iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 5 **Issue:** XI **Month of publication:** November 2017

DOI: http://doi.org/10.22214/ijraset.2017.11118

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor:6.887
Volume 5 Issue XI November 2017- Available at www.ijraset.com

Managing Big Data and Semantics (Wiki Pages) In NoSQL Format by Efficient Algorithms Implementing In Python

Aditi Choudhary[1], Kaveri Roy[1], Tadeesha Roy[1], Sushmitha Sekuboyina[1], Mrs. J. Dorathi Jayaseeli[2]
[1]*Bachelor of Technology, Department of Computer Science and Engineering, Third year, SRM University, Chennai, India*
[2]*Guide: Assistant Professor (S.G), Department of Computer Science and Engineering, SRM University, Chennai, India*

Abstract: Wikipedia is, in essence a giant encyclopedia- but with a twist. Wikipedia's content is created solely by the site's users, resulting in the world's largest online collaboration. Wikipedia articles are written, edited, and elaborated on by people of all types, from students, to subject-matter experts and professional researchers, to interested amateurs. It's true group collaboration. This paper presents algorithms that characterize the features of over 38 million web pages of Wikipedia by using Algorithms that are implemented in Python and developing a smart answer retrieving system out of that. This data-driven model involves demand-driven aggregation of information sources, mining and analysis. It also involves the usage of Mongo Db for the storage of the data.

Keywords: Smart answer, wiki matrix, eliminators, info box.

I. INTRODUCTION

Wikipedia is one of the open source knowledge repository that is used by people all over the web. What a wiki is – a collection of webpages where any user can contribute to modify the content. The first wiki was WikiWikiWeb, a website founded in 1995 to facilitate the exchange of ideas between computer programmers. Wikis enable users to not only write new articles, but also to comment on and edit existing articles [2]. Our capability for data generation has never been so powerful and enormous ever since the invention of the information technology in the early 19thcentury [1]. Every day, 2.5 quintillion bytes of data are created [4] and with over 7 billion pages already present and the knowledge of the mankind expanding at even sky rocketing pace, managing this amount of big data for use of people can be made easy. The era of Big Data has arrived. In just a single minute on the web 216,000 photos are shared on Instagram, there are 1.8 million likes on Facebook and three days' worth of video is uploaded to YouTube. Google performs 2 million searches each minute and 72 hours' worth of video is uploaded to YouTube within the space of 60 seconds. Along with the above examples, hundreds of websites are created within a minute online, at the same time 204 million emails are sent and millions of tweets are triggered [5]. Therefore, handling of Big Data has become the major breakthrough where the data has become the New Oil. Algorithms have been developed which will be seen here in this paper and how the data is managed in NoSQL(No Structured Query Language) in the MongoDb platform. MongoDb is a cross-platform document oriented database designed with both scalability and developer agility in mind. Instead of storing data in tables and rows as in relational database, MongoDb stores of JSON (JavaScript Object Notation) like documents with dynamic schemasmaking the integration of data in certain types of applications easier and faster. Classified as a NoSQL(No Structured Query Language) database, MongoDb eschews the traditional table-based relational database structure in favor MongoDb is more flexible to modification, efficient utilization of RAM(Random Access Memory), no complex joins, deep query ability and easy to setup [3]. The remainder of the paper is structured as follows: In second section we will see how the Wikipedia pages are retrieved in JSON(JavaScript Object Notation) format and how the keywords are extracted using Python scripts and matched with its synonyms of a Wiki page in order to exhibit smart answer retrieval. In the later part of the paper, some key initiatives, the scope and limitations ofthe project are discussed. And in the final section we conclude the paper.

II. WORKING

A. Retrieval of Wikipedia Pages (Module 1)

In order to extract the Wikipedia pages one needs to first look upon its structure. Each Wikipedia webpage is available in the JSON(JavaScript Object Notation) format. JSON(JavaScript Object Notation) is an open standard format that uses human-readable text to transmit data objects consisting of attribute–value pairs. It is the most common data format used for asynchronous

browser/server communication, largely replacing XML(Extensible Markup Language) which is used by AJAX (Asynchronous JavaScript and XML). JSON(JavaScript Object Notation) is a language-independent data format. It derived from JavaScript, but now code to generate and parse JSON(JavaScript Object Notation)format data is available in many programming languages. The official Internet media type for JSON(JavaScript Object Notation) is application/json. The JSON(JavaScript Object Notation) filename extension is .json. The JSON(JavaScript Object Notation) format was originally specified by Douglas Crockford. JSON(JavaScript Object Notation) is one way of sending data between various components of an application using data which can be serialized, or turned into a series of key-value pairs. Some applications use this method to send data from the application to the browser. All the webpages in the Wikipedia can be downloaded in JSON (JavaScript Object Notation) format which is a light weight data-interchange format.

Index of /other/wikidata/

../		
20150706.json.gz	06-Jul-2015 12:59	63
20150713.json.gz	13-Jul-2015 12:55	5098752607
20150720.json.gz	20-Jul-2015 13:23	5142415763
20150727.json.gz	27-Jul-2015 13:16	5173631898
20150805.json.gz	06-Aug-2015 08:07	5139576592
20150810.json.gz	10-Aug-2015 13:29	5168574560
20150817.json.gz	17-Aug-2015 12:57	5197349243
20150824.json.gz	24-Aug-2015 13:32	5237995846
20150831.json.gz	31-Aug-2015 13:33	5278481582
20150907.json.gz	07-Sep-2015 13:56	5313794062
20150914.json.gz	14-Sep-2015 13:57	5367334340

Fig. 1 Represents Wikipedia pages in JSON(JavaScript Object Notation) format [7].It contains terabytes of data and after downloading it from the website we can do further processing on it.

Generally, it is easy for humans to read and write while for machines it is easy to parse and generate results. It is well known that machines cannot understand human language and hence all the programs and commands are converted into binary language so that the machine can understand[6]. Similarly, the Wikipedia page contains page id, title, revision, ns and various other details about the web page. In order to find the required page from the vast network of webpages, page id and title are used for easy search. Since the content of Wikipedia is solely created by its users, there is a possibility for duplication of pages. Therefore only the user verified pages are displayed using their unique page id.

The following diagram shows the JSON(JavaScript Object Notation) format of a random page on the Wikipedia.


```
{
    "batchcomplete": "",
    "query": {
        "pages": {
            "15580374": {
                "pageid": 15580374,
                "ns": 0,
                "title": "Main Page",
                "revisions": [
                    {
                        "contentformat": "text/x-wiki",
                        "contentmodel": "wikitext",
                        "*": "<!--                BANNER ACROSS TOP OF PAGE          -->\
topbanner\" style=\"width:100%; background:#f9f9f9; margin:1.2em 0 6px 0; borde
#ddd;\"\n| style=\"width:61%; color:#000;\" |\n<!--                \"WELCOME TO WIKIPED
COUNT        -->\n{| style=\"width:280px; border:none; background:none;\"\n|
style=\"width:280px; text-align:center; white-space:nowrap; color:#000;\" |\n<d
size:162%; border:none; margin:0; padding:.1em; color:#000;\">Welcome to [[Wiki
div>\n<div style=\"top:+0.2em; font-size:95%;\">the [[free content|free]] [[enc
[[Wikipedia:Introduction|anyone can edit]].</div>\n<div id=\"articlecount\" sty
size:85%;\">[[Special:Statistics|{{NUMBEROFARTICLES}}]] articles in [[English l
</div>\n|}\n<!--              PORTAL LIST ON RIGHT-HAND SIDE              -->\n| style=\"wi
size:95%;\" |\n* [[Portal:Arts|Arts]]\n* [[Portal:Biography|Biography]]\n*
[[Portal:Geography|Geography]]\n| style=\"width:13%; font-size:95%;\" |\n*
[[Portal:History|History]]\n* [[Portal:Mathematics|Mathematics]]\n*
[[Portal:Science|Science]]\n| style=\"width:13%; font-size:95%;\" |\n*
[[Portal:Society|Society]]\n* [[Portal:Technology|Technology]]\n* '''[[Portal:C
Portals|All portals]]'''\n|}\n<!--                MAIN PAGE BANNER                -->\n{{#if:{{
banner}}|\n<table id=\"mp-banner\" style=\"width: 100%; margin:4px 0 0 0; backg
border-spacing: 0px;\">\n<tr><td class=\"MainPageBG\" style=\"padding:2px 8px;
color:#fffaf5; border:1px solid #f2e0ce; color:#000; font-size:100%;\">{{Main P
td></tr>\n</table>\n}}\n<!--              TODAY'S FEATURED CONTENT              -->\n{| id=
style=\"width: 100%; margin:4px 0 0 0; background:none; border-spacing: 0px;\"\
TODAY'S FEATURED ARTICLE; DID YOU KNOW        -->\n| class=\"MainPageBG\" style
border:1px solid #cef2e0; background:#f5fffa; vertical-align:top; color:#000;\"
left\" style=\"width:100%; vertical-align:top; background:#f5fffa;\"\n| style=\
<h2 id=\"mp-tfa-h2\" style=\"margin:3px; background:#cef2e0; font-family:inheri
size:120%; font-weight:bold; border:1px solid #a3bfb1; text-align:left; color:#
padding:0.2em 0.4em;\">{{#ifexpr:{{formatnum:{{PAGESIZE:Wikipedia:Today's featu
{{#time:F j, Y}}}}|R}}>150|From today's featured article|Featured article <span
size:85%; font-weight:normal;\">(Check back later for today's.)</span>}}</h2>\n
style=\"color:#000;\" |  <div id=\"mp-tfa\" style=\"padding:2px 5px\">{{#ifexpr:
{{PAGESIZE:Wikipedia:Today's featured article/{{#time:F j, Y}}}}|R}}>150|{{Wiki
featured article/{{#time:F j, Y}}}}|{{Wikipedia:Today's featured article/{{#tim
day}}}}}}</div>\n|-\n| style=\"padding:2px;\" | <h2 id=\"mp-dyk-h2\" style=\"ma
background:#cef2e0; font-family:inherit; font-size:120%; font-weight:bold; bord
#a3bfb1; text-align:left; color:#000; padding:0.2em 0.4em;\">Did you know...</h
style=\"color:#000; padding:2px 5px 5px;\" | <div id=\"mp-dyk\">{{Did you know}
style=\"border:1px solid transparent;\" |\n<!--              IN THE NEWS; ON THIS DAY
class=\"MainPageBG\" style=\"width:45%; border:1px solid #cedff2; background:#f
align:top;\"|\n{| id=\"mp-right\" style=\"width:100%; vertical-align:top;
background:#f5faff;\"\n| style=\"padding:2px;\" | <h2 id=\"mp-itn-h2\" style=\"
background:#cedff2; font-family:inherit; font-size:120%; font-weight:bold; bord
#a3b0bf; text-align:left; color:#000; padding:0.2em 0.4em;\">In the news</h2>\n
style=\"color:#000; padding:2px 5px;\" | <div id=\"mp-itn\">{{In the news}}</di
style=\"padding:2px;\" | <h2 id=\"mp-otd-h2\" style=\"margin:3px; background:#c
family:inherit; font-size:120%; font-weight:bold; border:1px solid #a3b0bf; tex
color:#000; padding:0.2em 0.4em;\">On this day...</h2>\n|-\n| style=\"color:#00
```


Fig. 2 Schematic representation of the JSON(JavaScript Object Notation)format of the Wikipedia pages.Text mining techniques are applied on JSON pages to extract the relevant text.

These pages are downloaded in normal format and are saved for further processing to be done on it. They are a huge amount of data and can be in terabytes.

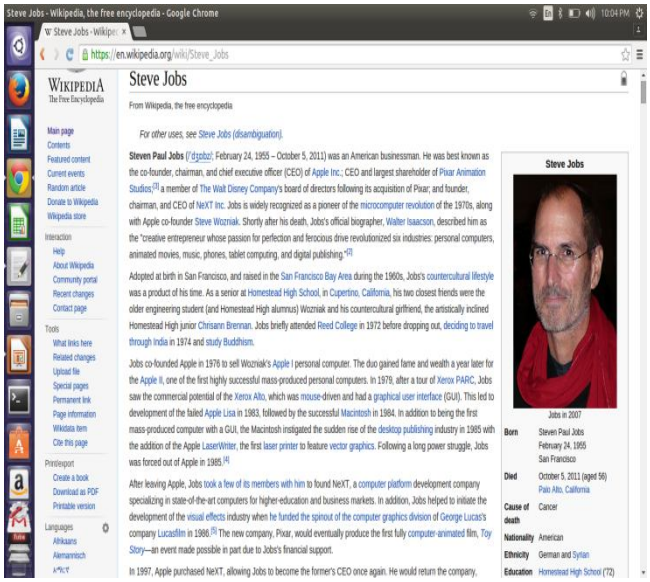


Fig. 3Thestructure of the Wikipedia page from where infobox is extracted out.

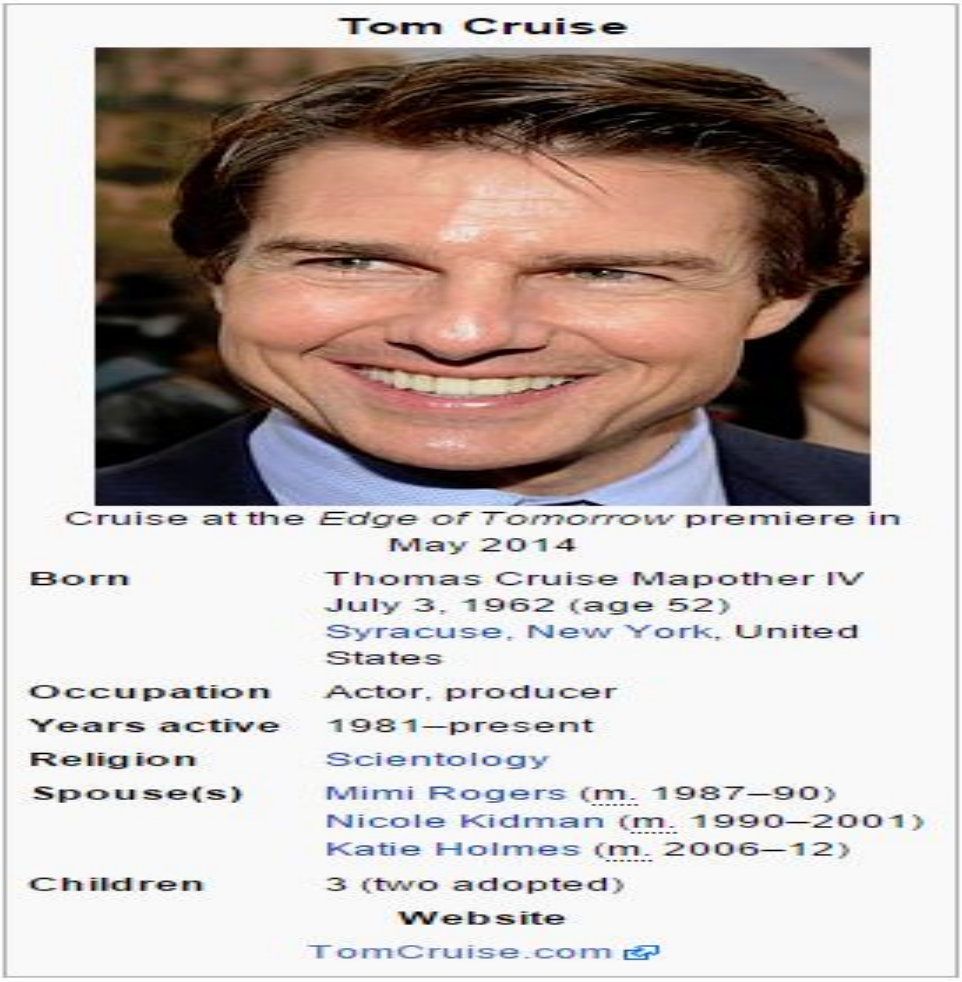


Fig.4The structure of infobox contained in the Wikipedia that will help in the data extraction.

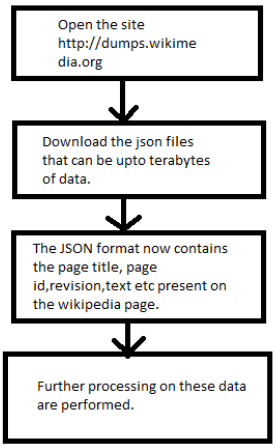


Fig.5 Flowchart representation of downloading JSON(JavaScript Object Notation) pages from the website. After downloading the JSON(JavaScript Object Notation) files further processing is performed on it.

B. Saving the data in MongoDb(Module 2)

MongoDb is one of several database types to arise in the mid-2000s under the NoSQL(No Structured Query Language) banner. Instead of using tables and rows as in relational databases, MongoDb is built on architecture of collections and documents. Documents comprise sets of key-value pairs and are the basic unit of data in MongoDb. Collections contain sets of documents and function as the equivalent of relational database tables.

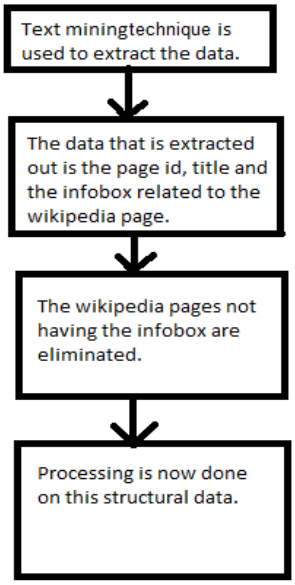


Fig.6Flowchart representation of applying text mining techniques on the JSON (JavaScript Object Notation) pages and saving it NoSQL format in MongoDb.

Like other NoSQL(No Structured Query Language) databases, MongoDb supports dynamic schema design, allowing the documents in a collection to have different fields and structures. The database uses a document storage and data interchange format called BSON(Binary JavaScript Object Notation), which provides a binary representation of JSON(JavaScript Object Notation) like documents. MongoDb is more flexible to modification, efficient utilization of RAM(Random Access Memory), no complex joins, and deep query ability and easy to setup [9]. When the JSON(JavaScript Object Notation) format of the webpage is extracted by using text mining technique, the page id, title of the page and also the info box is separated out and is saved in the MongoDb. If a particular page does not have info box then they are eliminated simultaneously. So now the MongoDb contains the page id, title and the infobox which is stored in form of classes. Data is now structural and mapping can be easily performed.

C. Storing of keywords in PHP My Admin (Module 3)

In order to design a user-friendly searching assistant, the need to give the user the freedom to use any form of the actual keyword stored in the database is of supreme importance. They must be allowed to use any synonym of the actual word stored in the database. Thus careful mapping of the synonyms to the correct answer comes into play. This is done by storing the keywords and their respective synonyms in the Php My Admin database using wiki matrix.

Wiki matrix is designed for business teams as a low-cost SharePoint Alternative; Central Desktop wikis are easily editable for corporate users. Style templates give you the flexibility you need to create your own professional branding/colors/css(cascading style sheet) look and feel. Put everything in one place. Work easily & securely with remote consultants or clients, anytime, from anywhere. Central Desktop is packed with powerful wiki features: embedded applications such as discussion threads, blogs, twitter-like micro blogging, file managers, task lists, milestone management, light calendaring, shared folders, online databases, workflow and permission management into a easy to use WYSIWYG(What You See Is What You Get) environment supporting gated (private - invitation only) or public (ungated) access. Enterprise LDAP(Lightweight Directly Access Protocol) single sign-on & API(Application Programming Interface)access available.The entire application is built around a WYSIWYG(What You See Is What You Get) edit environment. No wiki markup required [10].

The system now works as under: The user input is fetched from the browser and eliminators are removed instantly before feeding it to the database. The words are then matched with any of the words or their synonyms in the PhpMyAdmin database. The resulting matching keyword of the synonym is then fed to MongoDb in order to pick out the correct answer and thus display it to the user.

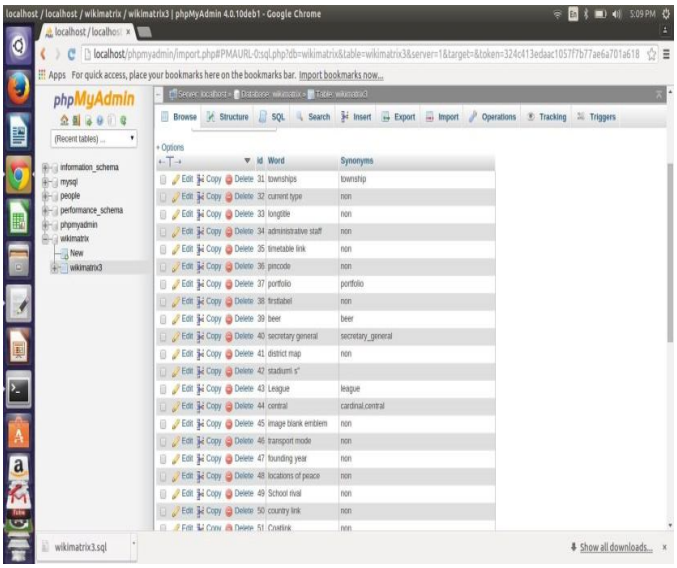


Fig.7The Wiki matrix database in Php MyAdmin downloaded from the website [9] that contains the keywords present in the infobox in Wikipedia.

D. Separation of key attributes (Module 4)

For an intelligent answer, speed is of utmost importance. For a speedy answer, the system must find the required data quickly, yet effectively. To solve it, we must feed the most important data to the querying system. Thus, filtering of questions is necessary. Filtering out of questions mean nothing but separation of keywords from the entire question which is normally a full sentence consisting of prepositions, pronouns, conjunctions, verbs, adverbs, adjectives or articles, otherwise known as “eliminators” in this context. Only the keywords are required for effective mapping of query to the required data. The eliminators must be removed instantly.

The algorithm thus designed for this solution can be explained as follows

An array is created which consists of all the possible prepositions, pronouns, conjunctions, verbs, adverbs, adjectives or articles (eliminators) which might be used by the user. The question asked by the user on the browser is taken in as an input. The input thus obtained is compared with the array of the set of eliminators. All the matches found from the user input to the array set of eliminators are thus eliminated.

The resulting filtered unmatched input is thus used as a set of keywords used to map with MongoDb and PhpMyAdmin, to find out and provide the answer to the user.If the synonym of a particular word is written in a query then the main word related to it is found in wiki matrix and synonym is replaced by that word and then the search results are obtained.

E. Wiki Matrix (Module 5)

WikiMatrix finds the Wikis that match the words left from the query with the synonyms and identifies the root word. It allows us to build a custom search query to find a Wiki matching exactly to our needs [10].

Wiki software and Wiki hosting services are listed at Wikimatrix. Both types differ in certain features. By default only general features are available for search.

It also compares the Wikis of our choice in a comfortable side-by-side table. It compares the features of many, many different wiki engines. A wiki is more than its list of features.

Why not use an online, hosted wiki? Because we are not always online, particularly when mobile. For personal notes intended only for us, taking the notes with us (either on a USB stick or a mobile device) is just as effective. If we do need to share our wiki with others, the wiki data can be hosted on a shared directory for a more conventional collaborative wiki.

F. Handling the query and Smart answer retrieval (Module 6)

The query asked by the user in the search box is divided into number of keywords. From it the words that are not required are eliminated whichare termed as eliminators. Main keywords are extracted and certain words not in use are eliminated using an array which specifies all the words that are not required. Mapping the keywords with all the pages through their object id in Mongo Dbis a very tedious task as each page contains a large amount of text. The time complexity of this process will be more and a large amount of time is wasted. In order to overcome this problem, matching of the title of the page and the first few lines of text with the keywords which are the attribute of the objected,is performed. If that keyword matches then we look for other keywords in the info box attribute and the result related to the query asked is given out. The info box has different parameters to define an object, so for every object it will differ. This form of data is unstructured, it cannot be saved in SQL(Structured Query Language) database as each page has different attributes and all attributes cannot be taken into account while creating the schema of SQL(Structured Query Language) database.

Whenever a person searches a query , the phrase is broken down into words and stored in an array. The words stored in the array are checked with the arraywhere the eliminators are stored. The matched words are eliminated. The next task is to find the synonym of the words left behind in the array. Synonyms can be found out by mapping the left out words with the wikimatrix that contains the root words and its synonyms as attributes of info box. If the synonym of a particular word is written in a query then the main word related to it is found in wiki matrix and the synonym is replaced by that word and then the search results are obtained. In this way we can build a smart retrieval system which may give the accurate results for complex queries.

Time complexity problem can be overcome by utilizing the flags. All the keywords obtained from the query are assigned as flag=1. Now mapping of these keywords takes place with the Wikimatrix. If some of the keywords are present in the Wikimatrix then the main word related to only those words are found and are replaced with them.The flag of those words that are present in the Wikimatrix and are replaced by their main word is set to 0. The keywords with flag=1 are mapped with the MongoDB and if it matches with the page titles of the pages then from the infobox attribute of that page the values of the keywords with flag=0 are extracted out and the value is displayed as an answer to the user. The algorithm is given below-

1) Keywords =a,b,c//these are the keywords extracted from the query asked by the user
2) flag of a,b,c is set to 1.
3) Mapping of a,b,c with Wikimatrix.
4) The main value related to the value a or b or c are found out in Wikimatrix and the previous values are replaced by those.
5) If the Keyword is already a main word then leave it as it is.
6) The flag of replaced values are set to 0.
7) The keywords having flag=1 are checked in the MongoDB database and is mapped with the title.
8) If it matches with the title then from the infobox attribute of that page the value of the keywords with flag=0 are extracted out.
9) The result is displayed.

Inthiswaywecanbuildasmartretrievalsystemwhichmay givetheaccurateresultsforcomplexqueries.

Smart answer is basically used to give accurate and precise results for queries. For example:

If user asks "Who is the CEO(Chief Executive Officer) of Maruti Suzuki?"
The answer should contain the precise result that is "Kenichi Ayukawa".
Wiki Matrix helps to match the left over words from the user query with the synonyms already stored to extract the root word. With the algorithm implemented in python, the root word is mapped with the page title stored in the Mongo DB to fetch the query result to the user usually displayed in the browser. The query result includes the information related to the root keyword that is stored in the infobox from the Wikipedia in JSON(Structured Query Language) format in MongoDb. The result obtained is available in a no compromise format and the user can easily view and download in his/her pc.
Why chooing infobox for smart answer? This is one of the advantages of retrieving smart answers: We have selected info box for this mechanism because it contains accurate results and matching of keywords with its keys are comparatively easier.
The keys of the infobox are present in the JSON(JavaScript Object Notation) format in MongoDb.

III. ADVANTAGES AND LIMITATIONS

A. *Merits*

1) It can handle more complex data.
2) Smart answers are a great tool for content designers to present complex information in a quick and simple way.
3) Indian search engines cannot handle complex queries and don't have a proper algorithm for generating the answers. There are two reasons why most people log on to the internet. One is for communication and the secondly it is to search for information that is appropriate to their question. People usually don't like to visit many sites to know the answer related to the question asked. Hence the smart answers help them.
4) Less time wastage (people do not need to go to various sites to search for their answers).
5) Accuracy is more.
6) Demerits Wikipedia Pages not having infobox is a disadvantage. The slides which do not have infobox cannot be incorporated in this mechanism because matching of keywords is not possible with such a large amount of text.

IV. SCOPE OF IMPROVEMENT AND FUTURE ASPECTS

A. *Scope of improvement*

For later up gradation of our technology, we will have hands on the bLADE Wiki.bLADE Wiki is a free mobile personal wiki to let us take and manage our notes. It runs on Windows desktops and Windows Mobile PDAs(Personal Digital Assistants) and smartphones - and can sync between them all. It can even be run on Windows from a USB(Universal Serial Bus) memory stick. Either way, it allows us to take our information with us - wherever we are!

It is a stand-alone application - that doesn't need or rely on a webserver. Files are stored locally with the wiki application. This means that using the wiki does not rely on network connectivity[10].

B. *Future aspects*

1) Research is going on this field and this might help researchers.
2) This technique can be effectively used in a search engine.
3) Governments and standards bodies at all levels are considering or adopting various foundational elements of the smart answers.
4) MongoDb is has recently been emerged and can prove its effectiveness in such cases. MongoDb is a cross-platform document-oriented database, making the integration of data in certain types of applications easier and faster.

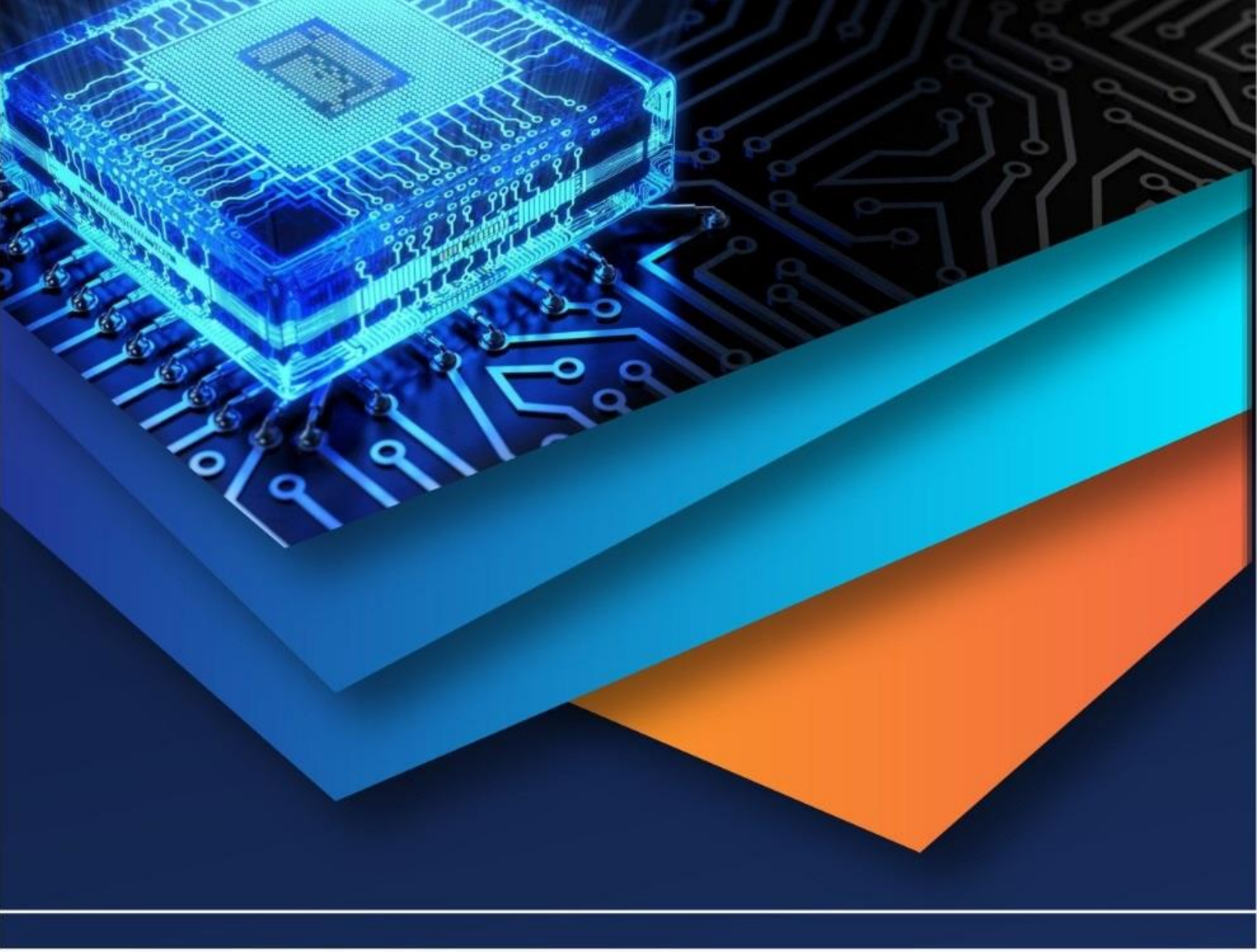
V. CONCLUSION

In a world with ever increasing demand for speed, there is an unending need to design new and better algorithms to satisfy such need. This paper hence is developed mainly to ace up the speed of the current ongoing search engine and to handle much more complex queries than those handled by the already existing algorithms in order to ease the search for the common user end. The algorithm described in this paper is developed in order to bring out *"Intelligent Answers"* to search queries. Also, this algorithm very efficiently handles the huge existing database of the well-known and essential Wikipedia, in a way that the end user can very easily and speedily get the desired answers. Usually it is a tedious job to go through every blue links we get by entering our query in the search box of a search engine in order to get the desired answer. Moreover, we can never be sure if a particular link will contain our answer and we have no chance of knowing that which particular link will have our answer. Mostly, we land up on many undesirable links before we finally get the link which will correctly answer our query. Hence this algorithm very efficiently picks

out the relevant answers and displays it, therefore eliminating the need to go through a lot of unnecessary links. This algorithm has become an essential part in enhancing the present search engines.

REFERENCES

[1] Xindong Wu, Fellow, IEEE, Xingquan Zhu, Senior Member, IEEE, Gong-Qing Wu, and Wei Ding, Senior Member, IEEE "Data Mining with Big Data".
[2] Michael Miller "Cloud Computing Web-Based Application that changes the Way You Work and Collaborate Online".
[3] "What is MongoDb?" http://searchdatamanagement.techtarget.com/definition/MongoDB
[4] VcloudNews April 5, 2015 "How much data is created each day?" http://www.vcloudnews.com/every-day-big-data-statistics-2-5-quintillion-bytes-of-data-created-daily/
[5] Victoria Woollaston, 30th July, 2013 "How many webpages are created each day?" http://www.dailymail.co.uk/sciencetech/article-2381188/Revealed-happens-just-ONE-minute-internet-216-000-photos-posted-278-000-Tweets-1-8m-Facebook-likes.html
[6] "What is a JSON file?" https://en.wikipedia.org/wiki/JSON
[7] "To download the Wikipedia pages in JSON format" http://dumps.wikimedia.org/other/wikidata/
[8] "What is MongoDb?" http://searchdatamanagement.techtarget.com/definition/MongoDB
[9] "Advantages of MongoDb" https://www.mongodb.com/mongodb-architecture
[10] "WikiMatrix" http://www.wikimatrix.org/



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY