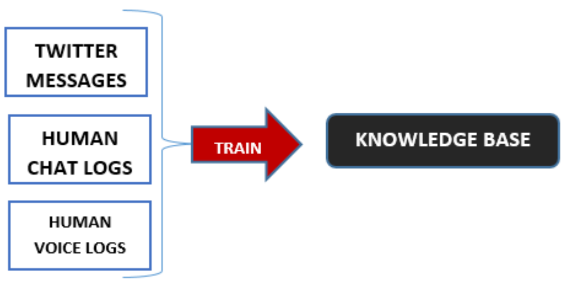
KBAI MidTerm

KBAI Techniques Leveraged:

Fames, Inductive learning, Case Based Reasoning and Concept Learning

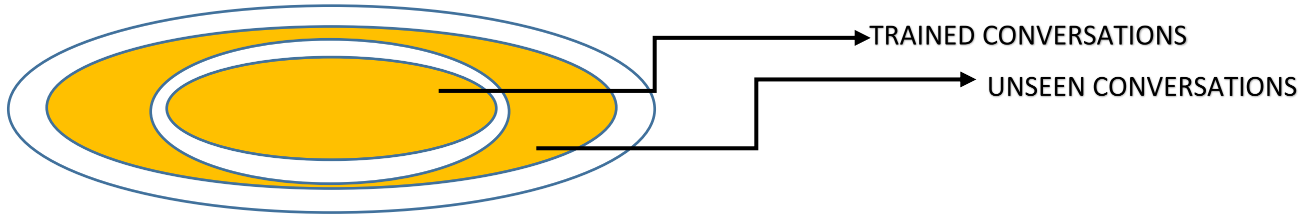
Problem Space Organization:

We are about to design an AI agent that will automatically post messages on a forum as a reply to a user's message. Such an agent can have a multitude of applications to enabling users accomplish a task to providing customer service. The agent will undergo a knowledge training phase where we will be synthesizing pre-existing knowledge to build the agent's knowledge base. The knowledge base is a repository of knowledge representations that the agent will exploit to perform reasoning to solve the conversation problem. The sources from which the input (Natural Language) is organized might include twitter conversations related to the problem domain, human chat logs, lexically encoded voice logs (as the voice issue is allowed to be assumed as resolved). Once the training phase is complete, the agent is deployed to the user environment and the agent begins to respond to the user's questions.



As with any computational learning task, we will be making some assumptions on the family of problems the agent will be tasked to solve to define the scope of generality of the conversational agent. This means that the data that is sourced to build the knowledge base during the training phase and the data that is presented by the user after deployment will be part of the same distribution. An agent trained on providing customer service can respond ***similar*** yet unseen conversations as long as they can be classified as belonging to the overall family of customer interaction conversations. I have chosen to illustrate with an example of an agent that will answer customer service requests.

`



Knowledge Representations, Composition (Questions 1 & 2)

Background Information

Dependency Structured Matrix (DSM) is a specialized representation for a constrained hierarchical graph which we will be using to build our conversational semantic network. Let's understand the basics of this knowledge representation:

System		1	2	3	4	5
CONVERSATION A	1	.		X		
CONVERSATION B	2	X	.	X	X	X
CONVERSATION C	3	X		.		X
CONVERSATION D	4	X			.	X
CONVERSATION E	5					.

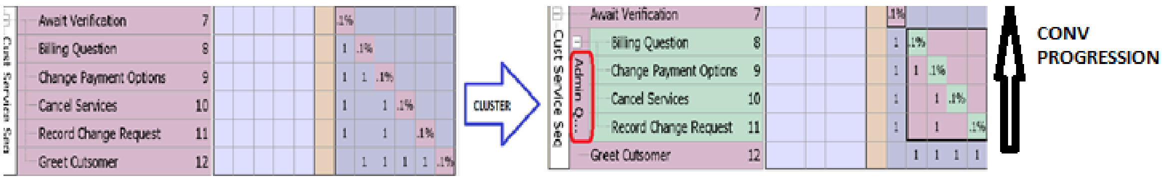
This figure shows a sample knowledge base has been decomposed into five conversations: conversation A, conversation B, conversation C, conversation D, and conversation E. The antecedent for conversation A are in column 1 which show that conversation A follows on conversations B, C, and D. This is shown by placing an X in rows 2, 3, and 4, which corresponds to conversations B, C, and D's rows. All of the indicated in the above DSM are as follows.

- ***conversation A*** *follows on conversation B, C, and D*
- ***conversation B*** *follows on No other conversations*
- ***conversation C*** *follows on conversation A and B*
- ***conversation D*** *follows on conversation B*
- ***conversation E*** *follows on conversations B, C, and D*

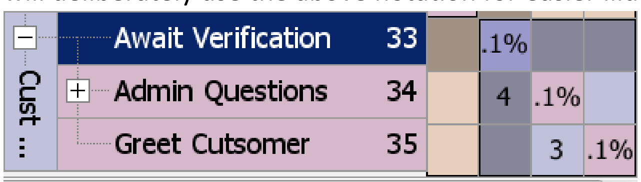
The basic idea behind design of this knowledge representation is to disambiguate the sequential position of an uttered sentence in the context of other sentence attributes such as nouns, verbs, phrases etc. Such an isolation enables representing conversational knowledge which is key to context sensitiveness of the agent.

Conversational Knowledge:

More critically, the DSM semantic network representation is conducive for performing **density** based clustering to create clusters of strongly connected conversations. As mentioned earlier, conversations are sourced from several instances of conversation threads between any dual combinations of people over the years. ***Hence density based clustering for maximizing connectedness of dual conversational threads imbibes the knowledge of all the people's utterances at a specific sequential depth in the conversation.*** This enables the agent to synthesize customized context labels (analogous to hash tags) that describe the level of specificity/generality of several people's conversations. In the below example, we show a DSM semantic network that lists various customer service conversations it has held. The data source does not attribute rich categorizations of the data to tune their specificity of application. The process of clustering automatically isolates a context that generalizes Billing, Change Payment, Cancel services and Record change request into **AdminQ .** A simple "graph reachability" algorithm that operates on the 1 values in the matrix and attempts to bring the ones under one cluster, would accomplish creating the new conversation abstraction.



While the above matrix can be transformed into nodes and edges to avoid O(n2) space consumption , we will deliberately use the above notation for easier illustration of downstream processing

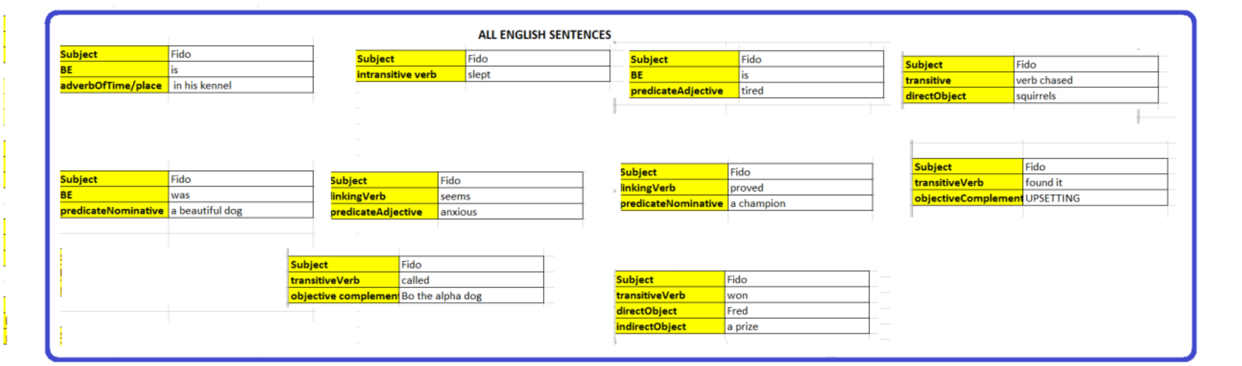


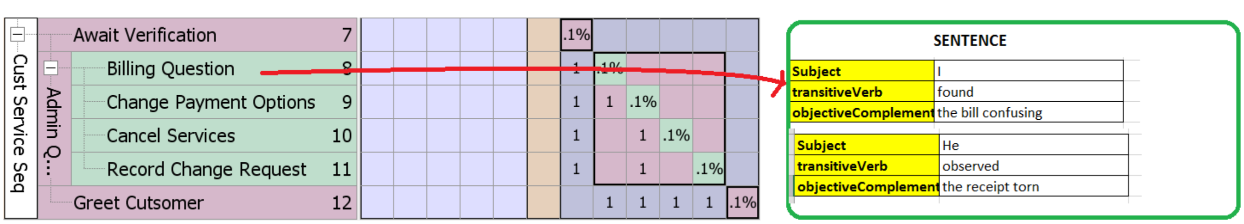
Looking at the above DSM Semantic Network, **"Await Verification"** has most tasks in Admin Questions as its antecedent, which was probably why the verification was pushed to the subsequent phase.

Sentential Knowledge

Ten Sentence Pattern

We will be representing frames for each sentence by using the ten sentence property of sentences in English. Every sentence in English can be ***phrase-structured*** as shown below. The yellow slots have values for each phrase-attribute associating with the utterance. The intent of the below diagram is to show that any sentence in English can be represented in any one of the 10 forms.We assume that such patterns can be generated using NLP parsers like shift-reduce.





The above diagram illustrates the complete knowledge representation we are leveraging

Each conversation has one or more associated frames that indicate the various sentences. That can be associated to a specific conversation topic. For example, the billing topic has one or more sentences clubbed together. Let's examine the values and understand why the semantic network brings the two sentence frames under a single cluster.

"I found the bill confusing" and "***He observed the receipt torn***"

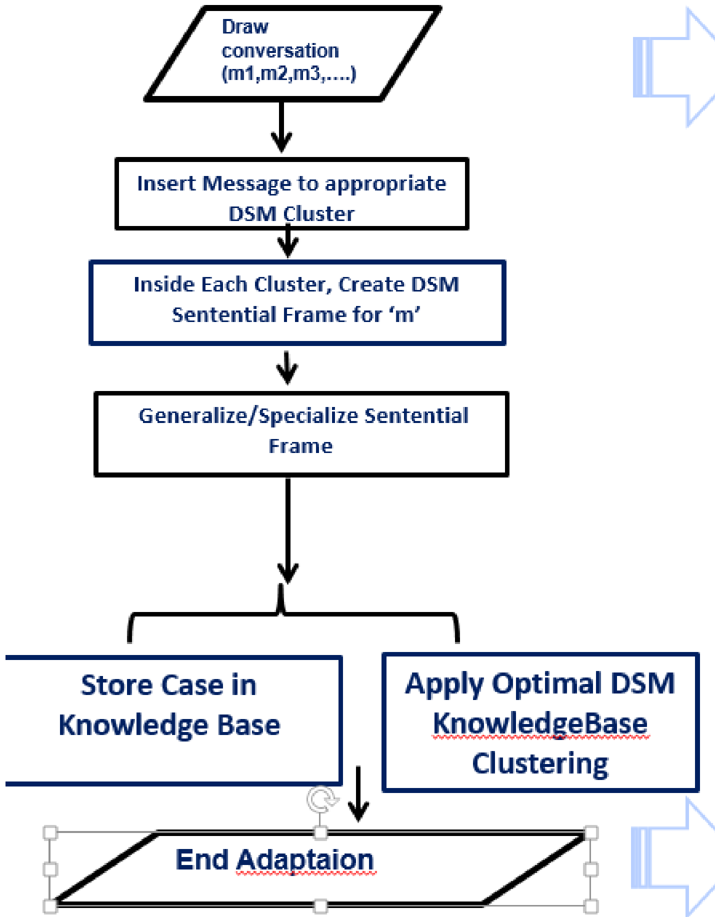
4 out of 5 words mismatch between these two sentences. However, the agent discovers the inherent similarity between the two sentences owing to the following factors:

1) They occurred in the same sequential depth of the entire conversation (right after the greetings). This is observable solely due to the fact that we maintain sequential order and cluster based on them. If the order of the conversations is not exclusively represented then the order may risk not getting weighted in a clustering process among other less significant attributes. The knowledge representation hence solves this specific problem solving goal.

2) The phrase structure is similar and transitive verbs and object complements contain more synonymous (NOT exact words). It is assumed that the agent has the option to look up synonymous words and hence form the association between these frames. Hence the knowledge representation supports the robustness by attributing grammatical phrase structure to the stored sentence. If not we would be associating frames based on matching words (not sentential contexts). We would be associating the paper "bill" and the person "Bill" without this knowledge representation

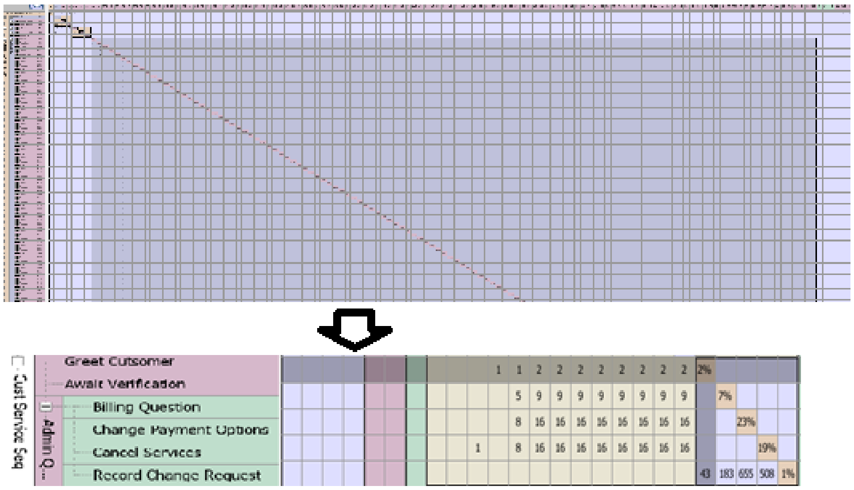
Incremental Knowledge Base Build-Up, Case Storage & Agent Understanding (Q5 ,Q6,Q9)

The below algorithm approximately sums up the process involved in building up a knowledge base of cases. The algorithm updates the knowledge base with each conversational training record that we have sourced.



Case Adaptation By Modularization

There arises a need for modularizing the structure of the semantic network in the graph as more complex conversations are stored. We apply density clustering algorithms to aggregation generate labels for each of the conversation to synthesize conversation context as explained earlier. But the larger issue is the choice of the clustering algorithm that we must apply to a given semantic network. On a graph sense, the clustering algorithm examines edge dependencies between a graph and assigns a connectedness metric. Some algorithms may target reachability, some may try to isolate entities. Hence, there arises a need to maximize the modularity of the knowledge base (see diagram below) for the conversation context to be more explicit.

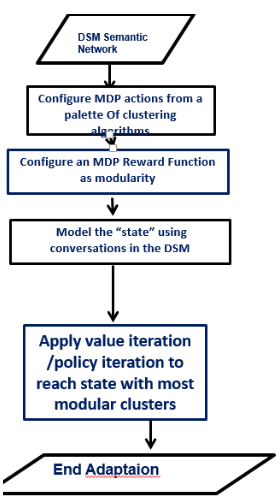


(Modularized DSM Semantic Network)

The idea of modularization/clustering is to enable the agent by itself derive generalizations in contexts of conversations. This is analogous to human cognition where we synthesize "generalized" knowledge from experience. The complexity arises out of the fact that such generalizations here are not performed on words but conversations accumulated from various sources, actors and timelines.

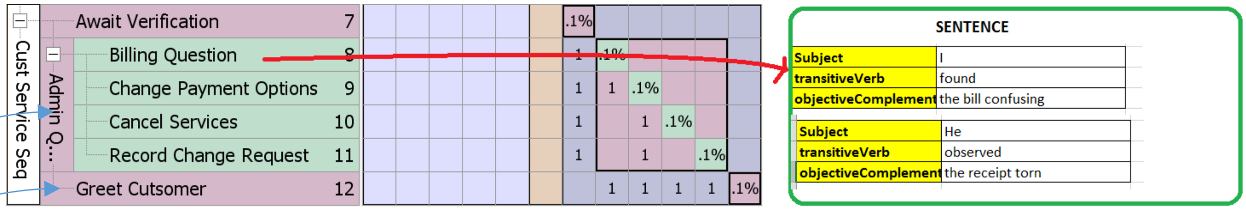
Modularization Implementation

Although modularization offers benefits, unless the right clustering algorithms is used to choose the labels of the DSM nodes, we would likely not achieve the expected autonomy of the agent to generalize. In order for such modularization to be effective, we need to set up a problem that examines the entire state of the DSM after every clustering operation, backtrack if necessary and apply the most optimal clustering algorithm that can maximize the modularity of the DSM semantic network. We might use a technique as stated below(Markov Decision processes) to maximize modularity in the DSM Semantic network



All of the above processes are aimed at attributing complex structure to natural language input to emulate human cognition of perceived conversations. ***Hence, such steps increase the understanding of the agent***

Sentence Generation



Lets assume that we have the knowledge shown in the screenshot as part of the knowledge base. Now we have to answer a query as

Agent: Hello How are you ?
Customer: I am fine. Thank you.

What happened to my bill ?

For the above question, the agent lands on the right sequential depth of the conversation (thanks to the DSM and its clustering maintenance). In this context, "bill" seems to hash this to a Billing Question Equivalent. The Agent surveys the answer node and returns the inference as the answer.

Agent: It is (appears twice) for the bill to be torn or **confusing**

Agent: It is quite common for the bill to be torn or **confusing**

There are more complex scenarios that are left unhandled for now. The decision tree for sentence generation might involve examining the sentence type of response to be provided