Prediction of Counter-Arguments using Indian Logic

MAHALAKSHMI G.S. AND GEETHA T.V.

Department of Computer Science and Engineering, Anna University, Chennai - 25
{mahalakshmi, tvgeedir}@cs.annauniv.edu

Abstract. Argument gaming is a process where arguments and counter-arguments are exchanged as moves and counter-moves during rational discussions. Predicting the direction of argumentation will facilitate the arguer to contribute to a maximum, if the objective of argumentation is knowledge sharing. In this paper, we propose an algorithm for prediction of counter-arguments during argumentative discussions by utilizing the system of syllogistic inference of Indian philosophy. In other words, this is a means to silently track the counter-moves of the opponent for every move from the proponent.

Keywords. Argumentation, Indian logic, prediction, reason fallacies

(Received June 12, 2008 / Accepted November 05, 2008)

1 Introduction

Argumentation can be defined as a process which involves exchange of arguments and counter-arguments. In this paper, we try to address persuasive kind of argumentation [41] which also includes the inquiry system of dialogues [4]. Persuasion aims at convincing others with one's own belief about the subject of discussion [42]. This form of inferencing seems to be identical to the five-membered syllogistic inference of Tarka sastra [40].

The objective of argumentation is to share valid knowledge among the discussion participants. The mechanism of sharing valid knowledge is not readily available as an argument or counter-argument; rather, the valid beliefs about the subject of discussion is said to evolve throughout the argumentation process. The evolving knowledge is recorded as learned definitions within the knowledgebase. For this to happen, the knowledge sharing volunteers taking part in argumentative discussion should adhere to certain rules for knowledge sharing [35]. These policies mainly regulate the argumentation community towards obtaining a definite conclusion.

The argumentation scenario goes like this: initially after adhering to the discussion criteria, the proponent submits an argument for discussion. The argument is analysed for defects by defect analysis procedures [20]. Defects are inconsistencies identified from the proponent's arguments. These inconsistencies are generated by referring to the individual belief base (or knowledge base) organized as Indian logic based ontologies, utilizing the inference and reasoning mechanisms [23].

In case of many defects arising, the defects are prioritized w.r. to the impact of inconsistency, and are populated into a 'defect set'. The resulting 'defect set' is evaluated and the most prominent defect is identified. The argument elements which are responsible in creating those defects are located and refutation (or opposition) strategies to act upon those areas are identified. This results in a 'refutation set'. The defect set and refutation set are analysed to identify a best suitable refutation strategy applicable over the defect set [24]. The counter-arguments are constructed and one counter-argument from the set of counter-arguments are chosen by game theoretic framework on argumentation. Optimal policy decisions are taken before deciding upon the output counter-argument. The performance of every arguer is measured by defect gain – the evaluation of defects, and reward values.

Before the generation of actual counter-argument from the opponent, the proponent may guess at the opponent's knowledge and the doubts (or defects) that will be raised in the counter-argument; this makes the proponent to calculate an anticipatory observation probability about the expected counter-argument. This probability is compared with the actual counter-argument after it is generated and the closeness of expectation is determined. The main contribution in this paper is the prediction algorithm for reasoning from argumentation.

2 Related work: Argumentation and Western Logic

Multi-robot task assignment and co-ordination is the hot research topic in current robot literature. Reinforcement learning is one of the major recommendations in multi-robot systems. Because uncertainty is unavoidable in robot perceptions, belief revision systems have been suggested as a good alternative in merging the co-ordination of multi-robot systems [2]. Partially Observable Markov Decision Processes (POMDP), whose solution, using reinforcement learning or other related techniques that approximate the optimal solution of stochastic decision-making problems is currently a hot topic in the literature [32]. Since, multi-robot co-ordination shall be looked at as a MAS (Multi-agent Systems) problem, we suggest argumentation as a fundamental communication model between the co-ordinating robots. POMDP based algorithms for argument based communication in MAS has been explored by Paquet et. al. [37].

In argument based communication, the agents accept the proposed arguments/ counter arguments only if it's unable to refute them with valid inferences [26]. Also even if it is unable to refute, the agents doesn't take the knowledge as it is, it does a trust and/ or reputation verification [44] with which the level of truth in argumentation process could be found, accordingly they will update their knowledge. Argument based communication looks at entities not only sending messages, but supporting them with reasons as to why those messages are appropriate [38]. This is done in order to convince the opponent about one's own belief.

Argumentation by persuasion is a kind of fundamental approach to dialogue based discussions, in which fallacies and other errors of reasoning can be analysed and evaluated [41 p.7, 43]. A persuasion is characterized by a conflict of opinions (of a claim) between the proponent and the opponent. The purpose of the participants is to resolve the conflict by persuading the other party to give up their opinions, to arrive at a final outcome of stable agreement in the end [14]. Black and Hunter [4] define a dialogue system for argument inquiry dialogues. This allows two agents to share beliefs in order to jointly construct arguments for a specific claim.

According to Toulmin [39], the different parts of a logical argument are as follows:

- **Claim**: what you believe your whole argument proves
- **Data**: what prompts you to make that claim; that is, the facts that lead you to believe your claim is true
- **Qualifier**: the part of the argument that measures the strength or force of the claim.
- **Warrant**: an assumption that supports the claim by connecting it to the data.
- **Backing**: any facts that give substance to the warrant.
- **Rebuttal**: the part of an argument that allows for exceptions without having to give up the claim as generally true.

These shall be represented as an Argument Graph, which is a network of nodes that represent propositions, and links that represent the inferences connecting these propositions [12]. This graph is then used to determine the best moves, to challenge the position of the other conversant. Thus, the question of what responses to generate is investigated. Robin Cohen proposes a model for the understanding of arguments in discourse. All the relations between arguments which are understood from the given text must be supported by conceptual knowledge of general or particular beliefs. Given a sequence of statements, the question is to figure out how the propositions relate to each other [6,7].

Preference-based argumentation framework [15] and Dialogue games [17] are worth-mentioning at this juncture. Behavior games which combine the dialogue games with shared plans with participants were found to be little more flexible than dialogue games [3]. Moore [29] gives a good review of similar approaches. Information sharing games have been already introduced [5]. A generic framework for dialogue games have also been attempted [28]. Protocols or dialogue systems, for argumentation, persuasion or debate have been discussed extensively [13, 18, 31, 34, 41]. The component of dialogue games are a combination of moves and counter-moves. Moves are classified as logical moves and dialogue moves with a stress on levels of commitment and the strategy problem. Parikh [30] uses game theory for cooperative dialogue [11].

However, every development in western logic towards argumentation focused on argument fallacies [8, 33], i.e. inconsistencies lying in the manner in which arguments are proposed. There is hardly any literature speaking on the developments of reason fallacies, as like Indian Logic. Reason fallacies shall be related to 'Data' part of Toulmin's argument structures [39]. Indian philosophical systems provide a deep analysis on classification and handling of reason fallacies in arguments. In this paper defects are alternate name given to 'reason fallacies' of arguments [20].

3 Related work: Argumentation and Indian logic

According to Indian logic, an argument is looked as a combination of three concepts; subject, reason and the object to be proved [40]. Therefore, problem in inferencing occurs if there is a problem in the 'reason' part of the argument, which is supposed to be the support of the claim on the 'subject'. Hence, the name 'reason fallacies' or defects in arguments. The methodology for finding defects is inspired by Tarka sastra [9]. However, we have attempted to further categorise the defects into concept based or relation based defects with inner categorizations [20]. This attempt of classifying the reason based defects in arguments will promote effective analysis over the arguments.

To refute a particular argument, one should have thorough knowledge about what defeat strategies to utilize in constructing the refutations [24]. Indian philosophy defines the various strategies of defeat which can be utilized to construct refutations [40]. However, we have further categorized the philosophical defeat strategies into five kinds: attack, introduce, expand, change and repeat.

The evaluation of arguments and choosing the best counter-argument is motivated by game theory [24]. The choice of best refutation is actually determined by the evaluation mechanisms [16, 27]. The belief-search algorithm [21] which we have utilized contributes to finding optimal refutations from the pool of recommended refutations. However, that best refutation should also satisfy two principles: 1. coverage of all defects 2. use of most recommended and applicable defeat strategy [24]. The reason behind this is that, most popular defects, when identified from an argument and when utilized to generate the next immediate counter-argument, will contribute greatly in interpretation of the submitted argument. Therefore, generation of defects is the driving force behind reasoning from arguments.

Generation of defects could be appropriate if and only if the submitted argument is interpreted in the right sense. Therefore, we have utilized the Indian logic based mechanism of argument representation [19] to have a correct interpretation of the argument elements. To analyse the input argument properly, the elements of arguments should have a correct mapping in the knowledgebase. In addition, the methodology with which the items of the knowledgebase are represented should also be convincing so that, there is no mis-map of world knowledge into the knowledge base. Quick as well as complete knowledge representation formalisms are required which play a good role in finding the defects of the submitted arguments. Indian philosophical method of knowledge representation [10, 40] comes to rescue at this juncture. It is a complete as well as descriptive kind of classification recommended from Nyaya Sastra, the famous Indian Philosophy. In this paper, we have utilized the classification recommendations [1, 10] of Nyaya sastra, and interpret the elements of arguments into specially enhanced knowledge representation formalisms called Nyaya logics. The overall idea is to address the mechanism of argumentation by finding flaws or defects in the arguments and thereby generate a suitable counter-argument [22].

4 Background: Argument Gaming

Definition 1: (Argument Gaming Model AG) Formally the argument gaming model for knowledge sharing AG is defined as a tuple *(S, Ω, Δ, Γ, R, T, O, B)* where,

- S is the set of all states; $S = \{s_1, s_2, ..., s_m\}$. Every proponent / opponent, proposing an argument is logically in one of these states {move, counter-move}.
- Ω is the set of all possible observations accumulated as a defect set or hole set; $\Omega = \{\omega_1, \omega_2, ..., \omega_n\}$. The observations are generally a kind of inconsistency analysis over the proposed arguments.
- Δ is the set of all possible actions or defeat strategies accumulated as defeat set; $\Delta = \{\delta_1, \delta_2, ..., \delta_p\}$. The actions are generally a kind of refutation mechanisms applied over the proposed arguments.
- Γ is the set of all possible counter-arguments constructed against an argument; $\Gamma = \{\gamma_1, \gamma_2, ..., \gamma_q\}$
- $R(s,\delta,\gamma,\omega,s')$ is the reward function that returns the immediate reward Rw for applying counter-argument γ of action δ to eliminate ω while in state s that resulted in state s'
- $T(s,\delta,\gamma,s') = Pr(s'|s,\delta,\gamma)$ is the transition probability, which gives the probability of moving to state s' given that the gaming agent applies counter-argument c of action δ from state s
- $O(\omega,\delta,\gamma,s') = Pr(\omega|s',\delta,\gamma)$ is the observation probability, which gives the probability of observing ω in the next state s' after projecting counter-argument γ of action δ. This is also referred as anticipatory probability, since the next probable observation is anticipated
- B is the belief vector; $b(s)$ returns the probability that the gaming agent is in state s. Since the currently made observations alone are not enough in deciding the nature of the current state, the gaming agent needs to take into account previous observations and actions as part of the procedural argument exchange to determine its current state, which is contained in B. The agent also needs to choose an action to be performed at every argument exchange. This action is determined by the policy $\pi: B \rightarrow \Delta$, which is a function that maps a belief state to the action the gaming agent should execute in this belief state.

A knowledge sharing (KS) game unfolds over a finite sequence of stages of argument exchanges, which is determined by the *horizon t*, the cumulative strength of Ω of every argument exchange, clearing which the discussion is assumed to have reached to a conclusion. Three important argument gaming functions are: observation, refutation and reward assignment [21]. The process of defect analysis is performed in observation, process of defeat strategy determination and optimal refutation generation is done in refutation, and the process of evaluating the counter-argument is carried out in reward analysis.

Definition 2: (Argument A) An argument is a set of propositions related to each other in such a way that all but at least one of them (the premise) are supported to provide support for the remaining (the conclusion). An argument A over argumentation framework AF is defined as a tuple

$$A = \langle A_{id}, f(c,r), A_{state}, A_{status}, A_{str} \rangle$$

where

$f(c,r) = c_{cat} \times r_{cat}$ is a function of argument concepts and argument relations

A_{id} is the argument index;
A_{state} , the state of argument; $A_{state} \subseteq$ {premise, inference, conclusion};
A_{status} , the defeat status of arguments; $A_{status} \subseteq$ {defeated, undefeated, ambiguous, undetermined and
A_{str} , the strength or conclusive force of the argument.

Definition 3: (Concept in Argument) A concept in the argumentation framework is defined as a combination of abstract concept with other categorical properties of concept existence in argument gaming.

$C_{AG} \equiv (c, C_{con}, C_{Cat}, C_{Cf})$ where c is the abstract concept,

C_{con} is the constraint set under which concept C is said to exist; $C_{con} \subseteq \Box O_D$

C_{Cat} is the category of concept in the procedural argumentation scenario; the category can be of three types;

$C_{Cat} \subseteq \{C_S, C_{OI}, C_R\}$

C_{Cf} is the confidence factor (a numeric value) associated with every abstract concept in the knowledgebase.

Definition 4: (Relation in Argument) A relation in the argumentation framework is defined as a combination of abstract relation with other categorical properties of relation existence in argument gaming.

$R_{AG} \equiv (r, r_q, r_{con}, r_{cat}, r_{cf})$

where r is the abstract relation,

$r \subseteq \{R^e, R^i, R^t, R^g\}$

r_q is the set of attributes of the abstract relation,

$$r_q \subseteq \{IC_i, D, X, Xp\}$$

IC_i is the set of invariable concomitance relations,

where $IC_i \subseteq \{symmetric, +IC, -IC, neutral\}$

D is the set of direct relations,

where $D \subseteq \{is-a, has-a, part-of\}$

[Note: For convenience, direct relations are notated by r throughout the rest of our work]

X is the set of exclusive relations where

$$x_i \subseteq X \;/\; (X \subseteq r) \wedge (X \neq \phi)$$

Xp is the set of exceptional relations where

$$xp_i \subseteq Xp \,/\, ((Xp \subseteq r) \wedge (Xp \neq \phi); (xp_i[V] = true)$$

for some element $c_k \subseteq c$; false for other elements of c.

r_{con} is the constraint set under which relation r is said to exist; $r_{con} \subseteq$ {reflexive, symmetric, anti-symmetric, asymmetric, transitive}

$r_{cat} \subseteq \{R_{S-OI}, R_{S-R}, R_{R-OI}\}$

r_{cf} is the confidence factor (a numeric value) associated with every abstract relation in the knowledgebase

Definition 5: (Defect) According to Nyaya school of Indian logic, a fallacy is an object of knowledge which obstructs an inference. It is known as defective reason and is of five kinds [36]. In general, these defects are a simple combination of concept and/or relation elements of the argument. From this perspective, we have categorised the defects into two major divisions: concept-originating, relation-originating [20]. The relation-originating defects may be arising either due to the presence of recommended relation at a wrong place or absence of a required relation at places where it is required. The defect exploration algorithm looks for existence of concepts and the nature of relations between concepts to identify the class and type of defects out of the submitted arguments. The attributes of concepts and relations are also analyzed for occurrence of defects [19].

Definition 6: (Refutation φ) A refutation φ is said to be a mapping existing between set of counter-arguments to set of arguments. It can be denoted as $\varphi : \{\Gamma\} \rightarrow \{\mathrm{A}\}$. The points of refutation (or the portions of argument which is prone to refutation) are denoted by **φ※**

Definition 7: (Concept Priority C_{pr})
The weight of a concept C denoted by C_{pr} is given by

$$C_{pr} = C \times \left(\sum_{k=0}^{q-1} Q_k \times \left(\sum_{con=0}^{3} Q_{con} \times \left(\sum_{v=0}^{val-1} value_v \right) \right) \right)$$

where, q denote the no. of concept qualities present with every concept, val denote the no. of values associated with every quality under the given quality constraints. (For example: Concept – crow; quality – color; value: black; quality constraint: mandatory).

Definition 8: (Relation Priority R_{pr})
The relation R is measured by relation priority factor which is given by

$$R_{pr} = R \times \left(\sum_{k=0}^{3} Q_k \times \left(\sum_{con=0}^{4} (Q_{con}) \right) \right)$$

where, $k \in r_q$.

Definition 9: (Knowledge Gain K↑) Let $K\uparrow$ denote the new knowledge gained (or the knowledge increment) during argument exchange. The increment in knowledgebase (or knowledge gain), measured by $K\uparrow$ is calculated as:

$$K_{\uparrow} = \sum_{k=0}^{w-1} IC\,(E_A[k])$$

where, IC is the information content of E_A, the elements of arguments harvested from the submitted argument and *w* being the no. of (concept and relation) words present in the argument. i.e. a word may be either a concept or a relation, when it is expressed as part of the argument.

The underlying assumption is that, the elements of arguments in the input argument triggers the listening KS volunteers to trace through their own knowledge bases in view of finding the suitable counter-argument; thereby, the knowledge units which lie in the trace path are refreshed and re-evaluated, which contributes to their knowledge gain. Therefore, the knowledge gain of the arguer (or proponent) is given by

$$K_{\uparrow pro} = \sum_{k=0}^{w-1} IC\,(E_A[k])$$

The respondent prefers to answer the argument only if it finds the details relevant to its' own knowledge base. For finding relevancy, let us assume the entity compares the harvested elements of arguments with that of its own knowledge dictionary denoted by *Dictionary*. The knowledge gain of the counter-arguer is given by

$$K_{\uparrow res} = \sum_{k=0}^{w-1} Dictionary \times IC\,(E_A[k])$$

where,

$$IC(E_A[k]) = C_{pr} + R_{pr}$$

Definition 10: (Defect Gain $\Omega_{\uparrow}$)

$\Omega_{\uparrow}$ is the defect gain of set of defects Ω for that particular counter-argument. If we assume $\Omega = \{\omega_1, \omega_2, \ldots\ldots, \omega_n\}$ [refer Definition 1], then the defect gain for that counter-argument is given by,

$$\Omega_{\uparrow} = \sum_{\omega=1}^{n} \left[\omega_{cdv} + \omega_{rdv}\right]$$

where ω_{cdv} is the concept-defect value and ω_{rdv} is the relation defect value , harvested out of that particular counter-argument, and , if $\wp$ is the defect gain constant in the argumentation system, then,

$$\omega_{cdv} = \wp * C_{pr}\ ;\ \omega_{rdv} = \wp * R_{pr}$$

Definition 11: (Trust σ_i^j)

Trust is calculated at regular intervals of transactions. Let σ_{old} indicate the trust history of an entity before the transaction. Let ρ be the number of transaction intervals, then, the trust of an entity *i,* (represented as E_i) with respect to the another entity *j,* (represented as E_j), at every interval ρ is given by,

$$\sigma_i^j = \frac{\left(\sigma_{old} + \sum_{k=0}^{\rho-1} \frac{gradeE_j(k) * rew_{cm}(k)}{\rho}\right)}{2},\ \rho < tot$$

where, $gradeE_j$ indicates the grade of entity *j* stored in the trust look-up table of entity *i.* Grades are a measure of evaluation of counter-arguments. When the counter-argument is enriched with fresh concepts and relations (from the view of proponent) about the world knowledge, the counter-argument of the entity is graded as "maximum". When the counter-argument reinstates the same knowledge of the proponent, the grading depends on the "*conceptualization distance*" (i.e. distance between the concepts in the knowledge base) of the elements of arguments with respect to the subject of discussion. If the counter-argument contains concept elements present around a single level of *conceptualization distance*, then the counter-argument is graded as "average"; counter-arguments with more than one *conceptualization distance* are graded as "minimum".

Definition 12: (Reward *rew*) The reward for the output counter-argument γ, denoted as *rew* is given by

$$rew = \frac{M_{tcf}}{AttemptD_m + AttemptD_n} \times \sum_{k=1}^{l-1} p_{tcf}\left(\frac{valid}{other}\right) \times \left[\frac{(\kappa * K_{\uparrow})}{\Omega_{\uparrow}}\right]$$

where

- $AttemptD_m$ represent the no. of domain members and $AttemptD_n$ represent the no. of non-members of the domain, who attempt to answer the submitted argument
- κ is the overall confidence factor / trust / reputation quotient calculated as a measure of total information shared at every counter-argument
- M_{tcf} is a measure of confident members of the community giving 'valid' counter-argument (i.e. uniform majority of the convincing responses) which is defined as

$$M_{tcf} = TC_m' + TC_n'$$

  - where, TC_m' denote the number of confident trusted domain entities belonging to majority who give valid counter-arguments
  - TC_n' denote the number of confident trusted non-domain entities belonging to majority, who give valid counter-arguments

- $p_{tcf}(valid/other)$ is the conditional probability of the answers given by the entities (both confident and not confident) being valid.

- $\Omega_{\uparrow}$ is the defect gain of set of defects Ω for that particular counter-argument.

From definition 12, it might be clear that, the reward for an argument is inversely proportional to the defect gain or defect weight. Therefore, a valid counter-argument which has no defects will render the reward to 'infinity'. This means, the reward is too high and the agent who has submitted the respective defect-free counter-argument has concluded the discussion. By following the definitions stated above, reasoning is performed over argumentative discussions.

5 Prediction Algorithm of Argument Gaming

During argumentative discussions, though knowledge representation is perfect, the components of entire knowledge base of the proponent are not entirely visible to the opponent and vice versa. Only through counter-arguments the opponent's knowledge shall be realised. Before the generation of actual counter-argument from the opponent, the proponent may guess at the opponent's knowledge and the doubts that will be rised in the counter-argument; this makes the proponent to calculate an anticipatory observation probability about the expected counter-argument. This probability is compared with the actual counter-argument after it is generated and the closeness of expectation is determined.

In other words, this is a means to silently track the counter-moves of the opponent for every move from the proponent. If there is approximately maximum closeness achieved in the expected probability, it means the proponent is capable of judging the level of opponent's knowledge and there would not be any confusion in responding to the counter-arguments (or) there would not be any unsuccessful moves to the future counter-moves. This helps the proponent to maximize its utility in due course of time during the discussion. Thus, the exchange of arguments is a game of reasoning which is defined over the moves and counter-moves through which valid knowledge is shared during discussions.

In a nutshell, the entire scenario can be traced like this: the proponent, at the time of generating every argument, generates an anticipatory 'argument tree' which contains all possible counter-arguments expected for its own argument. The tree can be up to any levels; it basically expresses how far the proponent is able to foresee the realm of discussion. At the start of discussion, having known the domain of the opponent, the proponent 'prunes' certain parts (counter-arguments) of the 'argument tree' which will never occur (in it's own perspective) while discussing with the opponent. (This 'opinion taking' about the nature of arguer has some serious issues, which we will address later).

<u>Algorithm: Argument Gaming</u>

```
Initialisation:
        Proponent generates argument
        Construct 'argument tree'
                Analyse argument of the self
                anticipate defects
                identify refutations
                Generate anticipatory counter-arguments to fit to the 'argument tree'
                Prune 'argument tree' to suit the opponent

Call opponent: Generate counter-argument

Proponent: Generate argument
   analyse counter-argument
   calculate Knowledge gain
   identify defects
   evaluate defects
   generate rewards
   select counter- counter-argument from 'argument tree' which matches defects
   if match found
        equals (anticipatory prob, observation prob.) = true
        improve trust for the opponent
   else
        equals (anticipatory prob, observation prob.) = false
        identify refutations
        select optimal refutation
        generate counter to counter-argument
   Improve 'argument tree'
        Analyse argument of the self
        anticipate defects
        identify refutations
        Generate anticipatory counter-arguments to fit to the 'argument tree'
        Prune 'argument tree' to suit the opponent

update trust table
update reputation table
Call opponent: Generate counter-argument

Opponent: Generate counter-argument
   analyse argument
   calculate Knowledge gain
   identify defects
   evaluate defects
   generate rewards
   select counter-argument from 'argument tree' which matches defects
   if match found
        equals (anticipatory prob, observation prob.) = true
        improve trust for the proponent
   else
        equals (anticipatory prob, observation prob.) = false
        identify refutations
        select optimal refutation
        generate counter-argument
   Improve 'argument tree'
        Analyse counter-argument of the self
        anticipate defects
        identify refutations
        Generate anticipatory counters to counter-argument to fit to the 'argument tree'
        Prune 'argument tree' to suit the proponent

update trust table
update reputation table
Call Proponent: Generate argument
```

When the discussion starts, and the argument is let out, the proponent expects a counter-argument which has already been identified in the anticipatory 'argument tree'. If one such counter-argument is actually obtained from the opponent, it means that the proponent is able to anticipate the motive of opponent accurately. (i.e. the proponent can pat itself at it's back!). This benefits the proponent in such a way that, it need not worry about computing counters online to the counter-argument at that instant; rather, it can choose one among them from the 'argument tree' (Fig. 1a). Theoretically, the proponent is able to predict the direction of argumentative discussion, such that, everything is now at it's control. In terms of partially observable argument gaming scenario, the anticipatory probability is equal to the observation probability [21].

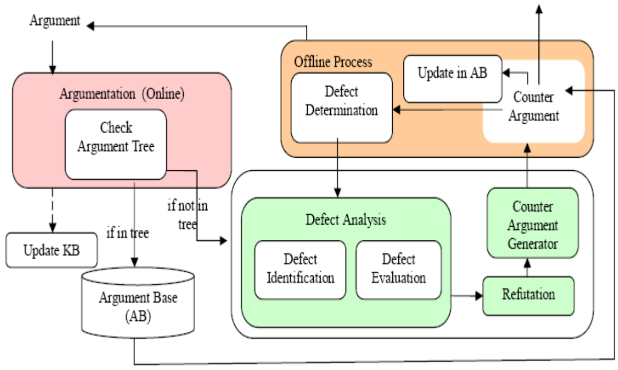

Fig. 1. (a) Prediction of Counter-arguments – Block diagram

A Running Example:

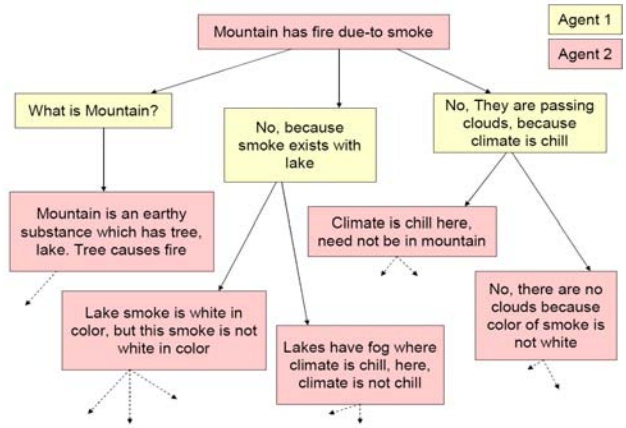

Fig. 1. (b) Anticipatory 'argument tree' for the argument "Mountain has fire due to smoke" in Agent 2

Two situations of failure may occur: 1. wrong opinion taken about pruning the 'argument tree' with respect to the domain details of the opponent. 2. The prediction of counter-arguments does fail at ease. In any of the above scenario, the proponent will face a counter-argument which is out of it's anticipatory 'argument tree'. This means, the proponent has no clues about where the argumentation is heading upon and therefore can't take the lead. It has to compute the counter-argument only at that instant which is time consuming. The observation probability does not match the anticipatory probability. This sort of feedback will help in generating appropriate counter-argument as the response (as explained in para 4 of Section: Introduction).

Let us assume, agent 1 and agent 2 discuss among themselves. Agent 2 proposes the argument: "Mountain has fire due to smoke" and starts the discussion. Before the argument is let out to the opponent (agent1), agent2 computes it's anticipatory 'argument tree' comprising all counter-arguments that it anticipates from the opponent. It also prepares, what would be the next immediate response from its own side, for any such future proposals from the opponent. The anticipatory 'argument tree' is in fig. 1.

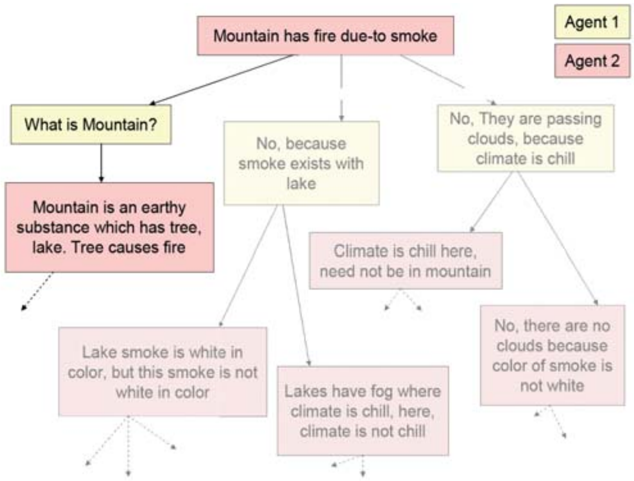

Fig. 2. Successful prediction and pruning in the anticipatory 'argument tree' of Agent 2

It can be seen that, agent 2 anticipates the future proposal that it could get from agent 1, through three anticipatory counter-arguments. The probability of expecting those counter-arguments (or anticipatory probability) is also attached with them. During discussion, if agent 1 generates, a different counter-argument out of the anticipations, agent 2 fails in its prediction; if agent 1 generates a counter-argument in the expectation set of agent 2, say, "What is Mountain?", the observation probability is now closer to the anticipatory probability and therefore, the anticipatory probability is doubled for that argument which proved the competency of agent 2 in predicting agent 1. Further downwards, the path of successful prediction is followed in generating future anticipations.

Fig. 2 shows the path of successful prediction and other counter-predictions being pruned by agent 2 after getting a successful prediction. Like agent 2, agent 1 also performs counter-argument predictions from agent 2 for each of its response to agent 2. This scheme of predictions and counter-predictions continues as long as the discussion comes to a halt.

6 Results

The knowledge base consisted of 78 Indian logic concepts (enriched with qualities and other special attributes as recommended by Nyaya classification system [40]) and 149 relations on a whole, comprising domains like bird, animal, geography, dairy, metal and nature. A more realistic implementation of argumentative discussion was carried out with two knowledge volunteers Agent 1 and Agent 2, discussing about the occurrence of fire over the mountain region on perceiving the smoke on that particular area. Agent 1 has concepts related to ‘nature’ domain, but it lacks information about ‘trees’. Agent 2 has also some concepts related to ‘nature’ domain but it lacks information about ‘falls’. The entire discussion is given in fig. 3b.

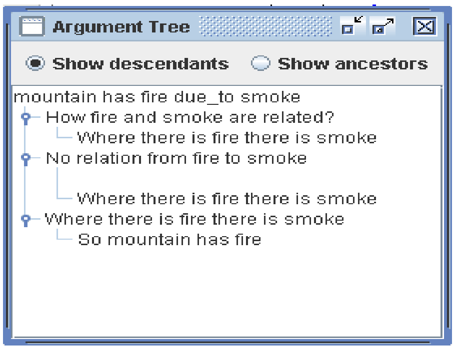

Fig. 3 (a) Argument Tree (for “mountain has fire due_to smoke”)

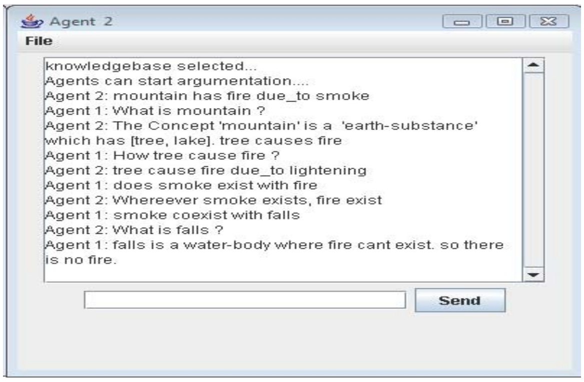

Figure 3. (b) Sample argument “Mountain has fire due to smoke” and related argumentative discussion

The discussion comprises of 10 arguments. The arguments and counter-arguments are generated after successful predictions over the opponent (refer Fig. 1 and Fig. 2). The offline argument tree predicted for argument 1 in Agent 2 (because agent 2 proposes argument 1) is shown in Fig. 3a. The fundamental idea is to select ‘n’ random defects (here, n=3) that would possibly be generated by the opponent for the proposed argument. These defects would ideally generate ‘n’ counter-arguments which is stored as predictions in the offline argument tree repository.

Argument proposed by Agent 2 is analysed for defects at Agent 1 and vice versa. Fig. 3b portrays the ‘miss’ in prediction of offline arguments in fig. 3a. This may be due to the lack of knowledge of agent 2 about the domain specialisations of agent 1. Based on the defects analysed, the defect value is computed which contributes to identifying parts of knowledge in the argument which form the source of generation of counter-argument. The concepts and relations are assembled in a particular format (i.e. in natural language sentences) and the counter-argument is constructed and let out to continue further discussion. (Note: We have not concentrated on natural language generation aspect while generating counter-arguments; instead, we have various structures of training sets of counter-arguments, based on which the new counter-arguments are constructed; However, doing natural language generation would be very much appreciable, but since that is a different research by its own which is not related to our scope, we tend to ignore it here)

At every argument exchange, defects are analysed out of the submitted arguments at both the agents; i.e. Agent 2 does defect analysis on the arguments proposed by Agent 1 and vice versa. When an argument is proposed and suppose, if the information is not found in the knowledgebase the maximum weight of a concept / relation in the knowledge base is given as a defect value. Generally, the defect value is maximum when the knowledge base is refreshed (and a defect is found) on a larger scale. The levels of knowledge refreshed in the knowledge base also contribute to the analysis of defects in arguments. The splitting of arguments into elements of arguments is listed in Table 1.

Table 1. Splitting of Argument Elements

Arg. Id	**Concept**			**Relation (generally 3 kinds, between the three concepts)**
	Subject	**object of inference**	**reason**	
1	mountain	fire	smoke	contact-contact (mountain, fire) invariable (smoke, fire)
2	mountain			
3	mountain	fire	tree	causal(tree, fire)
4	tree	fire		causal (tree, fire)
5	tree	fire	lightening	causal (tree, fire) causal (lightening, fire)
6	smoke	fire		invariable (smoke, fire)
7	fire	smoke		invariable (smoke, fire)
8	smoke	falls		locative (falls, smoke)
9	falls			
10	falls	No fire	Water body	Disjoint (water, fire)

Table 2. Argument Defects

The defects identified during the course of discussion is summarised in Table 2. The first argument tends to propose something about ‘mountain’ which is not known to the opponent.

Arg. Id	Defect Category & Type	Defect Value
1	Concept non-existence	15
2	Nil; Assertive	0
3	Absence of direct relation	28
4	Nil; Assertive	27
5	Absence of invariable relation	40
6	Nil; Assertive	27
7	Presence of direct relation	28
8	Concept non-existence	28
9	Nil; Assertive	0
10	No defect	0

Therefore, the defect 'concept non-existence' is generated. This defect contributes to generating the assertive counter-argument in return, "what is mountain?". The discussion continues with counter-arguments generated depending on the defects obtained from the previous argument.

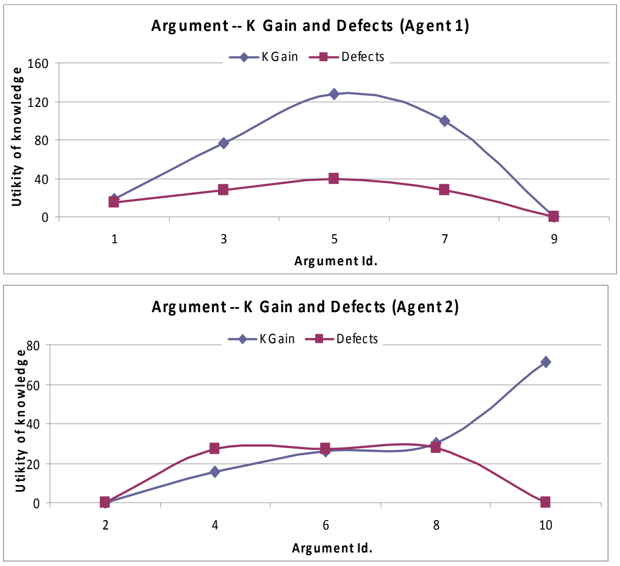

Figure 4. Knowledge Gain and Defect evaluation for Agent 1 and Agent 2 Interactions

The methodologies for calculation of defect gain, trust and rewards shall be roughly interpreted from Mahalakshmi and Geetha [25]. Ideally, if there is no defect gain, the reward tends to 'infinity'. But if there is incomplete information present in one's own knowledge base, obviously the counter-argument carries some form of 'assertive' statements for which the complete information is expected from the other end. In such cases of 'infinite' rewards, we analyse the counter-argument that has generated the reward, if it is assertive, we just allow the discussion to continue; or else, it shall be assumed that there is no defect found with the counter-argument and that the discussion has been concluded. The defect graph is shown in fig. 4. The graph displays the maximum defect gain observed from every argument during discussion. After defect evaluation, reward is calculated and the knowledge base is updated (refer fig. 5).

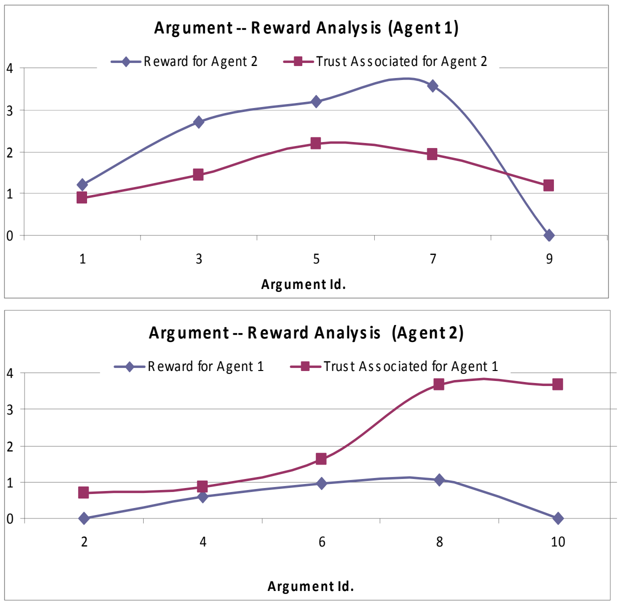

Figure 5. Trust improvement and Reward graph for Agent 1 and Agent 2

7 Conclusion

The aim of this paper is to propose an algorithm of reasoning by predicting the flow of arguments in argument gaming. The entire scenario of argumentation is motivated by Indian logic. The objective is to utilize the presence of reason fallacies in the submitted argument for further generation of counter-arguments. The notion of anticipating the counter-arguments beforehand, transform the entire argumentation scenario into a pattern of argument gaming. However, $\Xi < t-1$, where Ξ is the depth of prediction. In future, incorporating the natural language generation aspects into generation of arguments and introducing argument fallacies from the western philosophy is of our interest.

8 References

[1] G. Aghila, G.S. Mahalakshmi and Dr. T.V. Geetha, KRIL – A Knowledge Representation System based on Nyaya Shastra using Extended Description Logics, VIVEK journal, ISSN 0970-1618, Vol.15, No.3, 2003.

[2] Alessandro Farinelli, Distributed Task Assignment for Real World Environments, collection of abstracts: Multi-Robot Systems: Perception, Behaviors, Learning, and Action, Dagstuhl Seminar, edited by Hans-Dieter Burkhard, Martin Riedmiller, Uwe Schwiegelshohn and Manuela Veloso, 2006.

[3] G. Airenti, B. Bara, and M. Colombetti, Conversational and behavior games in the pragmatics of discourse. *Cognitive Science*, 17:197—256, 1993.

[4] Black E. and Hunter A., A Generative Inquiry Dialogue System, AAMAS, Honolulu, Hawaii, 2007.

[5] Carlson, Dialogue games: An approach to discourse analysis, D. Reidel publishing company, 1983.

[6] Cohen, R., A Computational Theory of the Function of Clue Words in Argument Understanding, COLING, pp. 251 --258, 1984.

[7] Cohen, R., Analyzing the Structure of Argumentative Discourse, Computational Linguistics, 13(1-2): 11-24, 1987.

[8] Engel, S. Morris, With Good Reason: An Introduction to Informal Fallacies, St. Martin Press, 1976.

[9] Gautama, The Nyaya Sutras, translated by S.C. Vidyabhusana, edited by Nanda Lal Sinha, Sacred Book of the Hindus, 1930. Reprinted: Motilal Banarsidass, 1990.

[10] Gradinarov, Phenomenology and Indian Epistemology - Studies in Nyaya-Vaisesika Transcendental Logic and Atomism, Sophia Indological Series, 1990.

[11] Grice, H. P., 'Logic and Conversation'. In: P.Cole and J. L. Morgan (eds.): Syntax and Semantics 3. Academic Press, pp. 41–58, 1975.

[12] T.F.Gordon and D. Walton, The Carneades Argumentation Framework, Paul E Dunne and T J M Bench-Capon (Ed.s), Computational models of argument : COMMA, IOS Press, 2006.

[13] C. L. Hamblin, Fallacies. London:Methuen, 1970.

[14] Jelle Gerbrandy, *Logics of Propositional Control.* AAMAS, 2006.

[15] Leila Amgoud and Claudette Cayrol, A reasoning model based on the production of acceptable Arguments Annals of Mathematics and Artificial Intelligence, v.34 n.1-3, p.197-215, 2002.

[16] Leila Amgoud and Henri Prade, Threat, reward and explanatory arguments: generation and evaluation CMNA IV, 4th Workshop on Computational Models of Natural Argument, Edited by Floriana Grasso, Chris Reed and Giuseppe Carenini, 2004.

[17] Levin, J. and Moore, J., Dialogue-games: meta-communication structure for natural language interaction. *Cognitive science*, 1(4):395-420, 1980.

[18] Mackenzie, J., 'Question Begging in Noncumulative Systems'. Journal of Philosophical Logic 8, 117–133, 1979.

[19] G.S. Mahalakshmi and T.V. Geetha, A Mathematical Model for Argument Procedures based on Indian Philosophy, Proc. of International Conf. Artificial Intelligence and Applications (AIA 2006) as part of the 24th IASTED International Multi-conference Applied Informatics (AI 2006), Austria, pp.311-316, 2006.

[20] G.S.Mahalakshmi and T.V.Geetha, Navya-Nyaya Approach to Defect Exploration in Argument Gaming for Knowledge Sharing, In proceedings of International Conference On Logic, Navya-Nyaya & Applications - A Homage To Bimal Krishna Matilal (ICLNNA '07), Calcutta, India, pp. 171-182, 2007.

[21] G.S. Mahalakshmi and T.V. Geetha, An Indian Logic-Based Knowledge-Sharing Architecture for Virtual Knowledge Communities, International Journal of Networking and Virtual Organisation (IJNVO)–special edition: Virtual Learning and Knowledge Sharing, Inderscience Publishers, 2008 (in press).

[22] Mahalakshmi G.S. and Geetha T.V., 'An algorithm for Knowledge Sharing by Procedural Argument Representations based on Indian Philosophy', *International Journal of Computer, Mathematical Sciences and Applications, IJCMSA,* Vol. 1, no. 3-4, pp. 253-267, 2008.

[23] Mahalakshmi G.S. and Geetha T.V., Reasoning and Evolution of consistent ontologies using NORM, International Journal of Artificial Intelligence (IJAI), Indian Society for Development and Environment Research (ISDER), India, Vol. 2, No.S09, pp.77-94, 2008.

[24] G.S. Mahalakshmi and T.V. Geetha, Modeling Uncertainty in Refutation Selection – A POMDP based approach, Journal of Uncertain Systems - special issue on "Advances in uncertain theory and its applications", Vol. 2, No. 3, pp.223-240, 2008.

[25] G.S. Mahalakshmi and T.V. Geetha, Argument Based Learning Communities – a "tarka" approach, KB Systems, spl issue on AI in blended learning, Elsevier, 2008 (in press).

[26] P. McBurney, Rational Interaction, PhD thesis, Department of Computer Science, University of Liverpool, 2002.

[27] Martin Caminada and Leila Amgoud, On the evaluation of argumentation formalisms, Artificial Intelligence, vol. 171,no. 5-6, pp.286-310, Elsevier Science Publishers Ltd., 2007.

[28] Maudet, N. and F. Evrard, `A generic framework for dialogue game implementation'. In: Proceedings of the Second Workshop on Formal Semantics and Pragmatics of Dialog. Universite Twente, The Netherlands, 1998.

[29] Moore and Hobbs, Computational use of philosophical dialogue theories, Journal of Informal logic, 18(2), 1998.

[30] Parikh, P., 'Communication and Strategic Inference'. Linguistics and Philosophy 14, 473–514, 1991.

[31] Parsons, S. and Hunter, A., A review of uncertainty handling formalisms. In "Applications of Uncertainty Formalisms", LNCS, vol. 1455, 1998.

[32] Pedro Lima, Discrete-Event-Based Modeling, Analysis and Design of Multi-Robot Systems, collection of abstracts: Multi-Robot Systems: Perception, Behaviors, Learning, and Action, Dagstuhl Seminar, edited by Hans-Dieter Burkhard, Martin Riedmiller, Uwe Schwiegelshohn and Manuela Veloso, 2006.

[33] Polycarp Ikuenobe, On the Theoretical Unification and Nature of Fallacies, Argumentation 18: 189–211, Kluwer Academic Publishers, Netherlands, 2004.

[34] Prakken, H., 'Formal systems for persuasion dialogue'. The Knowledge Engineering Review 21, 163–188, 2006.

[35] Sarvepalli Radhakrishnan and Charles A. Moore (edd.), A Source Book in Indian Philosophy - Chapter: The method of vada debate, pp. 361-365, Princeton University Press, 1989.

[36] Sathis Chandra Vidyabhusana, A History of Indian Logic – Ancient, Medieaeval and Modern Schools, Motilal Banarsidass Publishers, 1988.

[37] Sebastian Paquet, Ludovic Tobin and Brahim Chaib-draa, An online POMDP Algorithm for Complex Multiagent Environments, proc. of AAMAS, Netherlands, 2005.

[38] Simon Parsons and Peter McBurney, Argumentation based communication between agents, M.P.Huget (Ed.): Communication in Multi-agent Systems, LNAI 2650, pp. 164-178, 2003.

[39] Stephen Toulmin, *The Uses of Argument*, Cambridge University Press, 1959.

[40] Swami Virupakshananda, Tarka Samgraha, Sri Ramakrishna Math, Madras, 1994.

[41] D. N. Walton and E.C. W. Krabbe, Commitment in Dialogue: Basic Concepts of Interpersonal Reasoning. SUNY Press, pp.165-167, 1995.

[42] Doughlas Walton, Justification of Argumentation Schemes, The Australasian Journal of Logic, 3, 2005.

[43] Walton, D. and Woods, J., Argument: The Logic of the fallacies, Toronto: McGraw-Hill Ryerson, chapter 2, 1982.

[44] M.J. Wooldridge, Semantic issues in the verification of agent communication languages, Journal of Autonomous Agents and Multi-agent Systems, 3(1):9-31, 2000.