

Natural Language Based Report Generator

By K. D. Ranidu Nadeesha

169322P

Supervised By Mr. Saminda Premarathne

Faculty of Information Technology

University of Moratuwa

February 2019

Natural Language Based Report Generator

Software tool providing easy report generating for web-based applications

K. D. Ranidu Nadeesha

169322P

Dissertation submitted to the Faculty of Information Technology, University of Moratuwa,
Sri Lanka for the partial fulfillment of the requirements of the Degree of MSc in Information
Technology.

Faculty of Information Technology

University of Moratuwa

February 2019

Declaration

I declare that this thesis is my own work and has not been submitted in any form for another Masters, Degree or diploma at any university or other institution of tertiary education. Information derived from the published or unpublished work of others has been acknowledged in the text and a list of references is given.

Name of the Student:

Signature of the Student

K. D. Ranidu Nadeesha

.....
Date:

Supervised by:

Signature of the Supervisor

Mr. Saminda Premarathne

.....
Date:

Dedication

This dissertation is dedicated to my beloved parents, erandi and her parents, my teachers who gave me endless courage and support to achieve my task and goal in completing the research project.

Acknowledgment

My heartiest thanks go to my supervisor Mr. Saminda Premarathne for the guidance, assistance, encouragement, valuable bunch of advice on improving the research and providing this opportunity to carry out this research project.

Also, sincerely thanks to all my teachers who taught in MSc IT degree program. Things learned from these subjects made it easier to make this research project a successful one.

Last but not least, a sincere thank goes to everyone who supported especially teams from Redot Pvt Ltd and Venuerific Pte Ltd Singapore for contributing their valuable time on this research.

Abstract

Information plays a major role in our lives. It's a key element in day to day life. We use databases to store a large amount of data and retrieve it efficiently in less amount of time. It also plays a major role in computer systems. This information access by technical and nontechnical users. When it comes to non-technical users they should be able to know Structured Query Language to access data from databases. The goal of NLP[1] is to allow communication between people and computers without resorting to memorizing complex commands and procedures. In other words, NLP is a technique that can make the computer understand the languages used naturally by humans, but not by artificial language or created by man, as a programming language This minimize the communication gap between computer and humans. In order to archive this type of communication, we have to make the computer to understand what a human asks.

The natural language may be the symbol system easier to learn and use for people, it has proven to be the most difficult to master for a computer. Despite the challenges, natural language processing, or NLP, is widely regarded as a promising and critically important effort in the field of computer research. The general objective for the majority of computational linguists is to imbue the computer with the ability to understand and generate language so that, finally, people can approach their computers through text as although they were addressing another person. The applications that will be possible when NLP capabilities are fully realized are impressive computers that could process the natural language, translate languages with precision. and in real time, or extract and summarize information from a variety of data sources, depending on the requests of the users.

To use store data inside databases or retrieve information from databases, interaction with the database is important and for this purpose, technical knowledge is required. The problem is that it restricts the interaction between the naive user and the database. Only a few people who have knowledge of the formal language of the database can retrieve the

desired information from the database. To overcome this problem, this proposed system would have the ability to analyze the statements of users written in different ways and, consequently, gives a response to the user. In this way, it helps a normal educated person who has no knowledge of the query language to easily interact with the system.

This project proposes a customized solution for avoiding communication gap between non-technical user and database. Reducing the time spent on retrieve data is the main objective of this project which ultimately leads to a software application. The proposed system is capable of retrieve data from database by understanding human natural language questions. By keeping real time data structure JSON[2] format it easily adapts every database changes as well as the provide UI interface to enter questions and display results on the frontend. Written controllers and models are capable of understanding and generating results based on the question. Keeping in mind of the latest technologies and the mobile technology evolvment, Tool is capable of accessing it from desktop or mobile. For better performance, I have used Python3[3] scripting and flask frontend library for UI generating. I have tested the system with the collogues at my workplace which all of them are developers. And currently using the system for generating first and second level prototypes.

Table of Contents

1.	Introduction.....	1
1.1	Prolegomena	1
1.2	Background and Motivation	2
1.3	Problem domain.....	3
1.4	Hypothesis	4
1.4.1	Expected Features	4
1.5	Aim and Objectives	5
1.5.1	Aim	5
1.5.2	Objectives	5
1.6	Structure of the Thesis	5
1.7	Summary.....	6
2.	Current Development and Challenges	7
2.1	Introduction.....	7
2.2	Current developments	7
2.2.1	Quepy.....	8
2.2.2	CINDI	8
2.2.3	NLS-to-SQL.....	9
2.2.4	Fr2Sql.....	9
2.2.5	Natural Language Web Interface for Database (NLWIDB).....	10
2.3	Research Gap	10
	Quepy	10
2.4	Summary.....	11
3.	Technology foundation of the solution.....	12
3.1	Introduction.....	12
3.2	Technology used for the solution	12
3.2.1	Python	12
3.2.2	Natural Language toolkit	12
3.2.3	Django.....	13
3.2.4	Flask.....	13
3.2.5	JSON.....	13
3.2.6	Bitbucket.....	13
3.3	Development tools.....	13

3.3.1	Visual Code.....	14
3.3.2	PhpMyAdmin.....	14
3.3.3	MAMP	14
3.4	Hosting and Deployment technologies.....	15
3.4.1	AWS EC2.....	15
3.4.2	Beanstalk.....	15
3.5	Summary.....	15
4.	A new approach to natural language processing.....	16
4.1	Introduction.....	16
4.2	Requirement Gathering.....	16
4.3	Issues face by the clients in the industry	16
4.4	Gaps identified hasn't been addressed by other researches.....	17
4.5	Limitations with current development tools	17
4.6	Hypothesis	17
4.6.1	Natural Language Processing	18
4.6.2	Query generators.....	18
4.6.3	Rule-based database schema.....	18
4.7	User of the system	18
4.8	Input to the system.....	18
4.9	Outputs of the system	19
4.10	Features.....	19
4.10.1	System Specific Features	20
4.11	Summary.....	20
5.	Analysis and Design of the new natural language tool.....	21
5.1	Introduction.....	21
5.2	Research planning.....	21
5.2.1	Development methodology	22
5.3	Requirement analysis.....	23
5.3.1	Functional requirement of the solution.....	23
5.3.2	Non-functional requirements of the solution.....	23
5.4	Top level design architecture.....	23
5.5	Module architecture.....	24
5.5.1	Database Module / Database Driver	24
5.5.2	Natural Language Processing Module.....	25

5.6	Dependency management, ground up to work on generated prototype	28
5.6.1	Version controlling	28
5.6.2	Dependency management	28
5.6.3	Testing.....	28
5.7	Summary.....	28
6.	Implementation	29
6.1	Introduction.....	29
6.2	Overall solution	29
6.3	Implementation of the Solution	29
6.3.1	Programming.....	30
6.3.2	Database structure JSON Implementation.....	30
6.4	Stopword filter	32
6.5	Extraction of words.....	34
6.6	Join Query Handling.....	35
6.7	Generated sample user output.....	36
6.8	Summary.....	36
7.	Evaluation of the solution	37
7.1	Introduction.....	37
7.2	Participants	37
7.3	Summary.....	44
8.	Conclusion	45
8.1	Introduction.....	45
8.2	Conclusion	45
8.3	Limitations.....	45
8.4	Future works	46
8.5	Summary.....	46
9.	Reference	47

Table of Tables

Table 2-1: Evaluation done for existing tools.....	11
Table 5-1: Execution plan for the proposed system.....	21
Table 7-1: Test cases.....	39
Table 7-2: Test cases.....	41
Table 7-3: Test cases.....	44

Table of Figures

Figure 5-1: 2Evolutionary prototyping methodology used for proposed system	22
Figure 5-2: Top level design architecture of the proposed system	24
Figure 5-3: Modules contain in the projects	25
Figure 6-1: JSON table structure	31
Figure 6-2: Database connection	32
Figure 6-3: Stopword remove algorithm.....	33
Figure 6-4: Table identification	33
Figure 6-5: Word bag.....	34
Figure 6-6: System output.....	36
Figure 9-1: Join query handler	49
Figure 9-2: Order by query handler	50
Figure 9-1: User input capture code.....	51