**A Text-Based Dialogue System using AIML**

**Daniel Christopher Peric 25811460**

**Andisheh Partovi 22999825**

# The procedure

1. We started off by reading the provided link. We didn’t do any further readings, however, during the evaluation of the first assignment, we had to go over a lot of electricity articles.

2. We had several conversations with each other by imagining different scenarios and different paths the dialogue may take, always assuming the student character. These sample dialogues are provided in the appendix.

3. After that, we combined the distilling phase with implementation and while adding all the dialogues to the AIML file, we removed duplicates, tried to combine similar questions or answers and added alternative ways of saying the same questions.

Based on these dialogues and given our limited time, we decided to do three sub-topics, viz. fridge, Xbox, and solar panels by adding relevant questions and answers and also steering the user towards those topics through other mechanisms (see the Handling Different Phenomena Section).

We then abstracted out some keywords for each of these subtopics and added common synonyms of the keyword using a thesaurus dictionary.

Given the nature of AIML, we tried to merge as many questions as we can into one and took a greedy approach with responses. For instance, any question that includes the word “pay” will be responded with the amount of summer bill and winter bill. This greedy answer that tries to give as much information as they are relevant in one go, is very helpful in reducing the number of questions that must be answered, however, it sometimes make the answers mechanic and non-organic.

We used binary variables to keep track of discussed topics. Once a topic is discussed, its flag variable is set to true so the bot doesn’t neither initiates another talk on the topic nor allows the user to bring it up. A limitation of AIML is that, these variables should be initialised in a category and if for any reason that category isn’t called (the user never asked the question in that category), the variables will not hold any value. We placed the initialisation in the “HELLO” category and by adding a greetings as the first utterance of the bot, tried to encourage the user to say a greetings and therefore initialise the variables.

# 4. We did the WoZ experiment over a shared desktop chat environment. The wizard used the AIML file

# Handling Different Phenomena

1. Policy and handling unknown utterance: system has a strict policy as any known utterance must be hard-coded in the system. we have a wildcard category that catches all the unknown utterances and based on the discussed topics, tries to steer the conversation towards topics that are already coded. For instance, if the user hasn’t discussed the fridge yet, the system returns a fridge-related interruption as the answer for any unknown utterance, hoping that this will prompt the user to talk about the fridge. When the first topic is done, it will move to the next, using cue phrases such as “by the way”. When all the three topics are exhausted, the bot will mention that he has to go.

2. Initiatives: We have taken a mixed initiative approach, as the bot must both ask and answer questions. Because we can provide the first utterance through the python driver, and we leverage that to encourage the user to greet the bot, the very first interaction is user initiative. After that, interactions will remain user initiative until the user brings up one of the devices in the bot’s house (the fridge or the Xbox), then the bot will take control of the conversation for two or three turns. Because this is a fixed rule, the system is not variable initiative. This mixed initiative approach is natural to this scenario, therefore we kept it this way.

3. Dealing with social obligations: The system is coded to understand near to 80 different ways of greetings and farewells. This shows the core limitation of AIML that needs all the alternative ways of saying the same thing, be coded in. The system also includes answers to thanking.

4. Grounding: using the recurrence tool (srai), continuer utterances such as “yes” and “right”, are ignored and mapped to whatever comes after them. We are losing some grounding information in this way, but these tricks are necessary in AIML. The bot cannot ask clarification questions either so basically we couldn’t find any effective way of grounding in AIML. Considering that the system is text-based, grounding is not as vital as spoken dialogue system.

2.    Pronouns: this cannot really be handled in AIML as it doesn’t perform any natural language processing. We tried to include pronouns in the categories we designed just to give an illusion of intelligence.

# The bot

We chose the student characteristic and coded several attributes for the bot such as his name, number of people living in the house, their power bill amount, etc. These can be found in the “*assignment2.py*” file.

# Language Limitations

There are many limitations in AIML, some of which were very hard to work with:

1. Not being able to initialise variables safely.
2. The “*that”* tag is very limited.
3. The conditionings are very basic. Nested if-else structures are not allowed.
4. No natural language processing
5. Being punctuation blind
6. No language statistics or similarity measures

# Evaluation

AIML has many restrictions and these restrictions showed during the evaluation most uses had some issues many had a lot of issues and only one user was able to complete the conversation successfully to a reasonable length. The most successful feature was the use of dialogue acts and trying to catch digressions to steer the conversation onto topics the bot had information for. These are used in place of saying unhelpful things like “I do not understand, please rephrase” and would either ask or infer that the person should continue or hint at a topic that the person may like to talk about.

We have tried to use a naïve initiative by guessing secondary answers that will be given after questions are asked, leading answers were also given to give hints to the user so that they understood what areas the bot has knowledge in. This worked haphazardly and depended mostly on the way the user spoke to the bot, a conversational English was expected but some users used a more formal or less formal English and the bot failed to recognize any words from them. A thesaurus may have helped with word understanding or similarity.

The approach taken worked much better on smaller conversations and one out of the eight users had no trouble talking to the bot. The bot was approximately 80% correct and this was found to be too low and users were generally unhappy with the performance (rankings can be seen in figure 1). Even though User 4’s experience was good they still lacked confidence in the system and did not feel comfortable interacting with it; this was because the answers, while technically correct were not on topic but were graceful recoveries.

|  |  |  |  |
| --- | --- | --- | --- |
| User 1 |  | user 2 |  |
| conversation length | 26 | conversation length | 12 |
| bad utterances | 61.54% | bad utterances | 16.67% |
| good utterances | 38.46% | good utterances | 83.33% |
| user 3 |  | user 4 |  |
| conversation length | 19 | conversation length | 12 |
| bad utterances | 66.67% | bad utterances | 0.00% |
| good utterances | 33.33% | good utterances | 100.00% |
| user 5 |  | user 6 |  |
| conversation length | 4 | conversation length | 33 |
| bad utterances | 0.00% | bad utterances | 17.65% |
| good utterances | 100.00% | good utterances | 82.35% |
| user 7 |  | user 8 |  |
| conversation length | 30 | conversation length | 31 |
| bad utterances | 20.00% | bad utterances | 25.00% |
| good utterances | 80.00% | good utterances | 75.00% |
|  |  | Totals |  |
|  |  | conversation length | 132 |
|  |  | bad utterances | 19.70% |
|  |  | good utterances | 80.30% |

Figure 1



Figure 2a



Figure 2b

# Appendix I – Sample Human Dialogue

The following dialogue was recorded on a Google chat between the team members.

-Hi

-I am from a charity and I want to help you save money with your electricity bill

-save money, how?

-we have some general tips. but by asking some questions, I can get a better sense of your usage

-Oh, Ok that sounds cool

-so yeah one tip is unplug your phone after it's charged

-so should I unplug the whole charger or the phone

-just the phone is enough

-Ok

-Another tip is to check your fridge temperature. Make sure it's not too cold, especially if it's almost empty.

-Brrrr, my fridge is cold! What temp should it be set to

-between 0 and 5 degrees

-what is the correct temp for milk that is what I keep in it mostly

-let me google it for a second...between 1 and 4 should be good

-ok, thanks

-so how many people live in this household?

-2

-and how long each person spends at home on an average work day?

-14 hours

-what about weekends?

-all day

-ok so you never go out?

-seems that way. I work from homwe

-ah ok. so you should expect a big bill.

-no we jsut dont put on the lights at night we have LED lights

-cool. Those are actually more efficient.

-yes

-so just apply these tips and you will be alright

-OK thxs

-ok. Have a nice d