Three Technologies for Automated Trading

John Debenham and Simeon Simoff

University of Technology, Sydney, Australia {debenham,simeon}@it.uts.edu.au

Three core technologies are needed for automated trading: data mining, intelligent trading agents and virtual institutions in which informed trading agents can trade securely both with each other and with human agents in a natural way. This paper describes a demonstrable prototype that integrates these three technologies and is available on the World Wide Web. This is part of a larger project that aims to make informed automated trading a reality.

1 Introduction

Three core technologies are needed to fully automate the trading process:

– data mining — real-time data mining technology to tap information flows from the marketplace and the World Wide Web, and to deliver timely information at the right granularity.
– trading agents — intelligent agents that are designed to operate in tandem with the real-time information flows received from the data mining systems.
– virtual institutions — virtual places on the World Wide Web in which informed trading agents can trade securely both with each other and with human agents in a natural way.

This paper describes an e-trading system that integrates these three technologies. The e-Market Framework is available on the World Wide Web[1]. This project aims to make informed automated trading a reality, and develops further the "Curious Negotiator" framework [1]. The data mining systems that have been developed for mining information both from the virtual institution and from general sources from the World Wide Web are described in Sec. 2. Intelligent agent that are built on an architecture designed specifically to handle real-time information flows are described in Sec. 3. Sec. 4 describes the work on virtual institutions — this work has been carried out in collaboration with "Institut d'Investigacio en Intel.ligencia Artificial[2]", Spanish Scientific Research Council, UAB, Barcelona, Spain. Sec. 5 concludes.

2 Data Mining

We have designed information discovery and delivery agents that utilise text and network data mining for supporting real-time negotiation. This work has addressed the

---

[1] http://e-markets.org.au
[2] http://www.iiia.csic.es/

central issues of extracting relevant information from different on-line repositories with different formats, with possible duplicative and erroneous data. That is, we have addressed the central issues in extracting information from the World Wide Web. Our mining agents understand the influence that extracted information has on the subject of negotiation and takes that in account.

Real-time embedded data mining is an essential component of the proposed framework. In this framework the trading agents make their informed decisions, based on utilising two types of information (as illustrated in Figure 1): first, information extracted from the negotiation process (i.e. from the exchange of offers), and, second, information from external sources, extracted and provided in condensed form.
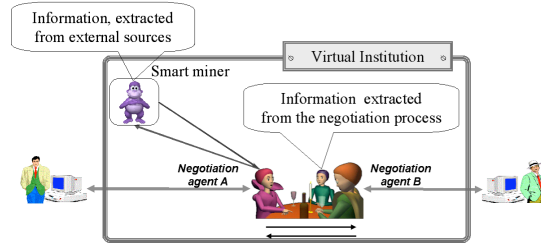

Fig. 1. The information that impacts trading negotiation

The embedded data mining system provides the information extracted from the external sources. The system complements and services the information-based architecture developed in [2] and [3]. The information request and the information delivery format is defined by the interaction ontology. As these agents operate with negotiation parameters with a discrete set of feasible values, the information request is formulated in terms of these values. As agents proceed with negotiation they have a topic of negotiation and a shared ontology that describes that topic. As the information-based architecture assumes that negotiation parameters are discrete, the information request can be formulated as a subset of the range of values for a negotiation parameter. The collection of parameter sets of the negotiation topic constitutes the input to the data mining system. Continuous numerical values are replaced by finite number of ranges of interest.

The data mining system initially constructs data sets that are "focused" on requested information, as illustrated in Figure 2. From the vast amount of information available in electronic form, we need to filter the information that is relevant to the information request. In our example, this will be the news, opinions, comments, white papers related to the five models of digital cameras. Technically, the automatic retrieval of the information pieces utilises the universal news bot architecture presented in [4]. Developed originally for news sites only, the approach is currently being extended to discussion boards and company white papers.

The "focused" data set is dynamically constructed in an iterative process. The data mining agent constructs the news data set according to the concepts in the query. Each concept is represented as a cluster of key terms (a term can include one or more words), defined by the proximity position of the frequent key terms. On each iteration the most frequent (terms) from the retrieved data set are extracted and considered to be related to the same concept. The extracted keywords are resubmitted to the search engine. The process of query submission, data retrieval and keyword extraction is repeated until the search results start to derail from the given topic.

The set of topics in the original request is used as a set of class labels. In our example we are interested in the evidence in support of each particular model camera model. A simple solution is for each model to introduce two labels — positive opinion and negative opinion, ending with ten labels. In the constructed focused data set, each news article is labelled with one of the values from this set of labels. An automated approach reported in [4] extends the tree-based approach proposed in [5].

Once the set is constructed, building the "advising model" is reduced to a classification data mining problem. As the model is communicated back to the information-based agent architecture, the classifier output should include all the possible class labels with an attached

Fig. 2. The pipeline of constructing "focused" data sets

probability estimates for each class. Hence, we use probabilistic classifiers (e.g. Naïve Bayes, Bayesian Network classifiers [6] without the min-max selection of the class output [e.g., in a classifier based on Naïve Bayes algorithm, we calculate the posterior probability $\mathbb{P}_p(i)$ of each class $c(i)$ with respect to combinations of key terms and then return the tuples $< c(i), \mathbb{P}_p(i) >$ for all classes, not just the one with maximum $\mathbb{P}_p(i)$. In the case when we deal with range variables the data mining system returns the range within which is the estimated value. For example, the response to a request for an estimate of the rate of change between two currencies over specified period of time will be done in three steps: (i) the relative focused news data set will be updated for the specified period; (ii) the model that takes these news in account is updated, and; (iii) the output of the model is compared with requested ranges and the matching one is returned. The details of this part of the data mining system are presented in [7]. The currently used model is a modified linear model with an additional term that incorporates a news index Inews, which reflects the news effect on exchange rate. The current architecture of the data mining system in the e-market environment is shown in Figure 3. The $\{\theta_1, \ldots, \theta_t\}$ denote the output of the system to the information-based agent architecture.
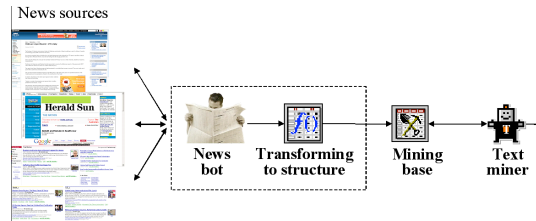
3 Trading Agents

We have designed a new agent architecture founded on information theory. These "information-based" agents operate in real-time in response to market information flows. We have addressed the central issues of trust in the execution of contracts, and the reliability of information [3]. Our agents understand the value of building business relationships as a foundation for reliable trade. An inherent difficulty in automated trading — including e-procurement — is that it is generally multi-issue. Even a simple trade, such as a quantity of steel, may involve: delivery date, settlement terms, as well as price and the quality of the steel. The "information-based" agent's reasoning is

Fig. 3. The architecture of the agent-based data mining system
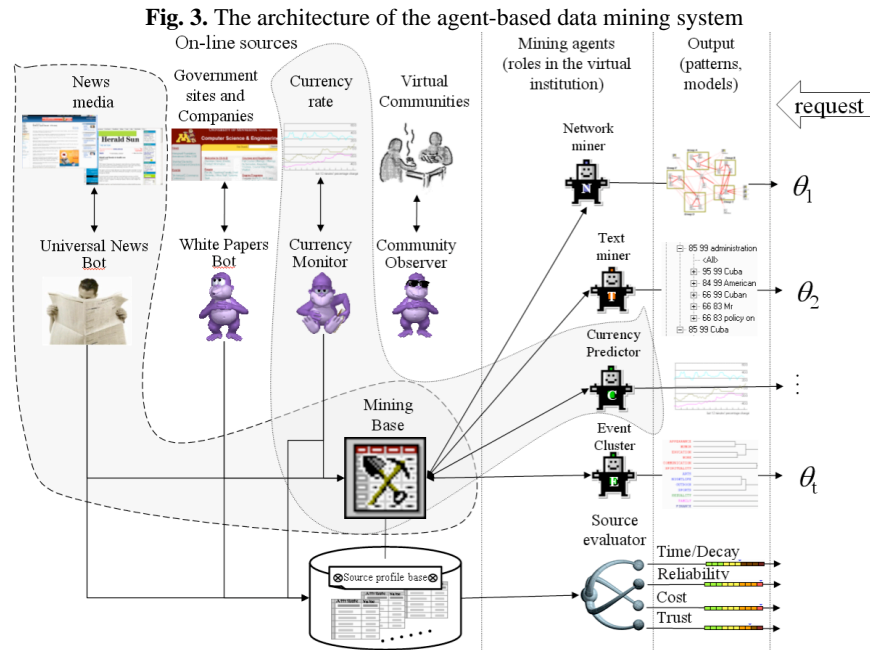
based on a first-order logic world model that manages multi-issue negotiation as easily as single-issue.

Most of the work on multi-issue negotiation has focussed on one-to-one bargaining — for example [8]. There has been rather less interest in one-to-many, multi-issue auctions — [9] analyzes some possibilities — despite the size of the e-procurement market which typically attempts to extend single-issue, reverse auctions to the multi-issue case by post-auction haggling. There has been even less interest in many-to-many, multi-issue exchanges.

The generic architecture of our "information-based" agents is presented in Sec. 3.1. The agent's reasoning employs entropy-based inference and is described in [2]. The integrity of the agent's information is in a permanent state of decay, [3] describes the agent's machinery for managing this decay leading to a characterization of the "value" of information. Sec. 3.2 describes metrics that bring order and structure to the agent's information with the aim of supporting its management.

3.1 Information-Based Agent Architecture

The essence of "information-based agency" is described as follows. An agent observes events in its environment including what other agents actually do. It chooses to represent some of those observations in its world model as beliefs. As time passes, an agent may not be prepared to accept such beliefs as being "true", and qualifies those representations with epistemic probabilities. Those qualified representations of prior obser-

vations are the agent's *information*. This information is primitive — it is the agent's representation of its beliefs about prior events in the environment and about the other agents prior actions. It is independent of what the agent is trying to achieve, or what the agent believes the other agents are trying to achieve. Given this information, an agent may then choose to adopt goals and strategies. Those strategies may be based on game theory, for example. To enable the agent's strategies to make good use of its information, tools from information theory are applied to summarize and process that information. Such an agent is called *information-based*.

An agent called Π is the subject of this discussion. Π engages in multi-issue negotiation with a set of other agents: $\{\Omega_1, \cdots, \Omega_o\}$. The foundation for Π's operation is the information that is generated both by and because of its negotiation exchanges. Any message from one agent to another reveals information about the sender. Π also acquires information from the environment — including general information sources —to support its actions. Π uses ideas from information theory to process and summarize its information. Π's aim may not be "utility optimization" — it may not be aware of a utility function. If Π *does* know its utility function *and* if it aims to optimize its utility

Fig. 4. Basic architecture of agent Π

then Π may apply the principles of game theory to achieve its aim. The information-based approach does not to reject utility optimization — in general, the selection of a goal and strategy is secondary to the processing and summarizing of the information.

In addition to the information derived from its opponents, Π has access to a set of information sources $\{\Theta_1, \cdots, \Theta_t\}$ that may include the marketplace in which trading takes place, and general information sources such as news-feeds accessed via the Internet. Together, Π, $\{\Omega_1, \cdots, \Omega_o\}$ and $\{\Theta_1, \cdots, \Theta_t\}$ make up a multiagent system. The integrity of Π's information, including information extracted from the Internet, will decay in time. The way in which this decay occurs will depend on the type of information, and on the source from which it was drawn. Little appears to be known about how the integrity of real information, such as news-feeds, decays, although its validity can often be checked — "Is company X taking over company Y?" — by pro-active action given a cooperative information source Θ_j. So Π has to consider how and when to refresh its decaying information.

Π has two languages: $\mathcal{C}$ and $\mathcal{L}$. $\mathcal{C}$ is an illocutionary-based language for communication. $\mathcal{L}$ is a first-order language for internal representation — precisely it is a first-order language with sentence probabilities optionally attached to each sentence representing Π's epistemic belief in the truth of that sentence. Fig. 4 shows a high-level view of how Π operates. Messages expressed in $\mathcal{C}$ from $\{\Theta_i\}$ and $\{\Omega_i\}$ are received,
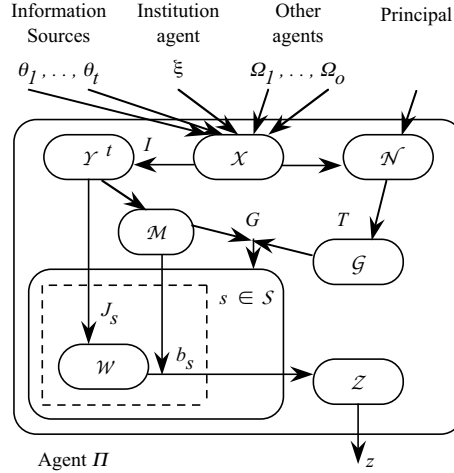
time-stamped, source-stamped and placed in an *in-box* $\mathcal{X}$. The messages in $\mathcal{X}$ are then translated using an *import function* I into sentences expressed in $\mathcal{L}$ that have integrity decay functions (usually of time) attached to each sentence, they are stored in a *repository* $\mathcal{Y}^t$. And that is all that happens until Π triggers a goal.

Π triggers a goal, $g \in \mathcal{G}$, in two ways: first in response to a message received from an opponent $\{\Omega_i\}$ "I offer you €1 in exchange for an apple", and second in response to some need, $\nu \in \mathcal{N}$, "goodness, we've run out of coffee". In either case, Π is motivated by a need — either a need to strike a deal with a particular feature (such as acquiring coffee) or a general need to trade. Π's goals could be short-term such as obtaining some information "what is the time?", medium-term such as striking a deal with one of its opponents, or, rather longer-term such as building a (business) relationship with one of its opponents. So Π has a trigger mechanism T where: $T : \{\mathcal{X} \cup \mathcal{N}\} \to G$.

For each goal that Π commits to, it has a mechanism, G, for selecting a strategy to achieve it where $G : \mathcal{G} \times \mathcal{M} \to \mathcal{S}$ where $\mathcal{S}$ is the strategy library. A *strategy* s maps an information base into an action, $s(\mathcal{Y}^t) = z \in \mathcal{Z}$. Given a goal, g, and the current state of the social model m^t, a strategy: $s = G(g, m^t)$. Each strategy, s, consists of a *plan*, b_s and a *world model* (construction and revision) *function*, J_s, that constructs, and maintains the currency of, the strategy's *world model* W_s^t that consists of a set of probability distributions. A *plan* derives the agent's next action, z, on the basis of the agent's world model for that strategy and the current state of the social model: $z = b_s(W_s^t, m^t)$, and $z = s(\mathcal{Y}^t)$. J_s employs two forms of entropy-based inference:

- Maximum entropy inference, J_s^+, first constructs an *information base* $\mathcal{I}_s^t$ as a set of sentences expressed in $\mathcal{L}$ derived from $\mathcal{Y}^t$, and then from $\mathcal{I}_s^t$ constructs the world model, W_s^t, as a set of complete probability distributions using maximum entropy inference.
- Given a prior world model, W_s^u, where $u < t$, minimum relative entropy inference, J_s^-, first constructs the incremental information base $\mathcal{I}_s^{(u,t)}$ of sentences derived from those in $\mathcal{Y}^t$ that were received between time u and time t, and then from W_s^u and $\mathcal{I}_s^{(u,t)}$ constructs a new world model, W_s^t using minimum relative entropy inference.

3.2 Valuing Information

A chunk of information is valued first by the way that it enables Π to do something. So information is valued in relation to the strategies that Π is executing. A strategy, s, is chosen for a particular goal g in the context of a particular representation, or environment, e. One way in which a chunk of information assists Π is by altering s's world model W_s^t — see Fig. 4. A model W_s^t consists of a set of probability distributions: $W_s^t = \{D_{s,i}^t\}_{i=1}^n$. As a chunk of information could be "good" for one distribution and "bad" for another, we first value information by its effect on each distribution. For a model W_s^t, the *value* to W_s^t of a message received at time t is the resulting decrease in entropy in the distributions $\{D_{s,i}^t\}$. In general, suppose that a set of stamped messages $X = \{x_i\}$ is received in $\mathcal{X}$. The *information* in X at time t with respect to a particular distribution $D_{s,i}^t \in W_s^t$, strategy s, goal g and environment e is:

$$\mathbb{I}(X \mid D_{s,i}^t, s, g, e) \triangleq \mathbb{H}(D_{s,i}^t(\mathcal{Y}^t)) - \mathbb{H}(D_{s,i}^t(\mathcal{Y}^t \cup I(X)))$$

for $i = 1, \cdots, n$, where the argument of the $D_{s,i}^t(\cdot)$ is the state of Π's repository from which $D_{s,i}^t$ was derived. The environment e could be determined by a need ν (if the evaluation is made in the context of a particular negotiation) or a relationship ρ (in a broader context). It is reasonable to aggregate the information in X over the distributions used by s. That is, the information in X at time t with respect to strategy s, goal g and environment e is:

$$\mathbb{I}(X \mid s, g, e) \triangleq \sum_i \mathbb{I}(X \mid D_{s,i}^t, s, g, e)$$

and to aggregate again over all strategies to obtain the value of the information in a statement. That is, the *value of the information* in X with respect to goal g and environment e is:

$$\mathbb{I}(X \mid g, e) \triangleq \sum_{s \in \mathcal{S}(g)} \mathbb{P}(s) \cdot \mathbb{I}(X \mid s, g, e)$$

where $\mathbb{P}(s)$ is a distribution over the set of strategies for goal g, $\mathcal{S}(g)$, denoting the probability that strategy s will be chosen for goal g based on historic frequency data. and to aggregate again over all goals to obtain the (potential) information in a statement. That is, the *potential information* in X with respect to environment e is:

$$\mathbb{I}(X \mid e) \triangleq \sum_{g \in \mathcal{G}} \mathbb{P}(g) \cdot \mathbb{I}(X \mid g, e) \tag{1}$$

where $\mathbb{P}(g)$ is a distribution over $\mathcal{G}$ denoting the probability that strategy g will be triggered based on historic frequency data.

4 Virtual Institutions

This work is done on collaboration with the Spanish Governments IIIA Laboratory[2] in Barcelona. Electronic Institutions are software systems composed of autonomous agents, that interact according to predefined conventions on language and protocol and that guarantee that certain norms of behaviour are enforced. Virtual Institutions enable rich interaction, based on natural language and embodiment of humans and software agents in a "liveable" vibrant environment. This view permits agents to behave autonomously and take their decisions freely up to the limits imposed by the set of *norms* of the institution. An important consequence of embedding agents in a virtual institution is that the predefined conventions on language and protocol greatly simplify the design of the agents. A Virtual Institution is in a sense a natural extension of the social concept of institutions as regulatory systems that shape human interactions [10].

Virtual Institutions are electronic environments designed to meet the following requirements towards their inhabitants:

1. enable institutional commitments including structured language and norms of behaviour which enable reliable interaction between autonomous agents and between human and autonomous agents;

2. enable rich interaction, based on natural language and embodiment of humans and software agents in a "liveable" vibrant environment.

The first requirement has been addressed to some extent by the Electronic Institutions (EI) methodology and technology for multi-agent systems, developed in the Spanish Government's IIIA Laboratory in Barcelona [10]. The EI environment is oriented towards the engineering of multiagent systems. The Electronic Institution is an environment populated by autonomous software agents that interact according to predefined conventions on language and protocol. Following the metaphor of social institutions, Electronic Institutions guarantee that certain norms of behaviour are enforced. This view permits that agents behave autonomously and make their decisions freely up to the limits imposed by the set of norms of the institution. The interaction in such environment is regulated for software agents. The human, however, is "excluded" from the electronic institution.

The second requirement is supported to some extent by the distributed 3D Virtual Worlds technology. Emulating and extending the physical world in which we live, Virtual Worlds offer rich environment for a variety of human activities and multi-mode interaction. Both humans and software agents are embedded and visualised in such 3D environments as avatars, through which they communicate. The inhabitants of virtual worlds are aware of where they are and who is there — elements of the presence that are excluded from the current paradigm of e-Commerce environments. Following the metaphor of the physical world, these environments do not impose any regulations (in terms of language) on the interactions and any restrictions (in terms of norms of behaviour). When this encourages the social aspect of interactions and establishment of networks, these environments do not provide means for enabling some behavioural norms, for example, fulfilling commitments, penalisation for misbehaviour and others.

Technologically, Virtual Institutions are implemented following a three-layered framework, which provides deep integration of Electronic Institution technology and Virtual Worlds technology [11]. The framework is illustrated in Figure 5. The Electronic Institution Layer hosts the environments that support the Electronic Institutions technological component: the graphical EI specification designer ISLANDER and the runtime component AMELI [12]. At runtime, the Electronic Institution layer loads the institution specification and mediates agents interactions while enforcing institutional rules and norms.

The Communication Layer connects causally the Electronic Institutions layer with the 3D representation of the institution, which resides in the Social layer. The causal connection is the integrator. It enables the Electronic Institution layer to respond to changes in the 3D representation (for example, to respond to the human activities there), and passes back the response of the Electronic Institution layer in order to modify the corresponding 3D environment and maintain the consistency of the Virtual Institution. The core technology — the Causal Connection Server, enables the Communication Layer to act in two directions. Technically, in direction from the Electronic Institution layer, messages uttered by an agent have immediate impact in the Social layer. Transition of the agent between scenes in the Electronic Institution layer, for example, must let the corresponding avatar move within the Virtual World space accordingly. In the other direction, events caused by the actions of the human avatar in

Fig. 5. The three layer architecture and its implementation
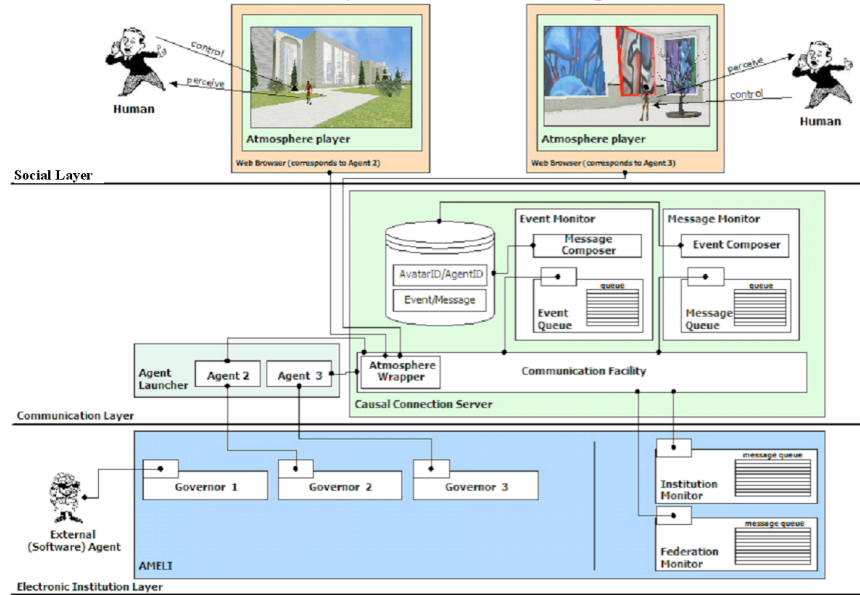
the Virtual World are are transferred to the Electronic Institution layer and passed to an agent. This implies that actions forbidden to the agent by the norms of the institution (encoded in the Electronic Institution layer), cannot be performed by the human. For example, if a human needs to register first before leaving for the auction space, the corresponding agent is not allowed to leave the registration scene. Consequently, the avatar is not permitted to open the corresponding door to the auction (see [11] for technical details of the implementation of the Causal Connection Server).

5 Conclusions

A demonstrable prototype e-Market system permits both human and software agents to trade with each other on the World Wide Web. The main contributions described are: the broadly-based and "focussed" data mining systems, the intelligent agent architecture founded on information theory, and the abstract synthesis of the virtual worlds and the electronic institutions paradigms to form "virtual institutions". These three technologies combine to present our vision of the World Wide Web marketplaces of tomorrow.

The implementation of the three components is described in greater detail on our e-Markets Group Site[1]. The implementation of the data mining systems is notable for the way in which it is integrated with the trading agents — this enables the agents to dynamically assess the integrity of the various information sources. The implementation of the trading agents is greatly simplified by the assumption that preferences for

each individual issue are common knowledge and are complementary for each a pair of traders. This assumption, together with the use of coarse discrete representations of continuous variables, reduces the number of possible worlds and simplifies the minimum relative entropy calculations. The implementation of the virtual institutions is an on-going research project with jointly with IIIA[2]. We have built a prototype with a proprietary game engine, and are now moving to modify an open source engine in an attempt to achieve acceptable performance. The whole project is at the 'demonstrable prototype' stage — although we are greatly encouraged by the performance observed. Much work remains to be done, notably implementing a scalable virtual institution.

References

1. Simoff, S., Debenham, J.: Curious negotiator. In M. Klusch, S.O., Shehory, O., eds.: proceedings 6th International Workshop Cooperative Information Agents VI CIA2002, Madrid, Spain, Springer-Verlag: Heidelberg, Germany (2002) 104–111
2. Debenham, J.: Bargaining with information. In Jennings, N., Sierra, C., Sonenberg, L., Tambe, M., eds.: Proceedings Third International Conference on Autonomous Agents and Multi Agent Systems AAMAS-2004, ACM (2004) 664 – 671
3. Sierra, C., Debenham, J.: An information-based model for trust. In Dignum, F., Dignum, V., Koenig, S., Kraus, S., Singh, M., Wooldridge, M., eds.: Proceedings Fourth International Conference on Autonomous Agents and Multi Agent Systems AAMAS-2005, Utrecht, The Netherlands, ACM Press, New York (2005) 497 – 504
4. Zhang, D., Simoff, S.: Informing the Curious Negotiator: Automatic news extraction from the Internet. In: Proceedings 3'rd Australasian Data Mining Conference, Cairns, Australia (2004) 55–72
5. Reis, D., Golgher, P.B., Silva, A., Laender, A.: Automatic web news extraction using tree edit distance. In: Proceedings of the 13'th International Conference on the World Wide Web, New York (2004) 502–511
6. Ramoni, M., Sebastiani, P.: Bayesian methods. In: Intelligent Data Analysis. Springer-Verlag: Heidelberg, Germany (2003) 132–168
7. Zhang, D., Simoff, S., Debenham, J.: Exchange rate modelling using news articles and economic data. In: Proceedings of The 18th Australian Joint Conference on Artificial Intelligence, Sydney, Australia, Springer-Verlag: Heidelberg, Germany (2005)
8. Faratin, P., Sierra, C., Jennings, N.: Using similarity criteria to make issue trade-offs in automated negotiation. Journal of Artificial Intelligence **142** (2003) 205–237
9. Debenham, J.: Auctions and bidding with information. In Faratin, P., Rodriguez-Aguilar, J., eds.: Proceedings Agent-Mediated Electronic Commerce VI: AMEC. (2004) 15 – 28
10. Arcos, J.L., Esteva, M., Noriega, P., Rodríguez, J.A., Sierra, C.: Environment engineering for multiagent systems. Journal on Engineering Applications of Artificial Intelligence **18** (2005)
11. Bogdanovych, A., Berger, H., Simoff, S., Sierra, C.: Narrowing the gap between humans and agents in e-commerce: 3D electronic institutions. In Bauknecht, K., Pröll, B., Werthner, H., eds.: E-Commerce and Web Technologies, Proceedings of the 6th International Conference, EC-Web 2005, Copenhagen, Denmark, Springer-Verlag: Heidelberg, Germany (2005) 128–137
12. : (Electronic institution development environment: http://e-institutor.iiia.csic.es/)