university of groningen / faculty of science and engineering / bernoulli institute

Knowledge Technology Practical Final Project

Mirkó Borszukovszki (s4745000), Levente Kis (s4765389)

18-01-2024

Contents

1 Problem

The task of our knowledge system is to give method and bait recommendations for inland fishing. While fishing may seem like a straightforward recreational activity, the success of it depends on complex strategic choices. For those new to the sport and the less experienced, even the method and bait selection can be quite ambiguous. We aimed to build a system that can give guidance to novice anglers based on a few evident questions about the chosen fishing environment. This is a relevant problem for improvement because in Hungary, a country of ten million people, there are 800,000 registered fishermen according to our expert.
Becoming a seasoned fisherman takes years and decades of gaining experience through trial and error. This experience includes knowledge of environmental factors like cold and warm fronts, biology and habits of fishes as well as where and when to fish. We learned from our expert that even the best working colour for baits changes with the lighting conditions.

2 Expert

The expert of our knowledge system is Péter Makay, Levente Kis' godfather. He is the co-owner and administrator of the Fishing World fishing store in Budapest, he has been working there since 2015. He has been actively fishing for over 25 years, even attending competitions occasionally. His favourite branch is big carp fishing, but he is experienced in practically all methods of inland fishing.
Since Péter is from a landlocked country he has not been able to master saltwater fishing. Consequently, our knowledge system is made for only inland/freshwater fishing. We were able to meet in person one time with the whole group and Péter. Otherwise, he was always available through phone calls when we had questions about the project.

3 Role of knowledge technology

Fishing is an activity where experience is crucial to getting good results. Veteran anglers know countless tips and tricks that even though seem unimportant, could be the difference between catching and not catching fish. The type, color and smell of the bait, the location and countless other factors influence the success of the fishing trip. If someone is new to this sport, they have no idea about any of these. This is where our system comes in handy. The average angler can now use the knowledge of an expert with more than 25 years of experience and they only have to answer a few basic questions. The implicit reasoning about the choice of a particular bait is made explicit in the knowledge system in the form of simple rules.

4 The knowledge models

4.1 Rule model

Our rule model is presented in an xml file using the rules in the format that is presented below. The first type of rule is used for the inference engine's internal chaining processes and the recommendation rules are used at the end of the chain to keep track of the final recommendations.

```
<!-- Rule for inference steps -->

<rule>
    <if>
        <air_pressure>decline</air_pressure>
```

```
    </if>
    <then>
        <oxygenlevel>low</oxygenlevel>
    </then>
</rule>
```

Listing 1: Example of rule used in the .xml file

```
<!-- Recommendation rule -->

<recRule>
    <if>
        <water>running water</water>
        <waterTemp>below 10C</waterTemp>
        <front>neutral</front>
        <time>day</time>
    </if>
    <then>
        <technique>flyfishing rod</technique>
        <bait>fly</bait>
        <technique>match rod</technique>
        <bait>maggot</bait>
        <technique>feeder rod</technique>
        <bait>corn</bait>
    </then>
</recRule>
```

Listing 2: Example of a recommendation rule used in the .xml file

As in the example, a list of preconditions can be satisfactory for multiple recommendations meaning that more than one type of fishing method and bait are suitable for the given body of water. In that case, the system displays all of them.

4.2 Domain model

Based on our expert's advice we chose the four most important features that determine what kind of fishing method and bait is recommended. These are summarized in table 1.

Question	**Options**		
Are you fishing at running water or standing water?	standing water	running water	
Is the water temperature below or above 10C?	below 10C	above 10C	
How did the temperature change in the previous days?	getting warmer	getting colder	did not change
Are you trying to fish during day or nighttime	day	night	

Table 1: Questions and Options

Apart from the table above, the system uses other rules that contain knowledge elements that are not visible to the user. This was implemented to capture the expert's thinking process in the rule model.
We found these questions to be quite descriptive as all the 24 (2*2*3*2) combinations of answers lead to different recommendations.

4.3 Inference and problem-solving model

We decided to use forward chaining in our inference engine because we thought that most amateur anglers do not have a specific fish in mind that they would like to catch, they would just want to experience catching any type of fish. Forward chaining displays all the recommendations that can be deduced from the rules so it was perfect for our application.

5 User interface, functionality

The user interface was written in Python using the tkinter library, which makes it convenient to display option menus that are used for every question. Apart from that, the user interface only makes use of basic text labels, buttons and displays images in a gridlike fashion. All of the GUI elements are encapsulated in one "Questions" class. Additionally, the program makes use of two other python files. The first one is used for parsing the .xml file containing all the rules that are in the form of the examples presented in section 4. For this, we used the xml.etree.ElementTree library. The rules are stored in the form of dictionaries. This is convenient because this way, our knowledge base becomes modular and is decoupled from the rest of the code. The second file that we used is for doing forward chaining on our knowledge base. It applies all the possible rules until there is no new fact that can be deduced. The algorithm distinguishes between facts and recommendations and when a recommendation rule like in section four is applied, it stores the recommendation in a separate list, making it easy for us to display all the recommendations at the end.

6 Walkthrough of a session

6.1 Answer questions

When starting the application, the user is presented with a question about the environment where they are trying to fish.

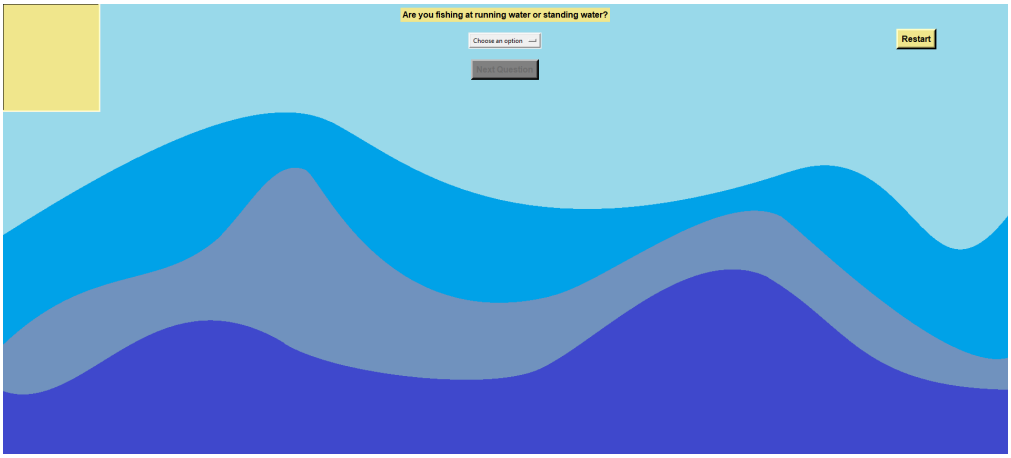

Figure 1: Initial user interface

6.2 Perform chaining

For every question, you can select from the list of options and when you select one, the "Next Question" button becomes green and clickable. After clicking on it, the next question appears with

a new set of options. The user answers are recorded in a list and added to the knowledge base. Before displaying the recommendations, the system processes the .xml file which contains all the rules from our expert. These rules are stored as a list of dictionaries where the key is the list of conditions and the value is a list of consequences. Then based on the answers of the user and the rules in the knowledge base, the system performs forward chaining and displays all the possible recommendations that can be deduced from the rules.

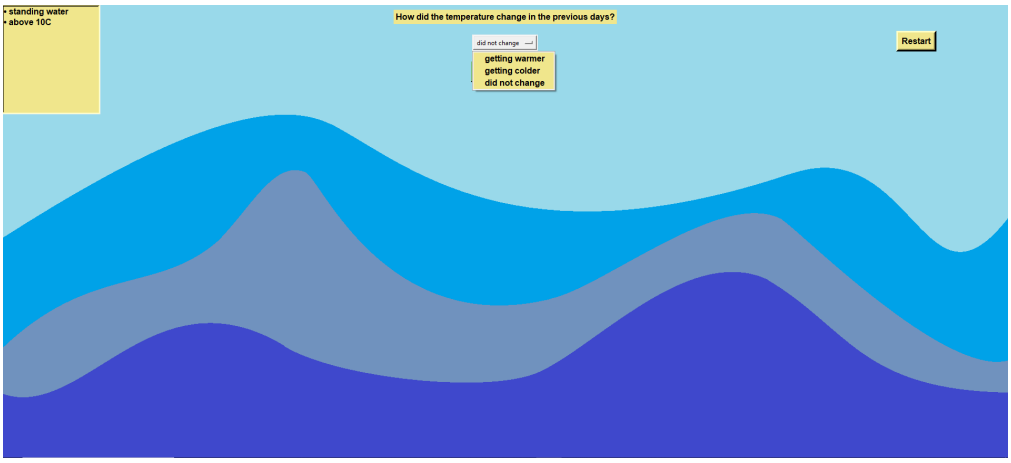

Figure 2: Selecting an answer to questions

6.3 Display recommendations

After all the questions have been answered, the system displays a list of pairs of recommendations. The first element in the pair is the type of fishing method that is recommended (this specifies what kind of rod you should use in this environment) and the second is the type of bait you should use for it. These are presented not just in text but also in image form.
In the top left corner, all the user's answers are recorded and displayed in list format and in the top right corner, there is a "Restart" button which can be clicked at any point during the session and it restarts all the questions and clears the options that have been selected until that point. The restart button empties the knowledge base, the list of recorded answers and resets the entire GUI by destroying all the widgets that were displayed.

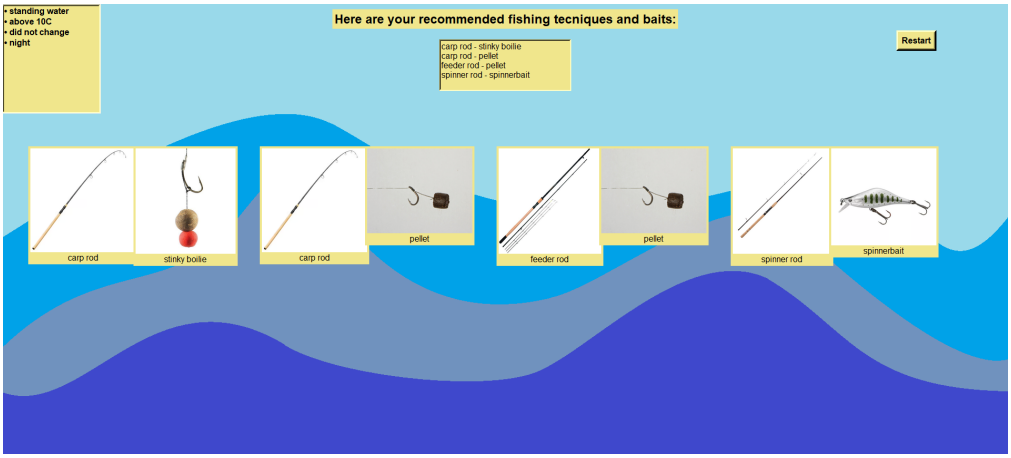

Figure 3: The system displays recommendations

7 Validation

During the development of the system, our expert was easy to reach and fast to respond. We were able to ask questions while working on the system when we were not certain about a choice being made. However, during the validation session, there were still a few things that our expert pointed out for improvement. We changed three pictures for baits as Péter suggested more representative ones and helped us choose them. The previous pictures did not have the position of the bait on the hook, only the bait by itself. Before the validation session, our system gave at most two methods and bait pairs as recommendations to avoid confusing the user. Péter suggested that the system should give as many recommendations as possible. He pointed out that more recommendations are a sign of better environmental conditions and that is important to learn for new anglers. In addition, not seeing their preferred fishing method may scare away less experienced fishermen from using it even if the conditions are suitable for it.

8 Task Division

Most of the tasks were done together and the design choices made were discussed. We made the initial system together then Mirkó finalized the code and made the GUI for the system. Questions for the expert were written by both of us. Interviews with the expert were done together. When we only had a few questions for the expert, they were asked by Levente through phone calls or texts. We made the knowledge model together. The report was written by both of us.

9 Reflection

Before this course, we had both encountered the world of fishing on a surface level, which is why we chose this topic. We found the experience of making the knowledge system engaging. How passionately our expert talked about the minor but really interesting details of fishing made us look at fishing from a new perspective.
The most challenging task was making a knowledge model from the vast amount of knowledge our expert has about fishing. What to include and what to leave out was hard to decide. Coming up with the right questions was a challenge for us because we were not familiar with the aspects of fishing method and bait selection. However, once the system started to come together the work was easier.
Our system could be expanded in multiple ways. More questions could lead to more specific recommendations. For example, if one could specify the kind of fish they wish to catch. These extra questions would make the system useful for the more experienced fishermen. However, they would also make the system more difficult for a beginner. Another way the system could be improved is by adding recommendations for saltwater fishing, this opens a whole new world of fishing.
In conclusion, we learned how an expert domain can be transformed into a knowledge system. Despite the challenges, this way of collaboration was rewarding. We were able to use our theoretical knowledge for a system we could see people use in everyday life while also experiencing a look into an expert's mind.