Zhuohang Li(zl299)

# CS 520 Final: Question 5 – The Big Picture

I pick the **PharmacistBot**. As a pharmacist, the bot should be able to:

1. Listen to and understand the patient.
2. Navigate itself though the shelves to get the right medicine or drug that the patient asked for.
3. Posses knowledge about different kinds of medicines.
4. Advise the patient about safe medicine use e.g. which two kind of drugs cannot be taken together.
5. Identify the patient's symptoms. Give patient tests if necessary, e.g. blood pressure measurement.
6. Based on the patient's symptoms, advise the patient about what drug should they take, and the dosage.
7. Make more sells! (Otherwise the pharmacy will close down). This means that it should be able to find alternative for drugs that they don't have at their pharmacy.

Issues that would be faced in and potential algorithm solutions:

1. First it needs to be able to move and navigates through the shelves inside the pharmacy in order to fetch the correct medicine that the patient needs. Simply store the layout of the pharmacy into its memory won't be sufficient because the layout might change, the location of drugs might change, and most important thing is that the obstacles in the pharmacy is dynamic. There might be people standing somewhere in the pharmacy and you cannot just run them over. So, we need to apply a path-finding algorithm to the bot, for example A * search algorithm. Comparing to DFS, A * can often provide a shorter path without having to cost as much computation and storage resource as BFS. The layout of the pharmacy can be represented in a 2-dimensional matrix, people and shelves in the pharmacy can be viewed as blocked, and the shelf where the intended drug locates would be the target point, and we could use the physical distance as the heuristic function.

2. To be able to identify different kinds of drugs or medicines, we will need some knowledge of computer vision. For example, we could train a CNN model to work as a classifier to separate the 26 letters in English alphabet, so that the bot can read the name of the medicine. Comparing to general neural network, CNN is better for image processing. An image is usually composed of thousands of pixels and the weights matrix for each NN layer would be really big. Propagating through those layers would consume a large amount of time. The weights of CNN on the other hand are shared by each layer, plus, we could use pooling as down sampling strategy to further reduce the computation cost. The CNN should take images of letters as input, and after some combination of convolution layer, pooling layer and FC layer, finally reaches the softmax layer where it votes towards a final recognition result of which letter it is. The training process requires a large number of labeled images data that needs to be collected in advance. We could also utilize CNN to enable the bot to recognize different trademarks through a similar procedure.

3. In order to understand the patient, the bot needs to possess ability of Nature Language Processing. For the part-of-speech tagging and name entity tagging, we could train a Hidden Markov Model. For example, the recorded sound signals are the observation sequence, and the information that patient is trying to convey is the hidden state sequence. We could apply forward and backward algorithms to get the probability of the observation sequence and use Baum-Welch Algorithm to estimate its parameters.

4. The most important ability of the PharmacistBot is that it should be able to give suggestions to the patient about what drug they should take after knowing their symptoms. We could use Bayesian Network. We could setup a basic BN with some theoretical knowledge of the rate, causes and symptoms. By asking the patient a few questions, the bot can make inference about the patient's disease and choose efficient drugs based on that. And as the number of patient cases grows, we can update the BN based on that. For example, if the patient is sneezing, he may either catch a cold or a flu. If the patient has been with another flu patient, he would have larger probability of catching flu than a cold. In this network, sneezing is related to cold or flu, and flu is related to having been with another patient.

5. To find alternatives for a drug, we could use clustering algorithms. We could represent different drugs in the format of a vector of features, where the features can be the proportion of different ingredients, the indication for that drug, the suggested dosage and so on. For the distance function, we could choose from Minkowski distance, Manhattan distance, Hamming distance, Euclidean distance, and Chebychev distance. For clustering algorithm, we can use DBSCAN instead of K-means so that we don't have to know the number of clusters in advance. If the two drugs are in the same cluster, we could consider they are somewhat fungible.

6. In order to give the robot the ability to know which two kinds of drug cannot be taken together, we can use a logical knowledge base. For example, we can first equip the knowledge based with some pre-stored information like aspirin and ibuprofen can not be taken together, drug X contains ibuprofen and the patient has just taken aspirin. If we query the knowledge base for P: the patient can take drug X, if {KB, P} is unsatisfiable, that means P is false, the patient cannot take drug X. If {KB, $\neg P$} is unsatisfiable, that means P is true, the patient can take drug X. If both are satisfiable, output whether the patient can take drug X is inconclusive.

Bonus:

i) Luna dressed as superman for Halloween.

ii) Luna Rose the Superdog: (Please forgive my horrible drawing skills)