SMART KITCHEN: RECIPE RECOMMENDATION SYSTEM

YAP LEE LENG

A report submitted in partial fulfilment
of the requirements for the award
of the degree of
Bachelor of Computer Science (Software Engineering)

Faculty of Computer Systems & Software Engineering
Universiti Malaysia Pahang (UMP)

MAY 2010

ABSTRACT

Today want to eat or cook what meals? This is a common question that everyone asking, especially the housewife. Many people busy on work, some of them no time to think the meal that they want prepare to their family at night. Children are very changeable, not easy to think the meal prepare to them every day. Therefore, this project is carried out to build a prototype for recipe recommendation. It gives recommender or suggest recipe to the user based on their preference or the ingredient that available. The system is a web based system, it develop for those who are in rush. The Linear Model of Expert System Development Life Cycle is implemented in the system development. The rule is design based on the recipe that collected from web sites and book.

ABSTRAK

Hari ini mahu makan atau memasak makanan apa? Ini adalah soalan umum yang semua orang bertanya, terutamanya suri rumah tangga. Banyak orang sibuk bekerja, beberapa dari mereka tidak ada masa untuk memikirkan makanan yang mereka ingin mempersiapkan untuk keluarga mereka pada malam. Cita rasa kanak-kanak sangat berbeza dan selalu tukar, tidak mudah untuk berfikir makanan yang sesuai untuk mereka setiap hari. Oleh kerana itu, projek ini dilakukan untuk membina satu prototaip yang member cadangan resipi. Ini memberi cadangan atau mencadangkan resipi bagi pengguna berdasarkan keutamaan mereka atau bahan yang sedia. Sistem ini adalah sistem berasaskan web. System ini dibangunkan untuk mereka yang terburu-buru atau tidak lapang. Model LMESDLC dilaksanakan dalam pembangunan sistem. Kandunganya adalah direkakan berdasarkan resipi yang dikumpul dari laman web dan buku.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Introduction

Cooking is a process that uses human knowledge to combine suitable ingredient. Recipes also a piece of knowledge, through the recipe that human cook, we can convey some information about their personality, culture and habits.

Now, less and less people have time and confident to cook. Recommender system provide a way to help people search for food and cook from recipes in a more flexible and easy way. Currently, there are so many websites for cooking recipes, and the recipe are regarded as "easy" to cook.[1] As the name suggests, recommender system is to recommend or suggest recipe to the users based on their preferences. Users can buy ingredient based on the recipe that recommender by the system. Making recommendations on which food to prepare based on recommending recipes is an interesting functionality in itself. [2]

1.2 Problem Statement

Nowadays, many people busy on work. Some of them did not know the meals they want to cook later need prepared what ingredient. It is normal to think that couples or family members who work at a company or a person who lives alone want to cook food for themselves as quickly as possible and no need to worry about what to cook when they are in rush. However, if everyday having same food, then they will get bored. They need an easy way to get more recipes.

Thinking of what to cook is also a difficult problem. To attract children liking, parent need to exchange the menu every day. Parents not only think to what recipe to changes, they also need to consider the nutrition that their children taken. Besides that, some people will forget buy ingredients to stock in their kitchen. This will become a problem when they want to prepare meal within short time. It is difficult to think what to cook with limited ingredient that in the kitchen.

1.3 Objective

The objectives to be achieved in this project are

i. To building an expert system to determine and recommend a recipe based on their preferred.
ii. To develop a system that give recommend recipe to users from the available ingredient.
iii. Enable the users upload the recipe to enlarge the database

1.4 Scope

There are four scopes in this project.

i This system is web based system.

ii This system is develop for those who in rush.

iii Users need to input or select the data by themselves

iv This system only have four types of meals

1.5 Thesis Organization

This thesis contains 6 chapters, which are Introduction, Literature Review, Methodology, Implementation, Result and Discussion, and the Conclusion. Chapter 1 gives an overview and basic concept of the study conducted. It consists of introduction, problem statements, objectives, scopes and thesis organization. Chapter 2 is Literature Review; it reviews the previous or existing works that was done by other researcher. All the resources that corrected such as, relevant article, journal, books, and reports will be discussed in detail in this chapter. Chapter 3 discuss about the method, techniques and the algorithm that will be apply in the development. Chapter 4 discussed about the detail of the research. Furthermore, the result, testing and discussion are explained in the chapter 5. Finally, the conclusion concludes the research and thesis.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter describes about some case studies related to the recipe recommend system. The purpose of this chapter is to increase knowledge, architecture and understanding in project background. Some of the case study and sources are come from several websites, articles, journals and books based on the research that had been done.

2.2 Expert System

An expert system is software that attempts to reproduce the performance of one or more human experts, most commonly in a specific problem domain, and is a traditional application or subfield of artificial intelligence (AI). [4]

Expert system simulates the judgments and behavior of a human or an organization that has expert knowledge and experience in a particular domain. Knowledge base for each particular situation that is described to the program is applied from the knowledge base that containing accumulated experience and a set of rules. Sophisticated expert systems can be enhanced with additions to the knowledge base or to the set of rules. [3]

Expert systems represent the expertise knowledge as data or rules within the computer. These rules and data can be called upon when needed to solve problems. [3]

2.2.1 Comparison of expert systems with human experts

Books and manuals have a tremendous amount of knowledge but a human has to read and interpret the knowledge for it to be used.

Table 2.1: Comparison of expert systems and human experts

Human experts	**Experts systems**
Use knowledge in the form of rules of thumb or heuristics to solve problems in a narrow domain.	Process knowledge expressed in the form of rules and use symbolic reasoning to solve problems in a narrow domain.
In a human brain, knowledge exists in a compiled form.	Provide a clear separation of knowledge from its processing.
Capable of explaining a line of reasoning and providing the details.	Trace the rules fired during a problem-solving session and explain how a particular conclusion was reached and why specific data was needed.

Table 2.1: Comparison of expert systems and human experts (cont...)

Human experts	**Experts systems**
Use inexact reasoning and can deal with incomplete, uncertain and fuzzy information.	Permit inexact reasoning and can deal with incomplete, uncertain and fuzzy data.
Can make mistakes when information is incomplete or fuzzy.	Can make mistakes when data is incomplete or fuzzy.
Enhance the quality of problem solving via years of learning and practical training. This process is slow, inefficient and expensive.	Enhance the quality of problem solving by adding new rules or adjusting old ones in the knowledge base. When new knowledge is acquired, changes are easy to accomplish.

2.2.2 Types of Expert system

There are several types of expert systems:-

- Rule based
- Object oriented
- Logic based
- Induction based
- Hybrid systems

This chapter will describe about the rule based expert system, object oriented and the hybrid systems.

2.2.3 Rule based expert systems

A rule-based expert system is an expert system which works as a production system in which rules encode expert knowledge.

Rule-based expert systems use human expert knowledge to solve real-world problems that normally would require human intelligence. Expert knowledge is often represented in the form of rules or as data within the computer.

Depending upon the problem requirement, these rules and data can be recalled to solve problems. Rule-based expert systems have played an important role in modern intelligent systems and their applications in strategic goal setting, planning, design, scheduling, fault monitoring, diagnosis and so on. [9]

2.2.3.1 Structure of rule based expert system

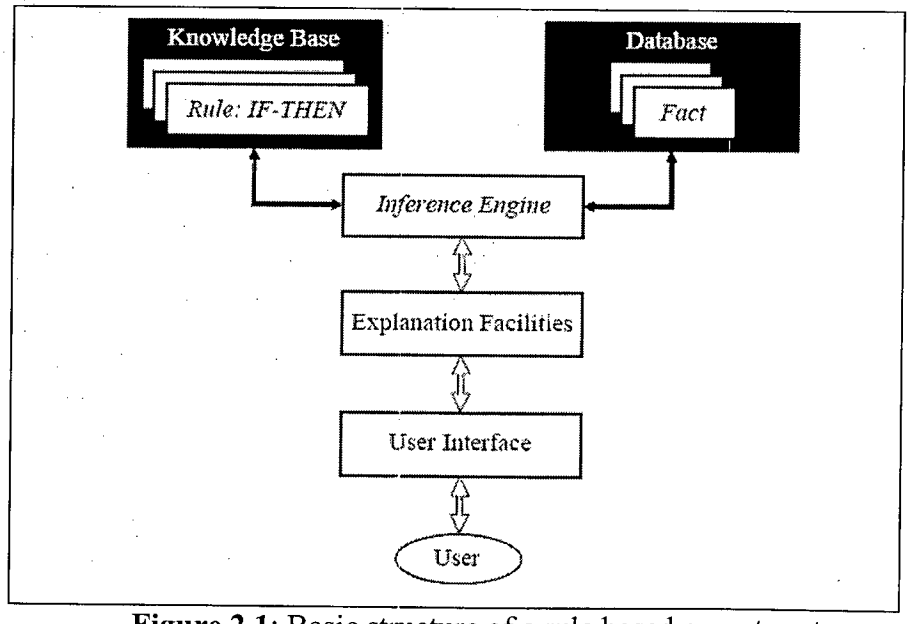

Figure 2.1: Basic structure of a rule based expert systems

A rule based expert system has five components:-

Knowledge base - contains the domain knowledge useful for problem solving. Each rule specifies a relation, recommendation, directive, strategy or heuristic and has the IF (condition) THEN (action) structure. When the condition part of a rule is satisfied, the rule is said to *fire* and the action part is executed. [15]

Database - includes a set of facts used to match against the IF (condition) parts of rules stored in the knowledge base. [15]

Inference engine - carries out the reasoning whereby the expert system reaches a solution. It links the rules given in the knowledge base with the facts provided in the database. [15]

Explanation facilities - enable the user to ask the expert system *how* a particular conclusion is reached and *why* a specific fact is needed. An expert system must be able to explain its reasoning and justify its advice, analysis or conclusion. [15]

User interface - is the means of communication between a user seeking a solution to the problem and an expert system. [15]

2.2.3.2 Chaining

There are two type of reasoning in rule based expert systems, forward chaining and backward chaining.

i. Forward chaining

   Forward chaining is the data-driven reasoning. Forward chaining starts with the data available and uses the inference rules to conclude more data until a desired goal is reached.

Each time only the topmost rule is executed. When fired, the rule adds a new fact in the database. Any rule can be executed only once. The match-fire cycle stops when no further rules can be fired.

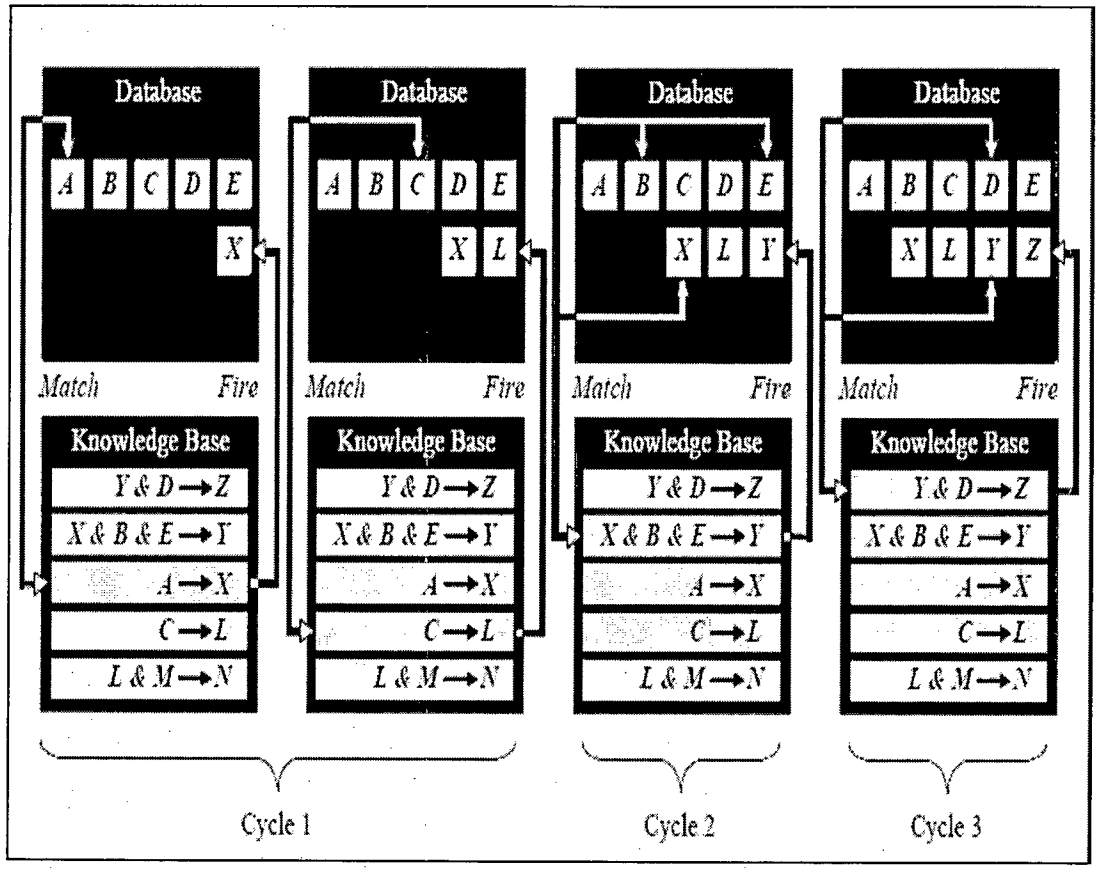

Figure 2.2: Forward chaining

To apply the forward chain, inference engine must:

1. Match the condition patterns of rules against facts in working memory.
2. If there is more than one rule that could be used (that could "fire"), select which one to apply.
3. Apply the rule, maybe causing new facts to be added to working memory
4. Halt when some useful conclusion or goal is added or until all possible conclusions have been drawn.

ii. Backward chaining

Backward chaining is the goal-driven reasoning. Backward chaining starts with a list of goals and works backwards to see if there is data which will allow it to conclude any of these goals. The expert system has the goal or a hypothetical solution and the inference engine attempts to find the evidence to prove it.

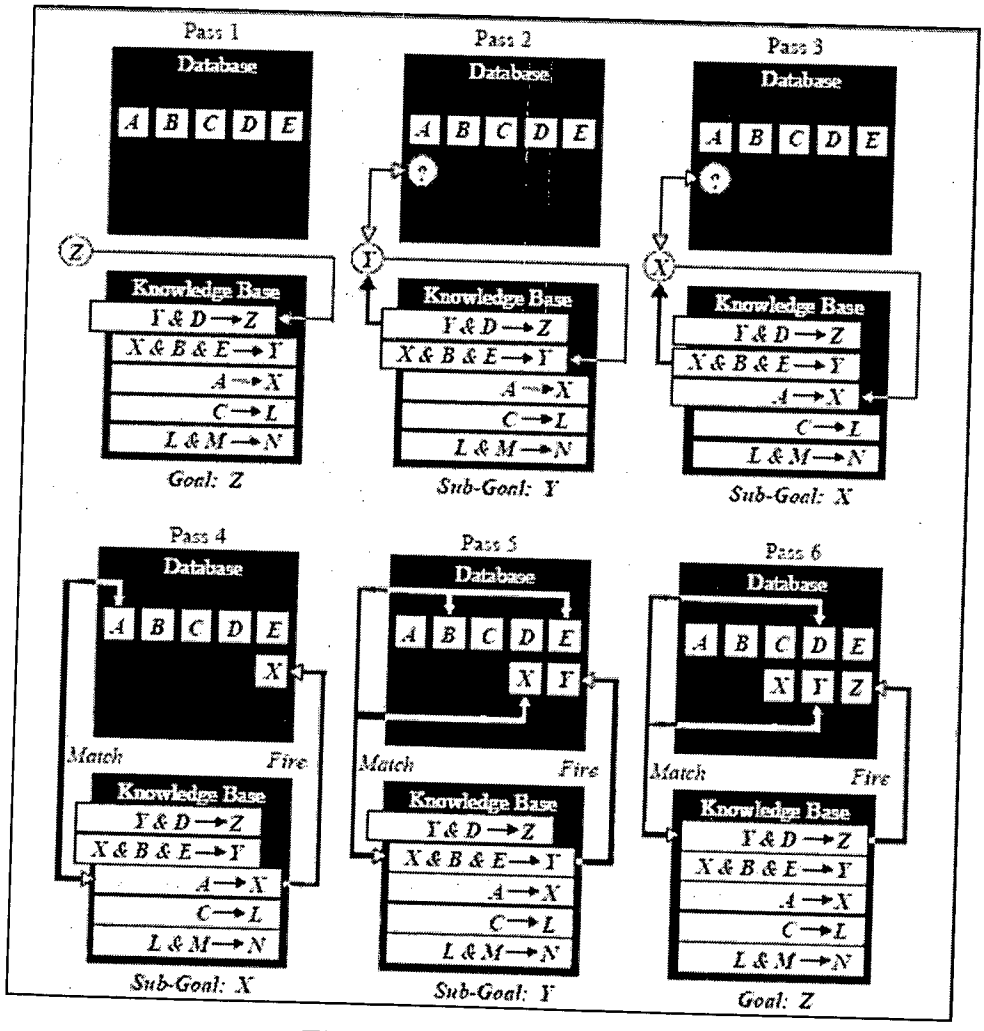

Figure 2.3: Backward chaining

To apply the backward chaining, the inference engine must:

1. Select rules with conclusions matching the goal.
2. Replace the goal by the rule's premises. These become sub-goals.

3. Work backwards till all sub-goals are known to be true, either they are facts or the user provides the information.

2.2.3.3 The advantages and disadvantages [15]

i. Advantages

- The structure of a rule-based expert system provides an effective separation of the knowledge base from the inference engine. This makes it possible to develop different applications using the same expert system shell.
- Dealing with incomplete and uncertain knowledge. Rule-based expert systems are able of representing and reasoning with incomplete and uncertain knowledge.
- Natural knowledge representation. An expert usually explains the problem-solving procedure with such expressions as this: "In such-and-such situation, I do so-and-so". These expressions can be represented quite naturally as IF-THEN production rules. (Negnevitsky, Pearson Education, 2005)
- Production rules have the uniform IF-THEN structure. Each rule is an independent piece of knowledge. The very syntax of production rules enables them to be self documented.
- The structure of a rule-based expert system provides an effective separation of the knowledge base from the inference engine. This makes it possible to develop different applications using the same expert system shell.

ii. Disadvantages

- Although the individual production rules are relatively simple and self-documented, their logical interactions within the large set of rules may be opaque. Rule-based systems make it difficult to observe how individual rules serve the overall strategy.
- The inference engine applies an exhaustive search through all the production rules during each cycle. Expert systems with a large set of rules (over 100 rules) can be slow, and thus large rule-based systems can be unsuitable for real-time applications.
- Rule-based expert systems do not have an ability to learn from the experience. Unlike a human expert, who knows when to "break the rules"; an expert system cannot automatically modify its knowledge base, or adjust existing rules or add new ones. The knowledge engineer is still responsible for revising and maintaining the system.

2.2.4 Frame-based Expert System (Object Oriented)

Frames are an application of object-oriented programming for expert system. (Negnevitsky, Pearson Education, 2005) Object oriented programming uses object as a basic for analysis, design and implementation. A knowledge engineer refers a frame as an object.

2.2.4.1 Frames as a Knowledge Representation Technique

i. Slot

The concept of a frame is defined by a collection of slots. Each slot describes a particular attribute or operation of the frame. (Negnevitsky, Pearson Education, 2002) Slots used to store values. It may contain a default value or a pointer to another frame, a set of rules or procedure by which the slot value is obtained. (Negnevitsky, Pearson Education, 2002)

The information included in a slot is as below:

- Frame name
- Relationship of the frame to the other frames
- Slot value
- Default slot value
- Range of the slot value\
- Procedural information

ii. Class and Instances

Frame may refer to a particular object or to a group of similar objects. Therefore, the word instance-frame use to refer to a particular object and the class-frame refer to a group of similar object with common attributes.

CLASS:	*Computer*		
[Str]	*Item Code:*		
[Str]	*Model:*		
[Str]	*Processor:*		
[Str]	*Memory:*		
[Str]	*Hard Drive:*		
[Str]	*Floppy:*	[Default]	← *3.5"; 1.44MB*
[Str]	*CD-ROM:*		
[Str]	*Mouse:*		
[Str]	*Keyboard:*		
[Str]	*Power Supply:*	[Default]	← *145 Watt*
[Str]	*Warranty:*	[Default]	← *3 years*
[N]	*Cost:*		
[Str]	*Stock:*	[Initial]	*In stock*

Figure 2.4: Example of class-frame

INSTANCE: *IBM Aptiva S35*		
Class:	*Computer*	
[Str]	*Item Code:*	*SY7975*
[Str]	*Model:*	*IBM Aptiva S35*
[Str]	*Processor:*	*Pentium 233MHz*
[Str]	*Memory:*	*48MB*
[Str]	*Hard Drive:*	*6.4GB*
[Str]	*Floppy:*	*3.5"; 1.44MB*
[Str]	*CD-ROM:*	*24X*
[Str]	*Mouse:*	*Cordless Mouse*
[Str]	*Keyboard:*	*104-key*
[Str]	*Power Supply:*	*145 Watt*
[Str]	*Warranty:*	*3 years*
[N]	*Cost:*	*1199.99*
[Str]	*Stock:*	*In stock*

INSTANCE: *IBM Aptiva S9C*		
Class:	*Computer*	
[Str]	*Item Code:*	*SY7975*
[Str]	*Model:*	*IBM S9C*
[Str]	*Processor:*	*Pentium 200MHz*
[Str]	*Memory:*	*32MB*
[Str]	*Hard Drive:*	*4.2GB*
[Str]	*Floppy:*	*3.5"; 1.44MB*
[Str]	*CD-ROM:*	*16X*
[Str]	*Mouse:*	*2-button mouse*
[Str]	*Keyboard:*	*104-key*
[Str]	*Power Supply:*	*145 Watt*
[Str]	*Warranty:*	*3 years*
[N]	*Cost:*	*999.99*
[Str]	*Stock:*	*In stock*

Figure 2.5: Example of instances-frame