Inferring Personality Types for Better Automated Negotiation

Sai Naveen Pucha(✉) and Praveen Paruchuri(✉)

International Institute of Information Technology, Hyderabad, India
sainaveen.pucha@research.iiit.ac.in, praveen.p@iiit.ac.in

Abstract. Automated negotiation between computational agents or between agents and humans has been a subject of active research with a focus on obtaining better quality solutions within reasonable time frames. The critical issue negotiators face during automated negotiation is that a negotiator may not always know the personality type of the opponent. Studies show that having information about the opponent improves the outcome of negotiation in general. However, unless there is prior knowledge, learning the opponent type in the limited amount of time or number of rounds in a negotiation is a difficult task. In this paper, we use a Partially Observable Markov Decision Process (POMDP) based modeling to perform better modeling of the opponent personality type. In particular, we focus on modeling the opponent into four different types to showcase that a better understanding of personality type can improve the outcome of automated negotiation. Our experiments performed using data sets generated from the IAGO software showcase that we indeed obtain better negotiation outcomes with a higher classification accuracy of the opponent personality type.

Keywords: Automated negotiation · Personality type · Belief tracking · POMDP

1 Introduction

The topic of automated negotiation has received wide attention in the literature due to its importance as one way of communication between agents and also between agents and humans [3,17,18]. In a general automated negotiation, an agent may need to negotiate with other agents or humans whose (personality) type may not be known [2,24]. However, knowing the types of other players can provide significant advantages to the agent in many situations in terms of having a better idea of how to approach or tailor the negotiation [6,9,10], potentially leading to better outcomes. The offer-counteroffer paradigm [8,12] is a popular way of modeling automated negotiations. In this study, we deviate from the offer-counteroffer paradigm since the other negotiators can take actions for information exchange and not just to make offers and counteroffers.

In the business world, people bring different negotiation styles and strategies to the bargaining table, based on their different personalities, experiences, and

D. C. Morais et al. (Eds.): GDN 2020, LNBIP 388, pp. 149–162, 2020.
https://doi.org/10.1007/978-3-030-48641-9_11

beliefs about the negotiation. When people with different negotiation styles are involved in a negotiation, the results can be a lot more unpredictable. On knowing more about the other parties, negotiation styles can have a significant impact on the way we plan our negotiation. Individual differences in social motives and preference for certain outcomes when agents interact can strongly affect how they would approach the negotiation [5]. Drawing on social motives that drive human behavior, [25] and others identify four basic personality types:

- **Individualists** concentrate primarily on maximizing their outcomes and do not show much concern for opponent outcomes.
- **Cooperators** focus on maximizing both their own and other's outcome. Cooperators tend to opt more for value creation strategies such as exchanging information than individualists.
- **Competitives** are motivated to maximize the difference between their own and other's outcome. Because of their strong desire to "win big", competitive agents tend to engage in behavior that is self-serving.
- **Altruists** are a rare breed of negotiators that strive to maximize other's outcome rather than their own. Although very few people may be pure altruists, most human negotiators behave altruistically under specific conditions such as when dealing with loved ones or with those who are less fortunate.

Given the above classification for personality type of negotiators, our goal is to build an automated negotiation agent that can adjust its belief regarding the type of opponents as the negotiation progresses and chooses appropriate actions to reach a high-quality negotiation agreement. There is quite some work on modeling and prediction of opponent type in negotiation literature [1,2] including usage of formal models such as game theory [4,7] and Bayesian learning [26,27] among others. However, there is relatively less work on analyzing the personality type of an opponent within the computation model of negotiation. POMDPs [11] have been used earlier in literature for modeling social interaction among humans in situations where information of the opponent has to be known [21]. More recently, preference elicitation in negotiation using the Gaussian uncertainty model has been developed in order to optimize the negotiation outcomes [14,15]. [22] presents a POMDP based model for the development of a strategic agent for human persuasion via the usage of argumentative dialogs.

The assumption we make in our work is that initially, the agent has no (or little) idea of the type of the other parties (hence we assume a uniform distribution over the possible types). Throughout the negotiation, the agent receives feedback and refines its belief accordingly. Given the characteristics, we would like to capture, prior work has shown that POMDP based modeling can be a good fit [11]. We focus on negotiation between two agents in the rest of the paper and present the POMDP framework for our negotiation problem as the next step, along with details of how it encodes our problem.

The rest of the paper is organized as follows: Sect. 2 presents details on how we model the Automated Negotiation problem as a POMDP. Section 3 describes usage of IAGO software to obtain a dataset that post some processing is used as input for the POMDP model. Section 4 performs a basic evaluation of our

POMDP model. In particular, the section compares the POMDP agent against corresponding MDP models with differing assumptions, performs reward function evaluation, and also studies the effect of misclassification of opponent type. Section 5 presents the experimental setups used and the results obtained. In particular, the section shows results on learning of opponent personality type in terms of classification accuracy, obtained via the POMDP belief updates. Section 6 presents the conclusions of our work.

2 Partially Observable Markov Decision Process Framework

Formally, POMDP can be defined by the tuple $\{S, A, T, \Omega, O, R\}$, where S is a finite set of states; A is a finite set of actions; $T(s,\, a,\, s')$ captures the probability of transitioning from state s to s' when taking action a; Ω is a finite set of observations; $O(s',\, a,\, o)$ is the probability of observing o when taking action a leads the agent to state s' and $R(s,\, a)$ represents the reward function, i.e., the reward obtained by taking action a, at state s. As part of modeling the POMDP, we also need to specify an initial belief where a belief state b is defined as a probability distribution over the set of states S. Once the negotiation problem is cast into the POMDP framework, many algorithms both heuristic and exact exist in the literature to find an approximate or optimal POMDP policy [11,20]. Note that a policy here refers to a mapping from a belief state b to an action a, for all the possible valid belief states.

2.1 Encoding a Negotiation Problem in the POMDP Framework

Using the ideas presented in prior work [18], we encode the negotiation problem into a POMDP tuple $\{S, A, T, \Omega, O, R\}$. As has been noted in prior works [18,19], the advantages of a POMDP-based modeling approach for negotiation are as follows: (a) POMDPs provide a natural way to capture the sequential nature of the negotiation process while reasoning about the new data observed (such as the actions of the other agent). (b) POMDPs enable to model and refine an agent's belief about other agents (in this work, the belief is over the personality type of the other agent).

A key issue we face here is that the negotiation transcripts typically have utterances, while a POMDP needs specific actions to be defined. To make the mapping of negotiation transcripts to a POMDP feasible, we first assign a set of codes to the actions of the players expressed in terms of dialogues [18]. Each utterance can correspond to one or more codes, and these codes form the action set that can be taken by each player. The POMDP would, therefore, be modeled in terms of codes. We now provide details of the POMDP encoding for a two-player negotiation.

2.2 State Space Definition

The state space of our POMDP has the following factors: <Type, MyProposal, OpponentProposal, MyAction, Observation, PreviousObservation>.

- **Type:** Modeled four personality types.
- **MyProposal:** The bid which the agent has to offer.
- **OpponentProposal:** The bid which the other player has to offer.
- **MyAction:** The action the agent has taken.
- **Observation:** The action which the other player takes, received by the agent as an observation.
- **PreviousObservation:** Observation of the agent in the last time step.

2.3 Action and Observation Set

There are two action categories from which the agents can choose. One category of actions is Information actions, while the other is Proposal actions. Information actions are those which do not involve a change in proposal values. Proposal actions, on the other hand, are used for changing the proposal value each agent makes. The negotiators can interact with each other using dialogues. The dialogue set that we use for experimentation is limited and is already a part of the IAGO software. The players get to pick their responses from this set.

We used the standard K-means clustering with tf-idf (term frequency–inverse document frequency) score as weights [23] to categorize each of the dialogues into 5 (code) categories. The tf-idf is a measure of the importance of a particular word with respect to the other words in a document. We assign the following five (action) codes to each of the five clusters generated by the K-means algorithm:

- **a1 - information** - This is a simple information exchange action.
- **a2 - same** - This indicates that the agent has the same value for the proposal again.
- **a3 - concede** - This means that the agents has conceded its proposal value.
- **a4 - agree** - This action indicates that the agent has agreed to the value/bid proposed by its opponent.
- **a5 - finish** - This action indicates that the negotiation has been completed/terminated.
- **a6 - NOOP** - This action has no impact on the negotiation process.

The key advantage of using action codes is that it makes the interaction amenable to being modeled using a POMDP due to their limited number. For this experiment, the rate at which the agent and the players concede is the same. The reservation value for the agent is dependent on its personality type and sets lower bound for acceptability of bid. Observation set (Ω) has the six codes that were part of the action set since observation for the agent is an action of the other player. In addition to the set of six codes, IAGO software also provides the emotion of the other player, which we model as part of the observations. In particular, we use four different emotions, namely Happy, Sad, Surprised, and

Angry, hence forming 20 action-emotion pairs as the observation set plus NOOP which leads to the following 21 observations in total: $a1 - happy$, $a1 - sad$, $a1 - angry$, $a1 - surprised$, $a2 - happy$, $a2 - sad$, $a2 - angry$, $a2 - surprised$, $a3 - happy$, $a3 - sad$, $a3 - angry$, $a3 - surprised$, $a4 - happy$, $a4 - sad$, $a4 - angry$, $a4 - surprised$, $a5 - happy$, $a5 - sad$, $a5 - angry$, $a5 - surprised$, $a6$

2.4 State Transition Function

The transition function represents the probability with which the agent reaches state s' when it takes action a from state s. The agent can take different actions from the action set, which can either be a proposal action or an information action. Upon taking these actions, the agent will make transitions to different states. The transitions in this domain are stochastic since information related to other players is part of our state, which we would not know beforehand.

2.5 Observation Function

The state in our POMDP model is a tuple, as described in subsect. 2.2. We follow the sequence of steps presented in [18] that show that when the state captures the previous observation, the observation function becomes deterministic. For explanation purposes, if ts is the current time step, we mathematically represent the observation function as follows [where $aa(ts)$ is the agent action taken and $oa(ts)$ is the opponent action taken at ts, i.e., the observation at ts]:

$$\begin{gathered}
pr(obs(ts)|s(ts+1), aa(ts)) = pr(obs(ts)|s(ts+1)) \\
= pr(obs| < Type, MyProposal(ts), OpponentProposal(ts), \\
MyAction(ts), Observation(oa(ts+1)), PreviousObservation(oa(ts)) >) \\
= pr(obs(ts)|oa(ts)) \\
= 1,\ if\ obs(ts) = oa(ts) \\
= 0,\ otherwise
\end{gathered}$$

2.6 Reward Function

The reward function is based on the personality type of the player. We introduced four personality types earlier, and each personality type has its reward function. As mentioned in subsect. 2.3, while emotions are part of the observations, in the current work, we do not condition reward values on the (observed) emotions. More specifically, the reward function for each personality type depends on whether a concession is made or not. For the Individualist, Cooperator and Competitor personality type, making a concession is not preferable for the agent; hence it is given a lower reward value. Please note that the meaning/value of concession by the same amount is different across the (3) personality types; hence the reward value is modeled accordingly. For example, between the Individualist and Cooperator agents, Individualists may never want to concede, but Cooperator will tend to concede at some time steps. For an Altruist, a concession is viewed as a positive since it helps its opponent, hence receives a high reward for a concede action.

2.7 Discount Factor

The value of the discount factor can vary from 0 to 1 in general. A discount factor close to 1 indicates that rewards in the distant future are of high priority while a value closer to 0 indicates that only immediate rewards are considered. We use a discount factor of 0.95 for our purposes.

3 Input Generation for the POMDP Model

We used the IAGO software [16] to obtain the input dataset. We then perform a processing step over this dataset to obtain state vectors in the form defined in Sect. 2.2. Please note that the dataset is not labeled on the personality types we model in this paper. We therefore went through the data set and labeled the data to include the personality types, e.g., if a player agent repeatedly agrees to a proposal made by the other agent without much bargaining involved, it is labeled as an Altruist. The state vectors of the POMDP are then built directly from the negotiation transcripts generated from the IAGO software.

4 Evaluation of the POMDP Agent

4.1 Sanity Check Experiment

In order to check whether the POMDP model we built is reasonable, we designed 2 MDP agents, namely MDP1 and MDP2. MDP1 doesn't capture the opponent personality type, whereas MDP2 knows the type of the opponent.

Figure 1 shows the results of the experiment, where the POMDP agent is evaluated using the MDP2 actions. There is a mapping step involved in this evaluation, since there is no observation feature in the MDP2 states. We perform this mapping using the remaining features of the state vector. Hence, it translates to a many to one mapping where multiple POMDP states can map to a single MDP2 state in which case the same optimal (MDP2) action is used for these states. The figure shows the number of timesteps on the x-axis and the Estimated Expected Total Rewards EETR [13] on the y-axis. From the figure, we deduce the following:

- EETR values for the POMDP and MDP2 agents are similar when the POMDP is provided information about the opponent's actions. Figure 1 shows the EETR values for POMDP and MDP2, however the lines overlap. We therefore represent MDP2 using a dotted green line and POMDP by an orange line with the + symbol for purposes of contrast.
- MDP1 obtains lesser EETR values, since it cannot make use of additional information about the opponent.

EETR is calculated as follows using Eq. 1:

$$V_t^*(s) = \max_{\mathbf{a}} \left[R(s,a) + \gamma \sum_{s \in S} T(s,a,s') V_{t-1}^*(s') \right] \tag{1}$$

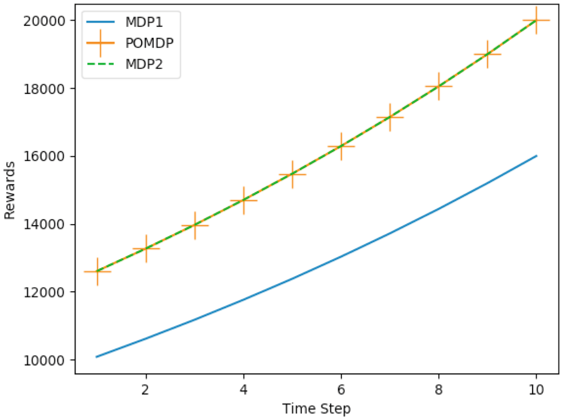
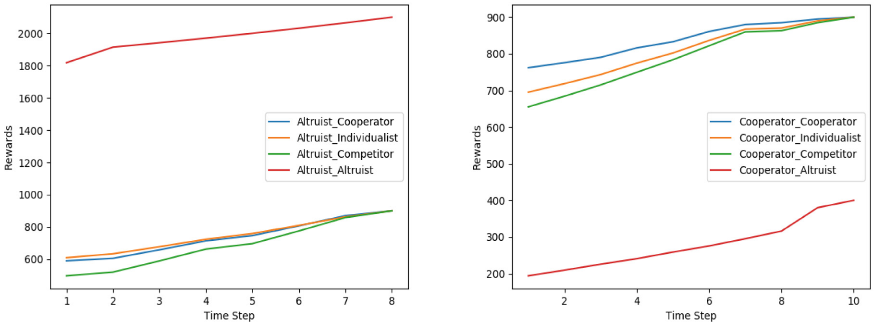


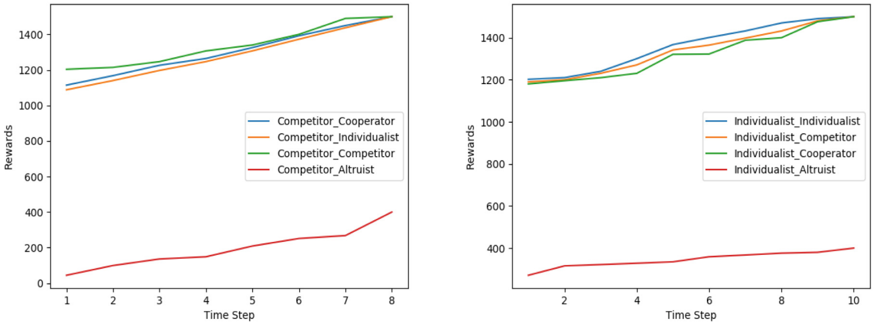
Fig. 1. Estimated expected total reward values for the computed policies of POMDP and the two MDP agents when given the same action sequences.

4.2 Reward Function Evaluation

The reward function is a component that is not present explicitly in the dataset, nor the data presents an obvious way to construct. As described in Sect. 2.6, the reward function is built, keeping in view the opponent types that need to be captured. With a well-designed reward function, the agent is expected to provide better EETR values if the reward function intended for that personality type is used. For example, if the POMDP uses the reward function of Individualist for an opponent who is Altruist due to incorrect modeling, it should be expected to return lesser EETR values than when it uses the correct reward function for Altruist. Figure 2 shows four plots, one per each personality type of the opponent. Each plot shows results for the experiment where we use the four different reward functions for a fixed personality type of the opponent [i.e., three incorrect and one correct reward function used]. Note that for purposes of this experiment, the state space of POMDP has only one opponent type information captured (instead of four as described above in State Space Definition), e.g., value for Type is Altruist in POMDP for plot (a), Cooperator in (b) and so on. The x-axis for each plot shows Time Steps, while the y-axis shows the reward obtained. From the plots, we can infer that when the correct reward function is used, the agent obtains better reward values as compared to when using incorrect one(s).



(a) Altruists with other functions (b) Cooperator with other functions



(c) Competitor with other functions (d) Individualist with other functions

Fig. 2. Each personality type tried with all the reward functions

4.3 Effect of Opponent Type Misclassification in Input Transcripts

This experiment while having similarities also differs from the previous experiment (on reward function evaluation) in the way we setup the experiment: In the previous experiment, we build POMDP with one value for (opponent) Type in the state space and then experiment by changing the reward function which should have been assigned for the other opponent types. In this experiment, we first make changes to the transcripts that are used to build the POMDP. In particular, we consider two setups here: Setup 1 where we (manually) change the opponent type in the first half of input transcripts where opponent type is Individualist and Setup 2 where we manually change the opponent type in the second half of transcripts where opponent type is Individualist, i.e., we introduce misclassification of opponent type in the input transcripts. For both the setups, we model two opponent types in state space i.e., value for Type in state space can be Individualist or Altruist. For both the setups we then perform three experiments: (a) Individualist correct i.e., no (manual) changes made to transcript and reward function used is the one that corresponds to when opponent type is individualist (b) Individualist Wrong as Altruists i.e., in the manually changed transcripts when Type is Altruist, POMDP continues to use Individualist reward since Altruist was introduced via a manual change (correct reward used when Type is Individualist) and (c) Individualist-Altruist correct (POMDP uses reward for Altruist when Type is Altruist).

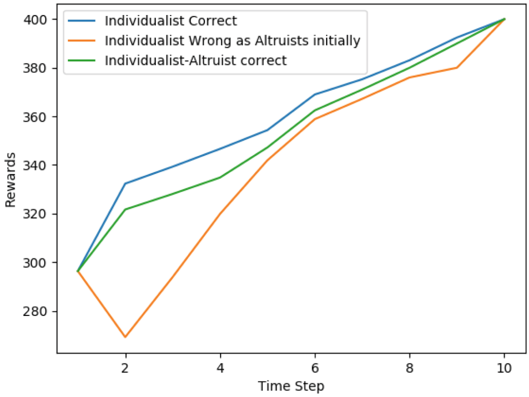


Fig. 3. Misclassification in the first half.

Figure 3 shows results for Setup 1, while Fig. 4 shows results for Setup 2. The x-axis for both the figures represent Timesteps while y-axis shows the EETR values. Both the figure shows that the orange line has the lowest EETR values due to the effect of type misclassification in input files as well as absence of correction in reward function, i.e., reward function used as if the misclassification was not present. The green line does better since, although misclassfication of opponent type was present, the reward function reflects this misclassification while the blue line performs best since there was no misclassfication in the input files (and hence no reward function correction is needed).

5 Experiment Results

5.1 Generation of Negotiation Data

We generate negotiation samples using the IAGO framework [16]. As described in [16], the negotiation samples are generated in IAGO when the player(s) negotiates with the IAGO agent. The IAGO agent can be part of a multi-issue bargaining task featuring four issues at five different levels. The agent utilizes the fixed list of utterances that the human may use, although it has its own set of responses. The agent attempts to gain the most value for itself in the negotiation by employing several human-negotiation techniques, such as appealing to

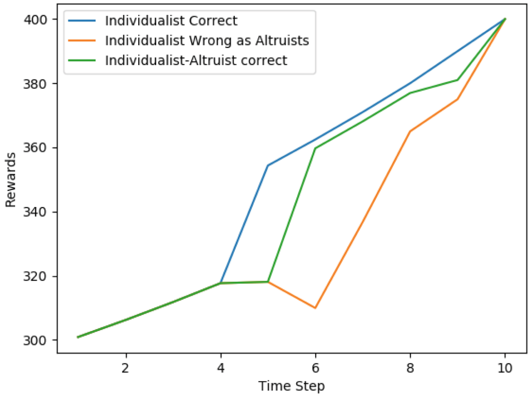


Fig. 4. Misclassification in the second half.

the principle of fairness by utilizing a fixed-pie lie. We will refer to the IAGO software agent as IAGOAgent.

We consider a scenario in which two players determine how to split a set of items between them. The items considered here are bars of gold, bars of iron, shipments of bananas and shipments of spices. The scenario was played out with the IAGOAgent and logs of the negotiation [16] were used to generate negotiation transcripts. The players had to negotiate with the IAGOAgent with a set time limit. Each of these players has different personality types. We used IAGOAgent versus the 4 personality types and generated a total of 30 transcripts for each of IAGOAgent-Individualist, IAGOAgent-Cooperator, IAGOAgent-Competitor and IAGOAgent-Altruist negotiation.

5.2 Opponent Personality Type Prediction

The key goal of our POMDP is to capture the personality type of the players and reason about them. In this experiment, we check whether it has indeed been successful in learning the behaviors using a belief tracking experiment [18]. To perform this experiment, we only use the POMDP belief update and ignore the rewards and optimal policy. We pre-specify the actions the POMDP agent is supposed to take as well as actions of the opponent player which are received as input directly from a dataset. Given this setup, we check if the POMDP can classify the opponent player type correctly using belief updates, i.e., if the type of other player is X, the sum of probabilities of all the states where type $=X$ would be the belief in player X. A high classification accuracy implies that the POMDP was able to learn the behaviors of the opponent correctly. In terms of setting up the experiment, the POMDP agent is trained using the actions which the IAGOAgent takes while the opponent player can be one of the four personality types namely Individualist, Cooperative, Competitive or Altruist (note that transcripts for each personality type are generated by IAGO via role play of each personality type during negotiation with the IAGOAgent).

- **Pair-wise** - In this experiment, we perform a pairwise comparison of the personality types, which leads to a total of 6 experimental settings, namely Individualist vs. Altruist, Individualist vs. Competitor, Individualist vs. Cooperator, Cooperator vs. Competitor, Cooperator vs. Altruist, Competitor vs. Altruist.
- **One vs. Rest Combined** - In this experiment, we perform a comparison of one personality type against a combination of the other three types, which leads to a total of 4 experimental settings, namely Individualist vs. Altruist & Competitor & Cooperator, Altruist vs. Competitor & Cooperator & Individualist, Competitor vs. Individualist & Cooperator & Altruist, Cooperator vs. Competitor & Altruist & Individualist.

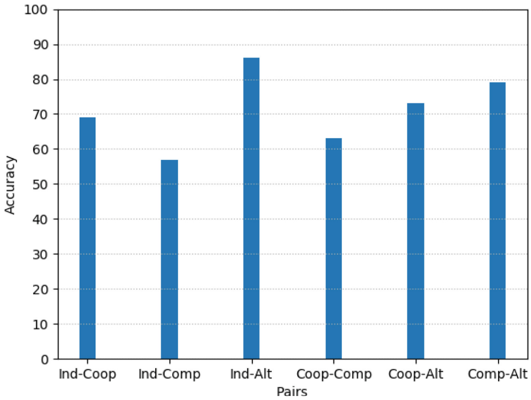


Fig. 5. Pair-wise comparison results.

For both the experiments, out of the 30 datasets generated for each personality type, we use 20 of them as training data and the other 10 datasets as test data. We use action pairs that are taken directly from the test data instead of allowing the POMDP to pick the optimal action. Figure 5 shows the classification accuracy values for the pair-wise comparison experiment. The X-axis indicates the pairs we have used as opponent players for testing, and Y-axis shows the accuracy in classification. Accuracy in classification refers to the number of times the POMDP belief update has correctly classified the opponent type.

In a Pair-wise tracking experiment, the POMDP initially believes that the opponent could be from any of the two personality types with a 0.5 belief; hence it is counted as a correct classification if the final belief at the end of the experiment is greater than 50%. Figure 5 shows that the Altruist type can be distinguished with high accuracy from the other types in a pair-wise comparison test with an accuracy greater than 70% in all the tests. Altruist is particularly distinguishable since they tend to maximize other agent's outcome rather than their own. The Cooperator type is second best with greater than 60% classification accuracy.

Individualists and Competitors fare worse (50% to 60%) since they have fairly similar characteristics (win vs. win big).

Table 1. Accuracy values

Personality type	Accuracy
Individualist	71%
Cooperator	75%
Competitor	65%
Altruist	81%

The second belief tracking experiment tries to distinguish between opponents where one opponent is of a single personality type while the other is a combination of 3 personality types. The goal of this experiment is to observe if we can get a better accuracy result than the pair-wise comparison. Table 1 shows the classification accuracy for this experiment. We observe that there is an improvement in classification accuracy for all the four types, possibly due to the consolidation of information into two categories instead of four categories earlier.

6 Conclusions

In this paper, we model an automated negotiation problem as a Partially Observable Markov Decision Process with a specific focus on personality types. We focus on four personality types, in particular, namely Individualist, Cooperator, Competitor, and Altruist. We generated 30 negotiation transcripts for each type for experimentation purposes using the IAGO software. For purposes of sanity check: (a) We compared our POMDP against two MDPs, MDP1, which does not model personality type and MDP2, which knows the personality type. We found MDP1 $\leq$ POMDP $\leq$ MDP2 in terms of performance (b) We introduced errors in POMDP model in multiple ways and found that they result in a decrease of reward obtained by the agent. Our classification accuracy results performed in two ways, i.e., Pair-wise and One vs. Rest, both show that we obtain good quality results, which showcase that transcripts containing personality-related information can indeed help to improve automated negotiation technology.

References

1. Baarslag, T., Hendrikx, M., Hindriks, K., Jonker, C.: Predicting the performance of opponent models in automated negotiation. In: 2013 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT), vol. 2, pp. 59–66. IEEE (2013)

2. Baarslag, T., Hendrikx, M.J.C., Hindriks, K.V., Jonker, C.M.: Learning about the opponent in automated bilateral negotiation: a comprehensive survey of opponent modeling techniques. Auton. Agent. Multi-Agent Syst. **30**(5), 849–898 (2015). https://doi.org/10.1007/s10458-015-9309-1
3. Baarslag, T., Kaisers, M., Gerding, E., Jonker, C.M., Gratch, J.: When will negotiation agents be able to represent us? The challenges and opportunities for autonomous negotiators. In: International Joint Conferences on Artificial Intelligence (2017)
4. Buron, C.L.R., Guessoum, Z., Ductor, S.: MCTS-based automated negotiation agent. In: Baldoni, M., Dastani, M., Liao, B., Sakurai, Y., Zalila Wenkstern, R. (eds.) PRIMA 2019. LNCS (LNAI), vol. 11873, pp. 186–201. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-33792-6_12
5. Camerer, C.F.: Behavioral game theory: predicting human behavior in strategic situations. In: Advances in Behavioral Economics, pp. 374–392 (2004)
6. Chen, S., Weiss, G.: OMAC: a discrete wavelet transformation based negotiation agent. In: Marsa-Maestre, I., Lopez-Carmona, M.A., Ito, T., Zhang, M., Bai, Q., Fujita, K. (eds.) Novel Insights in Agent-based Complex Automated Negotiation. SCI, vol. 535, pp. 187–196. Springer, Tokyo (2014). https://doi.org/10.1007/978-4-431-54758-7_13
7. De Jonge, D., Zhang, D.: Automated negotiations for general game playing. In: Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems, pp. 371–379 (2017)
8. Fatima, S.S., Wooldridge, M., Jennings, N.R.: Multi-issue negotiation under time constraints. In: Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems: Part 1, pp. 143–150 (2002)
9. Hao, J., Leung, H.: CUHKAgent: an adaptive negotiation strategy for bilateral negotiations over multiple items. In: Marsa-Maestre, I., Lopez-Carmona, M.A., Ito, T., Zhang, M., Bai, Q., Fujita, K. (eds.) Novel Insights in Agent-based Complex Automated Negotiation. SCI, vol. 535, pp. 171–179. Springer, Tokyo (2014). https://doi.org/10.1007/978-4-431-54758-7_11
10. Ji, S., Zhang, C., Sim, K.-M., Leung, H.: A one-shot bargaining strategy for dealing with multifarious opponents. Appl. Intell. **40**(4), 557–574 (2013). https://doi.org/10.1007/s10489-013-0497-6
11. Kaelbling, L.P., Littman, M.L., Cassandra, A.R.: Planning and acting in partially observable stochastic domains. Artif. Intell. **101**(1–2), 99–134 (1998)
12. Kraus, S., Arkin, R.C.: Strategic Negotiation in Multiagent Environments. MIT Press, Cambridge (2001)
13. Kurniawati, H., Hsu, D., Lee, W.S.: SARSOP: efficient point-based POMDP planning by approximating optimally reachable belief spaces. In: Robotics: Science and systems, Zurich, Switzerland, vol. 2008 (2008)
14. Leahu, H., Kaisers, M., Baarslag, T.: Automated negotiation with Gaussian process-based utility models. In: Proceedings of the Twenty-eighth International Joint Conference on Artificial Intelligence, IJCAI, vol. 19 (2019)
15. Leahu, H., Kaisers, M., Baarslag, T.: Preference learning in automated negotiation using Gaussian uncertainty models. In: Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems, pp. 2087–2089. International Foundation for Autonomous Agents and Multiagent Systems (2019)
16. Mell, J., Gratch, J.: IAGO: interactive arbitration guide online. In: AAMAS, pp. 1510–1512 (2016)

17. Mell, J., Gratch, J.: Grumpy & Pinocchio: answering human-agent negotiation questions through realistic agent design. In: Proceedings of the 16th Conference on Autonomous Agents and Multiagent Systems, pp. 401–409 (2017)
18. Paruchuri, P., Chakraborty, N., Gordon, G., Sycara, K., Brett, J., Adair, W.: Intercultural opponent behavior modeling in a POMDP based automated negotiating agent. In: Sycara, K., Gelfand, M., Abbe, A. (eds.) Models for Intercultural Collaboration and Negotiation. AGDN, vol. 6, pp. 165–182. Springer, Dordrecht (2013). https://doi.org/10.1007/978-94-007-5574-1_9
19. Paruchuri, P., Chakraborty, N., Zivan, R., Sycara, K., Dudik, M., Gordon, G.: POMDP based negotiation modeling. In: Proceedings of the IJCAI Workshop on Modeling Intercultural Collaboration and Negotiation (MICON) (2009)
20. Pineau, J., Gordon, G., Thrun, S., et al.: Point-based value iteration: An anytime algorithm for POMDPs. In: IJCAI, vol. 3, pp. 1025–1032 (2003)
21. Pynadath, D.V., Marsella, S.C.: Psychsim: modeling theory of mind with decision-theoretic agents. In: IJCAI, vol. 5, pp. 1181–1186 (2005)
22. Rosenfeld, A., Kraus, S.: Strategical argumentative agent for human persuasion. In: Proceedings of the Twenty-Second European Conference on Artificial Intelligence, pp. 320–328. IOS Press (2016)
23. Salnikov, M.: Text clustering with k-means and TF-IDF. https://medium.com/@MSalnikov/text-clustering-with-k-means-and-tf-idf-f099bcf95183 (2018). Accessed Aug 2019
24. Santos, R., Marreiros, G., Ramos, C., Neves, J., Bulas-Cruz, J.: Using personality types to support argumentation. In: McBurney, P., Rahwan, I., Parsons, S., Maudet, N. (eds.) ArgMAS 2009. LNCS (LNAI), vol. 6057, pp. 292–304. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-12805-9_17
25. Shonk, K.: Understanding different negotiation styles. https://www.pon.harvard.edu/daily/negotiation-skills-daily/understanding-different-negotiation-styles/ (2019). Accessed Oct 2019
26. Yu, C., Ren, F., Zhang, M.: An adaptive bilateral negotiation model based on Bayesian learning. In: Ito, T., Zhang, M., Robu, V., Matsuo, T. (eds.) Complex Automated Negotiations: Theories, Models, and Software Competitions. SCI, vol. 435, pp. 75–93. Springer, Berlin (2013). https://doi.org/10.1007/978-3-642-30737-9_5
27. Zeng, D., Sycara, K.: Bayesian learning in negotiation. Int. J. Hum Comput Stud. **48**(1), 125–141 (1998)