

Agent utterances in the dialogue

No Author Given

No Institute Given

In this document, I'll present the different cases of agent responses to a user utterance. I'll first define the utterances arguments to make the document readable.

- Lets x be a value of a criterion. Thus, we note for this document the fact of stating preference on x means either $p(x, i)$ or $p(i, x)$, where $x \neq y$ and $x, i \in \mathcal{D}_C$.
- We define the value y such that $x, y \in \mathcal{D}_C$.
- We define the value z such that $x \in \mathcal{D}_C$ and $z \notin \mathcal{D}_C$, which means that x and z represent the value of two different criteria.
- O is an option.

1 State Preference

In the dialogue, the user might state a preference on a value of a criterion named X . In this case, the agent can response with the utterances defined in the Fig. 1.

1. State.Preference(x): This is the most accurate response that I found in the example dialogue with lauriane and leonor. For example,
 - leo: state.Preference(Breton, *).
 - laur: State.Preference(*, Breton).
2. State.Preference(y): the agent can answer with its preferred value for the same criterion. For example,
 - User: I like the most japanese cuisine.
 - Agent: I like the most indian cuisine.
3. Propose (X): We can imagine that the agent prioritize the user preferences. Thus, is the user prefers x the agent would propose it. For example:
 - User : I like the V arrondissement.
 - Agent: Lets choose the V arrondissement.
4. Ask.Preference(Y): The agent cas ask the user to give more information about its preferences.
 - User: I like the most japanese cuisine.
 - Agent: Do you like indian cuisine?

2 Ask Preference

When the agent receive an ask.Preference utterance, it can use the following utterances which are illustrated in Fig. 3.

1. State.Preference(x): This is the most logical response is to response with the agent preferences on x . In the dialogue with lauriane and leonor, I found this example:
 - leo: Ask.Preference(Italian).
 - laur: State.Preference(Italian, *).
2. Ask.Preference(Y): Assuming that the agent is very submissive, he can prefer ask for the user preference on x rather than expressing its preference.
 - User: Do you like indian cuisine ?.
 - Agent: Do you like indian cuisine?

3 Propose

I distinguish here when the propose has as attribute a x or an option O . Take first the case where an option is proposed and, I assume that $V(O, C) = x$.

1. Reject(O): The agent can reject the proposal.
 - leo: Propose(africain).
 - laur: Reject (africain).
2. Accept(O): The agent can accept a proposal.
 - leo: Propose(africain).
 - lau: Accept(africain).

In this example of lauriane and leonor dialogue, we can notice that lauriane

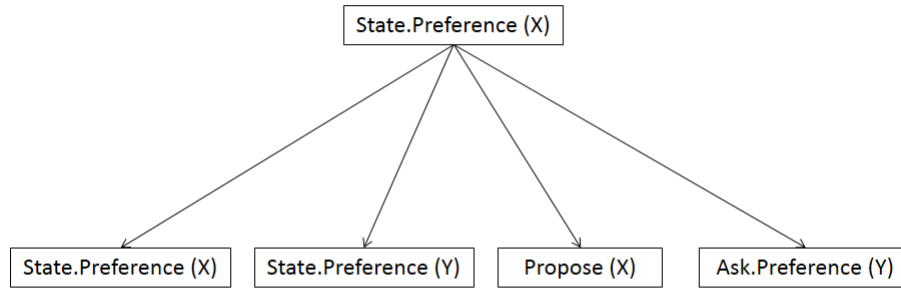


Fig. 1: state preference response.

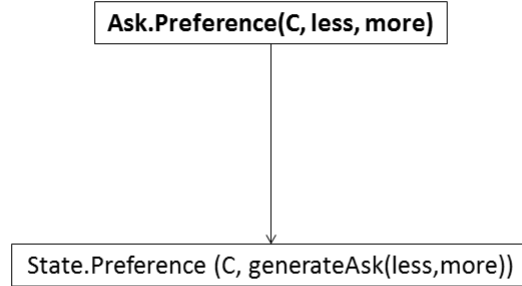


Fig. 2: agent possible answers to an ask Preference

accepts africain knowing that she rejected this proposal earlier in the dialogue. This can be explained by the fact that leonor used arguments to convince lauriane to like africain, and lauriane clearly said it "*Mais africain me tente bien, tu m'as bien vendu le truc*". For the moment, we are not able to express argumentation.

3. Propose (X): the agent can counter propose with only a value of a criterion x . For example:
 User: Lets go to Ginza restaurant
 Agent: Lets go to a french restaurant.
4. StatePreference(x): We can assume that the agent is not dominant enough to feel at his ease to express a reject. Therefore, he expresses its preference on the value that he doesn't like. For example
 User: Lets go to Ginza restaurant
 Agent: I don't like chinese cuisine.

The agent responses to a value are very similar to the option. I separate them to be easier to explain.

1. Reject(X): The agent can reject the proposal.
 - User: Propose(japanese).
 - agent: Reject (japanese).

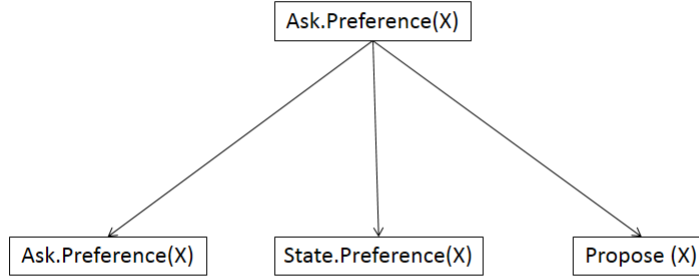


Fig. 3: ask preference response.

2. Accept(O): The agent can accept a proposal.
 - User: Propose(cheap).
 - Agent: Accept(cheap).
3. Propose (y): the agent counter propose with another value y . For example:
 - User: Propose(japanese).
 - agent: Propose (italian).
4. propose(O): the agent can counter propose with an option, like in the dialogue example where lauriane counter proposed leonor proposal with a restaurant proposal.
 - leo: Propose(Aveyronnais)
 - laur: propose(CheeseClub)
5. State.Preference(x/y): We can assume that the agent is not dominant enough to feel at his ease to express a reject or an counter proposal. Therefore, he expresses its preference on the value that he doesn't like to invite the user to reject this value, or to state his preference for something that he likes to invite the user to propose it. For example

User: Lets go to a chinese restaurant

Agent: I don't like chinese cuisine.

Or

agent: I really like french cuisine.
6. Ask.Preference(y): Debate wether model an ask here or not. I think that we don't have to keep it.

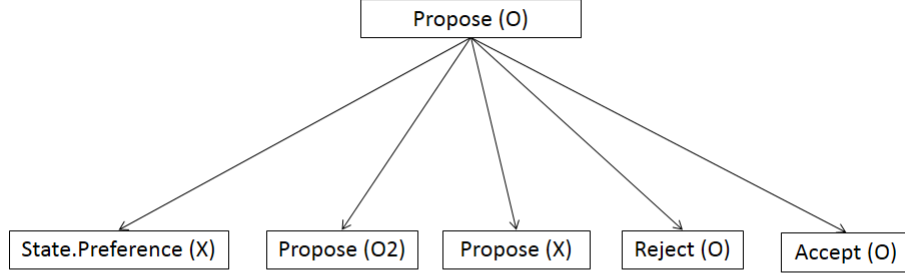


Fig. 4: agent possible answers to a proposing an option

4 Accept

If the user make an accept for an option O . It will close the dialogue because the main goal is satisfied. Whereas, accepting a value for a criterion may open the dialogue for the resting of the criteria or options.

1. propose(O/Z): the agent can either propose an option where $v(O, C) = x$ or propose a value for another criterion.
 - User: Accept(Aveyronnais)
 - Agent: propose(SAVEURS D'AVEYRON)
 Or
 Agent: propose(cheap)
2. the agent can move to the negotiation on other criteria. Thus, it can express his preferences or ask the user preferences on theses criteria.
3. State.Preference(z) for example
 User: Accept(Aveyronnais)
 Agent: State.Preference(calm, noisy): I prefer calm ambiance more than noisy ambiance.
4. Ask.Preference(z): For example
 User: Accept(Aveyronnais)
 Agent: ask.Preference(calm) : do you like calm ambiance?

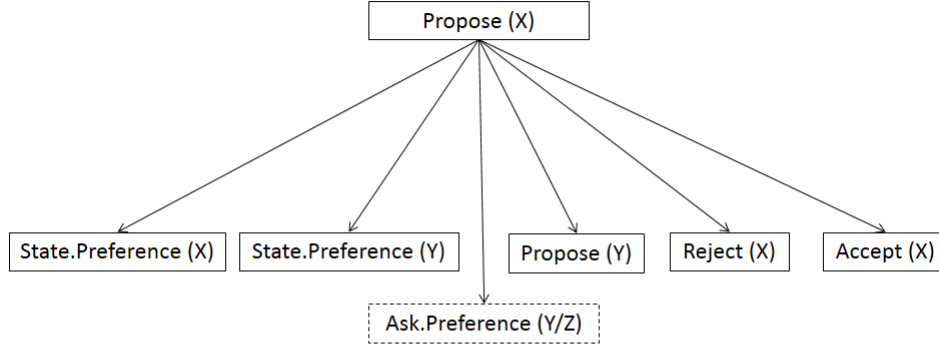


Fig. 5: agent possible answers to a propose value

5 Reject

5.1 Reject an option

When the user reject an option, I thought that it might be interesting to add an utterance to ask him why he rejected this option. I think to a simple utterance where the user can tell us for example that he doesn't like the cuisine or the ambiance of this restaurant.

1. `propose(O_2)`: the agent can propose another value to the user. For example
 - User: `Reject(cheeseClub)`: No, I'd rather choose another restaurant.
 - Agent: `propose(chez chuck)`: Let's choose chez chuck!
2. `propose(O)`: the agent can propose the same restaurant again because he's very dominant

warning: Loop, we have to stop the agent to make the same proposal several times.
3. `Propose (y)`: the agent can also propose another value y . For example:
 - User: `Reject(cheeseClub)`. where $(v(\text{cheeseClub}, \text{cuisine}) = \text{fromage})$
 - agent: `Propose (italian)`.
4. `State.Preference(z/y)`: the agent can state its preferences on other values for the different criteria. For example

User: `Reject(cheeseClub)`
 agent: I like the most french cuisine.

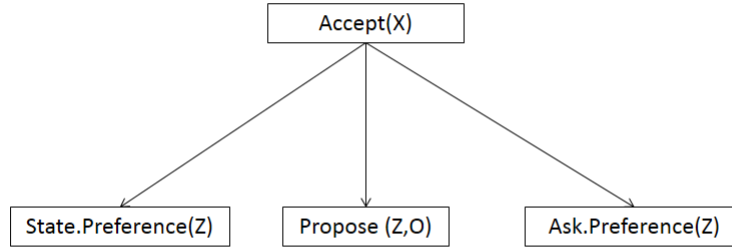


Fig. 6: Accept utterance

5.2 Reject a value

1. propose(O): the agent can propose an option where $v(O, C) \neq x$. For example
 - User: Reject(italian): No, I'd rather choose something else.
 - Agent: propose(chez chuck): Let's choose chez chuck!
2. propose(x): As explained before the agent can propose again the same value because he's very dominant, but we have to avoid a loop.
3. Propose(y/z): the agent can also propose other values. For example:
 - User: Reject(cheeseClub). where ($v(\text{cheeseClub}, \text{cuisine}) = \text{fromage}$)
 - agent: Propose (italian).
4. State.Preference(x): We can assume that the agent really like x and in order to report to the user, he express his preference. Or in the contrary, to support the user reject, he can state that he doesn't like x either. Example:
 - user: Reject(Italian): I'd rather choose something else.
 - Agent: State.Preference(Italian, *): I like the most italian
 - OR
 - agent: State.Preference(*, Italian): I like the least italian
5. State.Preference(z/y): the agent can state its preferences on other values for the different criteria. For example
 - User: Reject(italian)
 - agent: State.Preference(French, *): I like the most french cuisine.

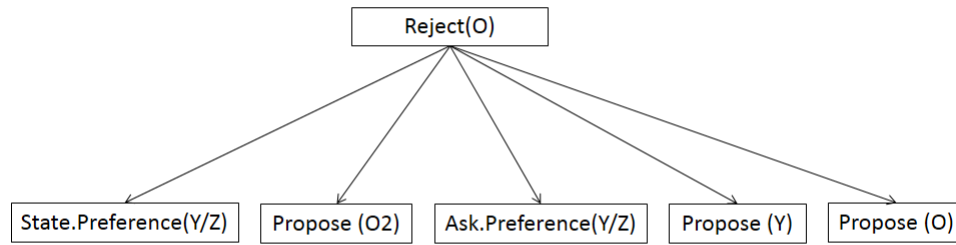


Fig. 7: agent possible answers to reject a an option

6. Ask.Preference(z/y): the agent can also ask for the user preferences in order to make the right proposal next time.
User: Reject(italian)
agent: Ask(French, *): Do you like french cuisine?

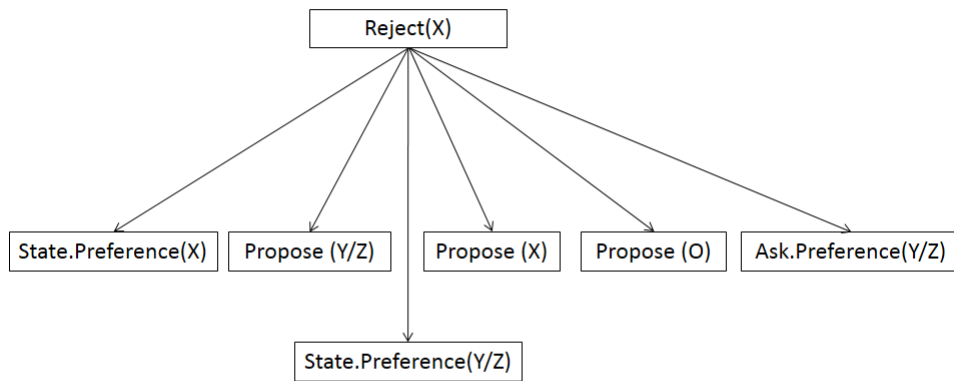


Fig. 8: agent possible answers to reject a value