Strategical Argumentative Agent for Human Persuasion: A Preliminary Report*

Ariel Rosenfeld
Bar-Ilan University, Ramat-Gan, Israel
rosenfa5@cs.biu.ac.il

Sarit Kraus
Bar-Ilan University, Ramat-Gan, Israel
sarit@cs.biu.ac.il

Abstract

Automated agents should be able to persuade people in the same way people persuade each other, namely via dialog. Today, automated persuasion modeling and investigation use unnatural assumptions in persuasive interactions which create doubt regarding their applicability in real world deployment with people. In this work we present a novel methodology for persuading people through argumentative dialog. Our methodology combines theoretical argumentation modeling, machine learning and Markovian optimization techniques, which together form an innovative agent named SPA. Preliminary field experiments indicate that SPA provides higher levels of attitude change among subjects compared to two baseline agents.

1 Introduction

Persuasion is designed to influence others by modifying their beliefs or actions. People often engage in persuasive interactions through dialog in which parties, who hold (partially) conflicting points of view, can exchange arguments. Automated agents should be able to persuade people in the same manner; namely, by presenting arguments during a dialog.

Persuasive technologies offer various techniques for an automated agent (the persuader) to persuade a human (the persuadee) to change how she thinks or what she does. Some of these techniques use argumentative dialogs as their persuasion mechanism. However, strategical aspects of argumentative persuasive dialogs are still under-developed (see (Thimm, 2014) for a review). Researchers in the field of Argumentation Theory have recently investigated the challenge of finding an optimal persuasion strategy in dialogs (Hunter, 2015; Hadoux et al., 2015). In particular, the proposed approaches do not assume that the opponent will play optimally, which is a common assumption in game theoretical analysis of persuasion dialogs such as in (Glazer and Rubinstein, 2008), and do not assume perfect knowledge of the persuadees' characteristics. The proposed methods have yet to be investigated with people, mainly due to their assumed strict protocols for the dialog which make their implementation with people very challenging.

In this paper we present a novel methodology for designing automated agents for human persuasion through argumentative dialog without assuming a restrictive protocol. Our agent assumes that the persuadee acts *stochastically* and models her behavior using Argumentation Theory modeling and Machine Learning techniques. The agent deploys a persuasion dialog policy generated by the approximation of an optimal solution to a Partially Observable Markov Decision Process (POMDP) (Kaelbling et al., 1998). Using the obtained policy, the agent presents arguments to its human interlocutor during a dialog. In this preliminary study we show that our agent was able to persuade subjects to change their opinion on a given topic more often than when equipped with two baseline agents. To the best of our knowledge, this is the first work within the context of strategical argumentation to consider the optimization of persuasive dialog with people.

2 Related Work and Background

Theoretical modeling and strategical studies of agent's behavior in persuasion interactions, both within argumentation theory and multi-agent systems, have presented logics, protocols and policies to enable agents to engage each other in a meaningful manner. In this realm, studies rely on the assumption that the engaging agents adhere to strict protocols and logics or that the agents are given un-

*We would like to thank the Intel Collaboration Research Institute for Computational Intelligence, ERC and ISF for their support in this research.

realistic prior knowledge on their opponent's knowledge and beliefs. Without these assumptions, strategic persuasion is inherently NP-complete (Governatori et al., 2013).

The literature on the optimization of persuasive strategies in argumentative dialog can be divided into 2 broad approaches; 1) Game Theory – in which the agent assumes that its interlocutor acts optimally. 2) Heuristic Based – in which the agent uses a strategy following some rule-of-thumb notion.

While the Game Theory approach provides theoretically founded methods and guarantees the computation of optimal argumentative strategies (e.g., (Rienstra et al., 2013)), it has been shown that people often do not adhere to optimal monolithic strategies that can be derived analytically in the argumentative context (Rosenfeld and Kraus, 2014; Rosenfeld and Kraus, 2015). Consequently, this work is categorized under the Heuristic Based approach.

In the Heuristic Based approach, the persuadee is not assumed to be strategical nor is she assumed to act optimally. Several heuristics for persuasive dialog policies have been proposed in the literature, for example, selecting arguments supporting the agents most important values (Bench-Capon, 2003). As observed in (Hadjinikolis et al., 2013), a history of previous dialogues can be used to predict the arguments that the persuadee might put forward. Naturally, this prediction (sometimes called persuadee or opponent modeling) is a key component in designing persuasive arguments. In this realm, the persuadee is usually assumed to act *stochastically*, an assumption we also make in this work. However, the persuader is not assumed to hold perfect knowledge of the persuadee's characteristics. To address this issue, we propose a methodology for predicting people's argumentative choices during a dialog using Machine Learning (ML) techniques. Rosenfeld and Kraus (2015) recently showed that ML can be extremely useful in predicting human argumentative behavior in the real world. We use the authors' suggested methodology in this work; given a certain state of the dialog, the persuader estimates the persuadee's next argument using ML.

Hadoux et al. (2015) suggested a variation of a Markovian model to optimize the persuasive dialog. As in previously suggested modeling, the authors imposed restrictive assumptions on the persuader and persuadee behavior which are relaxed in this work. Hunter (2015) also investigated probabilistic modeling of persuasive dialogs using an *asymmetrical* procedure, in which only the persuader can posit arguments. In this work, we assume a symmetric dialog in which both parties can posit arguments. Both works mentioned above, as well as most of the other works in this field, were not evaluated with people. This fact raises concerns regarding the applicability of the suggested theoretical models when considering human argumentative behavior.

3 Theoretical Modeling

In order to perform reasoning in a persuasive context, an argumentation framework needs to be defined. Throughout this work, we use a *Weighted Bipolar Argumentation Framework (WBAF)* which combines basic notions from the Bipolar Argumentation Framework (Cayrol and Lagasquie-Schiex, 2005), the Weighted Argumentation Framework (Dunne et al., 2011) and the Trust Argumentation Framework (Tang et al., 2012).

Definition 1. Let V be a completely ordered set with a minimum element (V_{min}) and a maximum element (V_{max}). A Weighted Bipolar Argumentation Framework (WBAF) $< A, R, S, W, B, \omega >$ consists of a finite set A called arguments, two binary relations on A called attack (R) and support (S), an interaction weighting function $W : R \cup S \rightarrow V$ and an argument belief function $B : A \rightarrow \mathcal{R}$. $\omega \in A$ is a designated argument which represents the discussed issue.

We will refer to the $WBAF$ as the "argumentation framework".

As defined for the Bipolar Argumentation Framework (Cayrol and Lagasquie-Schiex, 2005), we assume 2 types of possible interactions between arguments; attack and support. That is, if argument $a \in A$ relates to an argument $b \in A$, then aRb or aSb holds, with respect to the relation type.

Dunne et al. (2011) allowed relations to carry different weights. In our modeling, we assume a weighting function $W : R \cup S \rightarrow V$ which represents the degree to which one argument attacks or supports another.

Similar to the belief function defined in the Trust Argumentation Framework (Tang et al., 2012), the WBAF modeling assumes a belief function $B : A \rightarrow V$. The belief function represents the belief a reasoner has in regard to each argument individually, regardless of other arguments.

ω denotes the argument of interest. Specifically, a reasoner seeks to evaluate ω in the context of the argumentation framework. Dung (Dung, 1995) defined several fundamental principles that are the basis of most suggested reasoning rules in the literature. However, recent studies have shown that people do not adhere to many of them in the real world (Cerutti et al., 2014; Rosenfeld and Kraus, 2014), which poses a challenge in utilizing them in human interaction applications. Nevertheless, some reasoning rules have been shown to be effective in real world applications. We use these rules in our framework (see Definition 2).

In our framework we assume that a reasoner uses an evaluation function $v : A \rightarrow V$ which assigns a real-value to each argument while contemplating the argumentation framework. The evaluation function v can be defined in a large number of ways capturing different underpinning principles. In this work we define v based on the *gradual valuation* function (Cayrol and Lagasquie-Schiex, 2005) and the belief operations suggested in (Tang et al., 2012).

Definition 2 is an extension of the gradual valuation in Bipolar Argumentation Frameworks (Cayrol and Lagasquie-Schiex, 2005) to the gradual-belief valuation in $WBAF$s.

Definition 2. Let $AF =< A, R, S, W, B, \omega >$. Let $h : V \times V \rightarrow V$ be a **propagation function**, evaluating the quality of attack/support that one argument has over another, $f_{def} : V^* \rightarrow F_{def}$ (resp. $f_{sup} : V^* \rightarrow F_{sup}$) be the **summation function**, evaluating the quality of all the attacking (respectively supporting) arguments together, and $g : F_{att} \times F_{sup} \times V \rightarrow V$ be the **consolidation function** which combines the impact of the attacking arguments with the quality of the supporting arguments and the initial belief in the argument.

Consider $a \in A$ with $R(a) = \{b_1, \ldots, b_n\}$ and $S(a) = \{c_1, \ldots, c_m\}$. A **gradual-belief valuation function** on AF is $v : A \rightarrow V$ such that $v(a) = g(f_{sup}(h(v(b_1), w(b_1, a)), \ldots, h(v(b_n), w(b_n, a))), f_{att}(h(v(c_1), w(c_1, a)), \ldots, h(v(c_m), w(c_m, a))), B(a))$.

An instantiation f of f_{sup} (or f_{att}) must adhere to the following rules;

- $x_i > x'_i \rightarrow f(x_1, \ldots, x_i, \ldots, x_n) > f(x_1, \ldots, x'_i, \ldots, x_n)$
- $f(x_1, \ldots, x_n) > f(x_1, \ldots, x_n, x_{n+1})$
- $f() = \alpha \leq f(x_1, \ldots, x_n) \leq \beta$[1]
- $g(x, y, z)$ must be non-decreasing in x and z and non-increasing in y.
- $h(x, y)$ must be non-decreasing in both x and y.

Definition 2 creates a family of valuation functions. In this study we use the following instantiation:

Example. As defined in the ArgTrust application (Tang et al., 2012), let h be the min function. Also, extending the gradual definition from (Cayrol and Lagasquie-Schiex, 2005), we define $V = [-1, 1]$, $F_{sup} = F_{att} = [0, \infty]$, $f_{sup}(x_1, \ldots, x_n) = f_{att}(x_1, \ldots, x_n) = \Sigma_{i=0}^{n} \frac{x_i+1}{2}$ and $g(x, y, z) = max\{\frac{1}{1+y} - \frac{1}{1+x}, z\}$.

Proposition 3. The suggested instantiation is a gradual valuation-belief function.

4 Persuasive Dialog Optimization

A persuasive dialog is a finite sequence of arguments $< a_1, a_2, \ldots, a_n >$ where arguments with odd indices are presented by the persuader and arguments with even indices are presented by the persuadee. A dialog is terminated when the persuader uses the $terminate$ argument which is only available to him.

We used D to denote the set of all finite length dialogs. At every even index of the dialog, the agent observes the current state of the dialog $d \in D$ and posits an argument a according to a persuasive policy π. We assume that the persuader is *Omniscience*, namely, it is aware of all arguments affecting ω – the discussed issue. On the other hand, the persuadee may be aware of only a subset of the arguments of which the persuader is aware. The persuader can influence $v(\omega)$ under the persuadee's argumentation framework by introducing new arguments. Once presented with an argument of which the persuadee was unaware, we assume that the argument is added to persuadee's argumentation framework and $v(\omega)$ is updated accordingly.

The persaudee's argumentation framework is unknown to the persuader prior to or during the dialog. We assume that the persuader has a probability distribution χ of the persaudee's possible argumentation frameworks. However, due to the infinite number of possible argumentation frameworks, constructing and using χ is not straightforward. Note that the persuader can be certain that arguments presented in the dialog are in the persuadee's argumentation framework. We assume the persuadee's choice of arguments depends heavily on her argumentation framework. Namely, after an argument is presented by the persuadee, the persuader may change its estimations of the persuadee's argumentation framework as the persuadee's arguments acts as "signals" of her argumentation framework.

The agent seeks to execute an *optimal* persuasive dialog policy. Namely, a policy that will maximize the expected value of $v(\omega)$ by following it un-

[1] α (β) is the minimal (maximal) value of F_{sup} (resp. F_{att})

til the dialog terminates, given χ. Given any non-terminated dialog d, an optimal persuasive dialog policy π^* satisfies the following equation-

Definition 4. $\pi^*(d) = argmax_a \mathrm{E}_{\pi^*}[v(\omega)|da]$

Note that calculating π^* is infeasible, therefore a persuader can only approximate the optimal persuasive dialog policy.

5 Strategical POMDP Agent (SPA)

In this section we describe the Strategical POMDP Agent (SPA) which approximates the optimal persuasive dialog policy (Definition 4).

The persuasive dialog optimization problem (Section 4) can be modeled as the following Partially Observable Markov Decision Process (POMDP).

Definition 5. A Partially Observable Markov Decision Process is a tuple $< \mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, D, \Phi, \gamma >$ where:

- $\mathcal{S}$ is the set of all possible argumentation frameworks. $s \in \mathcal{S}$ is a persuadee's argumentation framework.
- $\mathcal{A}$ is the set of all arguments affecting ω and the argument $terminate$. $a \in \mathcal{A}$ is an argument that the persuader can posit.
- $\mathcal{T}$ represents the state transition dynamics, where $\mathcal{T} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \mapsto \{0, 1\}$. $\mathcal{T}(s, a, s')$ is an indicator function specifying whether or not a transition from s to s' using a is valid. Formally, $\mathcal{T}(s, a, s') = \begin{cases} 1 & \text{if } s' = s \oplus a \\ 0 & \text{otherwise} \end{cases}$ where $s \oplus a$ is the resulting framework from adding argument a to s.
- $\mathcal{R} : \mathcal{S} \times D \mapsto V$ is the reward function of arriving at state s with dialog d. We define $$\mathcal{R}(s, d) = \begin{cases} 0 & \text{if } d \text{ is non-terminated} \\ v(\Omega) & \text{otherwise} \end{cases}$$
- D is the set of all possible finite length dialogs. In our setting, D is also the set of all possible observations.
- Φ is the conditional probability $\Phi(d \mid s', a)$. We will discuss Φ later in this section.
- γ is the discounting factor, representing the likelihood for the persuadee to be bothered or annoyed by a prolonging dialog.

SPA approximates the optimal solution for the above POMDP using the Monte-Carlo Planning algorithm known as POMCP (Silver and Veness, 2010). POMCP is a general purpose algorithm for approximating an optimal policy in large POMDPs. The POCMP algorithm uses a Monte-Carlo search tree to evaluate each argument that the persuader can posit (at odd levels of the tree) given the state of the dialog (a node in the tree). The search tree is rooted in the empty dialog.

The deployment of the POMCP algorithm does not necessitate the explicit definition of Φ. Instead, POMCP requires 3 components:

1. $\mathcal{I}$, a black-box simulator for sampling $s \in \mathcal{S}$ according to each state's initial probability χ (discussed in Section 4).
2. $\mathcal{G}(s, d, a)$, a generative model of the POMDP. This simulator returns a sample of a successor state (s'), dialog (d') and reward (r) given (s, d, a), denoted $(s', d', r) \sim \mathcal{G}(s, d, a)$.
3. $\pi_{rollout}$, a policy that is deployed once leaving the scope of the search tree.

SPA approximates $\mathcal{I}$ using Algorithm 1. In other words, SPA is given an annotated corpus C of dialogs between humans (without any intervention by an agent) on a given topic ω. SPA assumes that the use of an argument a in C is an indicator of its likelihood to appear in persuadees' argumentation frameworks. Therefore, Algorithm 1 samples an argument subset A' from all arguments available in C according to each argument's Maximum Likelihood Estimation (MLE) (Scholz, 1985). The automatic identification of relations between arguments is still an open problem in Natural Language Processing (NLP) (Aharoni et al., 2014), therefore R and S are defined according to a manual annotation of the relation between each pair of arguments in C. More details regarding the annotation process are provided in Section 6. For the definition of B and W, SPA is given two answer sets of questionnaires answered by human participants, denoted Q_1 and Q_2. In Q_1, human participants rate the persuasiveness of each argument in C itself, namely, while disregarding all other arguments of which they may be aware for argument ω. We model each subject's answers in Q_1 as the subject's B function in the corresponding argumentation framework. In Q_2, the same participants whose answers were recorded in Q_1, rate the degree in which arguments affect others. That is, participants are presented with pairs of arguments from C for which a relation was annotated in the first place. Participants rate the degree to which the first argument affects the second one. We model each subject's answers in Q_2 as the subject's W function in the corresponding argumentation framework. In order to sample B and W, and thus complete the definition of the sampled argumentation framework, SPA uses the well-established Kernel Density Estimation (KDE) sampling method (Silverman, 1986): First, SPA samples a participant at random from the list of participants and retrieves her answers from both Q_1 and Q_2. Then, SPA uses a Gaussian KDE method to smooth the contribution of each of the subject's answers (in Q_1 and Q_2) over a local neighborhood of possible answers, resulting in a probability distribution centered around the subject's actual answers. Then, SPA samples the probability distribution and uses the sample as B and W. This process makes it possible to sample an infinite variety of B and W, while using a finite set of points as "anchors" for the sampling process.

Algorithm 1 Algorithm 1 for Simulating $\mathcal{I}$

Require: Dialog corpus C, answers sets Q_1, Q_2.
$A \leftarrow getArguments(C)$
$A' \leftarrow \{\Omega\}$ ▷ Create a set with the designated argument
for all $a \in A$ **do**
    $MLE(a) \leftarrow (\#a$ appears in $C)/|C|$
    Add a to A' with a probability of $MLE(a)$
$S, R \leftarrow$ manually annotated relations among argument in A'
$id \leftarrow$ uniformly sample a participant's id.
$B \leftarrow KDE(Q_1(id))$
$W \leftarrow KDE(Q_2(id))$
return $< A', R, S, W, B, \omega >$

SPA approximates $(s', d', r) \sim \mathcal{G}(s, d, a)$ using Algorithm 2. Namely, similar to the input of Algorithm 1, SPA is given (the same) annotated corpus C of dialogs between humans on a given topic ω. If $a = terminate$ then $s' = s$, $d' = da$ — denotes the concatenation of a to the end of dialog d, and $r = v_s(\omega)$ - the evaluation of ω in the argumentation framework s. Recall that once the persuader posits $terminate$ the dialog ends. Otherwise, the dialog continues with an argument by the persuadee. To simulate the persuadee's answer, a Machine Learning algorithm, $P(a'|d)$, is trained *offline* using C to predict the likelihood for each argument to be presented next given dialog d. Algorithm 2 returns $s' = s \oplus a$ denoting the addition of argument a to s and $d' = dab$ where b is an argument sampled according to $P(b|da)$. We define $r = 0$ for all arguments as we assume there is no direct cost for positing arguments. In our modeling, the cost of prolonging the dialog, is captured by γ (the discounting factor).

As for the rollout policy $\pi_{rollout}$, SPA uses a simple policy where an argument a is selected at random in odd indices of the dialog and the predication model $P(a|d)$ (see Algorithm 2) is used at even indices to simulate the persuadee's responses.

Algorithm 2 Algorithm 2 for Simulating $\mathcal{G}(s, d, a)$

Require: Dialog corpus C.
$P \leftarrow predModel(C)$ ▷ Constructed once.
if a=terminate **then**
    return $< s, da, v_s(\omega) >$
Add a to s.
$b \sim P(da)$ s.t $b \in s$.
return $< s, dab, 0 >$

For the training and evaluation of SPA we recruited more than 200 senior bachelor students in Computer Science. The topic we chose to focus on was the *"Computer Science Master's Degree"*, where the persuader's goal was to increase the likelihood that the persuadee will enroll in a master's program. The topic is of great interest to senior students and was thus selected.

6 Training SPA

Recall that in order to deploy SPA, we need to acquire an annotated dialog corpus C, and two answer sets Q_1 and Q_2 as defined in Section 5. We collected the needed data via two experiments. First we will describe the acquisition of C, Q_1 and Q_2 (Section 6.1) followed by the offline training of SPA (Section 6.2).

6.1 Data Collection

Phase 1 - We recruited 56 senior bachelor students studying Computer Science – 37 males and 19 females with an average age of 28. First, each student was asked to rate a series of five statements using an online questionnaire. The statements concerned the students' personal academic experience, such as "I would volunteer during my studies if I would get credit for it". The statement of interest to us was "I plan to enroll in studies for a Master's degree". For each statement, students provided a rating on the following Likert scale; 1-Strongly Agree, 2-Agree, 3-Nuetral, 4-Disagree and 5-Strongly Disagree.

Students were represented in the system using a special identification number that was given to them prior to the experiment by our research assistant. We ensured that the students were aware that the identifier could not be traced back to their identity in order to avoid possible influences on the students' behavior. Students were divided into 3 groups according to their answers to the question of interest: Positive, Neutral and Negative.

One week later, we matched the students such that each student was coupled with a student from outside her group. The students were asked to converse over the topic of a *"Master's degree"* for a minimum of 5 minutes, and try to convince their interlocutor to adopt their point of view. Dialogs ranged in length from 5 arguments to 11 (mean 7). Each dialog ended when one of the deliberants chose to exit the chat environment. The automatic extraction of arguments from texts and the automatic identification of relations between natural language arguments are still under-developed. Therefore, all dialogs were manually annotated for arguments and the relation between the arguments by a human expert using the annotation methodology used in (Rosenfeld and Kraus, 2015), resulting in an annotated dialog corpus C[2]. Immediately after the chat, students were again asked for their rating on the statement " I plan to enroll in studies for a Master's degree" using the same scale.

In our previous study (Rosenfeld and Kraus, 2015), we showed that people do not adhere to the reasoning rules proposed by the argumentation theory in real world deliberations. Apparently this result extends to persuasive interactions as well. For example, only 67% of the students participating in this phase of the data collection used a **conflict free** argument set in their dialog. Namely, 33% of the students used at least two arguments a and b such that a attacks b or vice versa during their dialog.

In **Phase 2 -** we recruited an additional 107 senior bachelor students studying Computer Science – 68 males and 39 females with an average age of 27. The students were asked to answer two online questionnaires, a week apart. In the first one, students were asked to rate the persuasiveness of each of the 16 arguments in C alone on a scale of 0 to 1, where 0 is "The argument is not persuasive at all" and 1 is "The argument is very persuasive". In the second questionnaire, students were asked to rate the degree one argument effects another over pairs of arguments. The scale that was used was again 0 to 1, where 0 stands for "No effect" and 1 a "Very strong effect".

6.2 Offline Training

In C, 16 distinct arguments were detected (8 pros and 8 cons). Using the methodology suggested in (Rosenfeld and Kraus, 2015), we trained a prediction model P to estimate the likelihood that an argument b will be presented next given dialog d. The prediction model, using a standard decision tree learning algorithm, returns a probability model that estimates the probability of each possible argument to be presented next. For comparison, we also considered using a *Bigram model* (Jelinek, 1990). In Bigram, the model calculates the probability $P(a_2|a_1)$ for every pair of arguments a_1, a_2. That is, the probability that a_2 will follow a_1. These probabilities are estimated using a Maximum Likelihood Estimator with smoothing on the dialogs from C. The perplexity measurement of P (using 10-cross validation) was significantly lower than that of Bigram, which makes it preferable.

P is used in the definition of $\mathcal{G}(s, d, a)$ – the generative process of the POMDP (Section 5). Note that the POMCP Algorithm maintains a search tree which keeps changing and expanding as long as the algorithm is running. Many POMCP applications that implement the POMCP Algorithm, especially in game playing (Heinrich and Silver, 2015), train the POMCP algorithm offline against itself. Namely, two instances of the POMCP algorithm are implemented and are trained simultaneously; the first POCMP learns to play against the second POMCP which in turn learns to play against the first. This methodology cannot be implemented in the scope of this work as we assume that the persuadee is not strategical and hence cannot be represented as a POMCP instance. However, the prediction model P does capture the non-strategic behavior of the persuadee and hence can be used in the definition of $G(\cdot)$.

7 Evaluation

7.1 Experimental Setting

For the evaluation of SPA we recruited 45 senior bachelor students studying Computer Science, 29 males and 16 females with an average age of 28. Students were first asked to rate two statements in an online questionnaire. The statements were: 1) " I plan to enroll in studies for a Master's degree", 2) "A Master's degree will help me in the future". For each statement, students provided a rating on the same Likert scale used in Section 6.1; 1-Strongly Agree, 2-Agree, 3-Nuetral, 4-Disagree and 5-Strongly Disagree. The agent's goal was to persuade the student to change her opinion and rate the 1^{st} statement higher. Given that the 1^{st} statement was initially rated as "Strongly Agree" (true for 3 subjects), the agent was evaluated on the basis of the 2^{nd} statement in which none of the participants provided the highest rating.

[2]We would be pleased to share the constructed corpus (in Hebrew) for future research.

We used a between-subjects experimental design with 3 conditions (agents), each tested with 15 students:

1. **SPA**. SPA was trained for 72 hours in which more than 22700 sessions were simulated. For the evaluation of SPA, we replaced the use of P with the persuadee's actual arguments. Recall that P was used to simulate the persuadee's response in the offline training of the POMCP. For the evaluation we used the student's actual arguments as presented in the dialog.
2. **Relevance Agent (RA)**. RA uses the relevance heuristic suggested in (Rosenfeld and Kraus, 2015) and presents a random argument that directly relates to the last argument presented in the dialog. Of course, RA only suggests arguments that positively relate to ω, that is, indirectly support it. If no such argument exists, RA suggests an argument which directly supports argument ω and does not relate to the last argument presented in the dialog. If all arguments directly supporting ω were already presented, RA terminates the dialog.
3. **Baseline**. Hunter (2015) suggest an asymmetrical approach for persuasive dialogs in which the persaudee's arguments are overlooked and the persuader posits arguments according to its estimation of what the persuadee is aware of. The suggested approach also assumes that the order in which arguments are presented has no bearing on the outcome, yet short dialogs are preferable. When analyzing C we noticed that the mean number of arguments suggested by each student was 4. Therefore, our baseline agent offline searched for the best 4 arguments it could present to a random persuadee. Due to the fact that the persuadee's arguments were overlooked, there was no reason for the agent to change its policy during the dialog. The agent searched through all argument subsets of size 4[3] and evaluated them in an equal number of simulations. In each simulation an argumentation framework was sampled (using $\mathcal{I}$, defined in Section 5). The evaluation of the 1820 subsets was carried out in 72 hours.

Subjects were pseudo randomly assigned to each of the 3 conditions, such that each of the 3 subjects who rated the 1^{st} statement in the questionnaire as "Strongly Agree" was assigned to a different agent.

A week after answering the questionnaire, each student was asked to engage in a chat with her agent. As discussed earlier in this paper, state-of-the-art automatic extraction of arguments from texts has not yet been accurately performed. Therefore, the identification of the arguments used by the students was done using a *Wizard of Oz* methodology, where during the chat a human expert[4] mapped each of the persuadee's sentences into an argument extracted from C (see Section 6.1). In case no argument in C suited the presented statement a designated "NULL" argument was selected. This rarely occurred. In order to bolster the natural flow of the dialog, the Wizard of Oz was also in charge of framing the agent's argument using discourse markers. Namely, the wizard was not allowed to alter the content of the argument but could add discourse markers such as "However", "Moreover", etc.

At the end of the dialog, subjects were asked to answer the online questionnaire once again.

7.2 Results

Of the 15 students who conducted their dialogs with SPA, 4 students (26.6%) changed their rating by one category. Three subjects changed from Positive to Very Positive and one from Neutral to Very Positive. Only one student (6.6%) of the 15 students who held their dialogs with the RA changed her rating (from Negative to Neutral). Among the students who held dialogs with the Baseline agent, 3 students (20%) changed their rating by one category. Surprisingly, all three of them changed their opinion from "Negative" to "Neutral".

Recall that during the collection of human dialogs (with no agent intervention, see Section 6.1) the students' changes in ratings were also recorded. Of the 56 students, 16 (29.6%) changed their opinion by at least 1 category. Of those 16 students, 4 (7.4%) changed their opinion by 2 categories. This result is only slightly better than the results obtained with SPA.

Due to the relatively small number of participants (45 participants over 3 conditions), the difference between the examined agents was not statistically significant nor was there a difference between the human-human dialogs (from Phase 1 of the data collection) and SPA.

[3] $\binom{16}{4}$=1820 subsets.

[4] In order to prevent the expert from being biased toward one of the agents, the expert was not involved in any other part of the research and, in particular, in building the agents.

8 Conclusions and Future Work

In this paper we present a preliminary report of our ongoing work. We present and evaluate a novel methodology for human persuasion through argumentative dialogs. Our methodology, based on argumentation theoretical modeling and machine learning on human generated dialogs and argumentative data, was evaluated in a field experiment with 45 people in which we revealed encouraging results that should be further investigated.

Currently, we are running large-scale experiments in which SPA aims to persuading people to change a practical decision. In our settings, subjects will be presented with a choice; either to receive a piece of chocolate cake or a healthy snack in return for their participation in our experiment. Given the subject's answer, SPA will try to persuade the subject to change her decision. SPA will be evaluated based on the subject's actual choices rather than their reported ratings or attitude.

This study is part of our ongoing effort to investigate the connections and challenges between Argumentation Theory and people (Rosenfeld and Kraus, 2014; Rosenfeld and Kraus, 2015).

References

Ehud Aharoni, Carlos Alzate, Roy Bar-Haim, Yonatan Bilu, Lena Dankin, Iris Eiron, Daniel Hershcovich, and Shay Hummel. 2014. Claims on demand–an initial demonstration of a system for automatic detection and polarity identification of context dependent claims in massive corpora. *COLING 2014*, page 6.

Trevor JM Bench-Capon. 2003. Persuasion in practical argument using value-based argumentation frameworks. *Journal of Logic and Computation*, 13(3):429–448.

Claudette Cayrol and Marie-Christine Lagasquie-Schiex. 2005. On the acceptability of arguments in bipolar argumentation frameworks. In *Symbolic and quantitative approaches to reasoning with uncertainty*, pages 378–389. Springer.

Federico Cerutti, Nava Tintarev, and Nir Oren. 2014. Formal arguments, preferences, and natural language interfaces to humans: an empirical evaluation. In *ECAI*.

Phan Minh Dung. 1995. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial intelligence*, 77(2):321–357.

Paul E Dunne, Anthony Hunter, Peter McBurney, Simon Parsons, and Michael Wooldridge. 2011. Weighted argument systems: Basic definitions, algorithms, and complexity results. *Artificial Intelligence*, 175(2):457–486.

Jacob Glazer and Ariel Rubinstein. 2008. A study in the pragmatics of persuasion: a game theoretical approach. *New perspectives on games and interaction*, pages 121–140.

Guido Governatori, Francesco Olivieri, Simone Scannapieco, Antonino Rotolo, and Matteo Cristani. 2013. Strategic argumentation is np-complete. *arXiv preprint arXiv:1312.4287*.

Christos Hadjinikolis, Yiannis Siantos, Sanjay Modgil, Elizabeth Black, and Peter McBurney. 2013. Opponent modelling in persuasion dialogues. In *IJCAI*.

Emmanuel Hadoux, Aurélie Beynier, Nicolas Maudet, Paul Weng, and Anthony Hunter. 2015. Optimization of probabilistic argumentation with markov decision models. In *IJCAI*.

Johannes Heinrich and David Silver. 2015. Smooth uct search in computer poker. In *IJCAI*.

Anthony Hunter. 2015. Modelling the persuadee in asymmetric argumentation dialogues for persuasion. In *IJCAI*, pages 3055–3061.

Fred Jelinek. 1990. Self-organized language modeling for speech recognition. *Readings in speech recognition*, pages 450–506.

Leslie Pack Kaelbling, Michael L Littman, and Anthony R Cassandra. 1998. Planning and acting in partially observable stochastic domains. *Artificial intelligence*, 101(1):99–134.

Tjitze Rienstra, Matthias Thimm, and Nir Oren. 2013. Opponent models with uncertainty for strategic argumentation. In *IJCAI*, pages 332–338.

Ariel Rosenfeld and Sarit Kraus. 2014. Argumentation theory in the field: An empirical study of fundamental notions. In *Frontiers and Connections between Argumentation Theory and Natural Language Processing*.

Ariel Rosenfeld and Sarit Kraus. 2015. Providing arguments in discussions based on the prediction of human argumentative behavior. In *AAAI*.

FW Scholz. 1985. Maximum likelihood estimation. *Encyclopedia of Statistical Sciences*.

David Silver and Joel Veness. 2010. Monte-carlo planning in large pomdps. In *Advances in neural information processing systems*, pages 2164–2172.

Bernard W Silverman. 1986. *Density Estimation for Statistics and Data Analysis*, volume 26. CRC press.

Yuqing Tang, Elizabeth Sklar, and Simon Parsons. 2012. An argumentation engine: Argtrust. In *Ninth International Workshop on Argumentation in Multiagent Systems*.

Matthias Thimm. 2014. Strategic argumentation in multi-agent systems. *KI-Künstliche Intelligenz*, 28(3):159–168.