Towards Social Argumentative Machines

Sergiu Indrie and Adrian Groza
Technical University of Cluj-Napoca
Department of Computer Science
Baritiu 28, RO-400391 Cluj-Napoca, Romania
Email: adrian.groza@cs.utcluj.ro

***Abstract*—This research advocates the idea of combining argumentation theory with the social web technology, aiming to enact large scale or mass argumentation. The proposed framework allows mass-collaborative editing of structured arguments in the style of semantic wikipedia. The Argnet system was developed based on the Semantic MediaWiki framework and on the Argument Interchange Format ontology.**

I. Introduction

The current research is the result of combining two visions: the *Argumentative Web*, proposed by Iyad Rahwan in 2007 [1], [2], and the *Social Machines* envisaged by Tim Berners-Lee [3]. On one hand, the *Argumentative Web* [1] is a large scale network of inter-connected arguments created by human agents in a structured manner. On the other hand, the *Social Machines* are the result of interaction of social web with the semantic technologies. The goal is to provide the users access to the next level of abstraction. Three key technologies are considered relevant [3] when designing the new generation of social software: Semantic Web, contextual mechanism for WWW, and information control.

In this paper we present the Argnet multi-user argumentation framework based on Semantic MediaWiki and on the Argument Interchange Format (AIF) ontology [1]. To model the interaction between arguments, the AIF ontology [4] is used. Here, an information node $I-node \in N_I$ represents passive information of an argument such as: claim, premise, data, locution, etc. A scheme node $S-node \in N_S$ captures active information or domain-independent patterns of reasoning. The schemes are split in three disjoint sets, whose elements are: rule of inference schemes ($RA-node$), conflict application node ($CA-node$), preference application node ($PA-node$). For modelling the debates, we use the theoretical model of Walton based on argumentation schemes [5]. The schemes are provided with detailed descriptions (premise, conclusion and critical question descriptors) that must (or can in the case of critical questions) be fulfilled.

II. Computational Model of Arguments in Argnet

An I-node, $\pi \in N_I$, is a tuple $\langle \varsigma, \delta, \tau, \phi, \varphi \rangle$, which captures the concept of a statement or fact, where ς is the summary of the node, δ is a user defined credibility value for the current piece of information, τ is a field containing the narrative free text description of the argument, ϕ is a web link supporting the node, and φ is a list of weights used to define contextual terms. Due to the structure and usage of a S-node, the concept of an argument can be embedded inside it. Unlike an I-node which can play different roles in an argumentation scenario (premise or conclusion), an instance of a S-node is uniquely associated with a set of premises and a conclusion and it defines a single reasoning action. There is no need to create another concept only to encapsulate these items under the name of an argument.

An S-node or argument, $\sigma \in N_S$, is represented as tuple $\langle \varsigma, \delta, \tau, \phi, \varphi, \omega, \pi \rangle$, where ς is the node summary, δ is the user defined credibility value, $\pi_i \in N_I$ is the list of I-nodes representing premises, $\gamma \in N$ is the conclusion node, ϕ is the supporting internet address, ϕ is the term defining the topic, ω encapsulates the type of the argumentation scheme used, and π contains the pattern used for graphical representation.

A. Semantic Templates for Argument Annotation

For the semantic annotation of I-nodes and S-nodes we use the semantic templates of the Semantic Media Wiki (SMW) framework.Semantic templates are a simple way of reusing content of parameterizable structures. The advantage that SMW brings is that semantic annotations can be used inside templates, thus allowing consistent annotation without having to learn specific syntax. In order to exploit the AIF ontology in the SMW, we used a mapping process [6] which translates the concepts and roles from the ontology to the internal structuring mechanisms of SMW, as follows: i) The class hierarchy, as well as class membership, are employed through the use of *categories*; ii) Object and datatype properties are defined through the use of the *property namespace*; iii) Attribute values are inserted using *semantic annotations*. To exemplify, consider the creation of the S-node hierarchy. In the Scheme Node category we type $[[imported\ from :: aif : S-node]][[category : Node]]$ which indicates that $S-node$ is an element from the AIF vocabulary and it is a subclass of *Node*. For creating a new conflict between arguments, in the Conflict Node category one has to enter: $[[imported\ from :: aif : CA-node]][[category : SchemeNode]]$

In this wiki implementation, the pieces of information will be created as articles belonging to the I-node category. The I-node semantic template can be used with the syntax described in figure 1. Here, in order to create an argument for weather prediction the user can populate the following fields: *summary* (ς), representing a short description of the argument, *certainty* (δ) representing the degree of belief the user would grant to the statement, *text* (τ), theoretically unlimited text from a wiki article that encapsulates the content of the argument,

978-1-4244-8230-6/10/$26.00 ©2010 IEEE

```
{{I-node
   |summary     = John said that it would rain tomorrow.
   |certainty   = High
   |text        = John, a weather man, carefully read ...
   |supportURL = http://www.theweathernetwork.com/weather
   |context     = [(1.0,cyc:weather),(0.8,foaf:topic)] }}
{{S-node
  |summary      = Considering the John's occupation and the
                  fact that he said it, proves it will rain.
  |certainty    = Very high
  |premises     = John is a weather man, John said it rains
  |conclusion   = On Friday (18.02.2010) it will rain.
  |supportURL   = http://en.wikipedia.org/wiki/Inference
  |topic        = Rain on Friday (18.02.2010)
  |scheme       = Argument from position to know
  |defaultform = Argument from position to know}}
```

Fig. 1. Semantic templates for I-node and S-node annotation.

supportURL (ϕ), a link to the source of the information (in this case the data upon which the weather prediction was made), and the *context* (φ), represented as a list of weights assigned to terms from the imported ontologies (the *cyc* ontology and the *foaf* vocabulary are used in the example).

For the S-node template, the two *premises* of the scheme regard the John's occupation and his affirmation that will rain, and they are represented as a list of I-nodes. The *conclusion* that "It will rain on Friday" is encapsulated as a type of Node, while *scheme* attribute encapsulates the type of the argumentation scheme used, in this case the *Argument from expert opinion*. The *default form* is used internally for the graphical representation of the argument chains, whilst the *topic* field is defined as a list of terms provided by the imported ontologies.

The prototype system will aid the argument creation process by providing users with the option of selecting existing argument schemes. However, to encourage further project growth, users will have the ability to create new argument schemes. The template required for scheme creation needs three attributes: a set of premise descriptions (A_i), a conclusion description (C), and a set of critical questions (CQ_i).

Having this argument model, arguments can be annotated by selecting a descendant of S-node using an appropriate template, selecting the existing Node types for premise and conclusion, and choosing a descendant specific argument scheme. The existing arguments are linked in argument networks based on the following actions: i) *create*, ii) *infer*, using *RA-node*, iii) *support*, using *PA-node*, and iv) *attack*, based on *CA-node*.

B. Argument Reasoning

The Jena tool is exploited to perform the following reasoning tasks: 1) *computing the argument validity*: the state of the argument is computed from its credibility value; 2) *generate explanations*: chaining of argument content; 3) *computing the degree of contradiction*: measure of argument subnetwork inconsistency or disapproval. An ontology model is created from the RDF file which serves as data for building an argument tree having the query element as root. This tree will contain all interaction with the query argument: argument components and all arguments inferring, supporting or attacking the argument or its descendants in the tree.

1) Computing the Argument Validity: In order to establish concepts like argument validity and explanation we need to define factors for evaluating the degree of argument support.

User-defined certainty (γ) consists of a numeric value selected by the user, symbolizing the certainty attached to the current *Node*. Here, the certainty values δ defined by the user are 1 for *very low*, 2 for *low*, 5 for *average*, 7 for *high*, and 9 for *very high*. To be able to determine argument validity we must define a function for calculating credibility: $credibility(node) = \gamma c + \upsilon u + \mu m + \alpha a + \psi p + \sigma s$ where γ, υ, μ, α, ψ and σ are the weighted factors and c, u, m, a, p and s are the certainty, usage (the number of participations in other arguments), minimum support (the minimum credibility value of a supporting S-node or premise), conflict attacks (a count of CA-nodes with the current node as target), preference supports (a count of PA-nodes with the current node as target) and scheme type values (the importance given to each argument scheme) of the targeted node.

Node usage (υ). An important factor in large-scale multiuser argumentation systems is element reuse. The idea is the more an argument is used, the more trustworthy it becomes. In order to capture this line of thought we keep track of element participation in arguments. In the case of I-nodes, we count the number of arguments in which it acts as a premise or as a conclusion. From this value we subtract the ones where the I-node is a conclusion for a preference or conflict node as those are not situations that support the item's credibility. Unlike I-nodes, S-nodes cannot take the role of premises, thus in this case we count the number of arguments in which the S-node is a conclusion, subtracting the ones where it is a conclusion for a PA-node or CA-node.

PA-nodes (ψ) are a means of allowing users to give extra support to an argumentation element. Using this concept, we can increase the credibility of facts (I-nodes) or reasoning acts (S-nodes). Thus, the PA-nodes factor is computed by summing up the credibilities of the number of preference nodes that have the current node as a conclusion. The final system allows parameter selection as well as weight attribution when computing the credibility computation. *CA-nodes (α)* present the user with the capability of contradicting arguments or statements. Like its opposite, the preference node, the CA-node can negatively influence the credibility of nodes. This value is obtained by counting the number of conflict nodes that have the current node as conclusion.

Minimum support (μ) contributes to the propagation of the credibility along the chain of arguments. The idea behind this factor is according to the *weakest link principle*, stating that an element is only as good as its lowest values support. For example, in the case of S-nodes, the node's credibility will take into consideration the smallest credibility of its premises, while in the case of a conclusion, its credibility is computed using the minimum credibility of its supporting arguments of type RA-node. When calculating the minimum support, PA-nodes and CA-nodes are not considered because their influence takes place at their specific factors.

Scheme (σ): Certain schemes, such as *Argument from expert opinion* might be valued highly than *Argument from example*, due to the reliability of the intrinsic source of the argument. Considering that assigning values to these schemes could be domain-dependant or even improperly used, special configuration pages will be available.

Concerns about these credibility factors would be whether their contribution must be bounded to an interval of values so that an element's credibility is also bounded or whether to set the factors to value argument structures like long chains of arguments instead of short ones. Another issue regards the introduction of new factors such as minimum support that differ in the selection function (average support, maximum support or Łukasiewicz. The motivation for these new terms comes from situations in which their absence might prove to be wasteful. For example, consider an argument with two premises, one has a high credibility value while the other has an average value. Is it the right choice to rule out the argument because the minimum support factor does not even consider the high value of the other premise? Argument validity is defined as good credibility, thus determining validity can be summed up by calculating a node's credibility and deciding whether it is above or below a balance point.

2) Generate Explanations: Explanation generation for accepting or defeating an argument rests on the computation of argument credibility. Since several approaches exist for providing explanations, in this implementation we opt for the *best explanation* strategy. Thus, if one wishes to obtain the explanation of an argument, the system must evaluate all nodes in the argument tree with the requested argument as root and select the most credible path from the root to a descendant. A more technical description would be that the algorithm starts from the root of the tree (the requested claim) and it selects recursively the most credible child until there are no more children. Once it has found a path, it concatenates the summaries of all passing nodes in a reverse order.

However, a few questions arise about the path selection. For instance, in our implementation we do not take into consideration preference nodes as a essential part in the explanation, but only as additional information. In our view, PA-nodes might supply irrelevant or unessential information in the reasoning chain. A workaround to a good explanation construction algorithm would be to simply provide all possible explanations, although this is not without its disadvantages in the context of large scale argumentation.

3) Computing the Degree of Contradiction: The degree of contradiction dc takes into consideration the credibility values of each argument $dc(c,r,p) = \sum_{i=1}^{n_c} credibility(c_i)/(\sum_{i=1}^{n_r} credibility(r_i) + \sum_{i=1}^{n_p} credibility(p_i))$ where c, r and p are vectors of conflict, rule and preference nodes, and n_c, n_r and n_p are the vector sizes.

III. Argnet System

The system architecture can be viewed as two sections merging to create a distributed semantic argumentation system. On one hand, from the technical viewpoint, the design relies on the core MediaWiki installation which takes care of the distributed aspect of the application as well as the presentation layer by providing a web interface with a database backup. This core is converted into a semantic system using the SemanticMediaWiki extension which allows annotation and thus further reuse of arguments exported in the RDF syntax. To increase usability, the Semantic Forms extension was added on top, aiming to ease semantic template usage with the help of HTML forms. The Graph extension facilitates the presentation of argumentation chains. On the other hand, from the conceptual point of view, Argnet has its foundation in Walton's argumentation schemes and their integration in the AIF. Using SMW we managed to model the AIF concepts as semantic templates. At this point the two branches merge with the help of our MediaWiki extension, customizable argumentation functions, and the Jena framework used for argument reasoning. This extension takes the user question as input, exports the wiki knowledge in RDF format and uses the Java reasoning application to supply the answer. We used Jena to read the wiki export and obtain information about element membership to create an internal model of the argument tree and provide answers such as argument validity, explanation, contradiction degree, and the graph description.

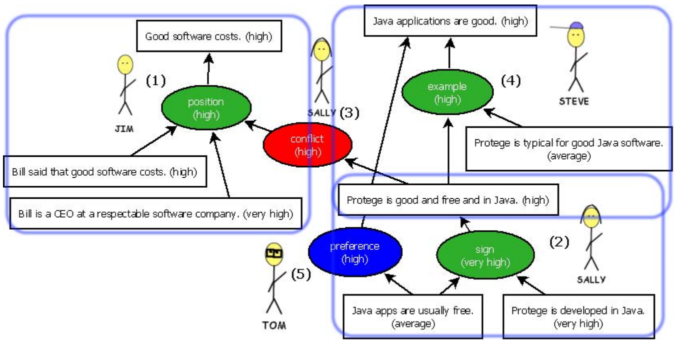

Fig. 2. A sample argumentation scenario.

The illustrative example is based on the network of arguments displayed in figure 2. Here, the information nodes are white, rule nodes are green, conflict nodes depicted with red, and preference nodes with blue. Each node type has its user defined certainty between parentheses, and argument scheme names were shortened from *Argument from position to know* to *position*, from *Argument from example* to *example* and from *Argument from sign* to *sign*.

Network construction. Suppose that Jim wishes to create an argument about the fact that good software costs more. He decides on what argument scheme fits best and chooses from the provided list the *Argument from position to know*. He builds the appropriate premises, conclusion and rule node, depicted as argument 1, where the credibility *high* is attached to the I-node $\prec$ *Good software costs more* $\succ$). Another argument is provided by Sally based on the *Argument from sign* scheme, stating that $\prec$ *Protege is developed in Java* $\succ$ and considering that $\prec$ *Java applications are usually free* $\succ$, so is Protege

TABLE I
MASS ARGUMENTATION TOOLS

Feature	Argnet	Debatepedia	Argumentum	Debategraph
Argument model	ASs and the AIF ontology	PRO/CON structure	Support/Oppose idea	A natural language debate
Semantic annotation	Yes	No	No	No
Graphical representation	Wiki pages and graph-based representation of he chains	Wiki pages with a PRO/CON structure	List of text articles	Graphic environment using concept maps
Degree of support	User defined	None	None	Element rating
Reasoning support	Validity, explanation and contradiction degree	None	None	None
Query capabilities	Use of SMW query language	Relying only on the MediaWiki mechanism	Implementation specific (by text, position, date)	Simple text based
User contribution	Only created arguments list (MediaWiki)	Simple user pages	Evaluation of user's contribution and comparison	Authors of elements displayed
Connectivity	RDF export	Simple MediaWiki export	None	RSS feeds

```
Argument   : Good software costs.
Status     : The fact that Good software costs is not
             sufficiently supported.
Explanation: The facts that Java apps are usually free and
             that Protege is developed in Java indicate that
             Protege is good and free and in Java, which is
             in conflict with the current claim.
```

Fig. 3. Argument explanation.

(argument 2). Consider that Sally sees on the wikipedia the claim that $\prec$ *Good software costs* $\succ$ and the fact that $\prec$ *Protege is a good and free piece of software* $\succ$. Considering that the claim is also true she creates a *CA-node*, attacking the *RA-node* of the argument 1 instead of its conclusion. Steve thinks $\prec$ *Protege is typical of good Java software* $\succ$ and builds argument 4 by reusing the conclusion of the argument 2. Lastly, Tom, a Java passionate, adds a *PA-node* (argument 5) on the argument that $\prec$ *Java applications are good* $\succ$, using one of the premises of the argument 2.

Network processing. Before one begins any processing, he or she must establish certain parameters needed for calculating node credibility. The weights of the argument schemes used in this scenario are: 2 for *Argument from example*, 3 for *Argument from sign*, 4 for *Argument from position to know*, 3 for *Preference* and 3 for *Conflict*. The weights of the credibility factors: 0.02 for *user defined certainty* (γ), 0.7 for *node usage* (υ), 1.5 for *PA-nodes* (ψ), -1.5 for *CA-nodes* (α), 0.18 for *minimum support* (μ) and 0.1 for *argument scheme* (σ).

If one should ask the *validity of the argument* 1 from figure 2, the system calculates all the credibility values from the claim to its descendants. After evaluating all credibilities, the claim's validity fails with a value of -1.08. The conclusion fails mainly because of the strong conflict node (credibility = 1.59), supported by reusable information, reusability being a heavily weighed factor. The *generation of explanation* also depends on the credibility values. In order to build an explanation, the system connects argument components found on the most credible chain of arguments from the claim to one of its premises. In this case the explanation for the argument $\prec$ *Good software costs* $\succ$ is constructed from the premise of the argument 3 and from the antecedents of the argument 2 (see figure 3). The obtained result for *contradiction degree* is $cd(c, r, p) = 0.53$.

IV. RELATED WORK

In order to provide a higher view of our system's characteristics we compare it with three state of the art large-scale mass argumentation systems: Debatepedia, Argumentum and Debategraph.The following features were considered (see table I): i) the *argument model*; ii) whether or not the system uses *semantic annotation*; iii) the *graphical representation*; iv) the possibility of assigning a *degree of support* to an element; v) the *reasoning support* provided by the software, this extending the usability of the system from an argumentation support system to an argumentation-capable system; vi) the *query capabilities* that help users to efficiently locate information; vii) the role and value of users, which can constitute into a credibility factor itself; viii) and the *connectivity*, an essential factor in the context of mass argumentation.

V. CONCLUSION

In this paper we have shown that using a prolific environment such as SMW as an argumentation platform could have numerous benefits and present a stable, yet flexible foundation for further development (the Argnet system is available at http://cs-gw.utcluj.ro/∼adrian/argnet1.html.). The AIF ontology provides the opportunity of integrating software agents within argumentative web, where the agents use the argument reasoning capabilities of our framework and structured facts extracted from DBpedia.

ACKNOWLEDGMENT

This work was supported by CNCSIS-UEFICSU, project number PNII-Idei 170/2009.

REFERENCES

[1] I. Rahwan, F. Zablith, and C. Reed, "Laying the foundations for a world wide argument web," *Artif. Intell.*, vol. 171, no. 10-15, pp. 897–921, 2007.
[2] I. Rahwan and B. Banihashemi, "Arguments in OWL: A progress report," in *COMMA*, 2008, pp. 297–310.
[3] J. Hendler and T. Berners-Lee, "From the semantic web to social machines: A research challenge for AI on the World Wide Web," *Artificial Intelligence*, vol. 174, no. 2, pp. 156 – 161, 2010, special Review Issue.
[4] C. Chesñevar, J. McGinnis, S. Modgil, I. Rahwan, C. Reed, G. Simari, M. South, G. Vreeswijk, and S. Willmott, "Towards an Argument Interchange Format," *Knowl. Eng. Rev.*, vol. 21, no. 4, pp. 293–316, 2006.
[5] D. Walton, "Argumentation schemes for presumptive reasoning," *Argumentation*, vol. 13, no. 3, pp. 338–343, 1999.
[6] D. Vrandečić and M. Krötzsch, "Reusing ontological background knowledge in semantic wikis," in *Proceedings of the First Workshop on Semantic Wikis – From Wiki To Semantics*, 2006.