Abstract

Proactive information sharing is a challenging issue in solving time-critical analysis tasks related to homeland security. This paper describes an ongoing research effort to support counterterrorism analysts using software agents that can interact with a Semantic Web reasoning infrastructure or other agents to retrieve and present relevant personalized information to the user. The approach is inspired by psychological studies suggesting effective human team behaviors are based on maintaining a shared mental model of the team. An agent architecture called CAST is used to support a computational shared mental model about the structure and the process of the team, enabling software agents to dynamically anticipate information needs of analysts, and to assist them by finding and delivering information relevant to their needs.

1. Introduction

Current collaboration technology consumes too much of the analyst’s valuable time. Shared whiteboards, chat rooms and email require significant active participation by multiple analysts to coordinate and negotiate information needs. Shared file systems and portals require the analyst to search for relevant information. Recommender systems require the analyst to read the recommended documents. Even automated “expert finder” tools still require significant communication overhead after the expert is identified. We are exploring the application of assistant agent technology to reduce the overhead effort of collaboration. The hypothesis is that if an assistant agent (AA) knows about the roles of individual analysts and the goals of the analyst team, they can proactively share specific relevant fine-grained information automatically. In our research, the assistant agent is empowered by CAST, a multi-agent architecture that supports a computational shared mental model for a team of agents. Based on the shared mental model, an agent can dynamically anticipate information needs of its teammates, and proactively deliver the needed information to the right recipient, thereby reduce the time delay in communication. The assistant agents will not completely eliminate the need for human to human interaction. The goal is simply to reduce the amount of time the analyst spends on the mechanics of collaboration.

2. SMM-based Agents for Supporting a Team of Analysts

CAST (Collaborative Agents for Simulating Teamwork) is an agent architecture that empowers agents in a team to have a shared mental model about the structure and the process of the team so they can anticipate potential information needs of teammates and proactively deliver it to them (Yen et al. 2001). Psychologists who studied teamwork have identified overlapping shared mental models as an important characteristic of high-performance teams (Cannon-Bowers et al. 1990). To enable agents with such capabilities, CAST provides five key features. First, it uses a high-level language, MALLET (Multi-Agent Logic-based Language for Encoding Teamwork), to capture knowledge about the team structure and the team process. Second, each agent establishes a computational shared mental model about the structure and the process of the team, enabling software agents to dynamically anticipate information needs of analysts, and to assist them by finding and delivering information relevant to their needs.
information needs of teammates or fuse large amount of low-level information.

The CAST agents play the role of AAs by establishing and maintaining shared mental models with the human analysts. The basic attributes such as name, interest and expertise of a human analyst is described in user profile. The notion of shared mental model broadens the scope of user profile to include dynamic process, ongoing tasks, current roles and responsibilities of the human user. In order to enable the collaboration between multiple AAs and multiple analysts, different AAs must have shared knowledge of their user profiles, which extends the profile of an individual user to a team. The shared knowledge of user profiles and each AA’s information gathering capability contribute to the SMM captured in MALLET, which enables one AA to anticipate the information needs of other AAs and triggers the proactive information delivery.

The capability of manipulating information dependency enables one AA to infer indirect information needs of teammates. For example, the lethality of a chemical weapon depends on weather conditions like the wind direction, the humidity, etc. If an AA knows such a dependency and that the analyst needs to estimate the lethality of the weapon under the current situation, it can infer that the analyst implicitly needs information like the wind direction, the humidity and other factors. The AA can also synthesize multiple pieces of information based on information dependency and present only the fused information to the human analyst to reduce the information load.

3. Semantic Web

The use of a Semantic Web (Antoniou and Harmelen 2004) infrastructure allows the agents to proactively share fine-grained information (e.g., a person’s citizenship or the range of a weapon) from documents rather than whole documents. The Web Ontology Language (OWL) became a World Wide Web Consortium standard in 2004. OWL supports more powerful queries than database technology via reasoning and inference over relations expressed in ontologies (e.g., subclass, subproperty, transitive...). We are currently using commercial Semantic Web products from Network Inference to support complex queries from heterogeneous sources as shown in figure 1. Domain specific ontologies are created via the Construct tool. Construct allows you to link classes (e.g., TerroristPerson) and properties (e.g., alias, citizenship, age, etc) in ontologies to instance data (e.g., a terrorist named Amed al Basra) stored in databases so that you can avoid having to load all instance data into the knowledgebase. Relevant upper and domain specific ontologies are loaded into Cerebra server. Text documents are parsed by AeroText (a natural language processing tool developed by Lockheed Martin) and the results are stored in databases. Assistant agents can now submit queries that are answered by reasoning over all ontologies and instance data. For example, find all known members of terrorist cells in south west Iraq.

![Figure 1. Semantic Web Infrastructure](image)

User profiles for analysts are entered by selecting items from ontologies that are presented in hierarchical trees. The analyst selects classes and properties of classes that are relevant to her role. The assistant agent uses these profiles to proactively retrieve and share fine-grained information relevant to the analyst’s interests as discussed in previous sections.

4. Conclusion

In solving homeland related tasks under time pressure, multiple HLS analysts with different expertise often need to work together and proactively share necessary information. We developed software assistant agents based on a novel agent architecture called CAST to support such a team of analysts. By establishing and maintaining shared mental models with the human analysts, those assistant agents can dynamically anticipate information needs of analysts, and collaborate on retrieving and presenting relevant information to them. In addition, the Semantic Web reasoning infrastructure enables agents to more effectively query and share fine-grained information from a well organized knowledgebase.

References


