Justin Hardy

Benjamin Frenkel

CS 4395.001

Dr. Mazidi

Chatbot Project Report

I. System Description

This Chatbot was designed to answer various basic questions about Texas Hold'em Poker, as well as to hold basic conversations with each of its users. The Chatbot was made 100% from scratch using a Machine Learning approach. The machine learning model that was used for this project was Logistic Regression, and various NLP techniques were used. This includes, but is not limited to text tokenization/detokenization, sentence negation, POS tagging & parsing, and word vectorization. These techniques were used at various points in the project for various reasons. POS tagging, for instance, was used to identify the subject of a sentence when discussing one's likes and dislikes. This isn't perfectly implemented but should work for a good chunk of sentences expressing interests.

In creating the model, at first we'd opted to use a combination of unigrams, bigrams, and trigrams in order to help train the model to recognize more sophisticated sentence structures. However, we'd created our knowledge base by hand, so we opted to only use unigrams to better generalize off of the small amount of data the model was being trained off of. The model will attempt to tag an input sentence, which will then be used as a lookup in the knowledge base for a suitable response to statements/questions of said tag. Obviously, the model is not perfect at tagging sentences, and will often times mis-tag sentences. This is for a number of reasons, such as that the model wasn't trained to recognize a specific sentence structure, or the model couldn't decide which two tags best fit the sentence structure.

II. Dialog Tree

A dialog tree was made prior to the implementation of the chatbot itself, which was made in an excel sheet. Below is a screenshot of the excel sheet. This gives a rough overview of how dialog is handled by the chatbot. In some instances, the chatbot must remember things previously stated, so a history of the chat session is kept helping facilitate its features.

For the most part, this tree closely represents the actual implementation. However, there are some instances where the actual implementation is missing parts of what is shown in the tree. For example, sentiment analysis is only partially implemented in the project. The model will try to map known negative and positive words to like/dislike expressions but does not actively attempt to detect sentence sentiments. This feature is essentially missing entirely from life-related conversational topics. Also, life-related conversational topics may not all be recognized, however inputting something such as “I just recently got a new dog” will be picked up and recognized as sharing a life-related event.

III. Sample Interactions

Here are some sample interactions with the chatbot. The first two are related to interests, and the last three are poker-related questions.

Interaction 1:

>How's it going?

I'm fine! I recall you said, "There's a lot to play" about gaming. Is there anything else that you like about it?

>I think it's a good pass-time.

Thank you for sharing, Justin!

Interaction 2:

>Pizza tastes pretty bad.

What don't you like about pizza?

>It's too gooey.

That's good to know.

Interaction 3:

>How is poker played?

The goal of a Texas hold'em game is to use your hole cards in combination with community cards to make the best possible five-card poker hand.

>What is a community card?

Three cards are dealt to the center on the flop, then 1 on both the turn and river. Everyone shares these cards.

Interaction 4:

>What is the best strategy to use on the flop?

There are many different considerations one must make on the flop. A couple include when to cbet, sizings, what category your hand is, and what type of flop it is.

>*Define "flop".*

The first 3 community cards dealt after preflop betting is completed.

Interaction 5:

>When should I go all in on the river?

It's important to note your sizings, and when you should lead, bluff, and bluffcatch on the river.

>Okay, so when should I lead?

Leading in the river involves not checking to the aggressor from the turn and instead betting into him, this is typically a good idea in certain types of spots where the aggressor from the turn will check a lot and you have value hands that can get called by hands the opponent will never bet if checked to.

IV. Knowledge Base Tags

Tag
unknown
chatbot
conversation_smalltalk_history
conversation_smalltalk_howareyou
smalltalk_greeting
gratitude
conversation_user_interests_like
conversation_user_interests_dislike
conversation_user_life
conversation_replies*
conversation_confused
conversation_no
poker
rules
stakes
where
strategies
phases
live
online
cbet
call
fold
raise
blinds
big blind
small blind
positions
bluffing
valuebet
handcategories
handrank
draw
preflop
flop
turn
river
cards
holecards
communitycards
preflop_strategies

preflop_hands
preflop_raise
preflop_call
preflop_3bet
preflop_4bet
flop_strategies
flop_cbet
flop_sizing
flop_handcategories
flop_types
turn_strategies
turn_cbet
turn_handcategories
turn_types
river_strategies
river_bluffing
river_bluffcatch
river_leading

* = machine learning model not trained to tag

The above table lists each tag that the machine learning model will try to categorize the input text by. For testing, this should make it more clear what sorts of things you can ask the poker chatbot. At the beginning of the program, the knowledge base (*kb.json*) is read in, and the machine learning model is trained on the patterns specified by the knowledge base to map inputs to tags. Outside of the model, the responses available in the knowledge base are mapped to the tags in a dictionary and used in the program for live lookup. After creating the model, it is pickled, and if the program is run again, the program will ask if the user would like to use the pickled model instead of creating a new one.

V. User Model Appendix

The user model is a python object that simply holds the user's name, as well as their likes, dislikes, and any general life information. Each time the program is run, the user is asked for their name. If they input a name that has been used before, the program will assume it is the same person they talked to before and load up their pickled user model. However, if the name is new, the program will create a new user model. After the program is exited (via the exit inputs), the program will pickle the user model's state for future reference.

VI. Chatbot Evaluation

Given that the chatbot was made from scratch, I feel as though it performs decently well for the hand selected data we gave it. The accuracy of its tagging could definitely see improvement, however, as it is its biggest weakness. In general, it'll tag things decently well, but there are some sentence structures it absolutely struggles with, and often times it will think you're asking for an explanation on one thing, when you really want an explanation on a different, albeit extremely similar thing. Although, this may be a fundamental flaw of the tagging approach. But using deep learning over logistic regression may have been a better idea for improving tagging accuracy.

I think the biggest strength of this poker chatbot is that it delivers poker information fairly well. If you ask it to define various poker terms, it will always do so. If you ask it for more information on when those terms apply to the game, it will always do so. I think its biggest weaknesses lie with the handling of the user model. That being said, all functions of the chatbot are usable, but one may encounter issues depending on ambiguity of the input sentences.