Hybrid Multiagent System for Automatic Object Learning Classification

Ana Gil, Fernando de la Prieta, and Vivian F. López

University of Salamanca, Computer Science Dpt,
Plaza de la Merced s/n, 37007Salamanca, Spain
{abg,fer,vivian}@usal.es

Abstract. The rapid evolution within the context of e-learning is closely linked to international efforts on the standardization of learning object metadata, which provides learners in a web-based educational system with ubiquitous access to multiple distributed repositories. This article presents a hybrid agent-based architecture that enables the recovery of learning objects tagged in Learning Object Metadata (LOM) and provides individualized help with selecting learning materials to make the most suitable choice among many alternatives.

Keywords: learning object metadata, learning object repositories, federated search, e-learning, emerging e-learning technologies, neural networks.

1   Introduction

One of the most widely accepted approaches within the context of e-learning is based on fragmenting the content into modular and self-contained units that can be reused in different environments and by different applications. The term learning objects (LO) is used to refer to these units. A LO can be considered a digital resource that is particularly apt for forming part of a course or other type of learning experience.

One of the characteristics of LOs is that, by adding metadata to resources, they can be more easily managed. This means that metadata are created independently from the resource to which they are joined, in order to turn them into LOs.

LOs are placed inside repositories so that they can be more easily stored and retrieved. The LO Repositories (LOR) are software systems that can store metadata either alone or together with educational resources Generally LORs provide some type of interface that allows for the recovery of LOs. Any interaction involved in the recovery of LOs can be carried out manually or automated across different software, such as an Agent architecture, or even by treating the LOs as Web Semantic Services.

LORs have a high degree of heterogeneity in their characterizations with the coexistence of different standards and definitions. This implies a need to formalize the common repositories architecture while making them more flexible. Additionally, LOs have the possibility of being stored with different metadata formats addressing different types of conceptualizations.

This paper is structured as follows: section 2 explains the main concepts and characteristics that establish learning objects as the fundamental base within the current context of web-based e-learning. Section 3 introduces the proposed multi-agent architecture and the mechanism applied to retrieve LOs from different repositories using a Service Oriented Architecture (SOA). These LOs will be processed according to certain classification criteria that have been personalized and are considered most appropriate for the user. We conclude with section 4, which explains some of the more relevant aspects and work in progress.

2   The Actual Context of the E-Learning

2.1   Learning Objects

The concept of learning objects has evolved into a central component within the current context of e-learning with web-based learning technology. The Learning Technology Standards Committee (LTSC) from the Institute of Electrical and Electronics Engineers defines a learning object as "any entity, digital or non-digital, which can be used, re-used or referenced during technology supported learning". The generality of the definition can give way to practically any educational resource being considered a LO. As a result, the IEEE´s definition has been severely criticized since it does not clearly distinguish or identify what an LO actually is. By not agreeing on a universally accepted definition for a learning object, there has been a proliferation of ideas to define and delimit the boundaries of the concept [1, 2, 3, 4].

(Chiappe et al., 2007) recently described[5] a learning object as a digital, self-contained and reusable entity with a clearly instructional content, containing at least three internal and editable components: content, learning activities, and elements of context. Additionally, learning objects should have an external information structure, the metadata, which can facilitate its identification, storage and retrieval.

Given all of these possible definitions, it is possible to arrive at a certain consensus regarding LOs: they must be a minimal content unit (self-contained) that intends to teach something (instructional purpose) and can be reused (resability) on different platforms without any compatibility problems. It is essential that the LOs contain information that allows them to be searched for and identified through automatic recovery techinques that facilitate the task for which they were created, enabling a single object to be used at a low cost in different levels and educational disciplines.

Existing standards and specifications about learning objects focus on facilitating the search, evaluation, acquisition, and reuse of learning objects so that they can be shared and exchanged across different learning systems. The most notable standards used for tagging LO with metadata are Dublin Core [6], MPEG-7[7] and, most importantly, Learning Object Metadata(LOM)[8].

Since 2002, the IEEE LOM has been the standard for specifying the syntaxis and semantics of learning object metadata. It uses a hierarchical structure that is commonly coded in XML, and includes element names, definitions, data types, taxonomies, vocabularies, and field lengths. LOM is focused on the minimal set of attributes needed to allow these learning objects to be managed, located and evaluated. LOM metadata descriptions support version management and maintenance,

resource storage and recovery (searching, location, instantiation, packaging, editing,etc.) and resource sharing.

2.2   The Learning Objects Repositories

In an attempt to facilitate its reusability, LOs are stored in public and private LOR. The previously mentioned LOR are highly heterogeneous, each with a different storage system, access to objects, query methods, etc. The heterogeneity is not in and of itself a problem, since there are currently different systems that are interoperable. [9]. One of the most important systems that has been increasingly used as an interface is SQI (*Simple Query Interface*) [10], which was normalized by CEN in 2005[11] and has been well defined by three APIs: Learning Objects Interoperability Framework, Authentication and Session Management and Simple Query Interface Specification.
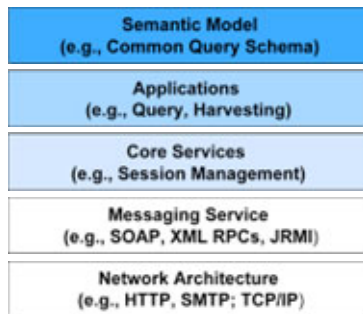

Fig. 1. Learning Objects Interoperability Framework

SQI is an abstraction level between the internal logic of a repository and the different external client; it is a middleware defined generally enough to be used in different fields, independent of technology and protocol. However, this definition is not only generic on a technical level, but on a conceptual level, allowing different types of queries (synchronous and asynchronous) and user requests. Even more important is the fact that it does not define any specific query language or LO packaging.

The basic functioning of a SQI interface is trivial; it is based on web services through which a client queries a LOR, usually in Very Simple Query Language (VSQL) [10] or Prolean Query Languaje (PLQL)[12]. The LOR then returns the LOs, usually packaged according to the LOM standard. This simple concept gave way to the birth of new types of applications dedicated to a federated search for learning objects in repositories. This software is used to perform simultaneous queries in different repositories, allowing a greater interoperability and, as a result, a better reusability of the resources where they are stored. As a result of these search applications, the topology of LO search systems has changed drastically. Figure 2 provides a graphical representation of the following classification for search systems:

- **Autonomous repositories.** Those that do not have a system allowing external searches and, as a result, require manual searches.
- **Repositories.** Those that have an external search interface and can be included in an automatic search system.
- **Repositories with Federated search system.** Those that, in addition to performing internal searches, can also perform automatic searches in other repositories.
- **Federated search systems.** Systems that can perform federated searches in different repositories; have the advantage of being able to perform filtering, cataloguing, etc. [13].
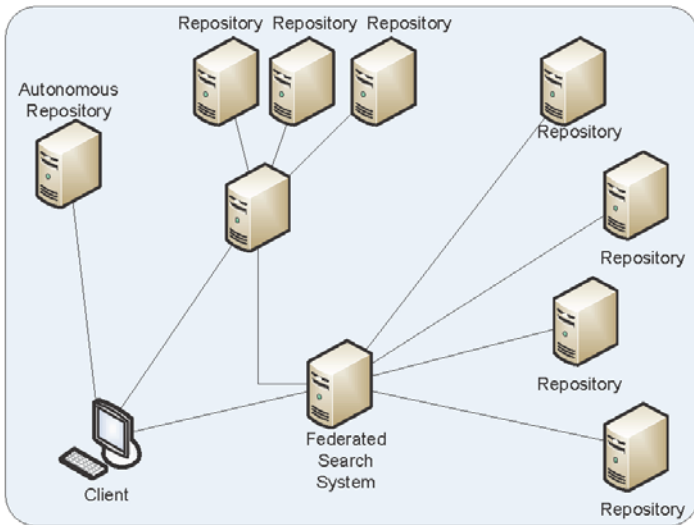

Fig. 2. Topology of Learning Object Repository

Because of continual research in search systems, the ability to create standardized and interoperability processes that can be applied to recovering LO has made it possible to formalize search and retrieval processes for LO in different repositories. One clear cut example would be the SILO search engine (the current Indexation and Query tool for learning objects) from the ARIADNE infrastructure [14]. Nevertheless, there are still many differences to overcome, among which the most important are:

- Excessive response time for the repositories.
- Limited number of results.
- An elevated percentage of errors when accessing the repositories, which is primarily due to the numerous occasions that the repositories either do not respond to the queries, or are simply not functioning, as shown in figure 3.

Beyond just the functioning of the repositories, which is partly due to the fact that they are young systems in constant evolution, is the fact that they lack other basic characteristics that are expected of any general search engine, such as classification
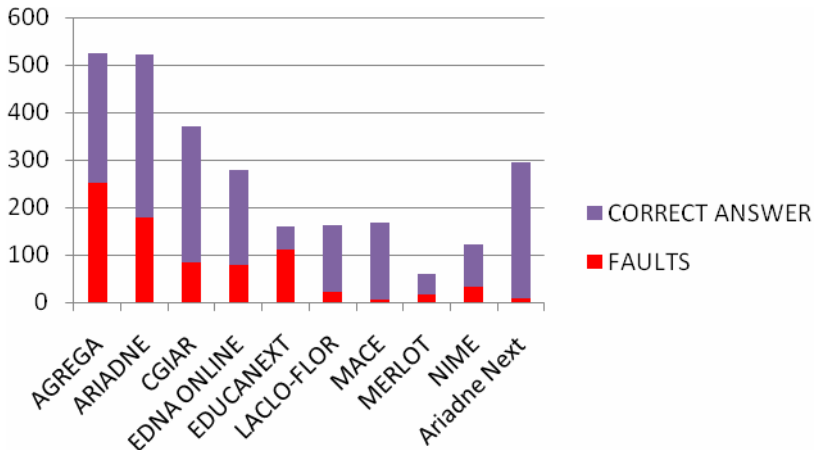
Fig. 3. Faults vs. correct answer in the LOR retrieval

tasks, sorting results, the use of different filtering techniques (such as the collaborative technique), the automated management of repositories and the extraction of statistics that serve to improve the global query process.

3 Hybrid Multiagent-System for Automatic Classification

Tools used to search and find Learning Objects in different systems do not provide a meaningful and scalable way to compare, rank and recommend learning material. As a solution to the problems observed with LOR search systems, we propose a hybrid system that integrates an agent based architecture that will serve to solve the issue of the federated search in repositories storing learning objects, and a neural network that sorts the obtained results.

The system will be designed with the primary goal of performing simultaneous searches in various LOR, with a subsequent filtering and sorting process based on criteria related to the quality of the recovered elements. The system design will also include secondary objectives such as a series of characteristics that can be found in any search system; the idea being to attempt to homogenize the heterogeneous environment previously presented in this article. The system will establish uniformity in the automated management repositories, incorporating a search history and a statistical system. Each of these functionalities complements the search tool and facilitates its use in the educational sector.

The system architecture is composed of two basic blocks: the interface, and the search system. These two blocks comprise the foundation of the system´s functionality. The primary interface is used for the communication between users and the search tool. A critical function in this block is the ability to take statistical data and used it to subsequently sort the results. The search system constitutes the core of the application and simultaneously performs tasks involving communication, extracting metadata, quality control, and sorting the LO.

Both blocks are designed using a hybrid system comprised of agents and a neural network that sorts the results. The agents are responsible for the communication, task flow, quality control, extrapolating statistical data, etc. The agents can modify their behavior to find the best solution for a problem, adjusting their behavior according to the knowledge they have acquired, a series of statistical data that they gather during each interaction with the repository containing learning objects, and what the end-user does with the results that have been provided.

The following list explains the pre-defined agents that provide the basic functionalities of the architecture, as illustrated in Figure 4:

- **Repository agent.** This agent is responsible for performing searches with the various repositories, extracting metadata, quality control for the LE received, and optimizing the search system. There will be one agent for each of the repositories so that multiple searches can be performed simultaneously.
- **Sort agent.** Responsible for verifying, controlling and coordinating the results from the neural network, and classifying and cataloguing the results.
- **Statistical agent.** This agent is responsible for gathering the statistical data obtained from the repositories and the interaction between the users and the search tool. It also provides the supervisor agent with the appropriate statistical data needed to effectively coordinate the tasks.
- **Supervisor agent.** Responsible for supervising the other agents, and for coordinating tasks. It obtains data from the statistical agent and adapts the tasks to the system according to different variables, such as the state of communication, the system load, etc.

A neural network is used for the sorting process, since it is specially designed for classification tasks such as those involved in the present study. In order to carry out the sorting process, it is necessary to establish a ranking system for the LO, indicating the rank for each LO. According to each position, the sort agent is responsible for presenting the results to the end-user working with the application. The learning process for
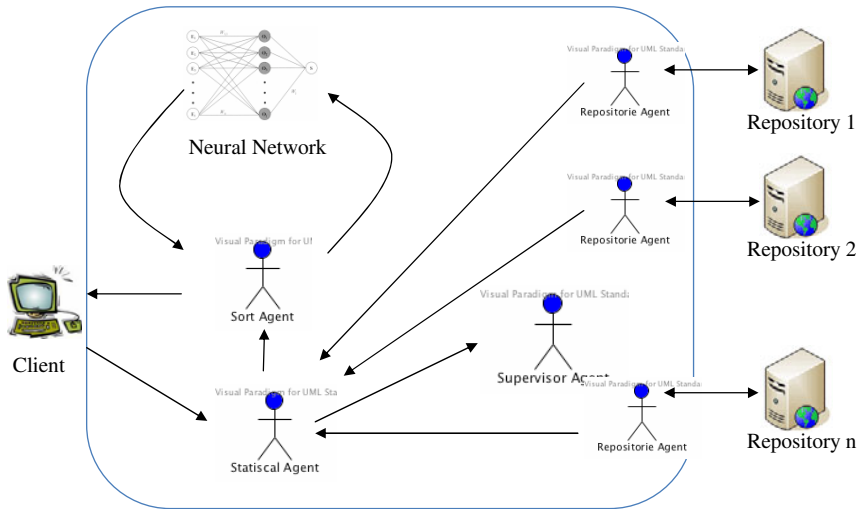

Fig. 4. Agent-Based architecture

the network is supervised since the data used are gathered from the iteration with the users to evaluate the proposed ranking, and the weight is adjusted for each iteration.

4 Results and Conclusions

The search and location services for educational content, and specifically LO, presented in this paper constitute the core of the development of distributed, open computer-based educational systems. For this reason the research in this area has been so active in recent years.
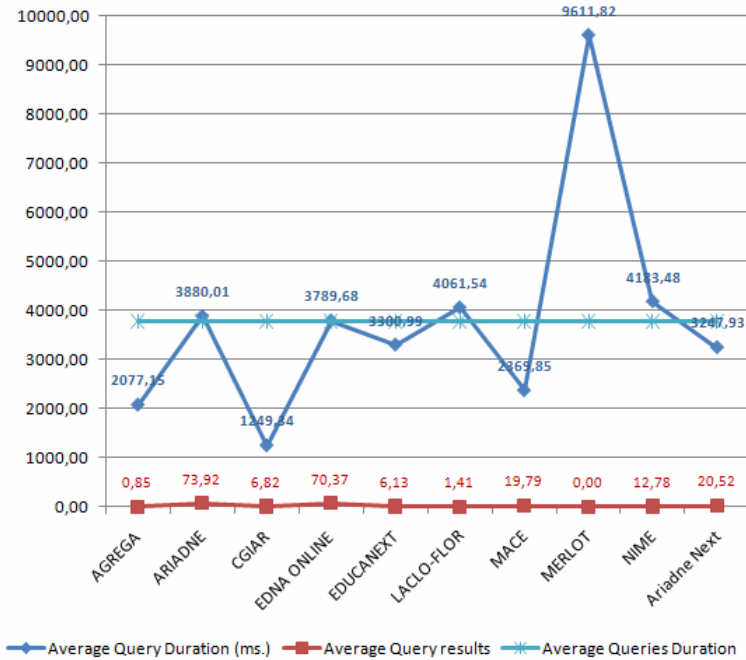

Fig. 5. Query duration and Query results

The design of the agent based architecture that we have constructed is ideal for solving problems that have been noted in LOR. It allows the system to adapt according to the workload and the state of the repositories with respect to the statistics data. One of the most significant advances that has been achieved is the reduction in the response time for the final results of the application, despite the high response time for LORs. Figure 5 clearly demonstrates these results.

The sorting system proposed is also very convenient, given that the LOM standard does not define a minimal set of fields that a LO must have; this makes it difficult to evaluate if a LO has a sufficient quality. Using the feedback provided by the users, from the daily use of the application, the network goes through a learning process, which allows it to continually improve its results.

If we consider the results obtained with the proposed system, we are confident that the adaption and learning features of an agent-based architecture makes it ideal for solving federated search problems in heterogeneous repositories.

References

1. Friesen, N.: Three Objections to Learning Objects. In: McGreal, R. (ed.) Online Education Using Learning Objects (2004), World Wide Web `http://learningspaces.org/n/papers/objections.html` (retrieved February 15, 2010)
2. Sosteric, M., Hesemeier, S.: When is a Learning Object not an Object: A First Step towards a theory of learning objects. International Review of Research in Open and Distance Learning 3(2) (October 2002)
3. Wiley, D.A.: Connecting learning objects to instructional design theory: A definition a metaphor, and a taxonomy. In: Wiley, D.A. (ed.) The Instructional Use of Learning Objects. Association for Educational Communications and Technology, Bloomington (2001), World Wide Web `http://www.reusability.org/read/` (retrieved March 23, 2009)
4. Polsani, P.: Use and Abuse of Reusable Learning Objects. Journal of Digital Information, Article No. 164, 3(4) (2003)
5. Chiappe, A., Segovia, Y., Rincon, H.Y.: Toward an instructional design model based on learning objects. Educational Technology Research and Development 55, 671–681 (2007)
6. DCMI Specifications, `http://dublincore.org/specifications/` (retrieved February 18, 2010)
7. MPEG-7, MPEG Home Page, `http://mpeg.chiariglione.org/standards/mpeg-7/mpeg-7.htm` (retrieved March 29)
8. IEEE 1484.12.1-2002, Final Draft Standard for Learning Object Metadata (LOM). The Institute of Electrical and Electronics Engineers, Inc., `http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf` (retrieved February 23, 2010)
9. Dagger, D., O'Connor, A., Lawless, S., Walsh, E., Wade, V.P.: Service-Oriented E-Learning Platforms: From Monolithic Systems to Flexible Services. IEEE Internet Computing 11(3), 28–35 (2007)
10. Simon, B., Massart, D., van Assche, F., Ternier, S., Duval, E., Brantner, S., Olmedilla, D.: A Simple Query Interface for Interoperable Learning Repositories. In: Workshop on Interoperability of Web-Based Educational Systems in conjunction with 14th International World Wide Web Conference (WWW 2005), Chiba, Japan (May 2005)
11. European Committe for Standardization, A Simple Query Interface Specification for Learning Repositories (November 2005)
12. Ternier, S., Massart, D., Campi, A., Guinea, S., Ceri, S., Duval, E.: Interoperability for Searching Learning Object Repositories. The ProLearn Query Language. D-Lib Magazine 14(1/2) (January/February 2008)
13. De la Prieta, F., Gil, A.: A Multi-agent System that Searches for Learning Objects in Heterogeneous repositories. In: Demazeau, Y., et al. (eds.) Trends in Practical Applications of Agents and Multiagent Systems: 8th International Conference on Practical Applications of agents and multiagent systems (PAAMS 2010). Advances in Intelligent and Soft Computing, pp. 355–362. Springer, Heidelberg (April 2010)
14. Ternier, S., Verbert, K., Parra, G., Vandeputte, B., Klerkx, J., Duval, E., Ordóñez, V., Ochoa, X.: The Ariadne Infrastructure for Managing and Storing Metadata. IEEE Internet Computing 13(4), 18–25 (2009)