Presented at: Proc. of WebNet '97, Nov., 1997.

An Adaptive Multi-Agent Architecture for the ProFusion* Meta Search System

Yizhong Fan
Sprint Corporation
yizhong.fan@mail.sprint.com

Susan Gauch
Department of Electrical Engineering and Computer Science
University of Kansas
sgauch@eecs.ukans.edu

*http://profusion.ittc.ukans.edu/

The goal of this project is to develop an intelligent, adaptive Web search tool. Our work is based on ProFusion, a Web meta-search engine developed at the University of Kansas. ProFusion analyzes incoming queries, categorizes them, and automatically picks the best search engines for the query based on a priori knowledge (confidence factors) which represents the suitability of each search engine for each category. It uses these confidence factors to merge the search results into a re-weight list of the returned documents, removes duplicates and, optionally, broken links and presents the final rank-ordered list to the user. The main goals of the current research are to 1) provide ProFusion with a multi-agent architecture which is easier to extend, maintain and distribute and 2) to include automatic adaptation algorithms to replace the hard-coded a priori knowledge.

The multi-agent system consists of four different types of agents, namely, a dispatch agent, a search agent, a learning agent, and a guarding agent. The *dispatch agent* communicates with the user and then dispatches queries to the search agent and the learning agent. The *search agent* interacts with the underlying search engines and is responsible for reporting search results, confidence factors, and time-out values of the underlying search engines to the dispatch agent, as well as invoking the guarding agent when necessary. The *learning agent* is in charge of the learning and development of the underlying search engines, in particular adjusting confidence factors. The *guarding agent* is invoked when a search engine is down and it is responsible for preventing the dispatch of future queries to a non-responsive search engine as well as detecting when the search engine is back online. *Figure 1* shows the control flow and intercommunication between agents in the ProFusion system.

Our multi-agent architecture demonstrates various desirable agent characteristics [Mae 1994] including: task-oriented modules, task-specific solutions, de-emphasized representations, decentralized control structure, and learning and development. The search agent, learning agent, and guarding agent each consists of a set of 6 identical competence modules, each of which is responsible for one of 6 underlying search engines (task-oriented modules). These competence modules are self-contained black boxes which handle all the representation, computation, "reasoning", and execution that is necessary for its particular search engine. Although all 6 competence modules for each of the 3 agents are implemented using identical code, each uses its own local configuration and knowledge files to achieve its competence (task-oriented competence). In other words, there is no central representation shared by the several modules. Instead, every task-oriented module represents locally whatever it needs to operate autonomously. The localized representations of different modules are not related (de-emphasized representations).

Figure 2 illustrates the ProFusion multi-agent system architecture view. This architecture is highly distributed and decentralized. Each search engine keeps its competence modules and local representations in a

separate directory. Except for the dispatch agent, all of the competence modules of the search agent, learning agent, and guarding agent operate in parallel. None of the modules is "in control" of other modules (decentralized control structure). Because of this distributed operation, the new system is able to react quickly to changes in the environment and make the corresponding adjustments.

The adjustments are made by the learning agent which uses adaptation algorithms. The new ProFusion adapts to changes in search engine's performance, to changes in search engine's response time, and to changes in search engine's result formats. The adaptation to performance is achieved by observing user behavior to provide feedback which dynamically changes the performance knowledge base, the adaptation to response time is achieved by using dynamically changing time-out values, and the adaptation to result formats is achieved by using a dynamic extraction pattern, or in other words, a parser.

With this adaptive multi-agent architecture, the ProFusion system is now more competitive in the dynamic Web environment since it automatically adjusts to changes in its environment. ProFusion is also much easier to maintain and extend because it no longer requires a priori knowledge of a new search engine's confidence factors for each category (this will be determined by the learning agent). Since the search agent incorporates a parser, no more custom code is needed for extracting the search results, only a description of the language the search engine currently speaks.

References

[Mae 1994]. Pattie Maes, "Modeling Adaptive Autonomous Agents", Artificial Life Journal, edited by C. Langton, Vol. 1, No. 1 & 2, pp. 135-162, MIT Press, 1994.