A PRELIMINARY REPORT ON

SMART SEARCH ENGINES: INSTEAD OF

SUBMITTED TO THE SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE
IN THE PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE AWARD OF THE DEGREE

OF

BACHELOR OF ENGINEERING (COMPUTER ENGINEERING)

SUBMITTED BY

HONEY TALREJA	**Exam Seat No.: 71714792E**
PRANAV SHIRUDE	**Exam Seat No.: 71714761E**
SHRIRANG MHALGI	**Exam Seat No.: 71714556F**
RIA MITTAL	**Exam Seat No.: 71714562L**

DEPARTMENT OF COMPUTER ENGINEERING

BRACT'S
VISHWAKARMA INSTITUTE OF INFORMATION TECHNOLOGY

SURVEY NO. 3/4, KONDHWA (BUDRUK), PUNE - 411048, MAHARASHTRA (INDIA).

SAVITRIBAI PHULE PUNE UNIVERSITY
2019 -2020

CERTIFICATE

This is to certify that the project report entitled

“ SMART SEARCH ENGINES: INSTEAD OF

Submitted by

HONEY TALREJA	**Exam Seat No.: 71714792E**
PRANAV SHIRUDE	**Exam Seat No.: 71714761E**
SHRIRANG MHALGI	**Exam Seat No.: 71714556F**
RIA MITTAL	**Exam Seat No.: 71714562L**

is a bonafide student of this institute and the work has been carried out by him/her under the supervision of **Prof. L. A. Bewoor** and it is approved for the partial fulfillment of the requirement of Savitribai Phule Pune University, for the award of the degree of **Bachelor of Engineering** (Computer Engineering).

Prof. L. A. Bewoor
Project Guide,
Department of Computer Engineering

Dr. S. R. Sakhare
Head,
Department of Computer Engineering

Dr. V. S.Deshpande
Director,
BRACT’s Vishwakarma Institute of Information Technology, Pune-48

Place : Pune
Date :

ACKNOWLEDGEMENT

It gives us great pleasure in presenting the preliminary project report on “Smart Search Engines: Instead Of “.

We would like to take this opportunity to thank our internal guide Prof. L. A. Bewoor for giving us all the help and guidance we needed. We are really grateful to her for her kind support. Her valuable suggestions were very helpful.

We are also grateful to Prof. S. R. Sakhare, Head of Computer Engineering Department, Vishwakarma Institute of Information Technology, for his indispensable support and suggestions.

At last, we are thankful to our parents, who have encouraged and inspired us with their blessings.

Honey Talreja

Pranav Shirude

Shrirang Mhalgi

Ria Mittal

(Dept. of Computer Engineering)

ABSTRACT

The current scenario of ever increasing information is rapidly progressing and demands a much efficient way of storage, organization and retrieval of information. The proposed system aspires to deliver relevant results which will be retrieved from the most gigantic information namespace over internet, stored in a format which will cater the needs of the system in the best possible way. A web based search system enriched with the concept of smart crawlers, feedback systems and a custom data structure to harvest a decisive search which would scale itself dynamically to attain relevancy is the primary goal of this project. The system also delivers a functionality of fetching alternatives for user specified search query. The user input query will be based on the category selected. We have added three categories to the system which are Automobiles, Colleges, Fruits and Vegetables. The alternatives that are provided will be based on the attributes of these categories individually as per user's choice.

Technical Keywords

Web Search Engines

- Web crawler
- Web indexing
- Page and site ranking

Machine Learning

- Algorithms
- Information extraction
- Nearest Neighbors
- Neural networks

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

1.1 Project Overview

This project harnesses the fundamental necessity of alternatives in the everyday life. Any person would fancy an opportunity where the replacement of a particular entity may be rewarding. The main ideology behind the project is replacing the existential search system with our relevant and dynamic system which will provide alternatives to users in frequently searched categories.

1.2 Motivation of the Project

Users frequently express their requirements in natural language which may not be understood by the system due to difference in language understanding. Also, many a times, users are in search of alternatives for many things. However, the existing system may sometimes fail to provide alternatives. This is due to the large volumes of data not stored in application specific format. Thus, a system that administers the requirements for frequent alternatives is pursued.

1.3 Problem Definition and Objectives

Today, people tend to consider alternative options for a lot of the things they come across in their daily lives. The traditional search engines regress due to their large volumes of data not stored in decent formats which inhibits retrieval of data for these types of specific applications. A web based search engine, which would assist the user to fetch alternatives only after understanding the context of the query is the prime intent.

The primary goal of system is to provide alternatives for user specified queries rooted within the scope of application. The system intends to maintain a custom data structure which stores the indexed information in a way so as to support easy and fast retrieval of alternatives. The fundamental objective is to create a pool of relevant options for users and provide assistance in decision making. In addition to the functionality of alternatives and options, the system also claims to be promising in terms of relevancy, regardless of the scale of data being processed.

1.4 Project Scope & Limitations

The system looks forward to ensure that it can serve various domains of the society. It finds major applications with students from diverse curriculum backgrounds. Researchers may also find the system useful who look for alternatives that may be more appropriate. Writers and novelists require a tool which can give them the best vocabulary according to their needs. In this new era of technology and continuously growing software and hardware standards, the IT sector needs the system to avail the alternatives that overcome the faults of an existing system. The users will have a choice to select categories from various options like fruits, vehicles and colleges.

1.5 Methodologies of Problem solving

The main methodologies used depend upon the user input as well as the requirements of the user. A database that stores relevant alternatives which are shaped using machine learning are involved. Nearest neighbors makes sure that all the relevant data is captured taking into consideration, all the attributes of the dataset. Following are the three categories in which the system specializes.

i. **Dietary Supplements:** Alternatives based on the vitamin and mineral contents of the food.
ii. **Automobile Recommendations:** Suggesting better options for cars and bikes based on various parameters such as cost, type, etc.
iii. **College Counseling:** Providing suggestion for colleges according to the courses offered, degree, etc.

CHAPTER 2: LITERATURE SURVEY

The internet has become the soul resource where all the user queries can be solved effortlessly. When it comes to finding any result, the search engines were not always as efficient as they are today. There has been a remarkable evolution in the working of search engines throughout the modern civilization. Today, the search engines have gained valuable smartness by recognizing human speech along with machine learning to process and rank the huge amount of information they access.

The working of any search engine includes many components like Information Retrieval, Web Crawling and Indexing. When a user provides the query, the IR tools help to access results that are ranked in a hierarchical format so as to provide relevance and trust. A web crawler is used to browse various existing websites in a methodical and logical format automatically. Indexing refers to the process of analyzing web pages based on their context and content like headings, titles or fields as per queries. Indexing is one of the best and fastest forms for searching the web.

A survey on trustworthiness of search engines results, which was carried out by the students of Kyoto University of Japan [1] Nakamura S. et al. (2007), consisted of 26 questions that were answered by over 1000 Internet users between December 25th and December 26th 2006, harvested the following results:

- About 68.7% of users make use of the search engines less than 10 times a day.
- Use of search engine to find basic information (68%) or detailed information (36.8%) about a specific topic. 7.4% of respondents to survey admitted the use of search engines for comparison.
- More than 50% of users only evaluate top 5 search results and only about 20% of them go beyond top 5 results.
- The other major results and analysis are depicted in the bar graphs below:

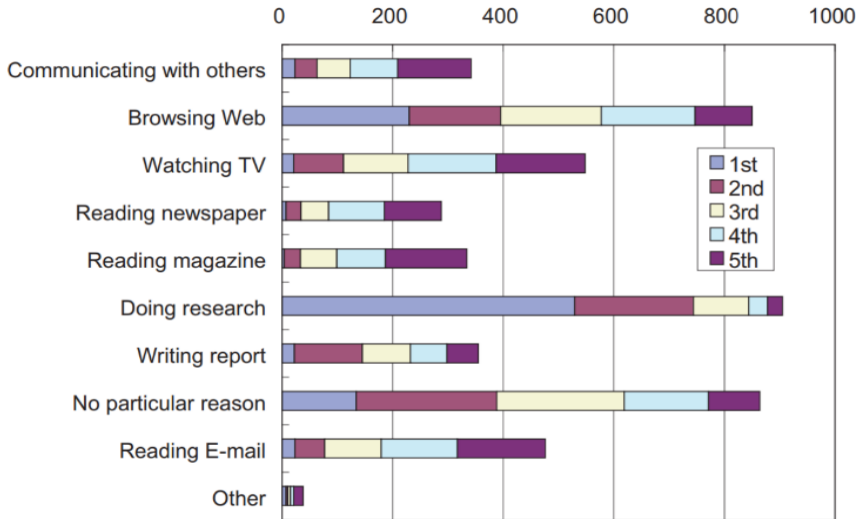

Figure 1: Situations when user search Web

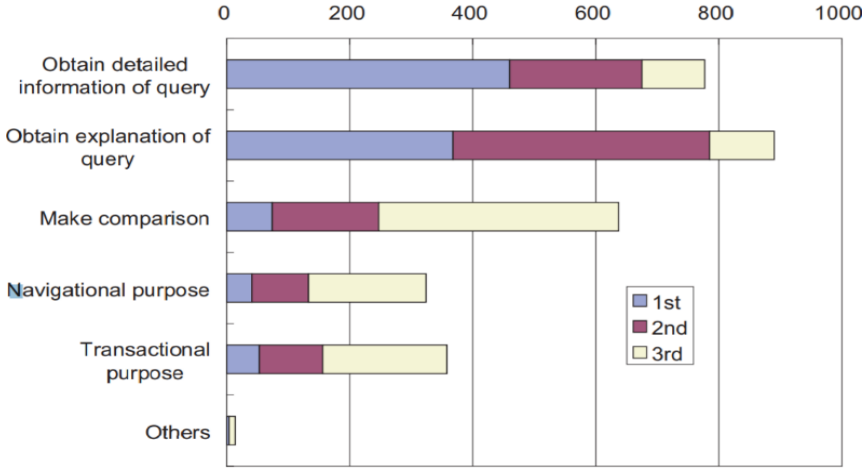

Figure 2: Reasons for Searching Web

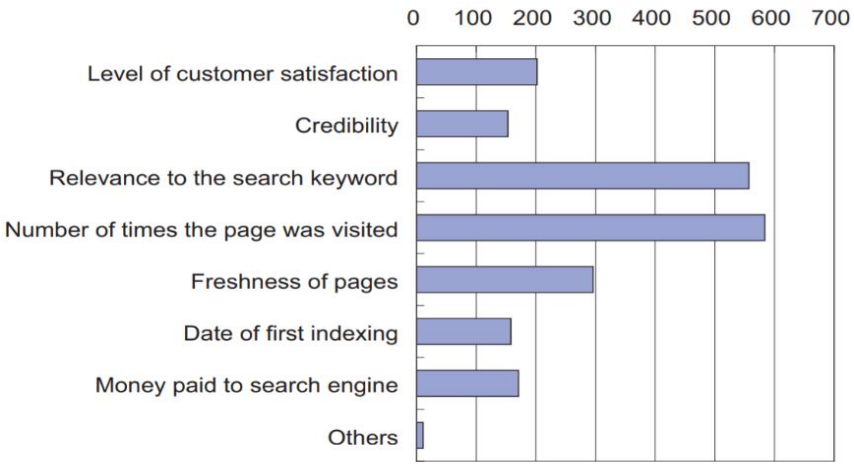

Figure 3: User beliefs about ranking

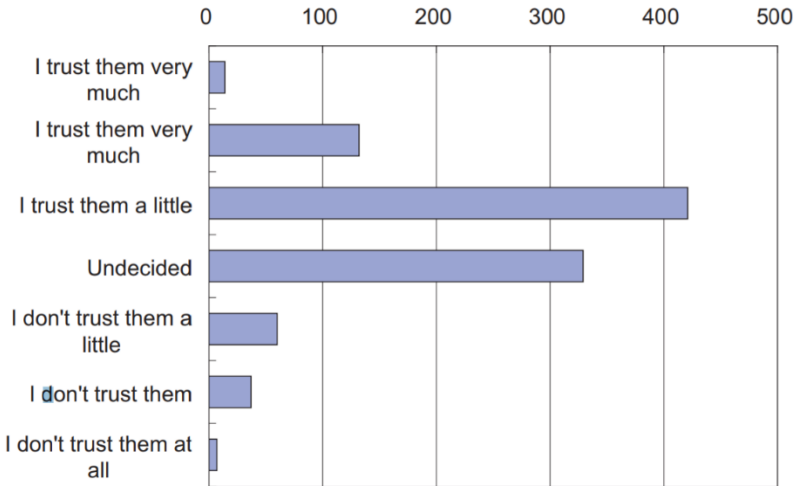

Figure 4: User Trust Levels

The survey concluded with an end note reflecting the inefficiency of search engines as well as the little understanding of user to make use of search engines properly. The user data gathered from this survey is of great significance to our proposed system supporting the need or revolution in search engine technology by introduction smartness to their model.

The prime task of a search engine is to retrieve relevant information from a huge repository of indexed database. Performance measures of a search engine include speed, variety, quantity and quality of fetched results (Google Search). A tradeoff is witnessed between these measures and thus sometimes leads to user dissatisfaction. The following paper discusses a comprehensive idea of working of the search engine and also proposes a system needed to overcome the gaps.

All the search engines harvest results by using different techniques which will be discussed in this paper. Before diving into the quality of results, it is necessary to understand how a search engine works to fetch those results for the users. The three most important phases of a search engine being- crawling, data indexing and ranking.

There are basically 3 types of search engines:

1. Crawler based search engines.
2. Human Powered directories.
3. Hybrid search engines.

The query “the color of darkness has 2 meanings”

1. Movie
2. The actual color of darkness

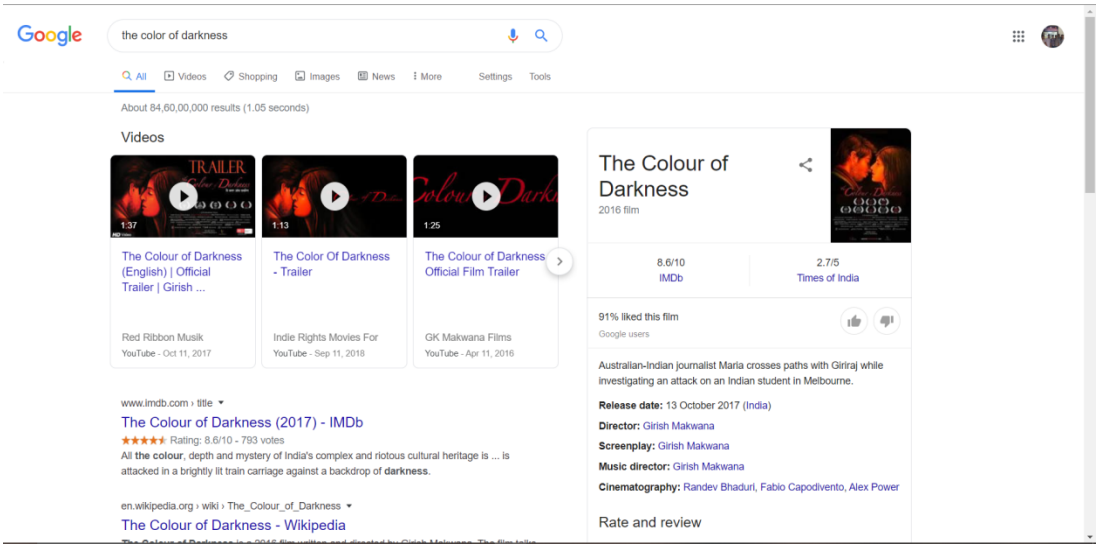

Figure 5: Google Search showing wrong result for Color of Darkness

The search engine was successful in giving the results for the movie. This is because the movie was searched quite a lot of time and is ranked high in Google search results. But if you scroll down the search results, you won’t find the actual color of darkness that is the color which you see when you close your eyes (Eigengrau) or the blackish color which you perceive when there is no light.

If we modify the same query, we find that the search engine provides a concrete result (White) which is totally irrelevant result as the color white is formed by combining red, blue and green light. How can darkness have white color!?

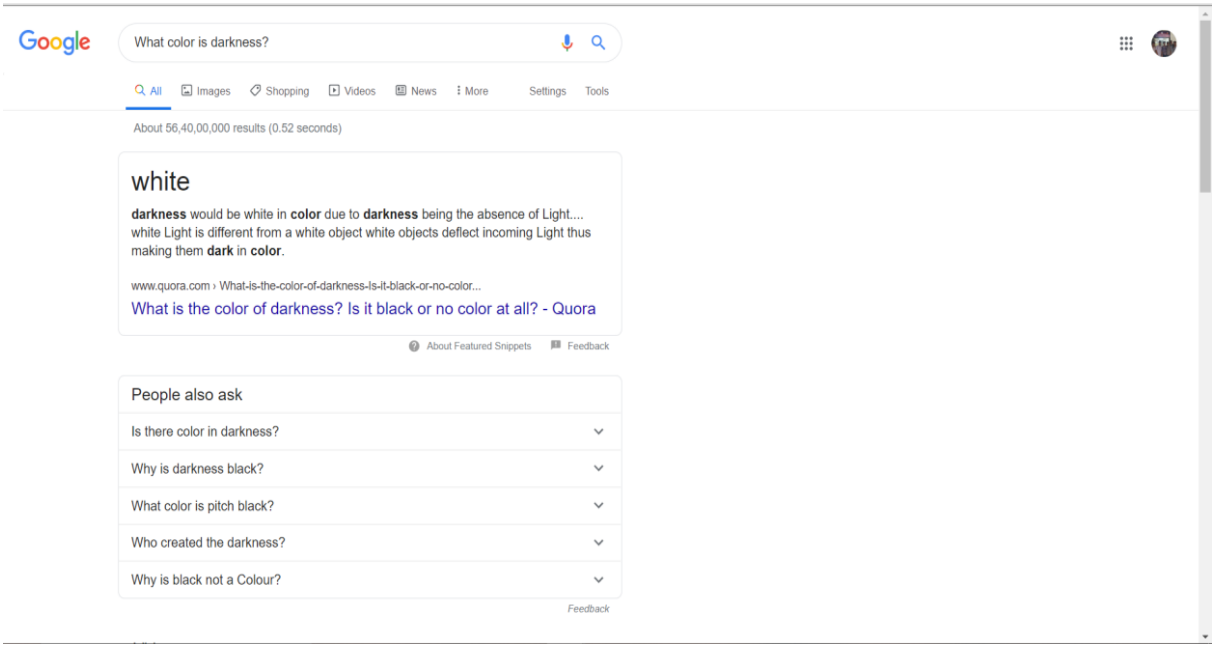

Figure 6: Modified Query for color of darkness

Through this study we have come across different aspects, advantages over other algorithms and limitations. Though lot of research has been done in the field of ranking, there is still immense scope available, since none of the ranking provides 100% percent relevance to the user information requirement for different queries. Although Google has introduced newer technological methods, it is impossible for it to provide a user friendly result always. Using these techniques have surely raised the bar of search engines, but when it comes to consistency and relevancy, it has shown many drawbacks. The major reason of it being the massive data that is present in this world which is not stored in user specific format. Thus, there is a need to enhance the existing search systems and inculcate smartness into it.

CHAPTER 3: SOFTWARE REQUIREMENT SPECIFICATION

3.1 Assumptions and Dependencies

The curiosity of many people looking for alternatives will be tackled in a relevant manner using the system. The system will scale its intelligence with more and more exposure to the queries. A variety of fields will take advantage of the system.

The proposed system will find its applications in the following fields:

i. **Dietary Supplements:** Alternatives based on the vitamin and mineral contents of the food.
ii. **Automobile Recommendations:** Suggesting better options for cars and bikes based on various parameters such as cost, type, etc.
iii. **College Counseling:** Providing suggestion for colleges according to the courses offered, degree, etc.

Depending upon the attributes in each category, alternatives will be provided. Each alternative would be extracted using a trained machine learning module.

3.2 Functional Requirements

3.2.1 Alternatives to Colleges

This feature enables the user to view similar colleges based upon all the attributes present in the dataset. Using a machine learning module, this will be possible. For example, if a user enters Indian Institute of Technology, Bombay, he/she should receive a set of colleges which offer the same or closest set of facilities, education, services and many more such attributes. All the attributes will be predefined in the dataset using which it will be possible to train the machine learning module.

3.2.2 Alternatives to Fruits and Vegetables

This feature is introduced for the user to view the fruits or vegetables which have the same or closest quality, contents, seasons and many more such attributes. If a user enters a query, for

example, apple, he/she should get the list of fruits and vegetables that have matching features. Based on a research, the contents of one apple, match the most to pears, thus, the user is able to view a list of fruits and vegetables which include pears.

3.2.3 Alternatives to Automobiles

It is a feature that displays the nearest match to the cars and similar vehicles based on user query. For example, if user enters Maruti Suzuki Swift, he/she gets the result having same engine type, fuel system, gear system and many more of such attributes, like, Maruti Suzuki Swift DZire, which is a similar car having a different model design.

3.2.4 Feedback

The user is also allowed to submit a feedback to the developers through an emoji feedback interface which will used to store results into the database. The feedbacks will be used to update the training module if users are not satisfied with the results.

3.3. External Interface Requirements

3.3.1 User Interface

The interface provided to the user will consist of a drop down menu for the user to select the required category for which the search should be performed. After selecting a category, the user enters the query in the text box and clicks on the search button. A list of alternatives is provided to the user thereafter.

3.3.2 Hardware Interface

To run the website on any device, a browser with CGI support working on any operating system (Windows, Linux or Mac). Since it is not a resource hungry program, it will run on most of the systems without hassle.

3.3.3 Software Interface

The system will be built using Django Framework (version 2.2 LTS) in Python (version 3.7.2). Various machine learning libraries like numpy, nltk and scikit-learn. Also, a database to store different categories as per requirements.

3.4. Non-Functional Requirements

3.4.1 Performance Requirements

The system will be available to the user through a website that will provide relevant and concrete results. All the results will be available according to analysis on recent trends and factors. Speed and efficiency are the key factors on which the system performs using GPUs. The website is robust and can handle multiple requests through the distributed architecture.

1. Relevant
2. Concrete Results
3. Robust
4. Efficient
5. Fast
6. Dynamic

3.4.2 Safety Requirements

The system will be distributed among various nodes to ensure maximum safety from threats and attacks. Secure Logins will protect user data and feedbacks provided by them.

3.5 System Requirements

3.5.1 Database Requirements

A classical database built in Microsoft SQL server is needed to be integrated within the project which is secure as well as portable over the internet. Its features should be such that there is ease of access of data from remote locations as well.

3.5.2 Software Requirements

The software should be compatible with all the new age browsers and operating systems like Windows, Linux and MacOS. It should be a web application which is deployed through a server. It should be accessible through a simple internet connection.

3.5.3 Hardware Requirements

No extra hardware is required for the system. Daily use laptop computers can be used to train and test the model. Hosting platform will provide the necessary hardware for hosting the progressive web application.

3.6 Detailed Use Cases

3.6.1 Select Category

Use case name	Select Category
Trigger	The user clicks on the drop down menu provided for selecting the category.
Precondition	The user has an internet connection.
Basic path	The user can see a list of categories. The user selects one of the categories for which they require an alternative.
Post condition	User can see the full list of categories.

Table 1: Use Case- Select Category

3.6.2 Enter Query

Use case name	Enter Query
Trigger	The user clicks on the text input box next to the category menu.
Precondition	The user has an internet connection.
Basic path	The user clicks on the text input provided and enters the query according to the category selected. The system identifies the query search and gives suggestions according to the input. The query is then sent to the back-end for processing.
Post condition	System starts looking for alternatives in the database.

Table 2: Use Case- Enter Query

3.6.3 View Alternatives

Use case name	View Alternatives
Trigger	The user has entered query or has clicked the search button.
Precondition	The user has an internet connection.
Basic path	The user gets a list of alternatives from the category selected. He/she gets a concrete answer based on recent trends and analysis done on the database.
Post condition	List of alternatives is displayed to the user.

Table 3: Use Case- View Alternatives

3.7 Analysis Models: SDLC Model to be applied

Classical methods of software development have many disadvantages:

- huge effort during the planning phase
- poor requirements conversion in a rapidly changing environment
- treatment of staff as a factor of production

New methods: Agile Software Development

Scrum - an agile process

SCRUM is an agile, lightweight process for managing and controlling software and product development in rapidly changing environments.

a. Iterative, incremental process

b. Team-based approach

c. developing systems/ products with rapidly changing requirements

d. Controls the confusion of conflicting interest and needs

e. Improve communication and maximize cooperation

f. Protecting the team form disruptions

g. A way to maximize productivity

Components of Scrum

a. Scrum Roles

b. The Process

c. Scrum Artifacts

Scrum Master

1. Represents management to the project
2. Typically filled by a Project Manager or Team Leader
3. Responsible for enacting Scrum values and practices
4. The main job is to remove impediments

The Scrum Team

1. Typically 5-10 people
2. Cross-functional (QA, Programmers, UI Designers, etc.)
3. Members should be full-time
4. Team is self-organizing
5. Membership can change only between sprints

Product Owner

1. Acts like one voice (in any case)
2. Knows what needs to be built and in what sequence this should be done
3. Typically a product manager

The Process

1. Sprint Planning Meeting
2. Sprint
3. Daily Scrum
4. Sprint Review Meeting

Sprint Planning Meeting

1. A collaborative meeting at the beginning of each Sprint between the Product Owner, the Scrum Master and the Team
2. Takes 8 hours and consists of 2 parts (“before lunch and after lunch”)

Parts of Sprint Planning Meeting

1. 1st Part:
    a. Creating Product Backlog
    b. Determining the Sprint Goal.
    c. Participants: Product Owner, Scrum Master, Scrum Team
2. 2nd Part:
    a. Participants: Scrum Master, Scrum Team

b. Creating Sprint Backlog

Pre-Project/Kickoff Meeting

1. A special form of Sprint Planning Meeting
2. Meeting before the begin of the Project

Sprint

1. A month-long iteration, during which is incremented a product functionality
2. NO outside influence can interference with the Scrum team during the Sprint
3. Each Sprint begins with the Daily Scrum Meeting

Daily Scrum

1. Is a short (15 minutes long) meeting, which is held every day before the Team starts working
2. Participants: Scrum Master (which is the chairman), Scrum Team
3. Every Team member should answer on 3 questions

Questions

1. What did you do since the last Scrum?
2. What are you doing until the next Scrum?
3. What is stopping you getting on with the work?

Sprint Review Meeting

1. Is held at the end of each Sprint
2. Business functionality which was created during the Sprint is demonstrated to the Product Owner
3. Informal, should not distract Team members of doing their work

Scrum Artifacts

1. Product Backlog

2. Sprint Backlog
3. Burn down Charts

Product Backlog

Requirements for a system expressed as a prioritized list of Backlog Items

1. Is managed and owned by a Product Owner
2. Spreadsheet (typically)
3. Usually is created during the Sprint Planning Meeting
4. Can be changed and re-prioritized before each Planning Meeting

Estimation of Product Backlog Items

1. Establishes team's velocity (how many efforts a Team can handle in one Sprint)
2. Determining units of complexity.
    a. Size-category ("T-Shirt size")
    b. Story points
    c. Work days/work hours
3. Methods of estimation:
    a. Expert Review
    b. Creating a Work Breakdown Structure (WBS)

Sprint Backlog

1. A subset of Product Backlog Items, which define the work for a Sprint
2. Is created ONLY by Team members
3. Each Item has its own status
4. Should be updated every day
5. No more than 300 tasks in the list
6. If a task requires more than 16 hours, it should be broken down
7. The team can add or subtract items from the list. Product Owner is not allowed to do it.

Burn down Charts

1. Are used to represent “work done”.
2. Are wonderful Information Radiators
3. 3 Types:
    a. Sprint Burndown Chart (progress of the Sprint)
    b. Release Burndown Chart (progress of release)
    c. Product Burndown chart (progress of the Product)

CHAPTER 4: SYSTEM DESIGN

4.1 System Architecture

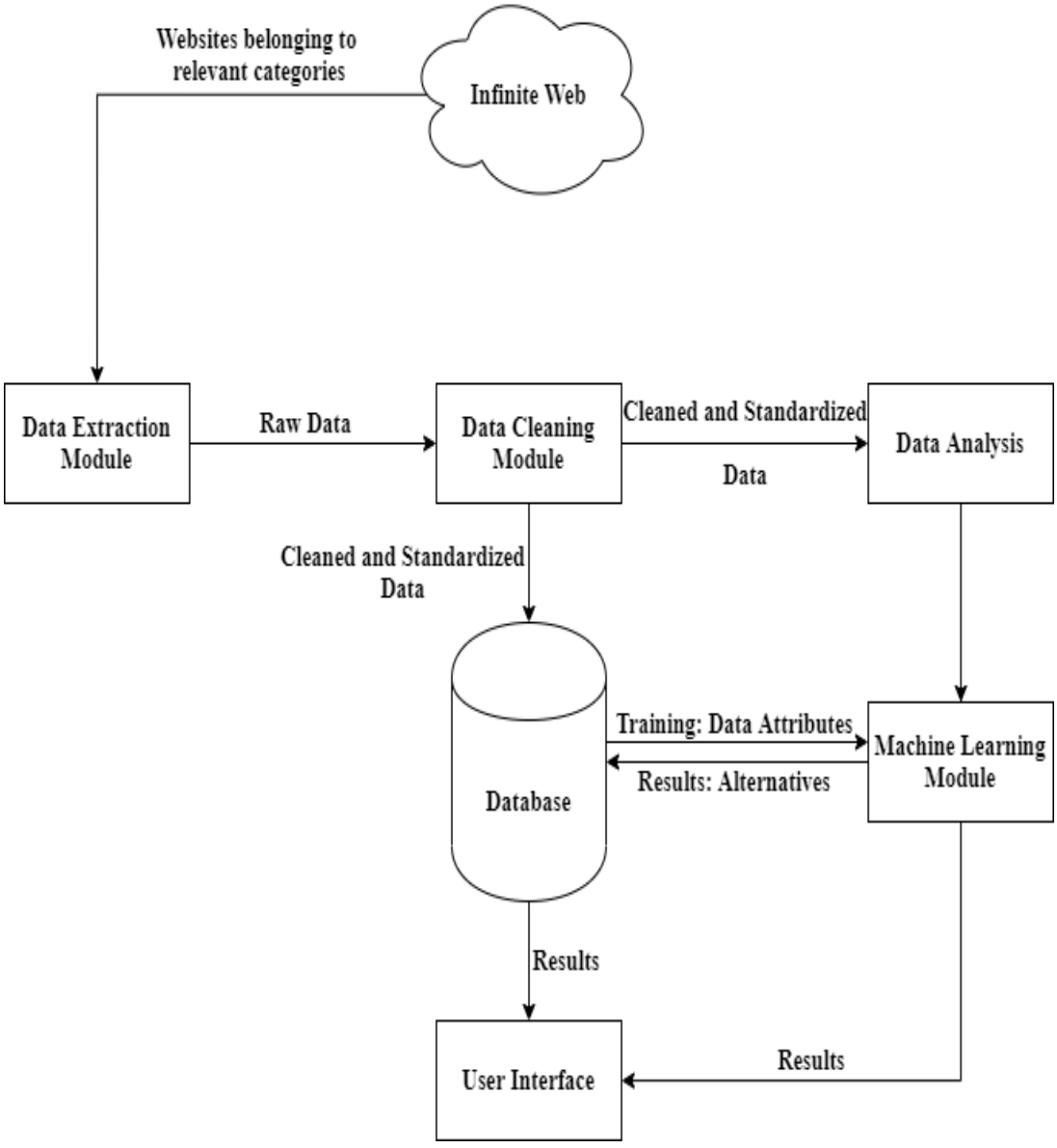

Figure 7: System Architecture

4.2 Data Flow Diagrams

4.2.1 Data Flow Diagram Level 0

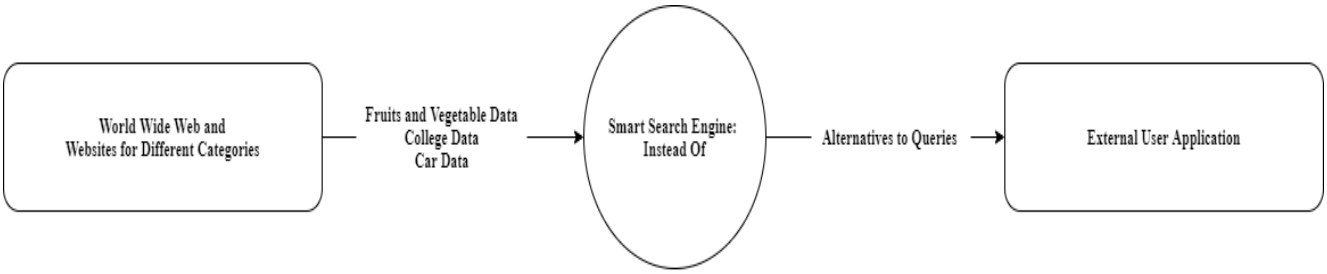

Figure 8: Data Flow Diagram Level 0

4.2.1 Data Flow Diagram Level 1

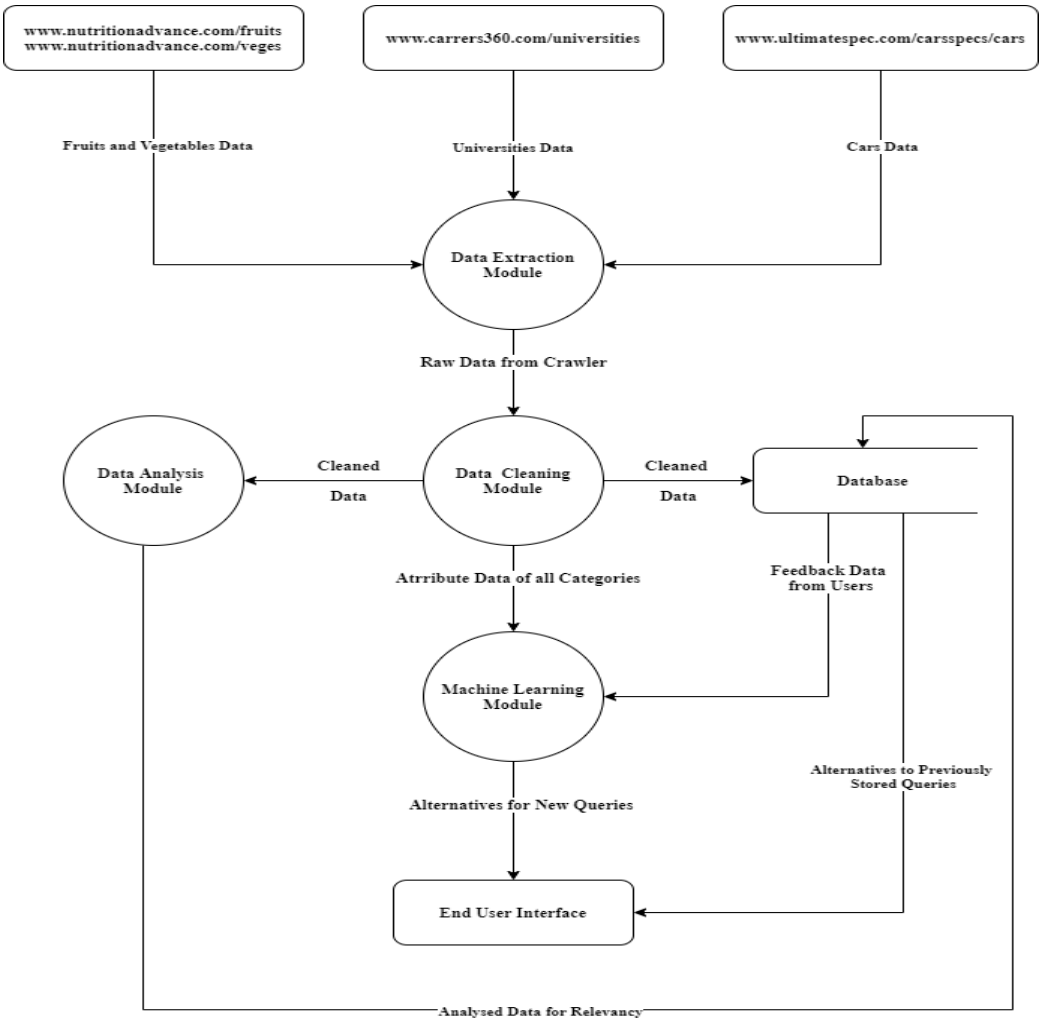

Figure 9: Data Flow Diagram Level 1

4.3 Entity Relationship Diagrams

4.3.1 College Dataset

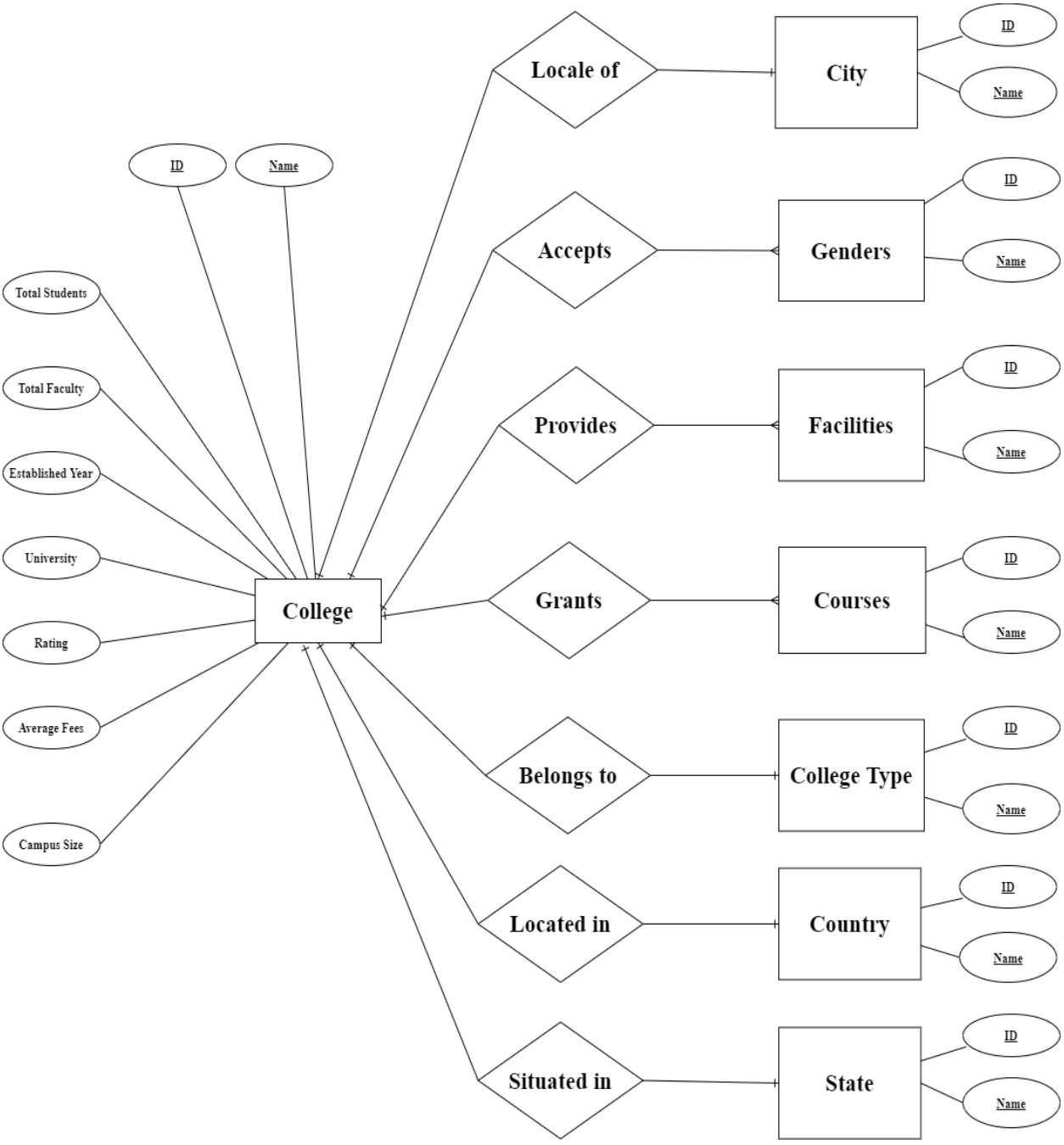

Figure 10: ER diagram for College Database

4.3.2 Fruits and Vegetable Dataset

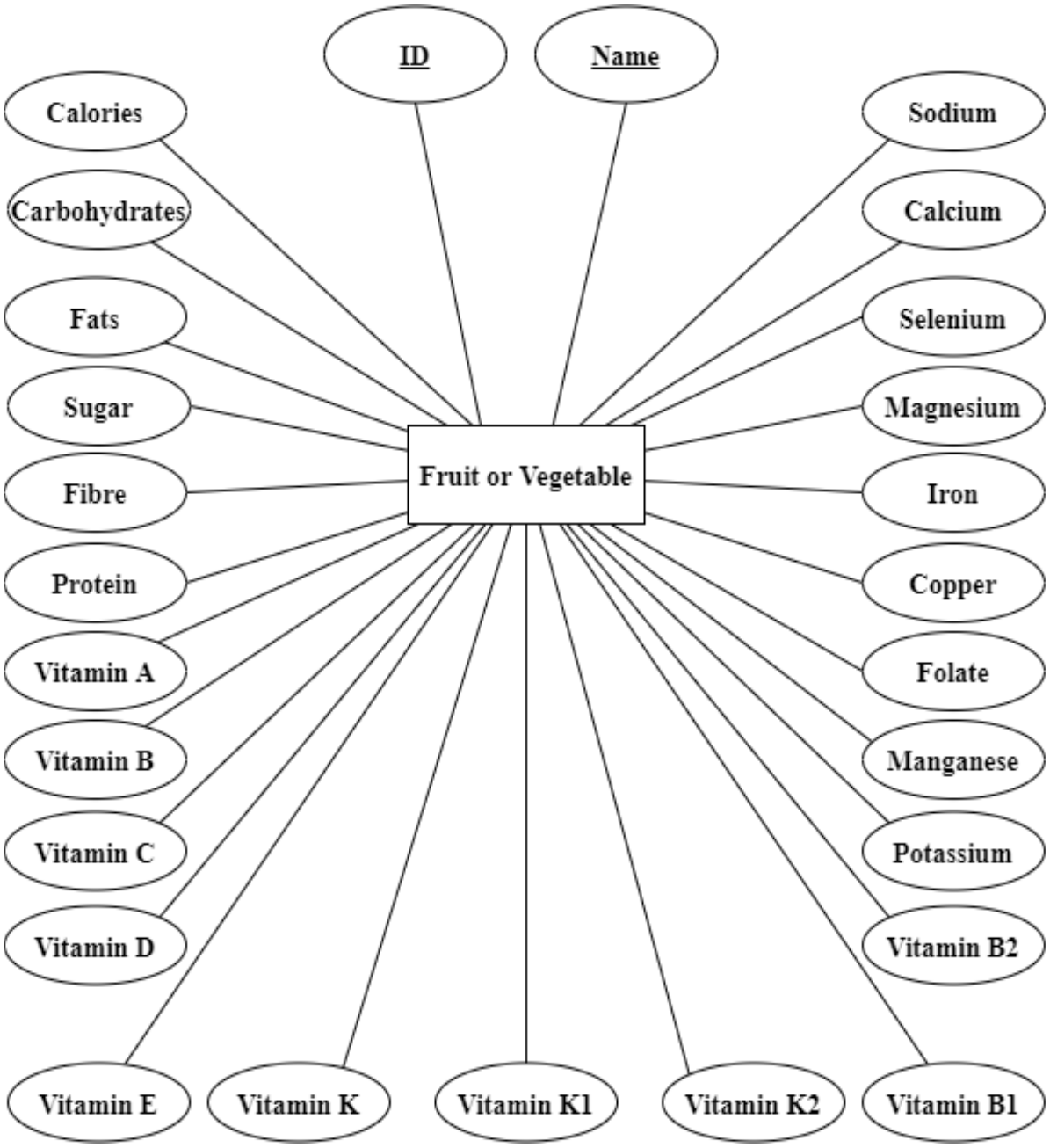

Figure 11: ER diagram for Fruit and Vegetable Database

4.3.2 Automobile Dataset

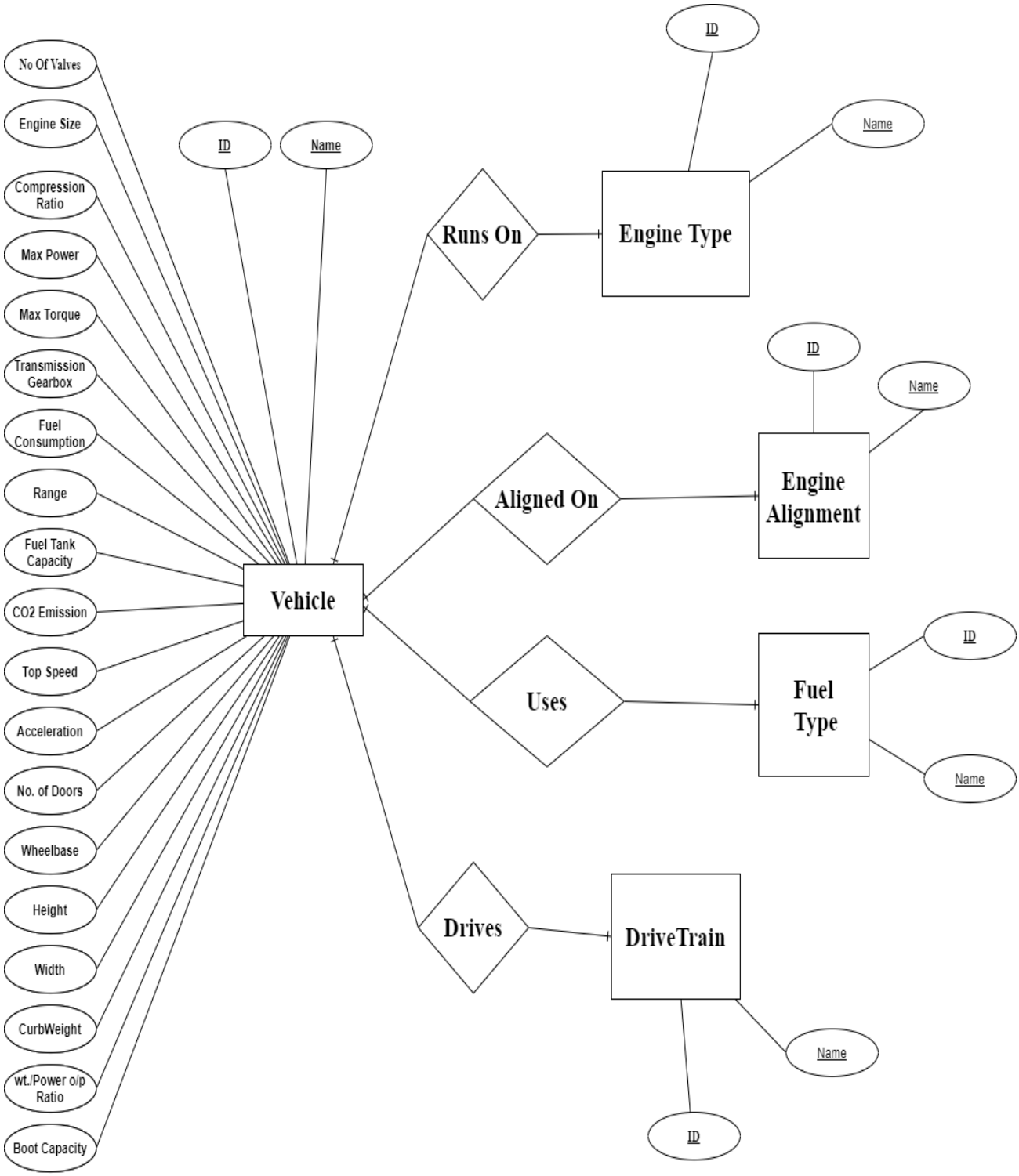

Figure 12: ER diagram for Automobile Database

4.4 UML diagrams

4.4.1 Activity Diagram

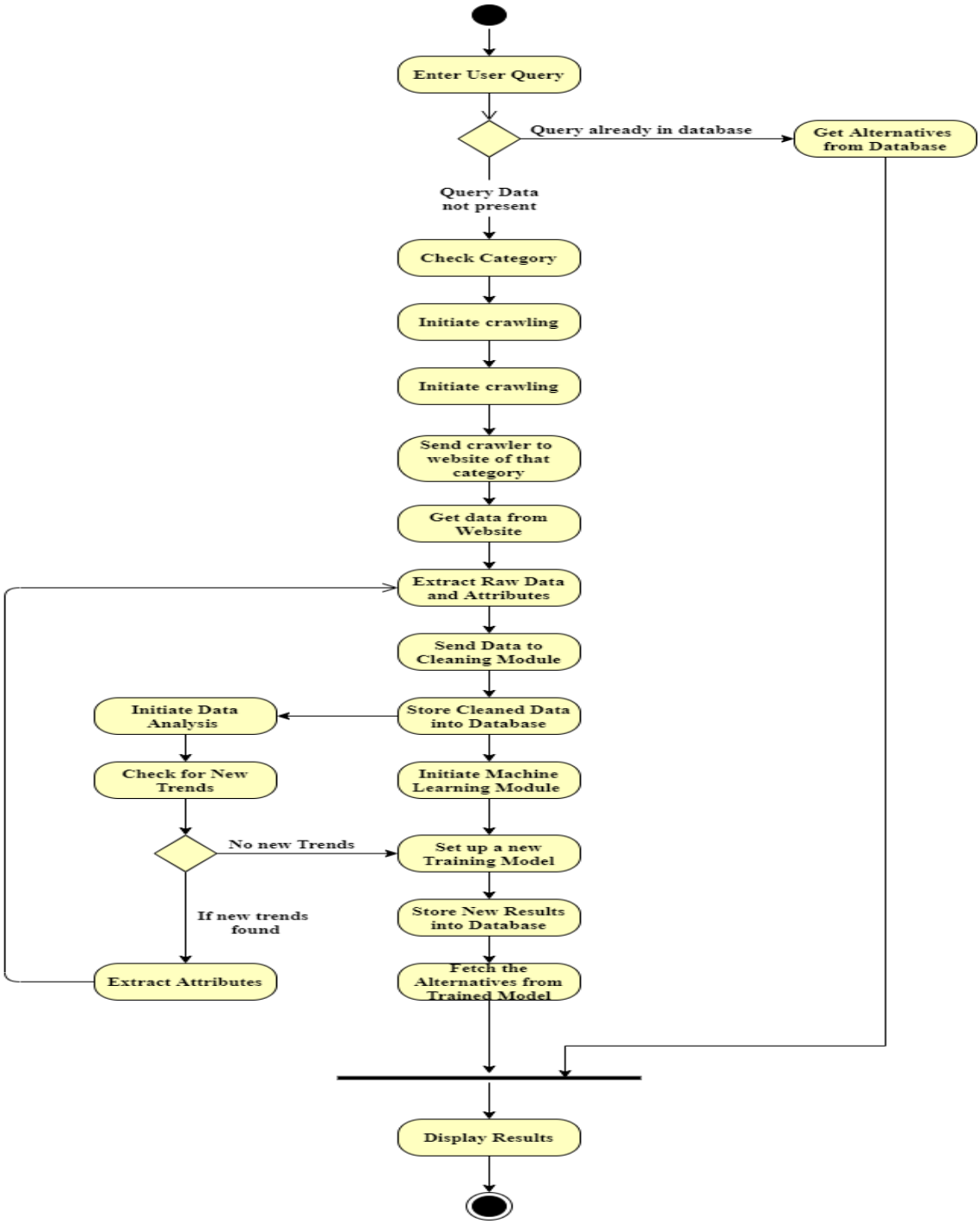

Figure 13: Activity Diagram

4.4.2 Component Diagram

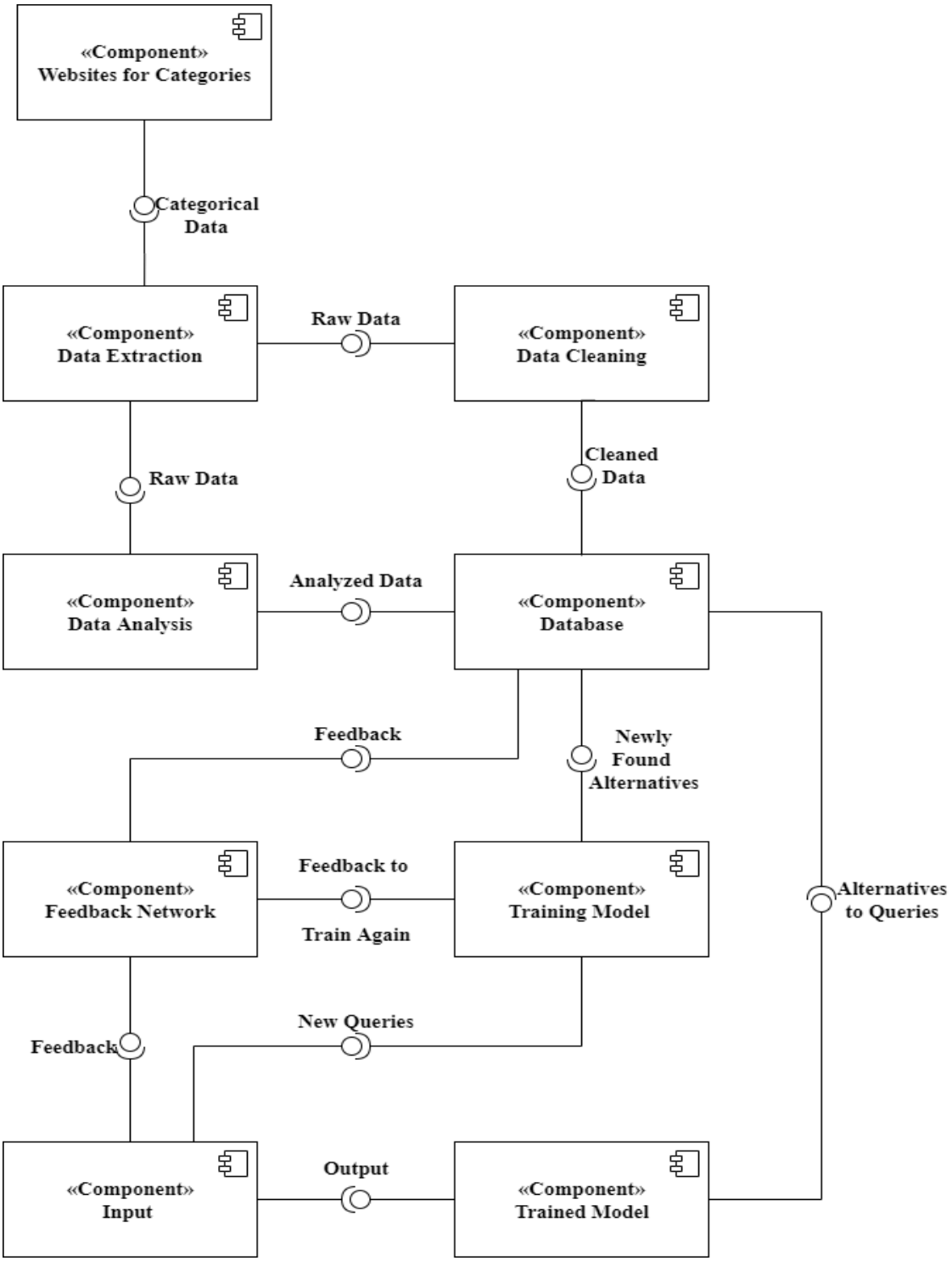

Figure 14: Component Diagram

4.4.3 Use Case Diagram

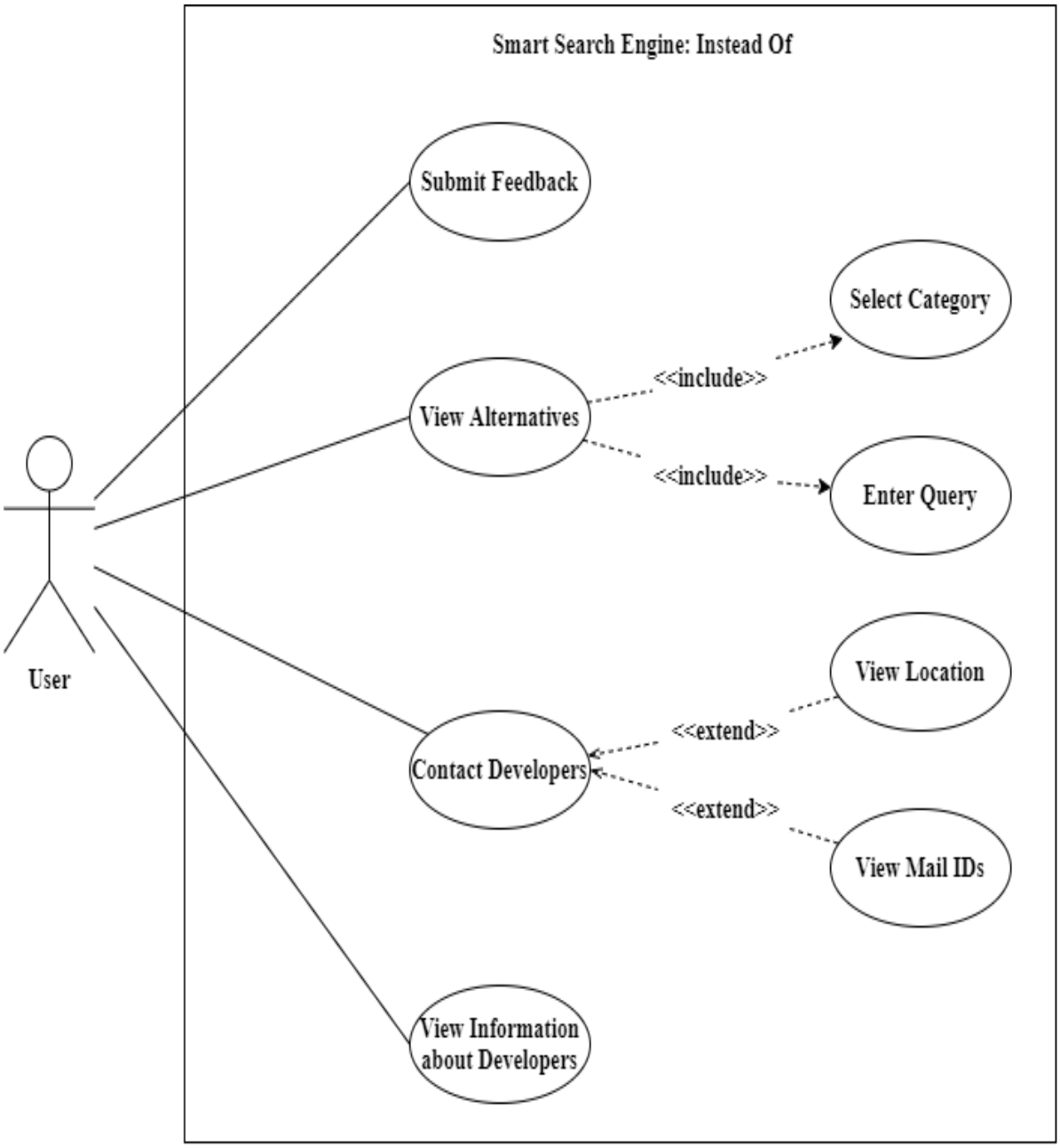

Figure 15: Use Case Diagram

4.5 Database Diagrams

4.5.1 College Master

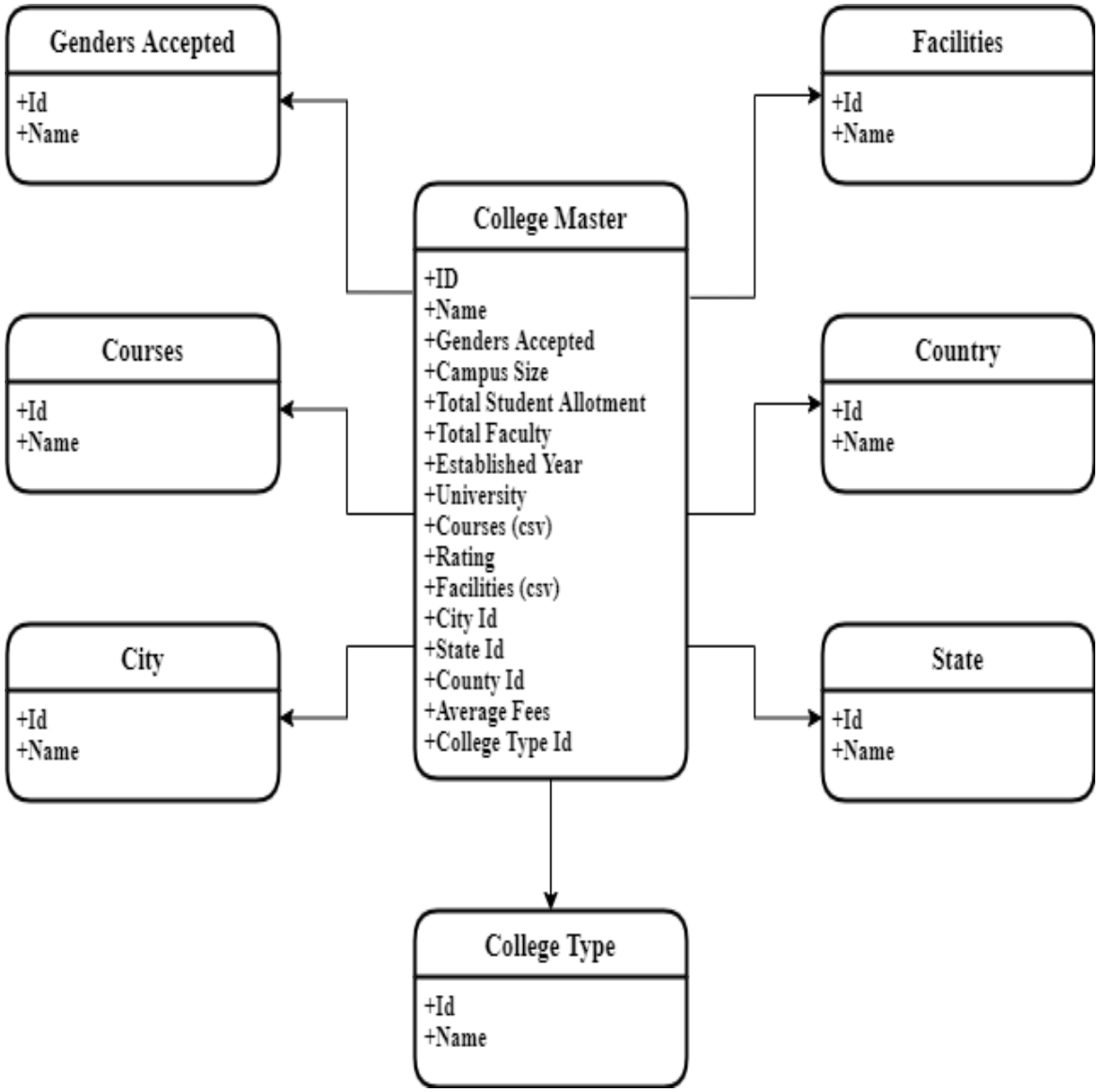

Figure 16: Database diagram for College Data

4.5.2 Fruit and Vegetable Master

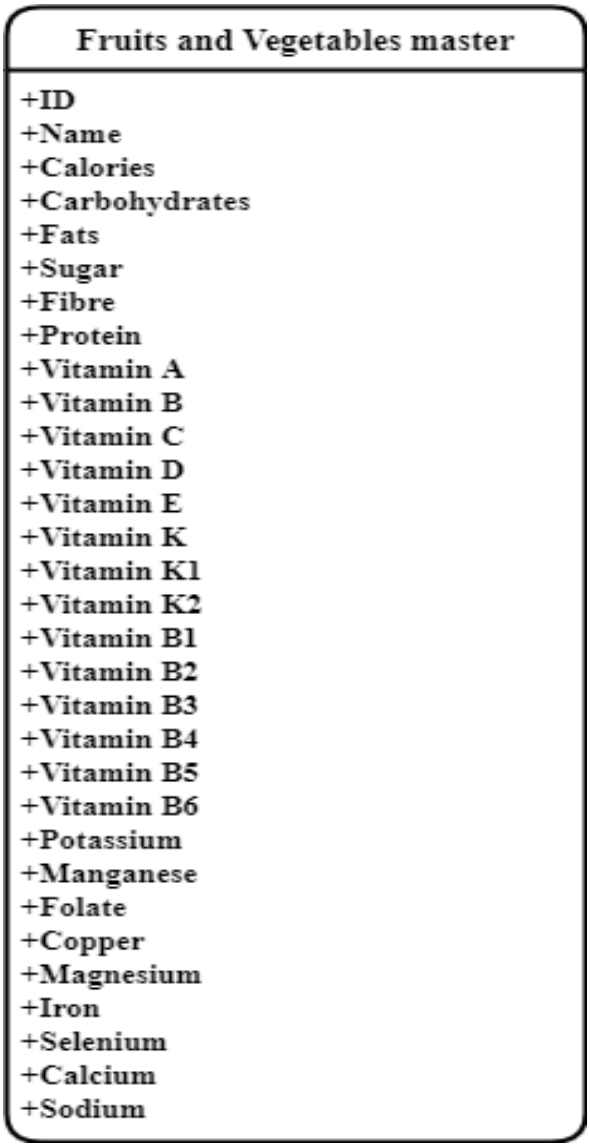

Figure 17: Database diagram for Fruit and Vegetable Data

4.5.3 Automobile Master

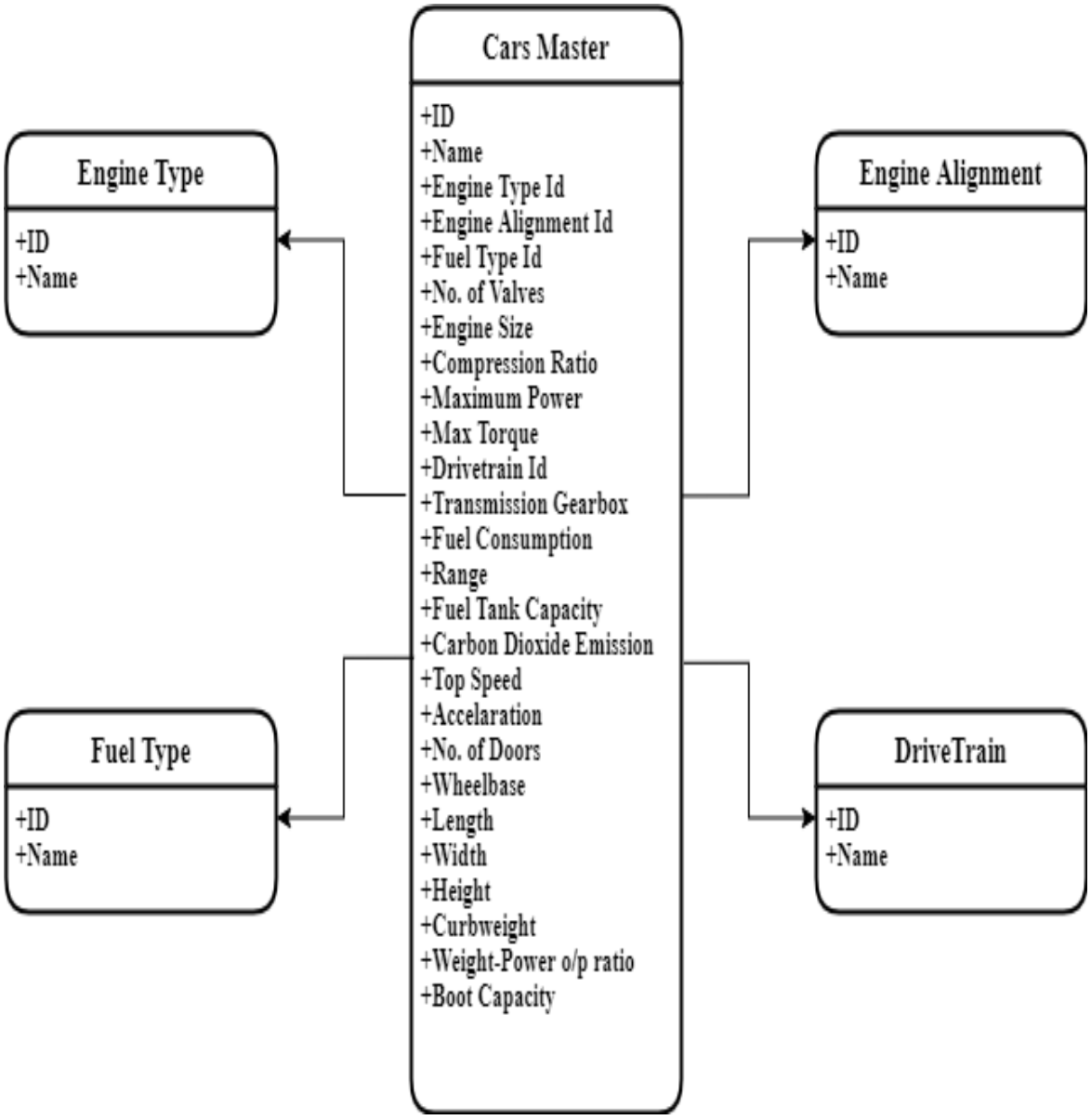

Figure 18: Database diagram for Automobile data

CHAPTER 5: PROJECT PLAN

5.1 Project Estimates

5.1.1 Reconciled Estimates

5.1.1.1 Cost Estimate

In the real world, there are 3 types of models included in estimation using the COCOMO Model. They can be described as follows:

1. **Organic:** Small team size, 2-50 KLOC, stable and little innovation required
2. **Semi Detached:** 50-300 KLOC, medium-sized, average abilities, medium time-constraints
3. **Embedded:** > 300 KLOC, large project team, complex, innovative, severe constraints

The proposed system will utilize the Organic Model for Cost Estimation.

Step 1:

$$E = a * (KLOC)^b$$

Where, E = Effort in staff-months

a, b = Coefficients to be determined

KLOC = Kilo lines of Code

Step 2:

For organic model a = 2.4 and b = 1.05

Assuming KLOC = 9

$$E = 2.4 * (9)^{1.05} = 24.11 \text{ staff} - \text{months}$$

Step 3:

Project Duration

$$TDEV = c * (E)^d$$

Where TDEV = Time for Development

For Organic c = 2.05 and d = 0.38

$$\text{TDEV} = 2.5 * (24.11)^{0.38} = 8.38 \text{ Months}$$

Step 4:

$$\text{Average Staff Size} = \frac{\text{E}}{\text{TDEV}} = \frac{24.11}{8.38} \approx 3 \text{ to } 4 \text{ staff}$$

Step 5:

$$\text{Productivity} = \text{P} = \frac{\text{KLOC}}{\text{E}} = \frac{9000}{24.11} = 373.29 \frac{\text{Loc}}{\text{staff}-\text{month}}$$

Step 6:

Assuming cost per month = ₹ 5000 /-

$$\text{Total Cost} = \text{E} * \text{cost per month} = 24.11 * 5000 = ₹\ \mathbf{1,20,550}/-$$

5.1.1.2 Time Estimates

Project Duration

$$\text{TDEV} = \text{c} * (\text{E})^{\text{d}}$$

Where TDEV = Time for Development

For Organic c = 2.05 and d = 0.38

$$\text{TDEV} = 2.5 * (24.11)^{0.38} = 8.38 \text{ Months}$$

5.2 Risk Management W.R.T. NP Hard Analysis

5.2.1 Risk Identification

			Impact		
	Risk Identification	**Probability**	**Schedule**	**Quality**	**Overall**
1	Continuous Changing of Web Data	Medium	Low	High	High
2	Project Modules Integration	Low	Low	High	High

Table 4: Risk Identification Table

5.2.2 Risk Analysis

Probability	**Value**	**Description**
High	Probability of occurrence is	> 75%
Medium	Probability of occurrence is	26 75%
Low	Probability of occurrence is	< 25%

Table 5: Risk Probability Definitions

Impact	Value	Description
Very high	> 10%	Schedule impact or Unacceptable quality
High	5 10%	Schedule impact or Some parts of the project have low quality
Medium	< 5%	Schedule impact or Barely noticeable degradation in quality. Low Impact on schedule or Quality can be incorporated

Table 6: Risk Impact Table

5.2.3 Overview of Risk Mitigation, Monitoring, Management

Following are the details for each risk.

Risk ID	1
Risk Description	Continuous changing of Web Data
Category	Requirements
Source	Software requirement Specification document.
Probability	Medium
Impact	High
Response	Mitigate
Strategy	Implementing Artificial Intelligence efficiently will resolve this issue
Risk Status	Identified

Table 7: Risk Table 1

Risk ID	2
Risk Description	Project Modules Integration
Category	Development Environment
Source	Software Design Specification documentation review.
Probability	Low
Impact	High
Response	Mitigate
Strategy	Implementing integration testing will resolve this issue
Risk Status	Identified

Table 8: Risk Table 2

5.3 Project Schedule

5.3.1 Project task set

- Requirement Gathering and Analysis
- Planning
- Design and Modeling
- Coding
- Testing
- Delivery
- Documentation
- Operations
- Meetings
- Scheduling and Formal Technical Reviews
- Cost and Effort Estimation

5.3.2 Timeline Chart

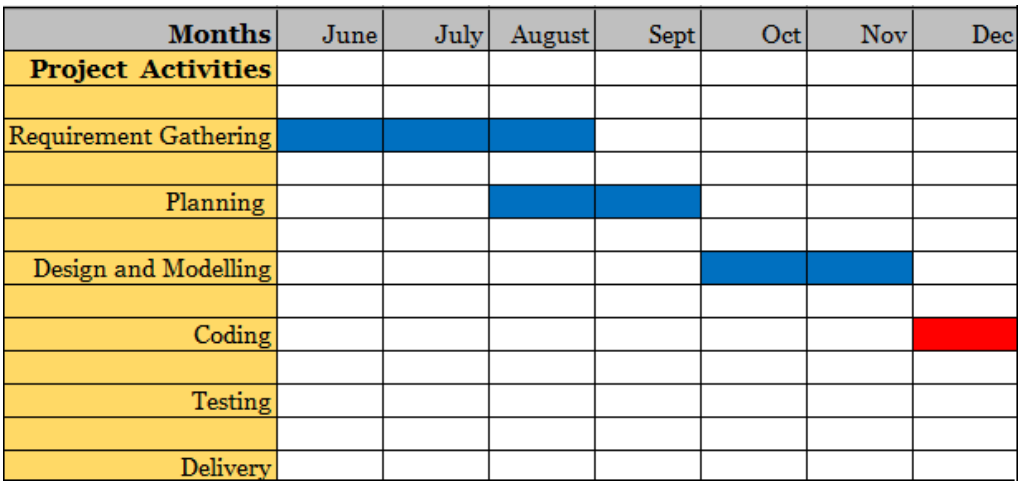

Figure 19: Timeline Chart

5.3.3 Task Network

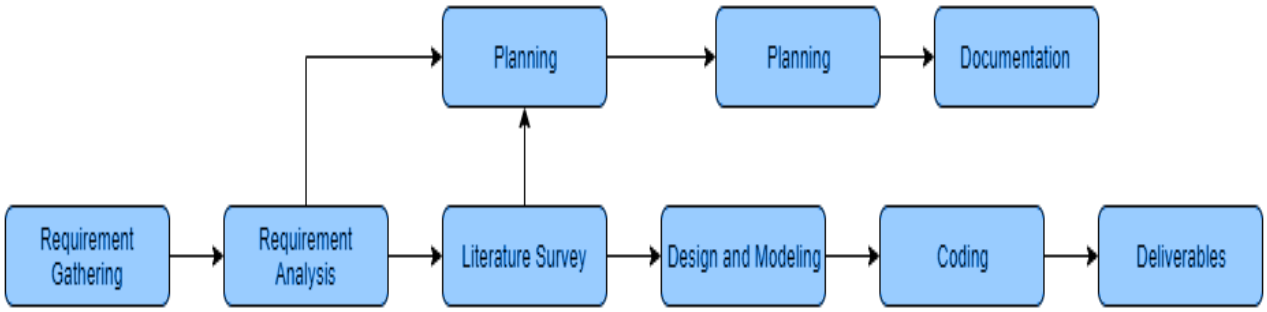

Figure 20: Task Network

5.4 Team Organization

5.4.1 Team structure

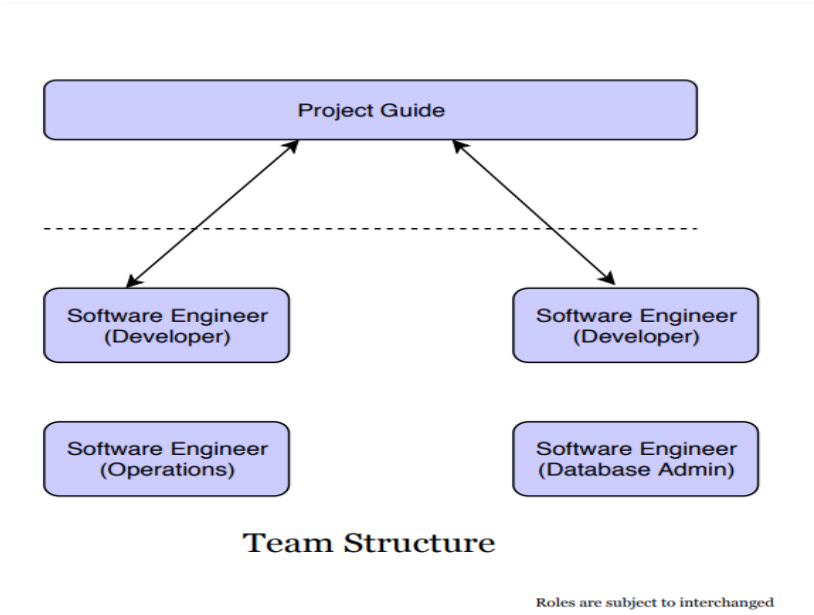

Figure 21: Team Structure

5.4.2 Management reporting and communication

To sustain improvements and creativity in the idea of project, communication with stakeholders was maintained throughout the lifecycle of the project development. Timely reporting and formal reviews regarding the project guide were submitted to the project guide, Prof. L. A. Bewoor. Interactions with domain experts were conducted to further refine the idea and scope of the project.

CHAPTER 6: PROJECT IMPLEMENTATION

6.1 Overview of Project Modules

6.1.1 Data Extraction and Crawler

This module is the first step towards collecting data related to the three categories in the project, viz. Colleges, Fruits and Vegetables and Automobiles. Raw data from websites was in unstructured formats. Noisy data extracted consisted of whitespaces, spelling errors, non-standard names and duplicates. Thus, there was a need to extract data first and then perform cleaning as well. The extraction unit consists of python scripts which run on different websites to collect information about the categories. The crawler is itself a collection of python programs which use BeautifulSoup-bs4 as the main html page extraction tool. All the data collected is stored in a csv file which in turn is used to update the database after being cleaned. The method of data extraction by the module is explained in the figure given below.

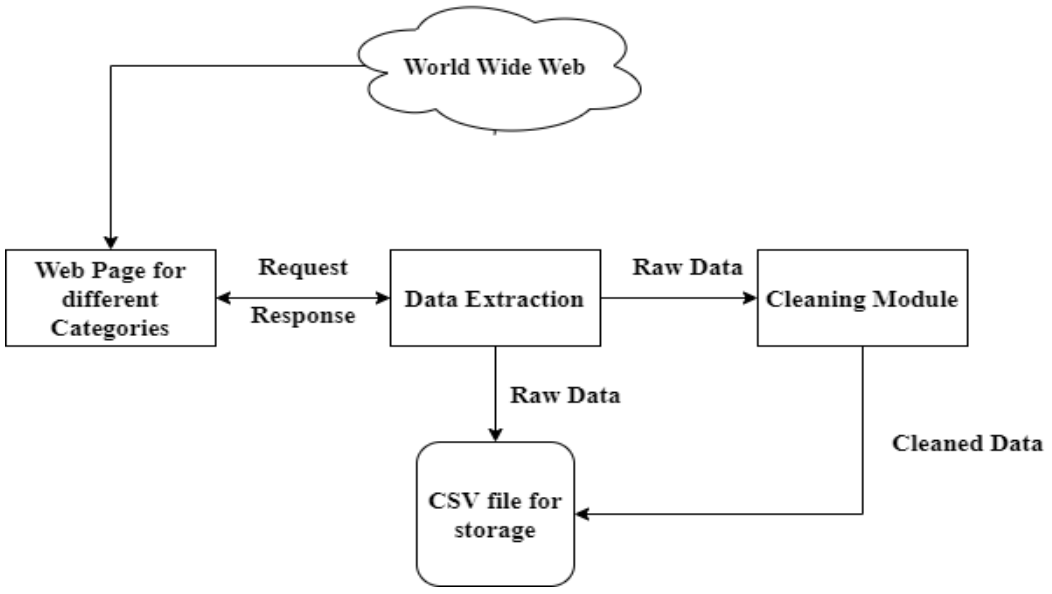

Figure 22: Crawler and Data Extraction Architecture

6.1.2 Data Cleaning

As the data is collected and stored, it needed to be analyzed, but, it contained whitespaces, spelling errors, non-standard names and duplicates, which needed to be cleaned. This module was used for removing noisy data and convert it in a standard format which could be used further for the application. In some cases, there were missing values for many attributes, thus, imputation was required. The missing values for the college dataset were filled using linear regression. In case of automobiles and fruits, it was filled using null data as the attributes were missing in reality. The major role of this module was introducing a standard into the data so that it becomes application specific. This module helps in overcoming the drawbacks of the traditional storage systems of search engines which are not application specific.

6.1.3 Data Analysis

Data analysis refers to the part where the components of data and their behavior is understood. Plotting of various graphs was helpful for the understanding about the form of data and its attributes. Since, the data was vast and multi-attributed, graphs played an important role. All three categories possess different natures and behavior. The relationship between datasets was identified through this module.

The results based on the analysis have been shown below:

1. Defining the number of Clusters for KNN based on the elbow method. (For Fruit Dataset)

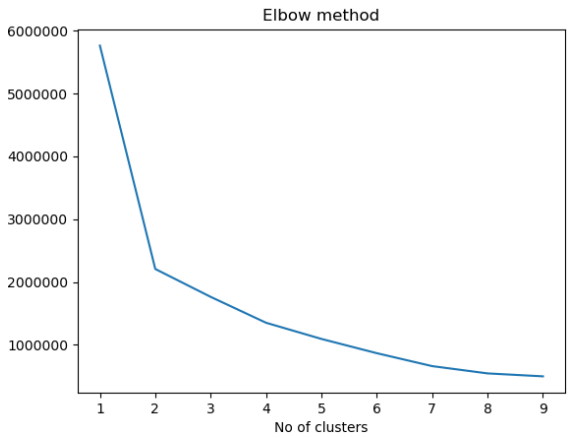

Figure 23: Elbow Method for Fruit Data

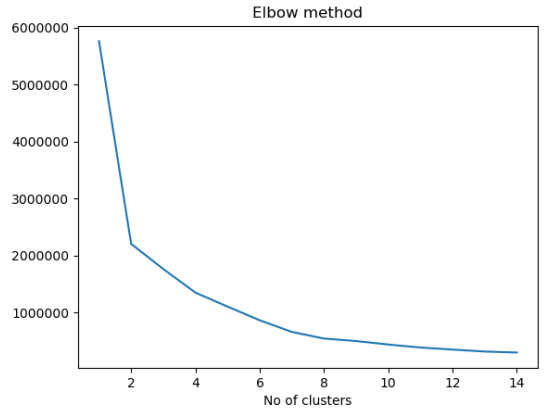

Figure 24: Elbow Method for Fruit Data

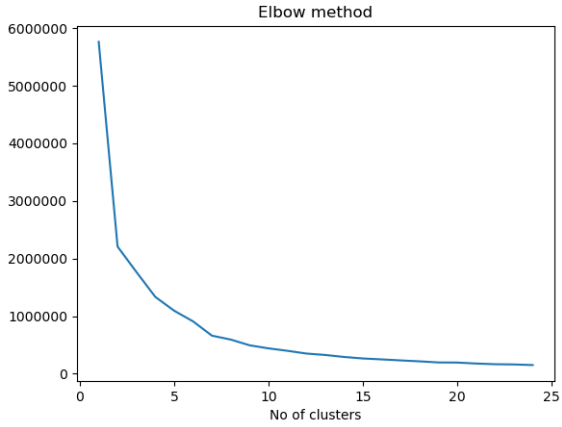

Figure 25: Elbow Method for Fruit Data

So based on this method, the most suitable value to divide the dataset into clusters is 5. This serves as the elbow point in the dataset.

2. Outlier Analysis

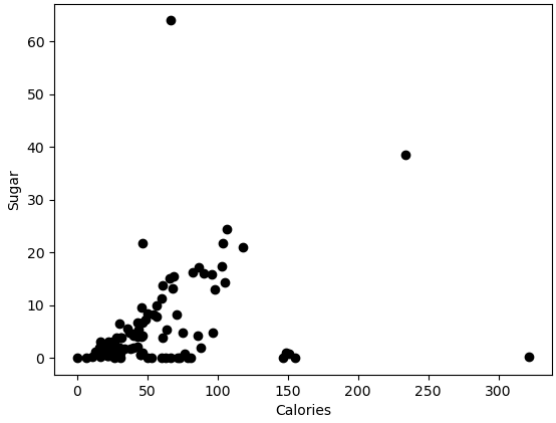

Figure 26: Outlier Analysis for Fruit Data

This analysis helped in determining if a particular group of data is acting in a weird manner. Through this analysis, some of the outliers identified were treated with removal of values as the data was inconsistent with the rest of the attributes.

6.1.4 Machine Learning Module

This module is mainly for the training purpose where data from the database is fed to the model. As per the application, Nearest Neighbor model, an application of the K-Nearest Neighbor technique was used for training. The principle behind nearest neighbor methods is to find a predefined number of training samples closest in distance to the new point, and predict the label from these. The number of samples can be a user-defined constant (k-nearest neighbor learning), or vary based on the local density of points (radius-based neighbor learning).

6.1.5 Database

The database is required for security as well as integrity of the project. The machine learning module requires input data which is fed from the database. All the results obtained from the machine learning module extract final data from the database. It provides a reliable groundwork for the machine learning to train and store important aspects of the data. The database consists of three different master tables of all categories which in turn are connected to their respective attributes.

6.1.6 User Interface

The user interface is the only module through which the users get connected to the system. It is an HTML and CSS based user interface using javascript. It is integrated into the system through the Django server.

6.2 Tools and Technologies Used

Tools and technologies used for Web Development:

1. Visual Studio Code
2. Django
3. Python 3.5

Tools and technologies used for Machine Learning:

1. Visual Studio Code
2. Python 3.5
3. Scikit-learn
4. Pandas
5. Nearest Neighbors Library
6. Matplotlib

6.3 Algorithm Details

6.3.1 Nearest Neighbors – Machine Learning

The principle behind nearest neighbor methods is to find a predefined number of training samples closest in distance to the new point, and predict the label from these. The number of samples can be a user-defined constant (k-nearest neighbor learning), or vary based on the local density of points (radius-based neighbor learning). The distance can, in general, be any metric measure: standard Euclidean distance is the most common choice. Neighbors-based methods are known as non-generalizing machine learning methods, since they simply "remember" all of its training data (possibly transformed into a fast indexing structure such as a Ball Tree or KD Tree).Despite its simplicity, nearest neighbors has been successful in a large number of classification and regression problems, including handwritten digits and satellite image scenes. Being a non-parametric method, it is often successful in classification situations where the decision boundary is very irregular.

In Python, **sklearn.neighbors** provides functionality for unsupervised and supervised neighbors-based learning methods. Unsupervised nearest neighbors is the foundation of many other learning methods, notably manifold learning and spectral clustering. Supervised neighbors-based learning comes in two flavors: classification for data with discrete labels, and regression for data with continuous labels.

We have used this model to train our unsupervised data to find out relevant alternatives to user input based on all the attributes present in the database pertaining to the category. This algorithm provides the number of nearest neighbors as specified. In our application, we have

extracted 5 neighbors. The nearest 5 matches as predicted by the model are returned to the user in the form of an index which needs to be traversed from the database to get output and relevant results.

6.3.2 Beautiful Soup – bs4 – Data Extraction

Beautiful Soup is a Python library for pulling data out of HTML and XML files. It works with your favorite parser to provide idiomatic ways of navigating, searching, and modifying the parse tree. It commonly saves programmers hours or days of work. It is used for easy extraction of data from the web pages in the form of text. This library was modified as per the requirements of the project. We used different attributes of this library in order to extract data which was stored on the web pages.

6.3.3 Database Entry

To store the data from csv file into the database, the attributes having one to many relationships or many to one relationships need to be stored using an algorithm which is customized by us.

6.3.4 Linear Regression

The college dataset consisted of many tuples having non-standard and missing values for attributes such as Rating, Campus Size, Faculty members and Courses. Thus, to overcome this issue, we used linear regression to tackle the noisy data. All the plots and results extracted were used in filling up values and thereby increasing relevancy.

CHAPTER 7: SOFTWARE TESTING

7.1 Types of Testing

7.1.1 Unit Testing

The entire project consists of many small modules where each one needs to be checked and tested if working as per design and requirement. Thus, unit testing was required. We tested each piece of code individually so that in the future, we could find fewer bugs.

7.1.2 Integration Testing

Every module was tested as and when integration was performed. After integration of each module into the main project, the altogether performance should meet expectations, hence this testing was carried out. We identified issues such as data type inconsistency while passing it through different modules was resolved after the testing. Major bug fixes took place during this phase of testing.

7.1.3 Compatibility Testing

As our system is a Web Application, browser compatibility testing was necessary. We as developers ensured that this web application runs in every browser regardless of the operating system including various versions of the browser. A rigorous testing of the application was performed in all the browsers on Windows, Linux and MacOS.

7.1.4 Back–end Testing

To check if the given input from GUI was appropriately stored in the database, this testing was carried out. Table structure, schema, stored procedure, data structure were the parameters considered while carrying out this testing. Few queries were given to authenticate the entire environment.

7.1.5 System Testing

Entire system was tested as per the requirements thoroughly. Overall requirement specifications were taken into consideration while performing this testing.

7.1.6 Alpha Testing

The main aim of this testing was identifying all the defects before releasing it to the end users. The development team of this project was responsible to do this testing. All bugs including font issues in the User Interface along with relevancy of the output displayed were checked.

7.1.7 Performance Testing

The overall performance including stress and load on the server using different platforms was done. Many requests were sent to the server together to check the amount of load it is capable of handling.

7.2 Test Cases and Test Results

Test ID	T001
Test Scenario	To check for alternatives of Colleges
Test Procedure	1. Go to Web Application 2. Select Category as Colleges 3. Enter the college name(may be selected from suggestions) 4. Submit Query
Test Data	College of Engineering Pune
Expected Result	
Actual Result	
Status	Pass

Table 9: Test Case Table 1

Test ID	T002
Test Scenario	To check for alternatives of Fruits and Vegetables
Test Procedure	1. Go to Web Application 2. Select Category Fruits and Vegetables 3. Enter the fruit or vegetable name(may be selected from suggestions) 4. Submit Query
Test Data	Apple
Expected Result	
Actual Result	
Status	Pass

Table 10: Test Case Table 2

Test ID	T003
Test Scenario	To check for alternatives of Automobiles
Test Procedure	1. Go to Web Application 2. Select Category Automobile 3. Enter the name of vehicle(may be selected from suggestions) 4. Submit Query
Test Data	Audi A3
Expected Result	
Actual Result	
Status	Pass

Table 11: Test Case Table 3

CHAPTER 8: RESULTS

8.1 Outcomes

The alternatives of queries entered by user are displayed in the form of a list. The sequence of display depends on the relevancy of the alternative, the closest one being on top.

8.2 Screenshots

1. Home Page

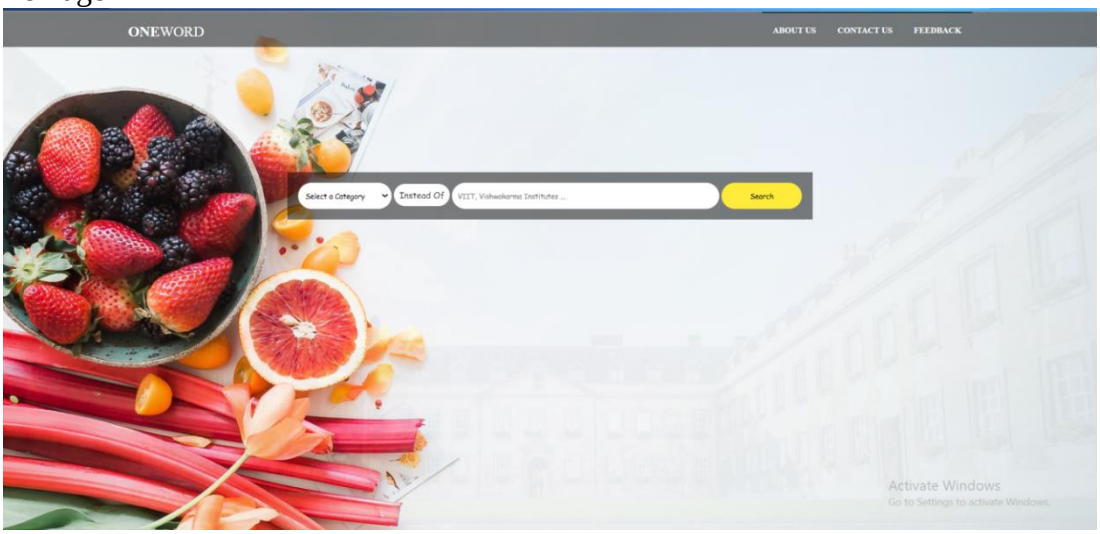

Figure 27: Home Page

2. Selecting a Category

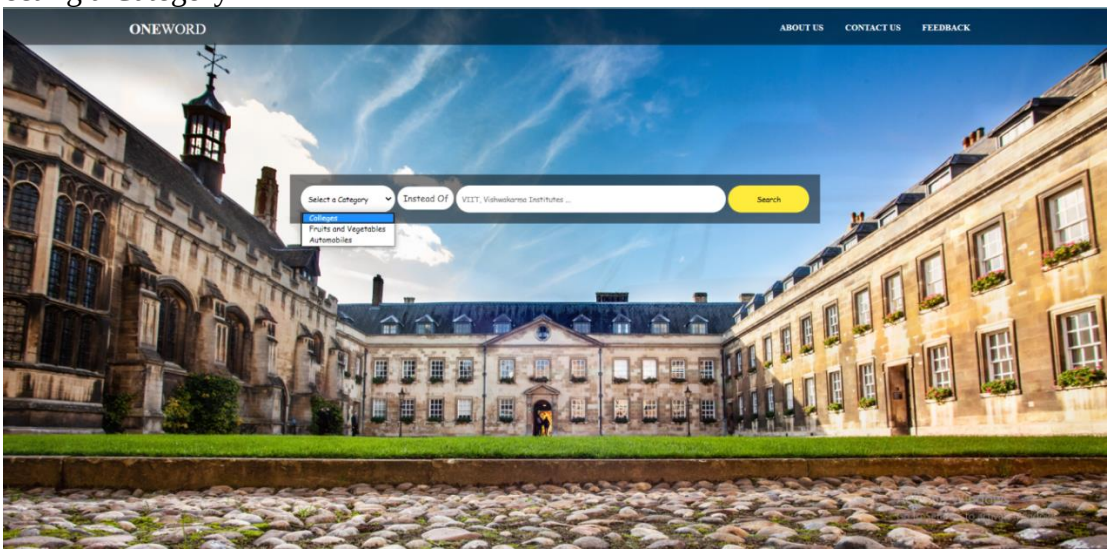

Figure 28: Selecting Category

3. Entering a Query and getting Suggestions

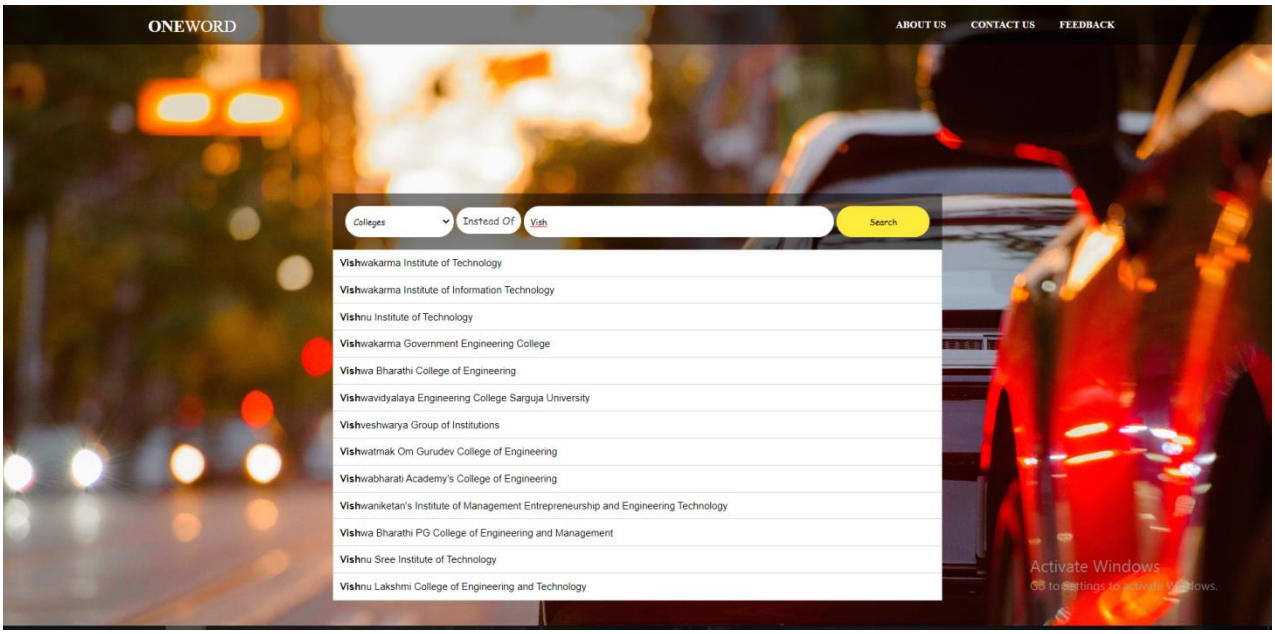

Figure 29: Query with Suggestions

4. Result Page with Alternatives

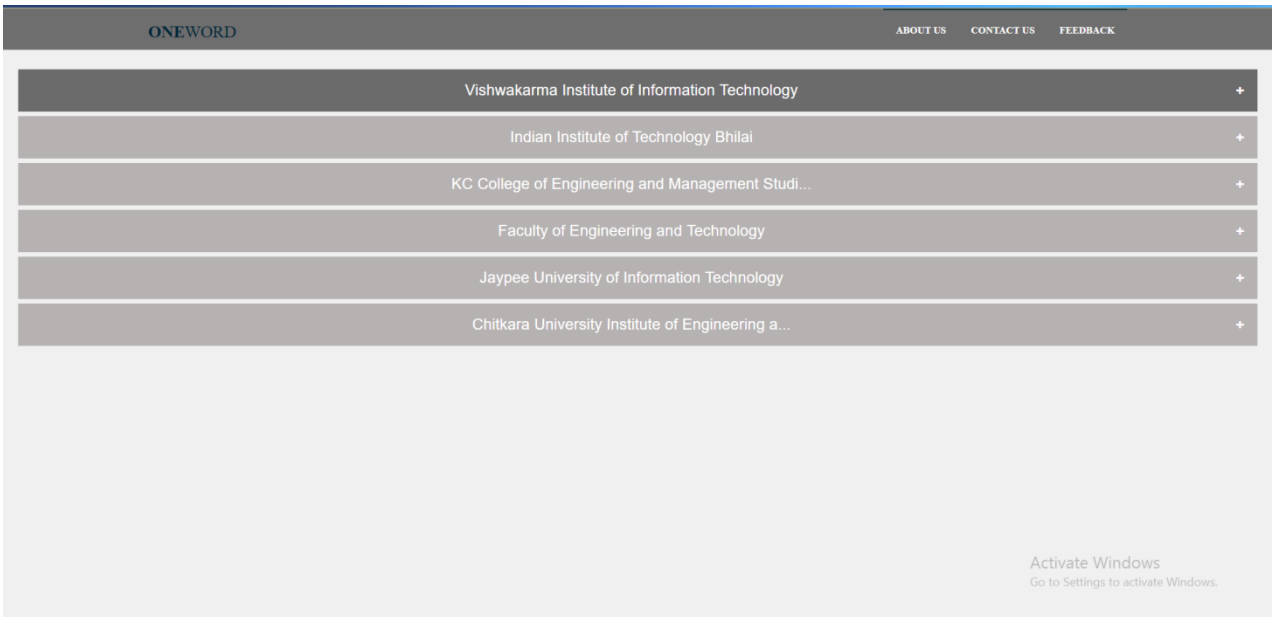

Figure 30: Result Page

5. Contact Us Page

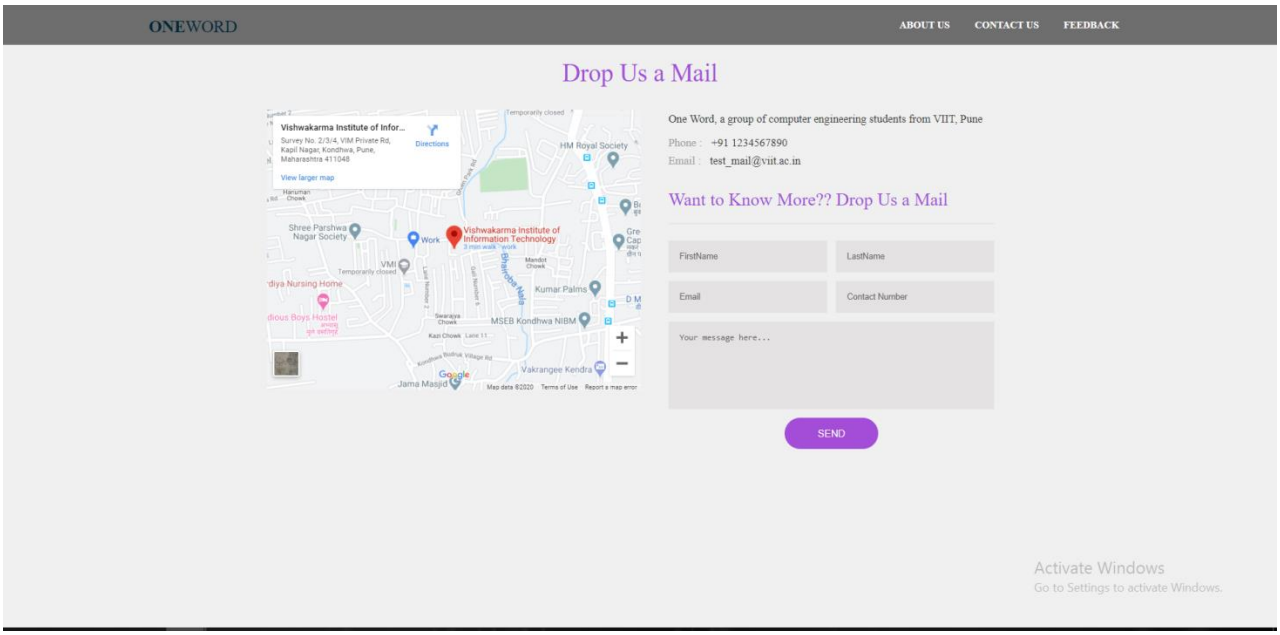

Figure 31: Contact Us Page

6. Feedback Page

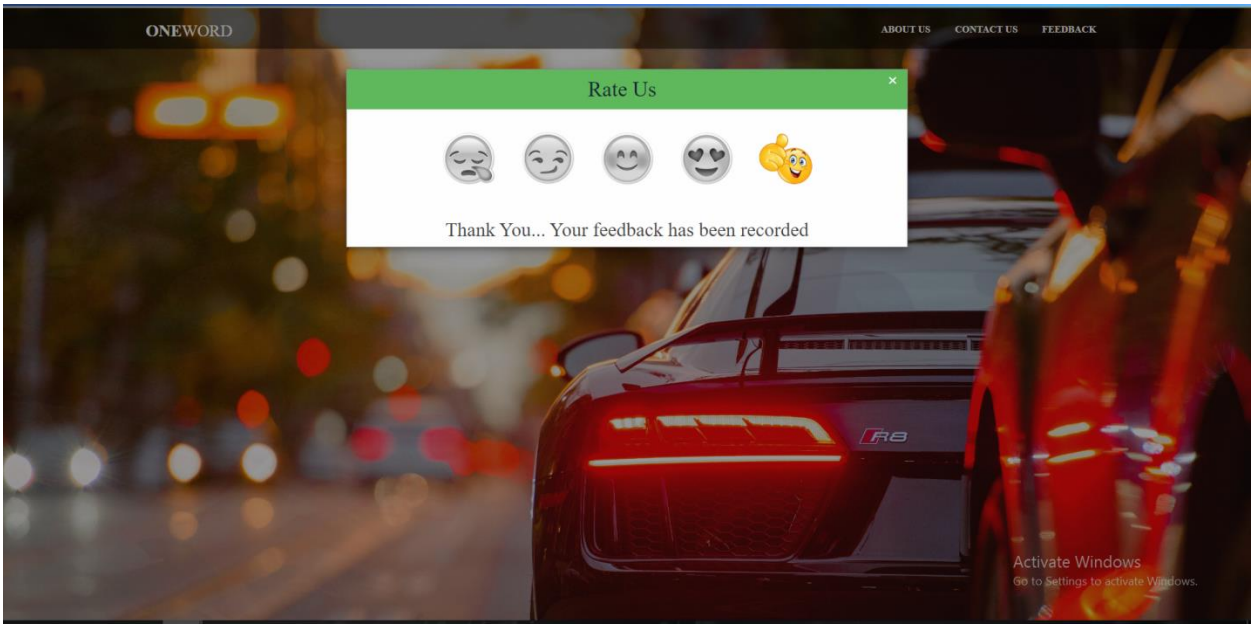

Figure 32: Feedback Page

CHAPTER 9: CONCLUSIONS

9.1 Conclusion

In this era of rapid growth of technology, it has become important to find alternatives for many things in everyday life. Most of the people tend to changeover to a better alternative if there is scope for enhanced results. The proposed system is an innovative platform that suggests trending alternatives to the most frequently required entities. “Instead Of” provides an insight to the user about recent lookups which are relevant to their category.

The user query is broken down into context specific information based on deep semantic understanding. Consequently, this relevant information is stored into the data structure according to the categories selected or specified by the user. The search system, thus, ensures that relevant information is extracted from the data structure. Different modules as specified in the report contribute towards full-fledged architecture of the project.

This report provides a comprehensive knowledge about the requirements, database and software models used. Major components of the system include the language processing engine based on deep semantic understanding of the user query, the search engine to fetch results from the proposed data structure or the machine learning module in turn accepts input from the enhancement module. All the relevant results are provided to the user in the form of a concrete answer which it can rely on.

A user interface in the form of a website will be catered to the end users, where they will be able to select the category, enter their query and view the relevant results as provided by the system. Additionally, useful links will also be provided along with the concrete answer so that the user’s requirements are fulfilled perfectly.

Thus, the proposed system indulges in finding alternatives for the query provided by the user by applying a customized approach to information retrieval and web crawling supported by enhancements using neural networks and understanding natural language.

9.2 Future Work

The feature of feedback including a neural network which will enhance the quality of results shared by the user is to be added. Also, new categories according to user's demand through feedback and mailing feature will be considered. In addition to this, looking at the large amount of data in the real world, it is possible that the query input by the user may not be clear for the system, thus, a need for query processing will be harnessed. More data from different websites is also to be collected. Dynamic ranking or results displayed to the user.

Continuous updates in the websites from which data has been gathered is inevitable, thus, a system which keeps track of the changes to the website so that the database can be changed in time. Along with that, a module tracking the login information of users to get personalized results according to age, trends or choices may be displayed pertaining to the category specified.

9.3 Applications

The system finds its applications in diverse domains of the society such as:

1. Students – for viewing alternatives to colleges in order to make career decisions. The can use the system for viewing similar fruits and vegetables or even automobiles for an investment.

2. Researchers – they may find the system useful to get dietary supplements in medicines in the form of fruits or vegetables in daily life. They can also use the results to formulate new practices in the medical industry.

3. Doctors – the system is helpful in terms of dietary measures where they continuously need quick results and recovery.

4. Investors – it is helpful in terms of investment in college trusts or even automobiles where they can try different options available.

5. Counselors – students often take the help of counselors to build their future career path. Counselors may use the system to suggest a list of colleges for the child based on their aptitude and conveniences.

6. Curious Minds – there are many curious people in this world that may belong to any field of profession who keep surfing the internet for options. This system provides the best solution to their problems on a single platform. They are also the major stakeholders in the development of this project and enhancements through feedbacks.

REFERENCES

[1] Nakamura S. et al. (2007) “Trustworthiness Analysis of Web Search Results”. In: Kovács L., Fuhr N., Meghini C. (eds) Research and Advanced Technology for Digital Libraries. ECDL 2007. Lecture Notes in Computer Science, Vol. 4675. Springer, Berlin, Heidelberg.

[2] Kancherla, Vinay, "A Smart Web Crawler for a Concept Based Semantic Search Engine" (2014). Master's Projects. 380.

[3]B. Adams and M. Raubal. Conceptual Space Markup Language (CSML): Towards the Cognitive Semantic Web. In International Conference on Semantic Computing, pages 253–260, Los Alamitos, CA, USA, 2009. IEEE Computer Society.

[4]Ria Mittal, Laxmi Bewoor, H. T. P. S. S. M. (2020). Search Engines: A Systematic Review. International Journal of Advanced Science and Technology, 29(4s), 1134 - 1141. Retrieved from http://sersc.org/journals/index.php/IJAST/article/view/6665

Links:

[1] https://www.wordstream.com/articles/internet-search-engines-history

[2] https://www.searchenginejournal.com/semantic-search-engines/9832/

[3]Google Search: How Search Works https://www.google.com/search/howsearchworks/

[4]Nearest Neighbors Documentation: https://scikit-learn.org/stable/modules/neighbors.html

[5]Beautiful Soup Documentation: https://www.crummy.com/software/BeautifulSoup/bs4/doc/

[6]Software Testing: https://www.softwaretestinghelp.com/types-of-software-testing/