Grammar Rules and Natural Language Understanding

– How the Final Patterns Script improves performance of chatbots

Ron C. LEE
Rong-chang ESL, Inc., 72 Quill,
Irvine, California, 92620

ABSTRACT

The Final Patterns Script (FPS) method was developed from the study of English language patterns and grammar rules, with the goal to improve machine understanding of natural language questions and shorten the time required to develop chatbots. FPS consists of over 100 groups of patterns, with each group containing a set of semantically equivalent patterns, and one pattern designated as Final Pattern. To answer questions posed by humans, the system invokes patterns from these groups to match the pattern in the natural language question from the user. Upon finding a match, the FPS then performs transformations onto other patterns in the group until it reaches the Final Pattern, which will finally retrieve the answer to the question. By employing the FPS method, chatbot developers only need to prepare answers to questions containing Final Patterns. Users can ask the same question in different ways and be able to retrieve the answer as long as that answer has been prepared for the relevant final-pattern question. This significantly reduces the time needed to create a functional chatbot. This article discusses how to choose a Final Pattern, the ways to organize patterns, as well as the limits of this Final Patterns Script method.

Keywords: Final Patterns Script, Grammar Rules and Natural Language Processing, Chatbot, Question Answering System

1. INTRODUCTION

As Barbara J. Grosz (2018) says, "the ability for a computer system to carry on a dialogue with a person has been a compelling goal of artificial intelligence (AI) research from its inception."[1] Nearly 70 years has passed since Turing Test was proposed, no computer systems have been able to pass the debatable test. Building a dialogue system is far more difficult than people originally thought.

In 2018 Loebner Prize Contest, one chatbot named "Tutor" won the first place in the selection contest, and the second place in the final Turing Test. According to the botmaster, "Tutor" was able to achieve such a result, one of the important factors was its application of the Final Patterns Script (FPS). (https://www.rong-chang.com/tutor_mike.htm)

2. WHAT IS THE FINAL PATTERNS SCRIPT

Languages have patterns, which are fixed phrases and grammar rules used by people, often subconsciously, to express various meanings. English, a natural language with a long history, has been widely used by many people in different countries in the world. It has been enriched so much during its development to meet the needs of the people to express subtle differences in meaning. Therefore, English can provide its users with various different ways to express the same or similar meaning.

The botmaster of "Tutor" has spent years studying English language patterns. He groups them according to the meaning they express. In each group, every pattern can express the same or similar meaning. Among all the patterns in the group, one pattern is designated as a Final Pattern; all the other patterns in the group are then converted to this Final Pattern. Since all the patterns in the group express the same or similar meaning, chatbot builders only need to create answers to questions that contain Final Patterns without worrying about the same question asked by users who use different language patterns. In other words, the same question can be asked by different users in different ways. If each group contains 30 patterns in average, the chatbot builder will be able to build a chatbot 30 times faster because only one pattern - the Final Pattern needs to be dealt with instead of all the 30 patterns in the group. The chatbot developer writes an answer to the question that contains the Final Pattern; questions using other 29 patterns will be converted to this final-pattern question. As a result of using the FPS, the 30 similar questions asked by different users can all retrieve the answer written for the final-pattern question by the chatbot developer. The Loebner Prize selection contest used 20 questions to test all the participating chatbots. The fact that "Tutor" won the 1st place with the total score much higher than other participants demonstrated the effectiveness of the FPS method.

ISBN: 978-1-950492-63-3
ISSN: 2771-6368

https://doi.org/10.54808/ICSIT2022.01.12

Over 100 groups of English patterns have been identified and included in the Final Patterns Script. Most grammar books on the market are written for language teaching or theoretical research of the English language. The Final Patterns Script is a grammar reference for people working in the field of computational linguistics. To help understand how the FPS works, some important patterns are introduced below.

3. THE PATTERN "WHAT IS ......"

In the FPS, "what is ......" is considered the most important Final Pattern as so many other patterns are converted to this pattern. If the chatbot builder has prepared an answer to "what is deep learning", users do not have to ask “what is deep learning” to get the answer. They can ask the question in various ways such as “could you tell me about (deep learning)”, “do you know (deep learning)”, “what do you know about (deep learning)”, “I want to know what (deep learning) is.” Depending on how the answer is written to the “what is ......” question, more patterns can be converted to “what is ......” question. Instead of asking “what is (deep learning)”, users can also ask “what do you think of (deep learning)”, “what is (deep learning) used for”, and so on.

In the English language, nouns can be countable or uncountable. If it is countable, usually the plural form is to add “s” to the noun. When users ask questions, they may use either plural form or singular form. Therefore, both possibilities need to be taken into consideration. If the noun is countable, the Semi-Final Pattern “what are ……” also needs to be included. For example, for the question “what is laptop”, the chatbot developer writes an answer to “what is laptop” and also enables “what are laptops” to share the same answer in case users use the plural form to ask the question.

When we ask about something, we use “what is ……”, but if we ask about a person, we use “who is ……” For example, “Who is Donald Trump?” In the FPS, “who is ……” is not a Final Pattern. It is converted to “what is ……” Instead of writing an answer to “who is Donald Trump”, the chatbot developer writes an answer to “what is Donald Trump”. Of course, people usually would not ask the question this way because Donald Trump is a person. The advantage of using the pattern “what is ……” instead of “who is ……” as a Final Pattern is that “what is ……” can handle more different ways of asking the question. If the answer is matched only to the question “who is Donald Trump”, the question “what do you know about Donald Trump” will not be able to retrieve the answer. If the answer is written for “what is Donald Trump”, the question “who is Donald Trump”, “what do you think of Donald Trump”, or “tell me about Donald Trump” can all retrieve the prepared answer.

4. INFORMATION QUESTIONS VS. YES/NO QUESTIONS

There are two kinds of questions: Information Questions and Yes/No Questions. The Information Questions are also called wh- questions because the questions start with pronouns “what”, “who”, “when”, “where”, “why” or “how”. All these words start with “wh-” with “how” as an exception. Most patterns in the FPS are about Information Questions. For example, a user may ask “how do you make coffee”, “how can I make coffee”, “how do people make coffee” and so on. The Final Pattern for them is “how to ……” A chatbot developer only needs to provide an answer to “how to make coffee” and users will be able to get the answer whether their question is “how can I make coffee” or “how do you make coffee”. The same way, the Final Patterns for other Information Questions are as follows: “what to ……” “who to ……” “where to ……” “when to ……” “why to ……” As to Yes/No Questions, they have only two answers, either “Yes” or “No”. They are mostly involved with modal verbs such as “can”, “may”, “will” or help verbs such as “have”, “does”, and “be”.

5. MODAL VERBS & HELP VERBS

Both modal verbs and help verbs are worth mentioning because of their functions in forming Yes/No Questions. In English, modal verbs are “must”, “may”, “might”, “will”, “would”, “shall”, “should”, “could”, and “can”; help verbs include “have”, “has”, “do”, “does”, “did”, “am”, “is”, “are”, “was”, and “were”. To form Yes/No Questions, we move a modal verb to the beginning of the sentence. If a sentence has no modal verb, we put a help verb “do”, “does”, or “did” at the beginning of the sentence. For example, we can change the statement “you can talk to a robot” into the question: “Can you talk to a robot?” If the predicate of a sentence is “be”, the predicate will be moved to the beginning of the sentence. With a sentence without a modal verb and "be" verb, such as “The Earth moves around the Sun”, it can be changed into a Yes/No Question by adding “does” to the beginning of the sentence and delete “s” from “moves.”

Some modal verbs are similar in meaning. The difference may lie only in politeness. For example, “would you show me your ID” is more polite than “Can you show me your ID?" When different modal verbs appear among the patterns in a group, "can" is often chosen to be included in the Final Pattern. The answer to "can you show me your ID" will also be retrieved if the user asks: "Would you show me your ID?"

6. THE SIMPLE STRUCTURE OF THE FINAL PATTERNS SCRIPT

The FPS is written in AIML (Artificial Intelligence Markup Language), whose structure is straightforward. The following, as an example, is extracted from a group with Semi-Final Pattern "tell me about ……" and Final Pattern "what is ……"

```
<category>
<pattern>GIVE ME INFORMATION ABOUT
*</pattern>
<template><srai>tell me about
<star/></srai></template>
</category>

<category>
<pattern>I NEED INFORMATION ABOUT
*</pattern>
<template><srai>tell me about
<star/></srai></template>
</category>

<category>
<pattern>SAY SOMETHING ABOUT *</pattern>
<template><srai>tell me about
<star/></srai></template>
</category>

<category>
<pattern>YOU MUST SAY SOMETHING ABOUT
*</pattern>
<template><srai>tell me about
<star/></srai></template>
</category>

<category>
<pattern>WHAT CAN YOU TELL ME ABOUT
*</pattern>
<template><srai>tell me about
<star/></srai></template>
</category>

<category>
<pattern>TELL ME ABOUT *</pattern>
<template><srai>what is <star/></srai></template>
</category>
```

In AIML, <category></category> is a unit, within which, <pattern></pattern> contains the question, and <template></template> provides the answer; between <srai></srai> within <template></template> is the recursion. "*" and <star/> are variables. The pattern "say something about ……" is converted to "tell me about ……" which in turn is converted to the Final Pattern "what is ……"

The chatbot "Tutor" is created in AIML; its database consists of large AIML files accumulated during the past more than 10 years. They can be converted to Excel format if needed. Any existing chatbots that want to improve their performance by using the FPS can easily convert the FPS into any formats they want and make it work with their database.

7. HOW TO CHOOSE A FINAL PATTERN

The number of patterns in a group varies. Some groups are larger than others. Picking a Final Pattern may not be an easy decision. One principle used in choosing a Final Pattern for a group in the FPS is to pick the most commonly used pattern. Searching a corpus that contains a large amount of English text can help you find the most commonly used patterns in the English language. Another principle in choosing a Final Pattern is to choose one that is easy for chatbot developers to remember. For example, for modal verbs, always choose "can"; for pronouns, choose "you". That is why "can you ......" is often seen in Final Patterns when both modal verbs and pronouns are involved.

8. TWO WAYS TO ORGANIZE PATTERNS IN A GROUP

Having grouped patterns, how to organize them can affect the efficiency and effectiveness of the FPS. There are two ways to organize patterns in a group. If a chatbot has already been created, and the Q&A database already exists, a top-down approach might be more useful, which means the pattern 1 is converted to pattern 2 and then pattern 2 is converted to pattern 3, and so on, until it reaches the Final Pattern at the bottom. The patterns in the group are ordered according to the frequency of their use in the English language, with the least used at the top and most used at the bottom.

When a user asks a question, but cannot get an answer from the system, there are two possibilities. One is there is no answer to the question in the database; the other is there is an answer in the database, but the pattern in the user's question does not match the pattern of the question in the database. If it is the second reason, the FPS may help the user to get the answer.

When the user cannot get the answer that exists in the database, the system will search through the patterns in the FPS to match the pattern of the user's question. If a match is found in a group, the system will go through all the patterns in that group to find a match to the pattern used by the question in the database. If it is found, the answer to the question will be retrieved and presented to the user. If the pattern checked does not match, the system will continue to check the next pattern in the group until the match to the pattern of the question in the database is found. If it reaches the Final Pattern at the bottom, but still cannot find a match, it may mean the answer to the user's question does not exist in the database or the pattern of the question in the database is not included in the FPS.

Another way to organize patterns in a group is to link every pattern directly to the Final Pattern instead of parsing every pattern in the group to reach the Final Pattern. The advantage of this approach is that it makes the program more efficient, provided that all the answers are written only for the final-pattern questions. Therefore, this approach is better to be used for chatbots being created from scratch.

For a chatbot whose Q&A database has already been created and its answers are not merely written for final-pattern questions, the top-down approach would be better. Adding the FPS to an existing Q&A database increases the chance of users getting answers from the Q&A database. If the developer of the Q&A system is willing to modify questions in the database using the Final Patterns, the chance for users to get answers in the database would be further increased.

9. REDUCTIONS

In a sentence, usually there are words that are unimportant and can be dropped without changing the meaning significantly. For example, "are there really ......", "my ...... is very ......", "show me some ......" Here words like "really", "very", and "some" are unimportant words. The patterns can be reduced to "are there ......", "my ...... is ......", and "show me ......". For example, if the answer in the database to the question is "are there ghosts", users can still retrieve the answer if they ask "are there really ghosts" because the word "really" is deleted from the user's question through reduction before pattern matching. One thing needs mentioning is the reduction of the articles "a", "an", and "the". These three articles are deleted from users' questions before pattern matching.

10. DISCUSSION

Years of work have been put into creating the FPS and its database. With a myriad of English language patterns collected and included, the system can understand users' questions at a fairly comprehensive level. However, the system is not perfect. The FPS is pattern-based, driven wholly by linguistic rules. You cannot claim it includes all the English language patterns. From time to time, users can find a pattern that is not included in its database.

The FPS is not developed to deal with Winograd schema questions, the kind of questions that can be answered easily by humans with common sense, but difficult for machines to answer. For example, "Paul tried to call George on the phone, but he was not in." It is obvious that George was not in, but this would be difficult for a chatbot to understand who was not in, Paul or George. In the same way, the FPS is not always helpful when users ask follow-up questions, especially when it involves a pronoun.

Further study of the FPS method is needed. Presumably the FPS will improve the understanding of natural language questions, but to what extent? How accurate are the answers? The questions like these are not fully answered yet. The human language is so complicated. In its current form, the FPS method has not fully dealt with the depth and breadth of language complexity.

While the FPS method is still under development, the framework has been built and new patterns are being discovered and added as part of ongoing efforts to improve the system. Meanwhile, chatbot developers or some educational training systems will continue to benefit from the way it saves them time in building new chatbots or question answering systems. Combined with new technologies developed in Artificial Intelligence, the FPS method will enable the development of breakthrough techniques in furthering the ability of robots to handle human-like conversations.

[1] B. J Grosz, Smart Enough to Talk with Us? Foundations and Challenges for Dialogue Capable AI Systems, **Computational Linguistics**, Vol. 44, No. 1, 2018, pp. 1–15.