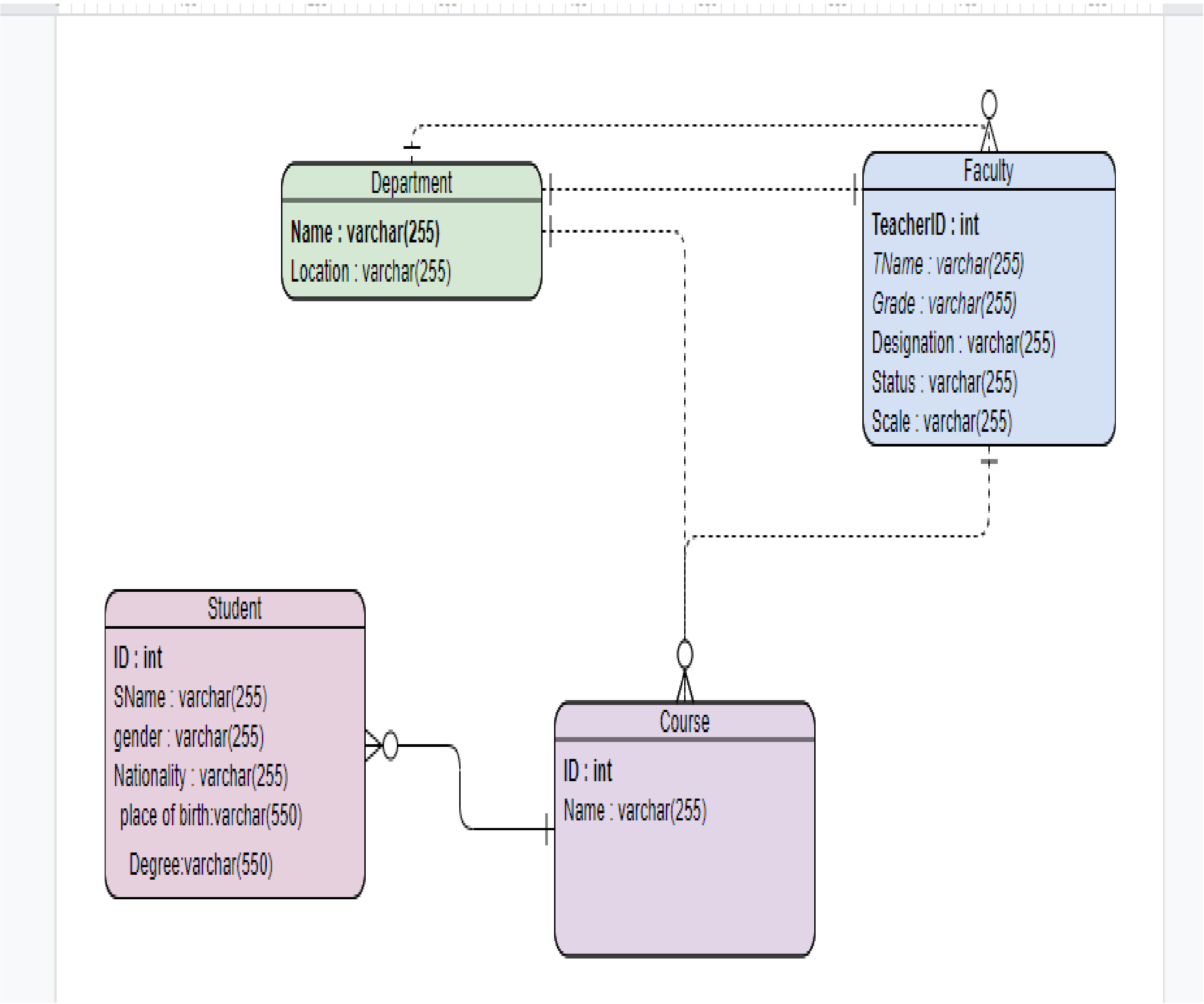


Figure 4.1: ERD of School Management System

Now for the testing data, dataset is developed. This dataset contains data of faculty, student, department and courses. This dataset is the mixture of data or we can say collection of data with no separate tables and relationship. This data set is finally ready to be used in different classifier. These classifiers differentiate data in the form of tables which depicts table of SQL. Implementation details are discussed further in next section of same chapter.

4.1.2. Hospital Management System

Next case study is related to Hospital management system we create a database of this case study. It is developed in MongoDB Compass community. Main purpose of picking this dataset is to check the efficiency of our proposed methodology. It contains 40 rows and 12 columns.it is collection of student teacher and course data.in next step collection is exported in the form of json (). **Figure 4.2** shows the ERD (entity relationship diagram) of hospital management system

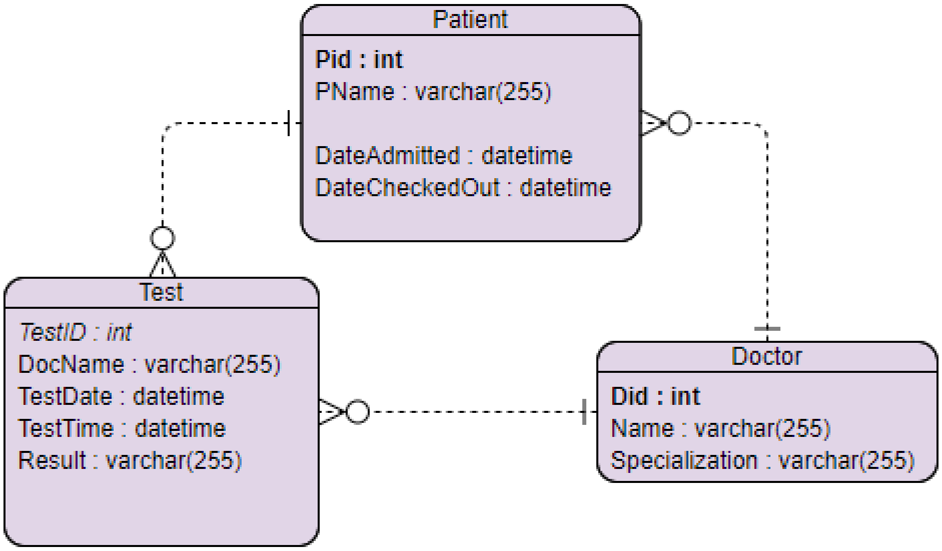


Figure 4.2: ERD of Hospital Management System

Now for the testing data, dataset is developed. This dataset contains data of Doctor, patient and test. This dataset is the mixture of data or we can say collection of data with no separate tables and relationship. This data set is finally ready to be used in different classifier. These classifiers differentiate data in the form of tables which depicts table of SQL. Implementation details are discussed further in next section of same chapter.

4.2. Implementation:

This part of research involves implementing code for clustering algorithms. Python is used to implement the algorithms. Jupyter notebook anaconda 3 is used for python, is used here for implementation in this study. It is very useful for providing almost everything that is related to python. It can be used for implementing core python. It offers hundreds and thousands of python libraries offering data suites and chunks of codes that can be used for developing other programs making the implementation easy and fast.

Starting with our implementation, we have imported a package *NumPy* that is mostly used for performing computations in scientific research. As our task has computational complexity, *NumPy* provides an excellent approach to resolve it. It has many things including Nd arrays (N-dimensional array objects), variety of different functions, even offers tools for integrating code written in C, C++ and Fortran. Besides, there are powerful tools for solving mathematical problems like linear algebra, Fourier transformation, and random number capabilities. As it has multi-dimensional array structures, it makes data manipulation easy. Ability to handle multiple data types lets *NumPy* to connect with different databases without any difficulty. *Pandas library is* also imported it deals with array. It also happens to be a tool for efficient data mining and machine learning. *seaborn* is also imported, it is used Seaborn is a Python data visualization library based on matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics. Next implementation step is to first export MongoDB data as JSON file, and then input data called in Jupyter notebook. To get result by applying a function. To get the idea of implementation, some main tasks along with the code are listed below.

4.2.1. Implementation of K-means:

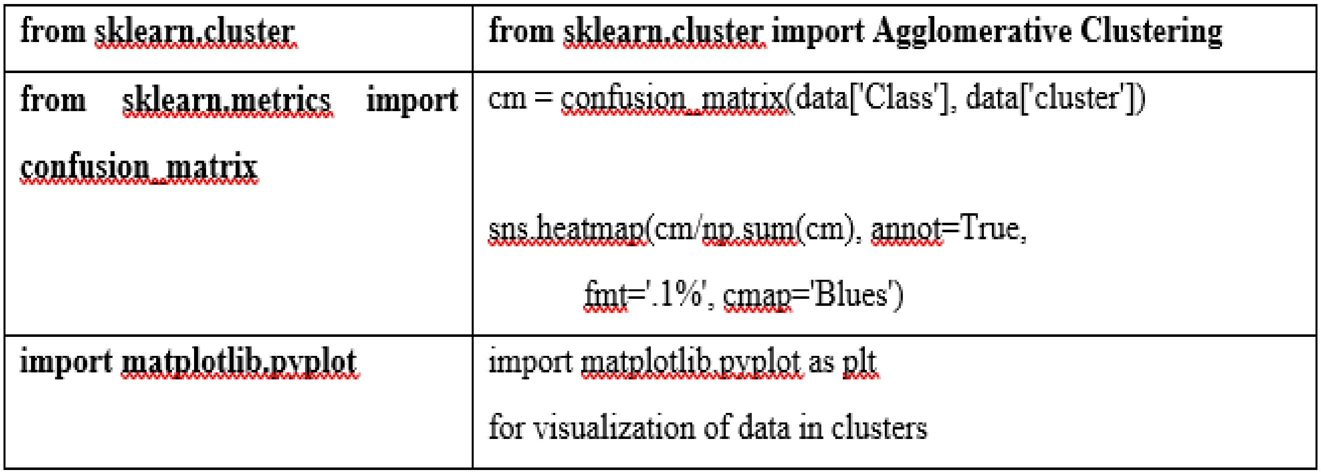
Importing pandas	import pandas as pd
Import numpy	import numpy as np
Import seaborn	import seaborn as sns
Create MongoDB	Create Dataset on MongoDB
pd.read_json	Read json file of Dataset

From sklearn.preprocessing	from sklearn.preprocessing import OneHotEncoder
transform(data[[c]]).toarray()	Convert all data to array
from sklearn.cluster	from sklearn.cluster import K-means
With the help of K-means predict data in clusters	k-means = K-means (n_clusters=2, random_state=0) k-means. fit (data) data['cluster'] = k-means. predict(data)

from sklearn.metrics import confusion_matrix	cm = confusion_matrix(data['Class'], data['cluster'])
	sns.heatmap(cm/np.sum(cm), annot=True, fmt='.1%', cmap='Blues')

import matplotlib.pyplot	import matplotlib.pyplot as plt for visualization of data in clusters

4.2.2. Implementation of Agglomerative Clustering:



from sklearn.cluster	**from sklearn.cluster import Agglomerative Clustering**

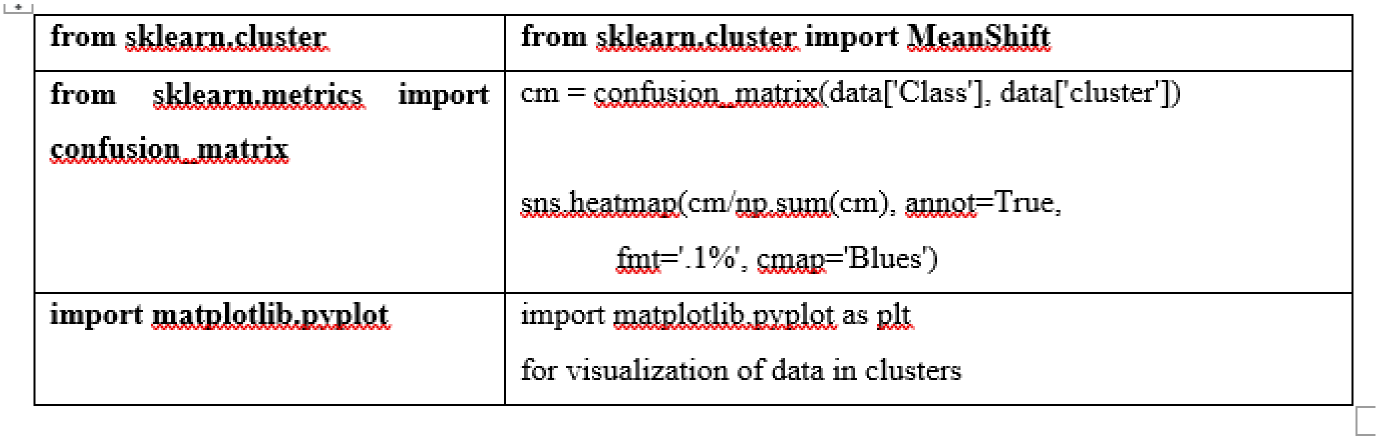
from sklearn.metrics import confusion_matrix	cm = confusion_matrix(data['Class'], data['cluster'])
	sns.heatmap(cm/np.sum(cm), annot=True, fmt='.1%', cmap='Blues')

import matplotlib.pyplot	import matplotlib.pyplot as plt for visualization of data in clusters

4.2.3. Implementation of Spectral Clustering:

from sklearn.cluster	from sklearn.cluster import Spectral Clustering
from sklearn.metrics import confusion_matrix	cm = confusion_matrix(data['Class'], data['cluster']) sns.heatmap(cm/np.sum(cm), annot=True,
	fmt='.1%', cmap='Blues')
import matplotlib.pyplot	import matplotlib.pyplot as plt for visualization of data in clusters

4.2.4 Implementation of Mean Shift:



from sklearn.cluster	from sklearn.cluster import MeanShift

from sklearn.metrics import confusion_matrix	cm = confusion_matrix(data['Class'], data['cluster'])
	sns.heatmap(cm/np.sum(cm), annot=True,
	fmt='.1%', cmap='Blues')

import matplotlib.pyplot	import matplotlib.pyplot as plt for visualization of data in clusters

Chapter 5

Results and Discussion

CHAPTER 5: RESULTS AND DISCUSSION

This chapter has two different sections. **Section 5.1** has details of result and some discussion **Section 5.2 5** includes comparison of different algorithms.

5.1. Results

In this research, clustering for database integration is performed using machine learning technique. Different algorithms are used to calculate the accuracy level of clustering. It must be mentioned that results obtained are highly dependent on dataset, rows and columns numbers and nature of data that they have. Slight change in their proportion can affect the results such as same case study can produce improved results if the number of data are increased. Similarly, if we have more number of rows and columns, then we obtain better results.

To calculate accuracy, we write down code in which correct predicted data is taken as input then divided by total number of data so we can get exact accuracy. To find whether our classifier prediction is true or not we used *confusion matrix* is used. It calculates accuracy in terms of percentage and visualize in the form of matrix. It basically calculates accuracy between prediction and actual data with the help of matrix.in confusion matrix the result on diagonal shows whether other prediction is closest to actual or not in the form of percentage. Different researchers have used different methods to compare the effectiveness of their proposed technique. We have used confusion matrix on four different algorithms. As a result, we achieved the best for clustering which will used in database integration as shown below in **Table 8**.

Table 8 : Comparison of algorithms based on accuracy

Sr. #	**Algorithm**	**Accuracy**
1.	K Means Algorithm	69.1%
2.	Agglomerative Clustering	40%
3.	Spectral Clustering	55%
4.	Mean Shift	40.8%

The above table shows that best results are achieved for K-means algorithm by using case study of student management. It has total 480 records and 17 fields. Results shows that K-means algorithm shows more accuracy as compared to other algorithms. But it all depends on nature of data.

5.2. Comparison

In Machine Learning, different algorithms are used for finding out the accuracy of predicted data which is used in clustering. After finding accuracy we apply confusion matrix to all algorithms so see clear picture of prediction and actual data of all machine learning algorithms. This section contains all detail of comparison

5.2.1. K-means algorithm:

This section presents the K-means algorithm results which apply on case study. **Figure 5.1** is the graph showing clustering results of algorithm. **Figure 5.2** is the graph showing accuracy matrix this will be used for comparison of five different algorithm.

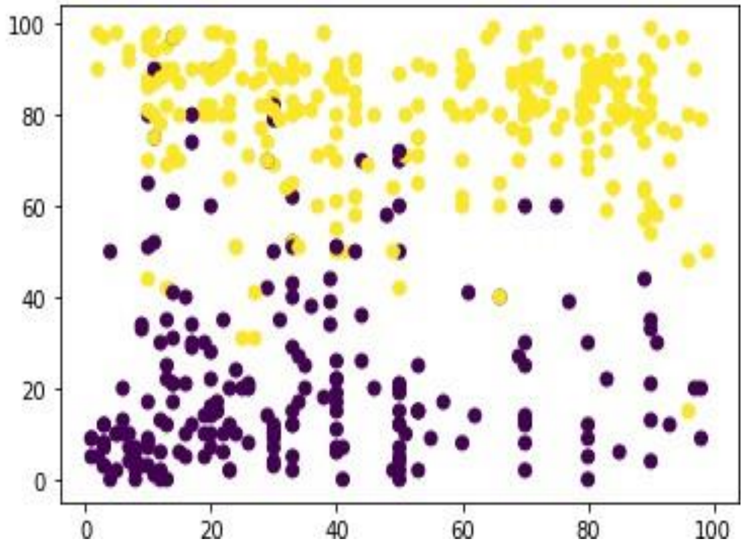


Figure 5.1: Clustering of K-means

The x-axis has the *student data* while y-axis depicts the faulty. This data is present in collection of student management system. With the help of K-means we separate data of student and faculty in the form of clusters. Yellow group of clusters shows student data while purple cluster depicts faculty data. These clusters will represent tables of SQ. So, from NoSQL database we can integrate data to SQL database with the help of K Means clustering.

5.2.1.1. Accuracy

We write piece of code for finding accuracy. We assigned one column as label then compare cluster with this label class if it will match stored in corrected data array. Then we calculate sum of this corrected data and divided it by total number of data for calculating accuracy. As result, we get accuracy of K-means is 69.1%.

5.2.1.2. Confusion Matrix for K Means

K Means as a classifier, is used to group same-class data together. How we can find out that classifier exactly classified data there is an easy way to understand with the help of confusion matrix.

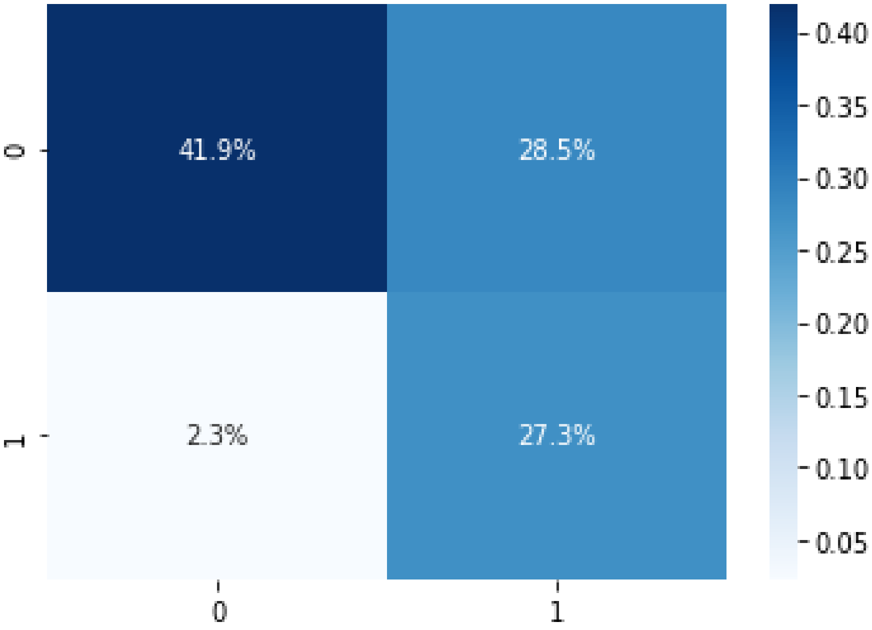


Figure 5.2: Confusion Matrix for K-means

This figure shows Confusion matrix which is used to show accuracy of actual data and prediction. Diagonal values like 41.9% and 27.3% shows 0,1 which shows that we get true values but at 1,1 the results are negative. same goes with 0, 0. But at 1,0 we get true values.

5.2.2. Agglomerative Clustering

We used agglomerative classifier for student management system to compare with k-means based on accuracy. This algorithm works classify similar data and represent it in the form of hierarchy.

5.2.2.1. Accuracy

We write piece of code for finding accuracy of agglomerative algorithm. We assigned one column as label then compare cluster with this label class if it will match stored in corrected data array. Then we calculate sum of this corrected data and divided it by total number of data for calculating accuracy. As a result, we get accuracy 40.0% which is less than K-means classifier.

5.2.2.2. Confusion Matrix

Agglomerative as a classifier, is used to group same-class data together. How we can find out that classifier exactly classified data there is an easy way to understand with the help of confusion matrix

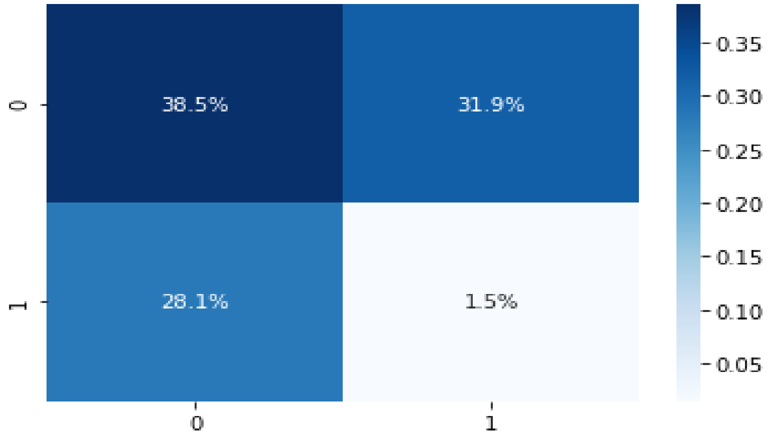


Figure 5.3: Agglomerative classifier

This figure represents Confusion matrix which is used to show accuracy of actual data and prediction of agglomerative clustering. Diagonal values like 38.5% and 1.5% displays at 0,1 which shows that we get true values but at 1,1 the results are negative. same goes with 0, 0. But at 1,0 axis we get true values.

5.2.3. Spectral Clustering

Spectral clustering is another technique of unsupervised machine learning. This portion represent accuracy of Spectral classifier. **Figure 5.4** is the confusion matrix for spectral clustering.so we can easily compare accuracy of this cluster with other algorithms.

5.2.3.1. Accuracy

We write piece of code for finding accuracy of spectral algorithm. We assigned one column as label then compare cluster with this label class if it will match stored in corrected data array. Then we calculate sum of this corrected data and divided it by total number of data for calculating accuracy. As a result, we get accuracy 63.5% which is less than K-means classifier.

5.2.3.2. Confusion Matrix:

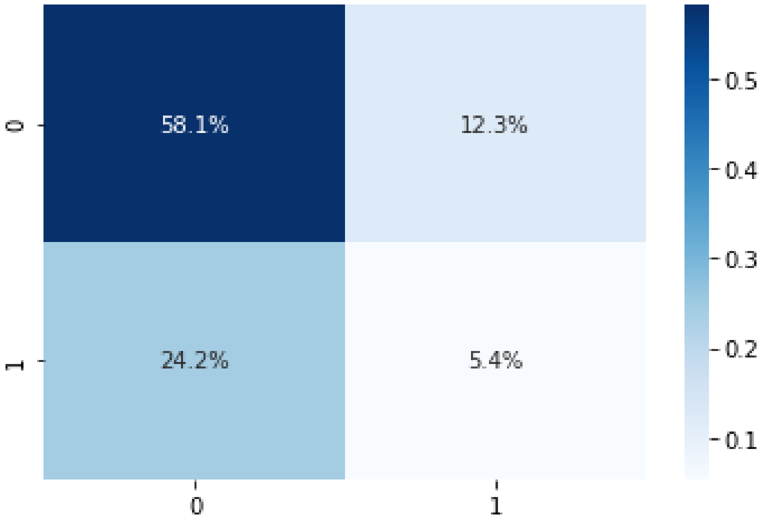


Figure 5.4: Spectral Clustering

This figure shows Confusion matrix which is used to show accuracy of actual data and prediction of agglomerative clustering. Diagonal values like 58.1% and 5.4% displays at 0,1 which

5.2.4. Mean Shift clustering

5.2.4.1. Accuracy:

We write piece of code for finding accuracy of Mean Shift algorithm. We assigned one column as label then compare cluster with this label class if it will match stored in corrected data array. Then we calculate sum of this corrected data and divided it by total number of data for calculating accuracy. As a result, we get accuracy 40.8% which is less than K-means classifier.

5.2.4.2. Confusion Matrix:

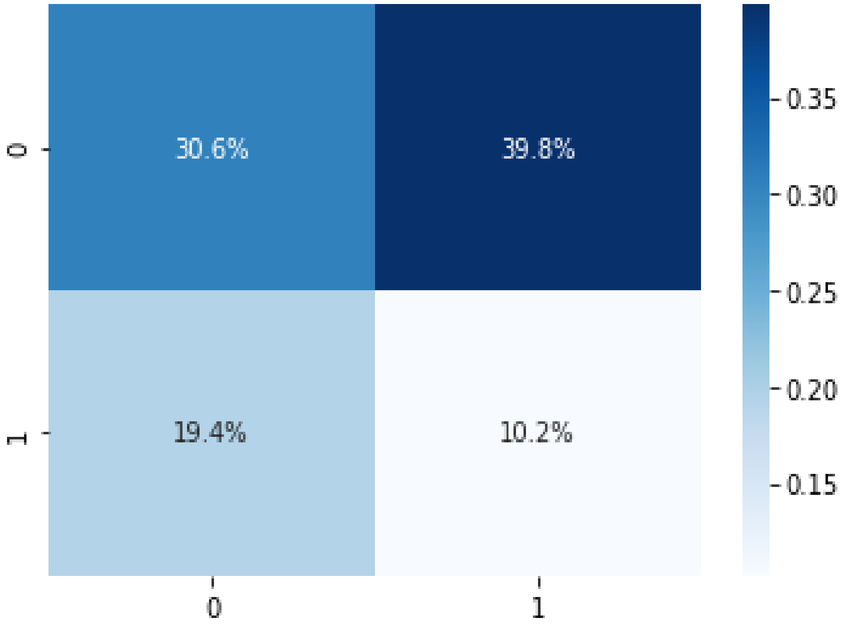


Figure 5.5: Mean Shift Clustering

It shows Confusion matrix which is used to show accuracy of actual data and prediction of agglomerative clustering. Diagonal values like 30.6% and 10.2% displays at 0,1 which shows that we get true values but at 1,1 the results are negative. same goes with 0, 0. But at 1,0 axis we get true values.

Chapter 6

Conclusion and Future Work

CHAPTER 6: CONCLUSION AND FUTURE WORK

Different classifier is used in this thesis for integration of heterogeneous databases. These algorithms are unsupervised machine learning algorithms. This technique is used to identify similar data from NoSQL in the form of clusters. As a result, clusters represent table f SQL.so we can integrate NoSQL data with SQL data with the help of unsupervised machine learning algorithms.

Two case studies (employee management system, and school management system) with different datasets have been conducted to check the efficiency of proposed method. The accuracy of clustering is measured with the help of confusion matrix. With the help of this matrix we can select which classifier is best for database integration. Results depends on nature of the data which are used in database.

6.1. Future Work:

In future work we will explore more unsupervised machine learning algorithms.so we can get better comparison of these algorithms and we can choose best for integration through all unsupervised machine learning algorithms**.**

6.2. Limitations:

Our solution towards database integration seemed to have few limitations. This method depends on amount of data available. Moreover, we only used four types of machine learning algorithms for clustering comparison. We only worked on unsupervised machine learning algorithm, but we can use supervised algorithm for better comparison. Our future focus will be to eliminate these limitations in the work.

REFERENCES:

[1] Sahbudin, MurtadhaArif Bin, Marco Scarpa, and Salvatore Serrano. "MongoDB Clustering using K- means for Real-Time Song Recognition." *2019 International Conference on Computing, Networking and Communications (ICNC).IEEE(2019)*

[2] Vyawahare, H. R., P. P. Karde, and V. M. Thakare. "A hybrid database approach using graph and relational database." *2018 International Conference on Research in Intelligent and Computing in Engineering (RICE).IEEE(2018)*

[3] Wu, Shanshan. "A Method for Building Shared Massive Heterogeneous IoT Data Environment."*2018 5th International Conference on Information Science and Control Engineering (ICISCE).IEEE(2018)*

[4] Sun, Yizhou, and Jiawei Han. "Mining heterogeneous information networks: a structural analysis approach." *Acm Explorations Newsletter 14.2 (2013):20-28.*

[5] Yang, Yongliang, S. James Adelstein, and Amin I. Kassis. "Target discovery from data mining approaches." *Drug discovery today 17 (2012):S16-S23.*

[6] Sun, Yizhou, Yintao Yu, and Jiawei Han. "Ranking-based clustering of heterogeneous information networks with star network schema." *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining.2009.*

[7] Sun, Yizhou, Charu C. Aggarwal, and Jiawei Han. "Relation strength-aware clustering of heterogeneous information networks with incomplete attributes." *Proceedings of the VLDB Endowment 5.5 (2012): 394- 405.*

[8] M.Plechawska-Wojcik,“ComparativeAnalysisofRelationalandNon-relational DatabasesintheContextofPerformanceinWebApplications,”*in Beyond Databases,Architecturesand Structures. Towards Efficient Solutions for Data Analysis and Knowledge Representation, vol. 716, S. Kozielski, D. Mrozek, P. Kasprowski, B. Małysiak-Mrozek, and D. Kostrzewa, Eds. Cham: Springer International Publishing, 2017, pp.153–164.*

[9] Huang, Yu, and Tie-jian Luo. "Nosql database: A scalable, availability, high performance storage for big data." *Joint International Conference on Pervasive Computing and the Networked World. Springer, Cham, (2013).*

[10] Bourennani, Farid, MouhcineGuennoun, and Ying Zhu. "Clustering relational database entities using k-means." *2010 2nd International Conference on Advances in Databases, Knowledge, and Data Applications.IEEE(2010).*

[11] Liu, Guofeng, Shaobin Huang, and Yuan Cheng. "Research on semantic integration across heterogeneous data sources in grid." *Frontiers in computer education. Springer, Berlin, Heidelberg, 2012. 397-404.*

[12] Bettoumi, Safa, ChirazJlassi, and NajetArous. "Comparative study of k-means variants for mono- view clustering." *2016 2nd International Conference on Advanced Technologies for Signal and Image Processing (ATSIP).IEEE(2016).*

[13] R. Lawrence, "Integration and Virtualization of Relational SQL and NoSQL Systems Including MySQLandMongoDB,"*2014 International Conferenceon Computational Scienceand Computational Intelligence, Las Vegas, NV, USA, 2014, pp. 285–290, doi:10.1109/CSCI.2014.56.*

[14] Christian Robert, "Machine Learning, a Probabilistic Perspective", *American Statistical Association, Vol. 27, pp 62-63, 2014*.

[15] Fabrizio Sebastiani, "Machine Learning in Automated Text Categorization", *ACM Computing Survey, Journal, Vol. 34, pp 1-47, USA, 2002*.

[16] Na, Shi, Liu Xumin, and Guan Yong. "Research on k-means clustering algorithm: An improved k- means clustering algorithm." *2010 Third International Symposium on intelligent information technology and security informatics.IEEE(2010)*

[17] Marco Barreno, Blaine Nelson, Anthony D, "The Security of Machine Learning", *Springer Link, Vol. 81, pp 121-148, 2010*.

[18] Reddy, M. P., et al. "A methodology for integration of heterogeneous databases." *IEEE transactions on knowledge and data engineering 6.6 (1994):920-933.*

[19] Liu, Yuelan, Xiaoming Liu, and Lu Yang. "Analysis and design of heterogeneous bioinformatics database integration system based on middleware." *2010 2nd IEEE International Conference on Information Management and Engineering.IEEE(2010)*

[20] Köpcke, Hanna, and Erhard Rahm. "Frameworks for entity matching: A comparison." *Data & Knowledge Engineering 69.2 (2010):197-210.*

[21] Ovaska, Kristian, et al. "Large-scale data integration framework provides a comprehensive view on glioblastoma multiforme." *Genome medicine 2.9 (2010):65.*

Liu, Zhan, et al. "Using semantic web technologies in heterogeneous distributed database system: A case study for managing energy data on mobile devices." *International Journal of New Computer Architectures and their Applications (IJNCAA) 4.2 (2014):56-69.*

[22] Li, Ming, et al. "Cloud-based ubiquitous object sharing platform for heterogeneous logistics system integration." *Advanced Engineering Informatics 38 (2018):343-356.*

[23] Chang, Fei, et al. "A universal heterogeneous data integration standard and parse algorithm in real- time database." *Proceedings of the 2012 International Conference on Information Technology and Software Engineering. Springer, Berlin,Heidelberg(2013).*

[24] Tang, Hao, et al. "Online application of science and technology program oriented distributed heterogeneous data integration." *2011 3rd International Conference on Computer Research and Development. Vol. 1.IEEE(2011).*

[25] Shvaiko, Pavel, and Jérôme Euzenat. "Ontology matching: state of the art and future challenges." *IEEE Transactions on knowledge and data engineering 25.1 (2011): 158-176.*

[26] Kumar, R. Saravana, and G. Tholk appia Arasu. "A Fast K-Modes Clustering Algorithm to Warehouse Very Large Heterogeneous Medical Databases." *Journal of Computers and Software (2013):1476.*

[27] Zhao, Huimin. "Matching attributes across overlapping heterogeneous data sources using mutual information." *Journal of Database Management (JDM) 21.4 (2010):91-110.*

[28] Faraj, Azhi, Bilal Rashid, and Twana Shareef. "Comparative study of relational and non-relations database performances using Oracle and MongoDB systems." *International Journal of Computer Engineering and Technology 5 11 (2014):11-22.*

[29] Sun, Yizhou, et al. "Pathselclus: Integrating meta-path selection with user-guided object clustering in heterogeneous information networks." *ACM Transactions on Knowledge Discovery from Data (TKDD) 7.3 (2013):1-23.*

[30] Vathy-Fogarassy, Ágnes, and TamásHugyák. "Uniform data access platform for SQL andNoSQL database systems*." Information Systems 69 (2017):93-105.*

[31] Usama, Muhammad, et al. "Unsupervised machine learning for networking: Techniques, applications and research challenges." *IEEE Access 7 (2019):65579-65615.*

[32] Ping, Lu, and Zhao An-Xin. "The design and research of the integration for heterogeneous database in campus card system."*International Conference of IEEE on Mechatronic Science, Electric Engineering(2011).*

Completion Certificate

It is certified that the contents of thesis document titled "Automated Integration of Heterogeneous Databases" submitted by Ms. Mamoona Safdar with Registration No. 00000171508 have been found satisfactory for the requirement of degree.

Thesis supervisor: ______________________

(Dr. Urooj Fatima)