

Efficient Process Retrieval Using Graph Database

Ravikumar More

Riya Chhajed

Madhav Dhoot

Vivek Bhandurge

Guided by Prof. S. P. Bholane

Department of Computer Engineering,
Sinhgad College of Engineering, Pune, Maharashtra, India.

Abstract

The aim of this paper is to retrieve a procedural text with the help of Natural Language Processing. Because existing methods for retrieving food recipes mainly focus on a procedural way. The procedures which need to be done simultaneously is difficult to show in the traditional system. It is utilized for various applications including recipe search, summarization and visualization. In order to solve this problem, it extends recipe-oriented problems by storing them in a structural way. This paper mainly focuses on identifying the dish and ingredients from user's query and retrieving the relevant process from the graph database. The major work flow is represented as a tree in which each leaf corresponds to a food or a cooking tool, each intermediate node corresponds to a cooking action, and the root node corresponds to the final cooking action. Here, we use new technology for storage purpose, Neo4j in which the dataset is stored in as a tree in which each leaf corresponds to a food or a cooking tool. So, it is a challenge to build this graph.

Keywords: Natural Language Processing, Neo4j.

Introduction

Background and Basics

This proposed paper is a graph-based approach for storing the recipes in the database. The motive of this project is to present the procedures in the structural manner. The previous system lacks in explaining the procedure because recipe text explains its procedure with a sequence of steps, a chef not always cooks them sequentially in practical

scene. The workflow of most of cooking recipes can be represented as a tree if we focus on the major stream composed of foods, tools, and actions. Trees can adopt various kind of effective algorithm such as computing the editing distance. Therefore, this paper aims to construct a system that extracts the major workflow as a tree whose leaves correspond to foods or tools and whose intermediate nodes correspond to cooking actions from the procedural text. Hereafter, the tree formatted workflow is being referred as a "Recipe Tree". Procedural representation often makes a similarity calculation more complicated. Therefore, this paper targets to extract the major workflow tree composed of only foods and tools as leaves and cooking actions as intermediate nodes.

Problem Definition

To build an Efficient Process Retrieval System using Neo4j for storage to store the process of recipe and ingredients of food in the graph format based on which process can be retrieved among many processes according to relevant query. Thus, retrieving two procedures simultaneously. Thus, making it easier for the user to prepare best relevant dish.

Scope Statement

It would more helpful for chef or who is interested in cooking to retrieve the recipe from the procedural text. And also, to learn new technology for storing the data such as Neo4j instead of MySQL. This paper can be used for storing any kind of processes and data. Any process which consists of a no. of steps can be stored and retrieved from database. This discussed idea can store textual data in a clustered graph format. Other data types can also be added other than text, i.e. images, video and audio.

Procedural programs can be stored for machine learning.

System Architecture

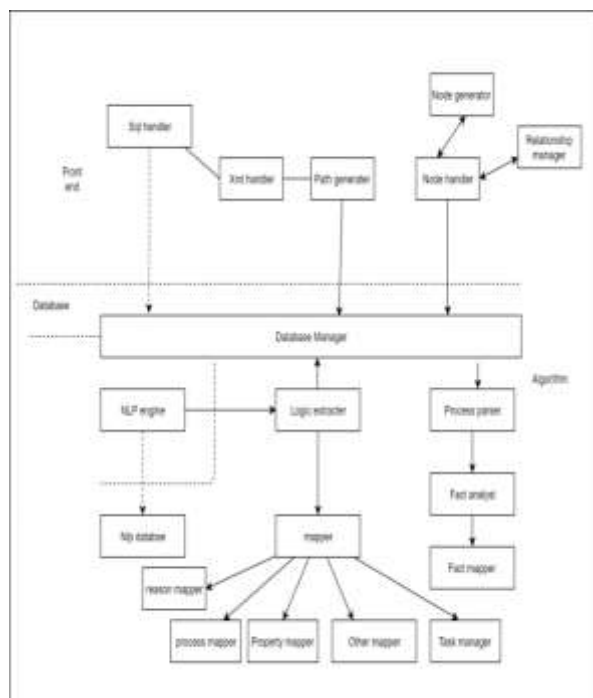


Fig 1.1 System Architecture

The working of the systems is as follows:

1. In the first step admin has to upload pdf file of recipe or can enter the recipe to the system.
2. The system will split the pdf and extract each procedure of recipe.
3. From procedure, by using Natural Language Processing, it will extract the noun and verb from procedure.
4. And finally, system will dynamically create an XML file in which the extracted nouns and verbs will be written. And that XML file will be pushed to the Neo4j database when user will search the recipe.

Related Work

1. Flow Graph Corpus from Recipe Texts

In this paper, we present our attempt at annotating procedural texts with a flow graph as a representation of understanding. The domain we

focus on is cooking recipe. The flow graphs are directed acyclic graphs with a special root node corresponding to the final dish.

2. A Framework for Procedural Text Understanding

In this paper we propose a framework for procedural text understanding. Procedural texts are relatively clear without modality nor dependence on viewpoints, etc. And have many potential applications in artificial intelligence. Thus, they are suitable as the first target of natural language understanding.

3. Feature Extraction and Summarization of Recipes Using Flow Graph

Our system extracts (i) a general way to cook as a summary of cooking procedures and (ii) the characteristic features of each recipe by analysing the work-flows of the top ten results.

4. Object Recognition based on Object's Identity for Cooking Recognition Task

In this paper, we address a novel task "cooking Recognition task". Cooking recognition task is to keep recognizing a cooking action of a cook and the target food product of the action at any time of the cooking.

HYPOTHESIS

The workflow of most of cooking recipes can be represented as a tree if we focus on the major stream composed of foods, tools, and actions. Trees can adopt various kind of effective algorithm such as computing the editing distance. Therefore, we aim to construct a system that extracts the major workflow as a tree whose leaves correspond to foods or tools and whose intermediate nodes correspond to cooking actions from the procedural text. Hereafter, we call the tree formatted workflow as a "Recipe Tree." Procedural representation often makes a similarity calculation more complicated. Therefore, in this paper, we target to extract the major workflow tree composed

of only foods and tools as leaves and cooking actions as intermediate nodes.

Design of the Study

Implementation phase focus over system design objectives. Software implementation is the process of designing, writing, testing, debugging/troubleshooting and maintaining the source code of computer programs.

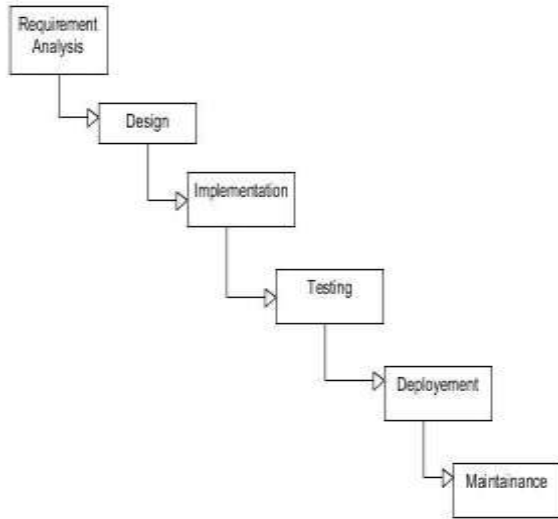


Fig1.2: Software Development Process

Implementation of system is divided into 7 main modules. With the help of Register and Login module user will register and login into the system. User will select source and destination location as an input. System will predict traffic weather and collision and will suggest the best route for that system uses weather, collision and traffic modules. Fore predication system uses Support Vector Machine and Hidden Markov Model.

SAMPLE OF THE STUDY

This includes of developing a system that helps a user to discover procedure of recipe in graphical way, for this we are using new technology of

database neo4j in which whole procedure of recipe will be displayed in neo4j database.

TOOLS USED

Software Requirement:

- Operating System: Windows 8 and above.
- Application: Server Tomcat 5.0/6.X
- Language: Java
- Front End: HTML, JSP

- Database: MySQL, Neo4j

Hardware Requirement: The hardware design of the system includes designing the hardware units and the interface between those units.

- Processor: Pentium –III
- RAM: 1 GB (min.)
- Hard Disk: 20 GB

STATISTICAL TECHNIQUE USED

It proposes a graph-based approach for storing the recipes in the database. The motive of this paper is to present the procedures in the structural manner. The previous system was lacking in explain the procedure because recipe text explains its procedure with a sequence of steps, a chef not always cooks them sequentially in practical scene.

Algorithm

NLP (Natural Language Processing)

NLP algorithms are typically based on machine learning algorithms NLP can rely on machine learning to automatically learn these rules by analysing a set of examples (i.e. a large corpus, like a book, down to a collection of sentences), and making a statically inference. In general, the more data analysed, the more accurate the model will be. Pseudo code:

1. While (there is a sentence) do
2. Term=nextTerm(sentence)

3. While (there are noun) do
4. Noun=findNoun()
5. If unprocessed(Term)
6. toNoun=countNoun(term)
7. If ToNoun>=Nr
8. Insert(Term,Noun)
9. End if

10. End if
11. End while
12. End while

Limitation of the Study

XML file of recipe should be present in the system before user search the recipe, so that xml file is to be pushed to the neo4j database so as to display graph in it.

Function:

F1=NLP to extract the recipe name

Output:

O1=Success Case (It is the case when all the inputs are given by system are entered correctly and accurate prediction is done)

O2=Failure Case (It is the case when the input does not match the validation Criteria)

Success:

1. When the desired output is generated
2. Data is stored in a node-edge format.
3. Step by step process is generated and displayed.

Failure:

1. Garbage output is generated
2. Wrong Process is generated
3. Failure in database Connectivity.

APPROACH

Our approach is to study the Neo4j database in which we can show the procedure of recipe in the graphical format.

Experiment Result:

The major functionality is to be performed by admin who will upload the recipe pdf to the system and extract the procedure and ingredient of each recipe and then it will be converted into XML file which will be passed to the Neo4j. This experiment has been done on multiple files of recipe and user functionality is to search the recipe name and system will show the procedure of whole recipe in neo4j database, it searches multiple recipe and to get the accurate result from the experiment.

Future Scope:

In this paper it uses Neo4j for storage of recipe. The main objective of this paper is to propose a method of storing the recipe and ingredients of food in the xml format and later on displaying the recipe in graph format to the user. In future, the system can be used in the field of food in which user can see the procedure of whole recipe instead of text.so that user can easily understand the procedure of recipe quickly.

Acknowledgment:

We are profoundly grateful to Prof. S. P. Bholane for her expert guidance and continuous encouragement throughout to see that this project rights its target since its commencement to its completion. We would like to express deepest appreciation towards Dr. S. D. Lokhande, Principal, Sinhgad College of Engineering, Pune, Prof. M. P. Wankhade, Head of Department of Computer Engineering. At last we must express our sincere heartfelt gratitude to all the staff members of Computer Engineering Department who helped me directly or indirectly during this course of work.

Conclusion:

With the methodology discussed earlier, the drawbacks of the existing systems have been

overcome. This gives an efficient, precise solution to the users, so that he/she gets detailed accurate process output from database available. Also, it's been proved that how NLP can extract the procedure from pdf and it gives best results. From that file we have generated the XML file so that it can extract the start node to end. And finally, the Cluster of the data gives processes that attached to the same node (object) and user can see the graph in the sequence in Neo4j.

Reference:

[1]. Shinsuke Mori, Hirokuni Maeta, Yoko Yamakata, and Tetsuro Sasada, "Flow graph corpus from recipe texts," in Proceedings of the Ninth International Conference on Language Resources and Evaluation, 2014, pp. 2370–2377.

[2]. Hirokuni Maeta, Tetsuro Sasada, and Shinsuke Mori, "A framework for procedural text understanding," in Proceedings of the 14th

International Conference on Parsing Technologies, 2015.

[3]. Yoko Yamakata, Shinji Imahori, Yuichi Sugiyama, Shinsuke Mori, and atsumi Tanaka, "Feature extraction and summarization of recipes using flow graph," in SocInfo, 2016, vol. 8238 of LNCS, pp. 241–254.

[4]. Yoko Yamakata, Koh Kakusho, and Michihiko Minoh, "Object recognition based on object's identity for cooking recognition task," in CEA, 2010, pp. 278–283.

[5]. Reiko Hamada, Jun Okabe, Ichiro Ide, Shin'ichi Satoh, Shuichi Sakai, and Hidehiko Tanaka, "Cooking navi: assistant for daily cooking in kitchen," in Proceedings of the 13th annual ACM international conference on Multimedia, 2005, MULTIMEDIA '05, pp. 371–374.

