Meet Percy: CS 221 Teaching Assistant Chatbot

Sahil Chopra, Rachel Gianforte, & John Sholar

CS 221: Artificial Intelligence: Principles and Techniques

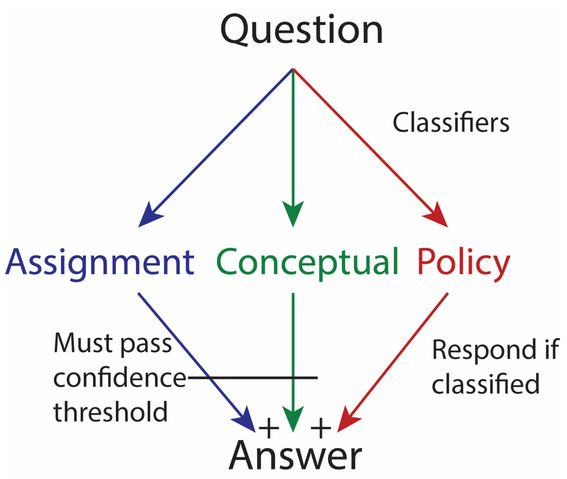
Overview

While in-person office hours are crucial in helping students receive unique in-depth explanations, the questions on Piazza are fairly repetitive and are meant for mass consumption. Leveraging this fact, we attempted building a TA chatbot to answer CS 221 Piazza questions.

Dataset and Processing

- Scraped Piazza data from CS 221 + CS 124
- Cleaned ~1000 questions
- Hand labeled ~ 1000 questions into three categories – Assignment, Conceptual, and Policy
- Converted PDF of Russel and Norvig's *Artificial Intelligence: A Modern Approach* into a text file and cleaned artifacts introduced by file conversion.

Chatbot Model



Approach

Question Types

Question Type	Description	Example
Assignment	Pertains to specific homework assignments from the class.	"I am receiving the following output probabilities form my Bayesian Network … What could be wrong?"
Conceptual	Pertains to artificial intelligence queries in general.	"How is value iteration different from policy iteration?"
Policy	Pertains to course logistics.	"How do I sign up for an alternative exam?"

Piazza posts for computer science classes typically pertain to assignments, concepts, or course policies. CS 221 questions were labeled accordingly.

Question Classification

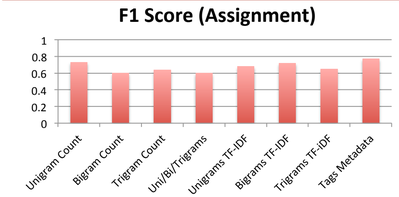
- Utilized Linear SVM to categorize questions as one of three categories.
- Feature extractors included unigrams, bigrams, trigrams, TF-IDF, and matches to hand written regular expressions.
- Replaced linear classifier for policy questions with regular expressions such as "office hours" or "alternate exam". Binary classifier misclassified policy questions into the assignment category, due to heavy bias in the data towards assignment questions.
- Instead, we prompt user to categorize the question for us.

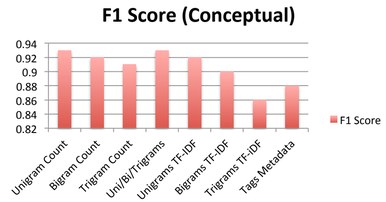
Question Answering

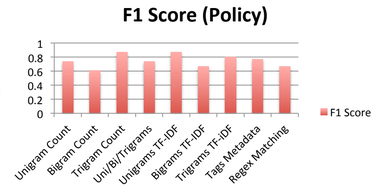
- Assignment – Compute TF-IDF representations of input question and (question, answer) tuples from the training set. Score previous answers according to weighted sum of cosine similarity between input and a question, answer pair from training set.
- Conceptual – First, compute TF-IDF on sections of the textbook to find the most similar section. Then compute TF-IDF on individual sentences to return the most similar paragraph from the textbook.
- Policy – Return a prewritten answer, depending on regular expression match.
- Logistics Database – The chatbot is also capable of accepting updated information (e.g. a change in location for office hours) directly via the CLI. User input is matched intelligently (using bigram and regular expression matching), and relevant updates are provided in addition to a standard reply.

Results

Question Classification







Unforunately, we did not have enough data to train our classifier successfully. The dataset was too small and had too much bias, as an overwhelming majority of questions were assignment related.

Question Answering

Question Type	Input	Output
Assignment	In class we discussed using the Monte Carlo method to help approximate the evaluation function. Will this work in this case since each time we call the evaluation function, we have a different game state?	I think the feedback backpropagates for a game like chess where the the only (and big) reward comes from moving a state to the terminal state (checkmate). The Qopt and Vopt recurrence would feed into each other such that you can infer rewards all the way back up to the start of the game.
Conceptual	I do not understand the motivation behind SARSA and other bootstrapping methods in the context of model-free learning. Why is it important to obtain feedback quickly if it is not used to modify the policy online?	For this reason, Q-learning is called a model-free method … As in the ADP learning agent, we can use this equation directly as an update equation for an iteration process that calculates exact Q-values, given an estimated model This does, however, require that a model also be learned, because the equation uses P (s# | s, a)
Policy	Will Code from lecture be posted?	Slides are posted on the website and lecture videos can be found at https://mvideox.stanford.edu/Course/787

Future Work

- Gather a larger dataset to properly train the classifiers.
- Apply K-Means or unsupervised learning for question classification.
- Develop online learning system that allows user to to mark returned answers as useful, increasing the probability they will be returned in the future.
- Implement SIRI-like slot filling system to help improve fact understanding and response.