A New Merging Algorithm Based on Semantic Relationships of Learning Objects

Elio Rivas-Sanchez[*1], **Jesús Serrano Guerrero**[2]**, Víctor Hugo Menéndez**[3]

[1]Universidad Nacional de Ingenieria, Managua, Nicaragua
[2]Universidad de Castilla la Mancha, Ciudad Real, Spain
[3]Unversidad de Yucatan, Merida, Mexico
E-mail: eliocarmen@hotmail.com

RESUMEN

Los Objetos de Aprendizajes son elementos muy importantes dentro de entornos de aprendizajes "e-learning" o bien cuando se utilizan recursos digitales para la educación porque describen el material educativo para los estudiantes, estos permiten la reutilización y la posibilidad de compartirse en diferentes sistemas de gestión del aprendizaje. Habitualmente cuando los docentes necesitan crear y estructurar experiencias educativas, ellos acuden a repositorios para recuperar recursos ajustados a sus intereses particulares para reducir el esfuerzo y el tiempo invertido en su elaboración. En este artículo, es presentado un método para la fusión o mezcla de Objetos de Aprendizajes recuperados desde repositorios heterogéneos; El modelo está basado en las relaciones semánticas que guardan los Objetos de Aprendizajes entre ellos los cuales son recuperados a través de un Meta-Buscador, como una alternativa para la localización o búsqueda efectiva de recursos educativos ajustados a los intereses de los docentes. El modelo expuesto en la propuesta ha sido implementado como prototipo inicial el cual recuperar Objetos de Aprendizajes de repositorios abiertos. Los resultados iniciales del estudio confirman la utilidad y efectividad del modelo.

Palabras Clave: Objetos de Aprendizajes; Algoritmo de Mezcla; Meta-Búsqueda; Relaciones Semánticas.

ABSTRACT

Learning Objects are key elements within e-Learning environment because describe the created educational material for students, besides, it permits the reusing and sharing in di_erent Learning Management Systems. Usually, when teachers need to create and structure educational experiences, they attend to repositories for retrieving resources fitted to their interest, for reducing the e_ort and the computational time. In this paper, a proposal is presented for merging Learning Objects from heterogeneous repositories; the model is based on semantic relationships between Learning Objects retrieved from a meta-search engine, as an alternative for locating fitted educational resources for teacher's interest. The model exposed in the proposal has been implemented as initial prototype, which retrieves Learning Objects from open repositories. An initial study results confirm the usefulness of the model.

Keywords: Learning Objects, Merging Algorithm, Meta-Search, Semantic Relationship.

---

* Autor para la correspondencia.

1. INTRODUCTION

E-Learning environments and Information Retrieval Systems promote the digital learning resource creation for structuring educational experiences for the student. Hence, teachers supported by learning tools are able to develop educational resources for specific purpose. However, constructing and structuring these digital resources imply a significant effort, because these resources are located in several repositories, implying that teachers need to attend to many search engines simultaneously. To reduce this effort, teachers are supported by educational content repositories available on Web, where they are able to find, reuse, and choose educational resources, considered as adequate to their educational objective.

Therefore, the search of educational resources of relevant content in repositories and search engines is a key task, to develop a successfully educational material for students. The teachers, besides searching, among repositories they must discriminate manually the useful of the found objects. So this task is hard and exhaustive, besides the wasted of time, and most of the time the retrieved Learning Objetcs are incorrect.

A possible solution to this drawback is the using of the meta-search, this approach groups several search engines and/or repositories into a unique search mechanism, like if them were one search engine, so the user would not have to invest time and effort in searching Learning Objects in several independents repositories and search engines, having all the databases of objects integrated into an unique search mechanism.

The outline for this paper is as follows: Section 2 related Los searching, sharing and reuse initiatives; Section 3 the approach for merging and locating Learning Objects based on semantic relationships, Section 4 experimental design on information retrieval measures, result and discussion and Section 5 conclusion and future work.

2. STATE OF ART

2.1. Learning Objects

The using of digital Learning Objects in e-Learning environmentis essential and plays an important role, because they allow to teachers, to structure activities and educational experiences for the student. For this paper argues that a Learning Object (LO) is a knowledge component, consisting of a collection of digital files and metadata that describe it in technical and pedagogical terms [1]. The purpose of the LO is to provide a component model based on standards that allow flexibility, platform independence and reuse of content [2]. Metadata are fundamental to those aims because allow the user to classify, locate, develop, combine, install and maintain objects [3]. The SCORM specification [4] and the IEEE-LOM standard [5] have contributed to standardization of metadata used to describe the Learning Objects making feasible implementing the proposal.

2.2. Metadata

IEEE-LOM is the current standard for Learning Objects, which describes a set of labels allowing to represent the metadata of a Learning Object. These metadata have an orientation towards the learning and instruction, but insufficient for the needs of several Learning Management Systems. The table 2.2 shows the metadata classification and their intrinsic features.

In this standard, the labels can be filled with two types of values: the values corresponding to controlled vocabularies and free text. The labels are formalized into a XML multi-schema that the specification implements, therefore the metadata of a LO are associated with creating an instance of XML multischema defined. Besides another important characteristic of the LOs, is that they can be exchange between e-Learning platforms without loss of structure and information.

Table 2: Scheme of metadata relation

Metadata				Data type
Relation	Kind	Relationship		Controlled Vocabulary
	esource	Identifier	Catalog Entry	Langstring
		Description		Langstring

The relationships between Learning Objects are defined by IEEE LOM standard, which establishes that relation metadata group defines the relationship between a Learning Objects and other Learning Objects, if any. To define multiple relationships, there may be multiple instances of this category. If there is more than one target Learning Object, then each target shall have a new relationship instance.

The relationships between objects are stored in their relation metadata group, but are not always present, because the standard has not defined it as a required metadata to fulfill. The relation metadata group is composed at the same time by two main elements such as

70

kind and resource, which permit to define and locate the target resource (see table 2). kind identifier is the metadata in which relationships type are stored, twelve relation- ships are define taken by Dublin Core Standard [6], resource metadata is decomposed by Identifier and Description, Identifier metadata are composed by Catalog and Entry which are the IDs of the target related resources, and Description metadata is a description of the target Learning Object.

2.3. Learning Objects Repositories

The Learning Objects Repositories (LOR) are one type of digital libraries, specialized in educational resources and are based on metadata standard. They are technologically prepared to interoperate between others repositories and e-Learning applications [7]. Learning Objects Repositories are available tools for developing e-Learning courses and store completed educational experiences, this strength is exploited by e-Learning teachers because they take advantage of the available educational resources than these repositories expose for reuse and sharing [8].

The LORs play an important role because they stored the educational resources, besides for publishing an educational re- source is needed to fulfill many quality requirements of information completeness, involves defining a values collection that permit to establish the title, size, format, educational purpose, audience, difficulty, etc. These values or descriptors can discern the usefulness of the resource for any specification or educational need.

The quality of description or documentation of LOs is one factor that influences in their location, reuse and search [9]. The repositories establish many quality criteria to have the most metadata information completeness, helping users to discriminate the usefulness of LOs. This information completeness can help to metasearch engines that links several heterogeneous repositories, in order to give the higher quality Learning Objects to users.

One of the main problems of LOs is that they are stored in different LORs or databases. For that reason, tools such as meta-search engines can be an interesting solution because they are able to send requests and connect simultaneously on several search engines, LORs or databases, and merge the results from all of them into a unique list. Hence the results of this kind of tools are more complete due to it is possible to extract the best LOs from each source.

2.4. Metasearch engines

Metasearch engines are powerful search tools; they use several mechanisms for information retrieval. Metasearch engines have become important, since the wide available repositories and databases; due the fastest growth of information. The available information resources have become unmanaged; this situation is turning difficult for the users. A metasearch engine is an alternative solution because they can execute several searches simultaneously, sending requests to several search engines in real time, for after merge the obtained results into one combine list [10].

Metasearch engines are widely known both for general and specific purposes, it's retrieve better results when is employed for specific domains; it's strength lies on it's three components which are: Database Selector, Query Dispatcher and Result Merger, these are detailed in following:

1. Database Selector: The main goal is to choose and exe- cute queries simultaneously, in suitable search engines or databases for the requested information.

2. Query Dispatcher: The Query Dispatcher receives the query and extracts its members: terms and operators. This phase implements several strategies to transform the original query to a set of related queries. At last, the final query set is sent to the selected search engines to execute the search.

3. Result Merger: After the results have been processed, the objective is to take the whole results into a merging and ranking algorithm, and finally, present the results in a final combined list to users.

The figure 4 shows the basic metasearch engine architecture which establishes connection to several search engines. The shown architecture has been taken from [11].

A key component within the Result Merger is the merging of the retrieved resources. Once the results from various search engines are collected, the metasearch engine merges them into a single ranked list. The effectiveness is closely related to the result merging algorithm it employs [12].

2.5. Metasearch engines for Learning Objects

As was mentioned above about the key phases of a metasearch engine, we will summarize in the following some approaches found in e-Learning environments.

2.5.1. Selection of Databases

For a metasearch engine, the selection of the database is an important factor, because it will help to choose the data sources, where educational resources are located. In this stage [13] a metasearch engine has been developed as a federated solutio, called SEARCHY, which is a multi-agent metasearch engine that uses Dublin Core as the metadata standard to search and describe documents.
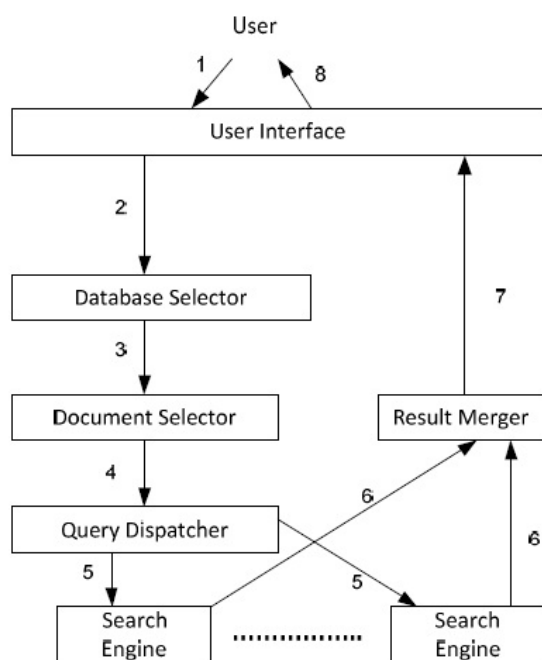
The above approach is a solution for search Learning Objects in a federated network, also integrates several databases and providers in order to translate the user query, into the particular format for each database provider, taking into account the communication and information exchange protocols.

The ARIADNE is a distributed repository for Learning Objects. It encourages the share and reuse of such objects. An indexation and query tool uses a set of metadata elements to describe and enable search functionality on Learning Objects. To increase the interoperability of ARIADNE with other repositories within the LOM community, ARIADNE represented metadata according to the LOM standard, which enables other repositories to share this metadata. In spite of having search federated tool, a weighting algorithm is not explicit for merging and ranking Learning Objects.

In the ARIADNE experience [14] the core initiative consist in the integration of several repositories among federated instance, in this sense, some problems like communication technologies and standardization of protocol exchange information can be try homogeneously. Also a unique search mechanism can be used for searching in the whole available educational repositories [8], which unify ARIADNE and other Learning Object Repositories that are based on the standard IEEE LOM.

2.5.2. Merging and Ranking Algorithm

Due to the merging and ranking strategies are essential, because these manage the final results returned to users. In [15] a metasearch engine approach for Lecture Notes has been developed called LESSON, which final ending, consist in merging results from multiple components search engines into a single ranked list, the RSF strategy was also designed, which takes into account rank, similarity and features of lectures notes.

The LESSON approach also takes in count the Round-Robin (RR) merging strategy, which arranges the elements from the result lists of all components engines in ranking order, from the top to the bottom in turn. This strategy only improves recall not precision, to overcome this drawback: rank, similarity and features are merged.

In [16] the main idea is to push users motivation to submit suggestions and comments, in order to assign credits to users and set a value cost for each Learning Object, permitting increase the valued of the most popular LOs, and also LOs rankings. The users gain credits when they submit LO, evaluate, review and add valuable information to existing LOs. Once the LO has been submitted and accepted to be published in the repository, reviewers submit their comments and rate the LO in the range [1-5]. The rate of the Learning Objects is based on the relevant opinion on LO's quality. Also the ranking mechanism is based on the purchases of the LOs. So LOs with higher purchases has a high value of rank.

Also metrics for determining the importance of relevance documents has been expose by [17] which proposes a several metrics in order to develop the concept of relevance in the context of Learning Object searching. The set of metrics, try to estimate the topical, personal, and situational relevance dimensions. However, the tools

used in the middle for search and find Learning Objects in different systems do not provide a scalable way to rank learning material. In order to resolve this drawback, there is a proposal using Contextual Attention Metadata (CAM) [18], this approach take in count the lifecycle of the Learning Object, ranking and recommending metrics to improve the user experience and retrieve the desired results that user is needing. Four types of metrics are detailed: Link Analysis Ranking, Similarity Recommendation, Personalized Ranking and Contextual Recommendation.

The importance to determine the relevance of the Learning Objects within a collection, is an approach widely addressed in [19], the relevance is modeled using the number of references, downloads of the object and similarities between objects, all this variables are captured, in a controlled environment, making use of the system called CORDA. The approach takes into account a data mining technique for the recovery of data.

2.5.3. Query Processing

As was mention above, the Query Dispatcher components receive the query and extract its members: terms and operators. Other additional process is to fit the query format for each particular search engine. In this stage, previous researches are concentrated for the interoperability between Learning Objects Repositories in order to achieve interoperability; the Simple Query Interface (SQI) was developed [20] as a universal interoperability layer for educational networks. The proposal can be used for sending several queries to a Learning Objects Repositories into a federated network for retrieving results from different sources. The SQI approach works on ARIADNE federated platform, having an user interface to retrieve the results.

2.5.4. Performance measures in information retrieval

Within the Information retrieval area, which task consists in searching for documents, for information within documents and for metadata about documents. Many different measures have been proposed, in order to evaluate the performance of information retrieval systems. These measures require a collection of documents and a query. The used measures to determine the usefulness of the propose model has been taken from the work called Building efficient an effective metasearch engines [11]. For purposes of this work we will use Learning Objects instead of

documents. The performance measures are the following:

1. Precision: Precision is defined as the proportion of retrieved material actually relevant, from the total number of retrieved documents. The formula to model this measurement is the following:

$$Precision = \frac{Number of retrieved relevant objects}{Total retrieved objects} \quad (1)$$

2. *Recall:* Recall is defined as the proportion of relevant material recovered from the total relevant documents in the collection of documents, regardless of whether they are recovered or not.

$$Recall = \frac{Number of retrieved relevant objects}{Number of relevant objects in collection} \quad (2)$$

3. *Average Precision (AveP):* The *Average Precision* tries to estimate the precision for ranked objects in list. From the retrieved documents, compute the position it's occupying, also if the document is relevant for the user assigning the value of 1 through a binary function P(r), until the last relevant document in collection.

$$AveP = \frac{\sum_{i=1}^{N} (P(r) * rel(r))}{Number of relevants documents} \quad (3)$$

Where *r* is the rank, *N* the number retrieved, *rel()* a binary function on the relevance of a given rank, and *P(r)* precision at a given cut-off rank:

$$Pr = \frac{relevant of retrieved documents of rank r or less}{r} \quad (4)$$

4. *Mean Average Precision (MAP): Mean Average Precision* is the mean of the average precision scores for whole queries.

$$MAP = \frac{\sum_{q=1}^{Q} AveP(q)}{Q} \quad (5)$$

Where *Q* is the number of queries.

3. A Model for Merging Learning Objects from Different Sources Based on Their Semantic Relationships

The semantics relationships between Learning Objects can provide information, about, how objects are related to other objects in a collection; this situation can denote which objects within a collection are important for a user. In order to model the importance of the related objects, the model takes into account the semantic relationships between objects. This model is defined for a series of metrics such as Weighted Completeness Index [21], which is a measure that determines the degree of metadata completion, the Chorus Effect [Thompson], which considers that a resource found in various lists is significant for the merging process and the Number of Relationships which considers the number of objects related with the target object. These factors give information for cataloguing the recovered objects when they have not semantic relationships among them.

The relationships between objects are created during generation and composition processes [23]. In generation, basic objects are created with a simple educational objective. New objects are created with a fine level of granularity for immediate use and achieve a specific instructional objective. The main activities for generation are: cataloging and storage. At this stage simple instructional resources are located (text documents, images, videos) and are labeled to have a set of basic objects.

The composition is the required process to build a new Learning Object from other ones; it implies recovery activities, processing and composition, therefore more complex LOs are created from other objects that already exist. This type of complex object known as SCO allows the exchange of information in e-Learning environments and pursuing a higher-level educational goal. The LO's creation task takes into account combining several educational resources; this task of composing object from other objects defines the granularity grade of an educational resource.

The final products of the processes of composition and generation are the combined resources. Simple or combined objects which describe the granularity level; therefore, the granularity is the size of the object created considering other [1].

The relationships between objects have a proportional relationship with a granularity degree, because objects with higher number of relationships have a higher degree of granularity and objects with smaller number of relationships have lower degree of granularity, therefore the model can be tuned in order to refine the search towards a certain granularity degree or with certain relationships.

In order to obtain the desired results, the model can be focused on objects that have some relationships types; in this case a weight value w_i for each relationship type has been assigned. The formula to model the weights is the following:

$$w_i = \frac{e^{1/n}}{max(e^{1/n})} \qquad (6)$$

Where n is the relationship position.

IEEE-LOM provides twelve relationships types for the relation metadata, the model groups these twelve relationships into six types, because the relationships are bidirectional, therefore they are grouped in pairs. In table table: 3, the weights are listed and ordered in decreasing order in terms of weights:

The higher weight is assigned to the *version* relationship because the model is considering retrieving the latest educational resources version to users. The weights are assigned to each element based on its definition and sorted based on their position. For example, the weight of *version* relationship means that a resource is an updated version of another application, the *part* relationship means that an object is a physical or logical part of another, so that the relationship *version* is more relevant since is a version of another, if only it were a part of it. Thus the object reference to *version* receives a higher weight of 1, against the reference *part* which receives a smaller weight of 0.44.

Table 3: Weights of the relationships pairs

| Position	Weight	Relationship pairs
1	1	version
2	0.60	base
3	0.51	require
4	0.47	reference
5	0.44	part
6	0.43	format

These weights can be adjusted to refine searches or improve performance of the retrieved results, using Machine Learning, which can provide and store the consulted resources and users behavior.

When objects are related, is possible under supervision watch the behavior they describe, in order to catalogue or provide a classification type of a related Learning Object within a collection. The model takes into account this behavior and provides a classification of 5 objects types, which are shown in following:

1. *Pointing (P)*: A LO *A* points at other object *B* in a collection of *N* objects. $A \rightarrow B$.
2. *Pointed (Pd)*: A LO *B* is defined as pointed, when being part of the collection of *N* objects, the object *A* points at the object *B*. $A \rightarrow B$.
3. *Pointed and Pointing (P,Pd)*: Given three LOs *A, B* and *C* within a collection *N* of objects, *B* is a pointed and pointing object, because *A* is pointing to *B* and *B* is pointing at *C*. $A \rightarrow B \rightarrow C$.
4. *Orphan (Or)*: An object that is not related with other Learning Objects.
5. *Repeated (Re)*: A LO present in several lists.

The figure examplerelation shows an example of the semantic relationships between objects and help to understand the classification made below. LO *A* in its metadata relation describes the type of relationship (*kind*) by the relationship *isversionof*, Also the related resource identifiers detailed (*identifier/catalog, identifier/entry*) which describes the LO *B*, thus the LO *A* is a pointing object. Also can be seen that LO *B* is pointing to the LO *C*, therefore *B* is a pointed and pointing object, and *C* is a pointed object.

The formula used to model the Weighted Completeness Index (Wci) is as follows:

Figure 2: Example of relationships between objects.
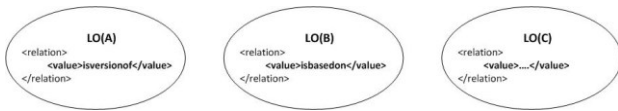
$$W_{ci} = \frac{\sum_{i=1}^{N} \alpha_i P(i)}{\sum_{i=1}^{N} \alpha_i} \qquad (7)$$

Where *P(i)* is 1 if the i-th metadata has a non-null value, 0 otherwise. *N* is the number of existing metadata in a LO, α_i is the relative importance of i-th metadata.

The formula to model the Chorus Effect (C_e) is as follows:

$$C_e = \frac{\sum_{i=1}^{L} A_i}{L} \qquad (8)$$

Where A_i is the number of occurrences of the i-th object. *L* is the number of lists retrieved by the metasearch engine.

The formula to model the number of related objects with the target object (N_r) is as follows:

$$N_r = \sum_{i=1}^{N} r_i \qquad (9)$$

Where r_i is the number of related objects with the target object.

For each Learning Object *O* inside a collection of *N* resources, and retrieved *L* lists, the formula to model the Relationship Index (RI) is as follows:

$$RI(O_iL_k) = \begin{pmatrix} Max(\frac{(\sum_{i=1}^{n} w_i)^2}{\sum_{i=1}^{n} (1+w_i)}) & O_i \in Pointing \cup Pointed \\ \\ 0 & O_i \in Orphan \end{pmatrix} \qquad (10)$$

Where O_i is i-th object in a collection of *N* resources, L_k is the k-th list in a collection, I_i is the relationship weight for each object related with O_i, *n* is the number of LOs related with object O_i.

The Index Merging permits calculate the importance for each Learning Object. Therefore, the Index Merging (I_m) is the combination of formulas (wci), (ce), (Nr) and (RI), giving as result a normalized metric for this index which is the following:

$$I_m(O_i) = \frac{(W_{ci} + (RI(O_iL_k) * N_r)/2 + C_e}{Max((W_{ci} + (RI(O_iL_k) * N_r)/2 + C_e)} \qquad (11)$$

3.1 *Examples*

In order to test the effectiveness of the proposed Model, two scenarios are shown in this section, one simple and one complex. These scenarios are at the ends and their importance lies to confirm if the Model is able to consider any given situation: LOs with and without relationships.

Scenario 1: Three lists with objects are retrieved from different sources with repeated Learning Objects and do not have relationships among them, so the Relationship Index and Number of Relationships are zero and the Model will only take in count the Weighted Completeness Index and Chorus Effect, to determine the importance of objects in the final combined list. In this scenario the Index Merging is the sum of both factors.

Table 4: Final scores for each object retrieved

No.	Objects	RI	C_e	W_{ci}	N_r	I_m
1	O_5L_3	1.15	0.25	0.62	7	1.00
2	O_5L_1	1.05	0	0.80	7	0.88
3	O_2L_2	0.92	0	0.73	6	0.67
4	O_2L_3	0.69	0	0.62	5	0.44
5	O_2L_4	0.77	0	0.85	4	0.42
6	O_5L_4	0.50	0	0.85	1	0.14
7	O_1L_1	0.36	0	0.80	1	0.12
8	O_4L_4	0.17	0	0.85	1	0.11
9	O_3L_4	0.14	0	0.85	1	0.10
10	O_3L_1	0.17	0	0.80	1	0.10

Scenario 2: There are four lists of Learning Objects, where 13 objects are pointed, 5 objects are pointed and pointing objects, 1 repeated object and one orphan object. The relationships between the objects are described in figure Scenario2. The type of each relationship is expressed by a line type. The boxes are grouping the lists of retrieved objects from each repository and each object is represented by a circle (the nomenclature O_iL_k means the object i in the list k).
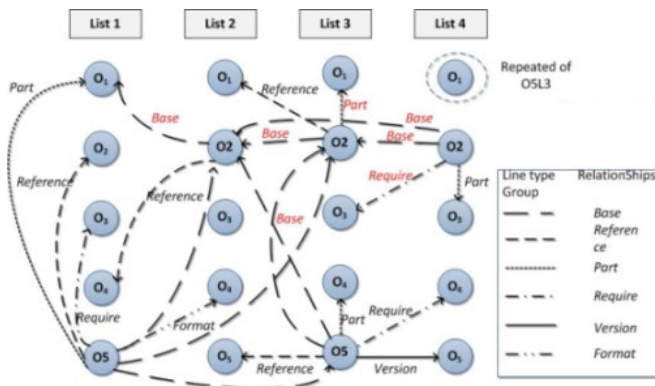

Figure 3: Scenario 2, relationships between objects.

The final step is calculate the importance for every object in the collection through Index Merging using formula (Im). Table table:rexamp shows the top ten objects, ordered by higher weights in a merged list and its positions in decreasing order in terms of Index Merging.

The examples have proved that objects with relationships of higher weight, high number of relationships, and high Weighted Completeness Index will be in higher position in the merged list (see seventh column in table table:rexamp). There is correlation among the final results and the factors employed in the model, such as weight of relationships, number of relationships and weighted completeness objects. For example the object O_5L_3 is in first position in list, because it has 7 relationships, the Relationships Index is 1.15, Chorus Effect is 0.25 and the Weighted Completeness Index is 0.62, in spite of the object O_5L_1 which is in second position in list, because it has 7 relationships, the Relationship Index is 1.05, Chorus Effect is 0 and the Weighted Completeness Index is 0.82. The objects without relationships are in the lower positions in term of Index Merging. Therefore the model has proved that will rise in first position in list, those objects with a high value in: number of relationships, weight of relationships, Weighted Completeness Index and Chorus effect.

The results show that it is necessary to encourage the filling of metadata in repositories, because objects more completed will have better position in list. Also when new educational resources are created from other objects, the numbers of relationships increase. When objects are used in several e-Learning environments, Chorus effect will have a higher value at the time of retrieving objects from different sources. Because the Model takes in count the Chorus effect and is a factor which add relevance to objects.

4. Experimental Work

4.1 A Metasearch Architecture for Learning Objects

From the model described previously [24], an architecture has been developed in order to determine the importance of an educational resource and retrieve the best results for user, therefore the system uses Learning Objects merging algorithm from different sources, taking into account the semantic relationships between Learning Objects, being supported by a metasearch engine [25]. The system consists of a flexible architecture and permits incorporate new modules and consulted repositories.

The metasearch engine system architecture describes the 8 following abstracts processes in particular (see figure

4), in which was conceived, these are described in following:

1. *User query*: The user information need is expressed in keywords, typed into the system interface.
2. *Repositories requests:*: The system send request to educational repositories available on the web, with the original keywords, adapting query format for each search engine of the repository.
3. *LO retrieval*: The objects retrieved by the search engines are captured and tried with their different information exchange formats.
4. *Vectorization*: The objects retrieved of each repository are inserted into a vector of vectors, giving as final result a matrix of objects and lists.
5. *Metadata standardization*: The platforms of educational repositories use different exchange information format, which can results as data inconsistence, for that reason all metadata are homogenized into a unique format.
6. *Metrics (W_{ci} , C_e, RI , N_r and I_m)*:The metrics of Weighted Completeness Index (W_{ci}), Chorus Effect (C_e), Relationship Index (RI), Number of Relationships (N_r) and Index Merging (I_m) are computed, in order to have information of the importance for each object and merge them in the final and combined list.
7. *Relationships creation*:Generally the retrieved objects do not have filled the relation metadata. For that reason are completed under supervision as first approach using the module: *Relationship Creator*.
8. *Objects returned*:The objects are ordered in decreasing order in terms of Index Merging, and are returned to users. After, the user can make his final objects selection.
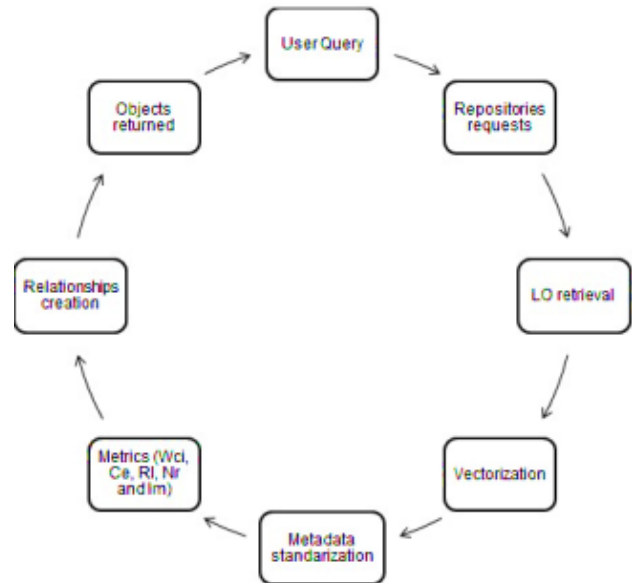

Figure 4: Metasearch engine architecture processes.

The whole processes described previously, are summarized in two core architecture modules, the Search in Repositories and Merging Process are describe below and are shown in figure 5.
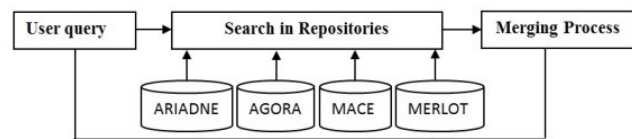

Figure 5: System General Processes.

4.1.1 Search in Repositories Module

Search in Repositories module consists in several processes which are described in details in the figure 6; the metasearch engine process begins from a user request through the typing of keywords in the system as *computing*. After the system have executed the queries in the repositories and have retrieved the Learning Objects. The system considers each retrieved list as an object's vector and merges all vectors into a matrix composed by lists and objects. At this moment the implemented prototype architecture retrieved Learning Objects from ARIADNE [14], AGORA [26], MACE [27] and MERLOT [28] repositories as first approach.

Search in Repositories is possible through making connection to heterogeneous repositories using an URL connection pointing directly to searching services of repositories available on Web.

4.1.2 Merging Process Module
The merging process consists of 2 stages:

1. The searching process above, transfers the matrix composed by objects and lists into the merging process, which consists in reading and store all objects into memory, and given that the retrieved objects from several sources and heterogeneous repositories are available in different standard formats, generally presents data inconsistencies when is needed to interpret them automatically for example, special characters. At this stage data are tried through a *metadata validator* module or debug process for homogenizing of formats and data, given the whole results as a homogenized objects vector.

2. The second stage is about that, each object from homogenized objects vectors is transferred into the following functions in order to determine: Weighted Completeness Index, Chorus Effect, Relationship Index, Number of Relationships and Index Merging. The Relationship Index is determined by the *Relationship Analyzer* sub-module using the relation metadata. At this stage exists the limiting that relation metadata is not filled, for this reason an additional Model *Relationship Creator* was created to complete the relationships between objects.

As final step, the merging algorithm in order to determine the importance for each object, use the Index Merging formula into the Merging Process for merging the whole objects into a unique and final scored list, sorted in decrease order in terms of Index Merging to finally present the list to users for the selection of Learning Objects.
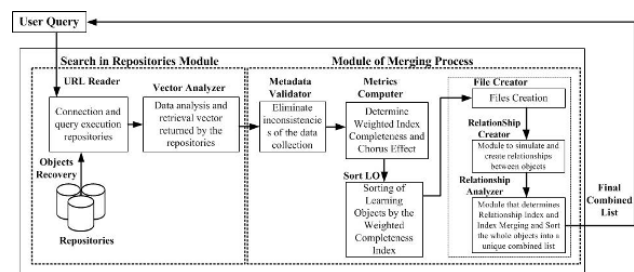

Figure 6: Architecture of the metasearch engine.

The proposed architecture is an initial developed prototype. The tool retrieves Learning Objects from the repositories of ARIADNE, AGORA, MACE and MERLOT as first approach, taking in count the

querying services exposed by the LORs. The JAVA based architecture, is the enough flexible to incorporate new connections of heterogeneous repositories using an URL connection.

For this first version, the prototype uses the JSON [29] and XML, as exchanges formats to retrieve Learning Objects from LOR, using their query services based on AJAX [30] technology. However, some queries protocols are been integrated into developing like SQI [20] and OAI-HMP [31].

4.2 Data settings

The purpose of this section is to evaluate the effectiveness of the proposed model for merging Learning Objects from different sources based on their semantic relationships, under the context of metasearch on e-learning environments. So we select four Learning Objects Repositories. They are: ARIADNE, AGORA, MACE and MERLOT. The reasons these repositories are selected are: (1) they are used by nearly all the e-learning community; (2) each of them has indexed a relatively large number of Learning Objects; and (3) they are integrated into a federated network in order to encourage the reuse of Learning Objects between e-learning environments. The results merging algorithm we proposed in this paper are completely independent of the repositories.

Each query is submitted to repositories. For each query user and each search in repositories, the whole results are collected (some repositories may return less than 20 results for certain queries). Totally there are 2,886 Learning Objects retrieved. The relevancy of each object is manually checked by the expert criteria, also based on the criteria specified in the object description and the narrative part of the corresponding query.

4.2.1 Evaluation criteria

Because it is difficult to know all the relevant documents to a query in a search engine, the traditional recall and precision for evaluating IR systems cannot be used for evaluating search/metasearch engines. A popular measure for evaluating the effectiveness of search engines is the Average Precision (AveP) [11] and the Mean Average Precision, which will be use in this paper in order to evaluate the effectiveness of the merging algorithm.

4.2.2 Results Analysis

In order to test the system performance we have sent 10 queries for each repository, in table table: 5 are listed the queries used.

The reader can see the results in table table: 6 of: 1) the retrieved objects for every query, 2) the total objects retrieved, 3) relevant objects for each query and 4) relevant objects retrieved.

Table 5: List of queries employed

No.	Queries
Q_1	Make Function Java
Q_2	Java Programming
Q_3	Management and Monitoring of Projects
Q_4	Principles of Marketing
Q_5	Agriculture and Nutrition
Q_6	Consequence of Climate Change
Q_7	Bussines Competitiveness
Q_8	Water Pollution
Q_9	Programming Visual Basic
Q_{10}	Greek History

The table table: 6 shows that ARIADNE, for the query number 2 *Java Programming*, retrieved 737 objects (see third column), 7 objects are considered as relevant in the whole collection (see fourth column). The same case for AGORA which retrieved 22 objects, only 3 objects are relevant, MACE retrieved 36 objects and 3 are relevant and finally MERLOT which retrieved 62 objects and 14 of them are relevant.

The table table: 7 shows the merged results, in fourth column the number of useful objects for the user for each query is detailed, in fith column the precision measurement gives an overall measure about the useful objects, over all objects retrieved.

In table table: 7, the reader can see the overall precision for each query; however, the obtained percentage is very low, this means that we are recovering a lot of Learning Objects not useful for users; only a little amount is useful. Therefore, the system is able to get and discriminate the quality and usefulness of objects and put them in the higher position of the list due the contribution of their factors. Results in table table: 8 support this affirmation, because in case of precision for first ten objects P(10) the data are very high, therefore the system puts in the first positions the top ten relevant objects for users.

After we evaluated the precision of the 10 queries used in the metasearch system. The results are reported in Table table: 8. Since for each query, we collect the whole objects from each repository, we compute the effectiveness of the model for each query at three N levels only, i.e., N=10, 20 and 50. Second the Average

Precision for each query and finally the Mean Average Precision.

As can be seen in table table: 8, the precision result for P(10) is very high (see second column). For the queries: 2,3,4,5,6,7 and 9, precision score for the first 10 objects in the combined list is greater or equal to 50. This means that the system is placing the most important elements in first positions; these data are supported by the AveP measurement, also MAP is very high (see fifth column), indicating that for all the used queries, the results have been very successful but, it is still needed further experiments.

The system is able to put in higher position in the combined list, those objects with mayor number of relationships, high value of completeness and chorus effect. The system is able to work even when objects do not have relationships, in this case, the unique factor the model takes into account is the *Weighted Completeness Index* and *Chorus effect*, therefore objects with a high value of completeness and Chorus effect will be in higher position in the combined list, this approach does not consider possible useful objects, that not necessary are completed but are of frequently use in the e-learning community.

Table 6: Result of the queries

Queries	ARIADNE	AGORA	MACE	MERLOT	Retrieved	Relevant
Q_1	18	78	3	2	101	8
Q_2	737	22	36	62	857	27
Q_3	79	7	101	1	188	16
Q_4	52	2	83	25	162	22
Q_5	224	85	1	6	316	9
Q_6	106	1	27	3	137	17
Q_7	59	1	171	3	234	10
Q_8	316	2	151	10	479	5
Q_9	31	46	35	10	122	10
Q_{10}	217	12	44	17	290	9
Total	1839	256	652	139	2886	133

One of the factors, the system takes into account is the Weighted Completeness Index and is very significant when the importance is obtained for one objects, however, if the objects have relationships, the W_{ci} is not the most important factor, because the model is considering the number of relationships that one object is having between other objects. Therefore, the model put in higher position objects with mayor number of relationships, even if the completeness is medium or low, at least 50.

The system is not able to detect those objects not related with the user query, besides sometimes these objects are related, therefore the algorithm put in a high position

objects not related with the query. This is an exception that will be considered in future works.

From the repositories consulted, MERLOT retrieves in first positions Learning Objects better adjusted to the user query, for example, from the 7 of 10 queries executed, the relevant objects, were found in the first 20 objects, followed by MACE, after AGORA, and in the last position ARIADNE.

The relation metadata is not frequently complete in the consulted repositories, even this condition; we present in decrease order the most frequent relationships found during the experiment in percentage: Base (32), Part (27), Version (17), Reference (16), Format (7) and Require (1). This means, when teachers create new simples object or complex object, they are basing their educational material from other objects.

Table 7: List of Merged Result

Queries	Retrieved	Relevant Collection	Relevant Retrieved	Precision
Q_1	101	8	3	8
Q_2	857	27	7	3
Q_3	188	16	7	8
Q_4	162	22	12	13
Q_5	316	9	5	3
Q_6	137	17	7	12
Q_7	234	10	5	4
Q_8	479	5	3	1
Q_9	122	10	8	8
Q_{10}	290	9	4	3
Total	2886	133	61	5

Table 8: Effectiveness of the model for merging Learning Objects

Queries	P(10)	P(20)	P(50)	AveP
Q_1	0.3	0.15	0.06	1
Q_2	0.7	0.35	0.14	83
Q_3	0.7	0.35	0.14	57
Q_4	1	0.6	0.24	73
Q_5	0.5	0.45	0.1	82
Q_6	0.7	0.35	0.14	71
Q_7	0.5	0.25	0.1	71
Q_8	0.3	0.15	0.06	89
Q_9	0.8	0.4	0.16	81
Q_{10}	0.4	0.2	0.08	56
				MAP = 76

5. CONCLUSION AND FUTURE WORK

In this paper has presented a model that takes into account the richness of the semantics relationships between Learning Objects and the information that can provide, about, how objects are related to other objects in a collection; this situation can denote which objects within a collection are important for a user. In order to model the importance of the related objects, the model takes into account the semantic relationships between objects. This model was defined with a series of metrics such as Weighted Completeness Index, which is a measure that determines the degree of metadata completion, the Chorus Effect, which considers that a resource found in various lists is significant for the merging process and the Number of Relationships which considers the number of objects related with the target object. These factors give information for cataloguing the recovered objects when they have not semantic relationships among them.

The model has proved that is able to recover and put in first positions relevant objects for users, the P(10) measurement, AveP and MAP are supporting this affirmation. Also the output expected results between the system and the supported by consulting experts are consistent, because objects with a high value of relationships, completeness and Chorus effect have been placed in the highest positions in the list.

The system works under any condition: objects with and without relationships. The experiment results show that it is necessary to encourage the filling of metadata in repositories, because objects more completed will have better position in list. Also when new educational resources are created from other objects, the numbers of relationships increase. When objects are used in several e-Learning environments, Chorus effect will have a higher value at the time of retrieving objects from different sources.

In order to refine the searches for different users with particular needs the model is flexible and the weights of relationships can be adjusted to obtain desired results.

As future work, Since relation metadata is not filled required for the IEEE-LOM standard we are going to incorporate against the *Relationship Creator* module, the process uses into Learning Object Generation Using and Assisted Approach [26], which objective tool is to use Learning Objects similarities to propose the filling of metadata and do easier to describe the resource. Besides make the system able to detect objects which are not related with the user query.

Another future work will consist in incorporate new repositories, in order to extend the queries of the user and obtain better results when multiples sources are being

consulted. To improve the user interface and the merging algorithm, in order to make easy for the users the usage of interface and the information retrieved, and to manage exceptions from real cases.

In order to resolve the drawback about the Learning Objects which do not have relationship with the user query, the system will take in count other metadata like *title*, *description* and *keywords* from which will obtain the frequency of occurrence of the terms for every metadata, resulting a bag of words. These words will be checked against the keywords introduced by the user. The system will consider that objects are into the query context, if the user keywords exist into the Learning Objects metadata. This approach will be an additional factor to include into the merging model, which will put in higher position those objects that belong to context, and put in last position those objects do not belong to query context.

6. BIBLIOGRAPHY

[1] D. A. Wiley, *Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy,* The Instructional Use of Learning Objects: Online Version.
URL http://reusability.org/read/chapters/wiley.doc

[2] V. H. Menéndez, M. E. Prieto, A. Zapata, *Sistemas de gestión integral de objetos de aprendizaje.*, IEEE-RITA 5 (2) (2010) 56–62.

[3] J. Vargo, J. C. Nesbit, K. Belfer, A. Archambault, Learning object evalu- ation: Computer-mediated collaboration and inter-rater reliability, Inter- national Journal of Computers and Applications 25 (3) (2003) 198–205.

[4] T. Andrews, C. Daly, Using moodle, an open source learning management system to support a national teaching and learning collaboration, in: AAEE Conference, Yeppon, 2008.

[5] L. T. S. Committee, et al., *Ieee standard for learning object metadata*, IEEE Standard 1484 (1) (2002) 2007–04.

[6] S. L. Weibel, T. Koch, *The dublin core metadata initiative*, D-lib magazine 6 (12) (2000) 1082–9873.

[7] C. López Guzmán, F. J. García Peñalvo, *Repositorios de objetos de aprendizaje: bibliotecas para compartir y reutilizar recursos en los entornos e-learning*, Biblioteca Universitaria 9 (2) (2011) 9.

[8] J. Klerkx, B. Vandeputte, G. Parra, J. L. Santos, F. Van Assche, E. Du- val, *How to share and reuse learning resources: the ariadne experience, in: Sustaining TEL: From Innovation to Learning and Practice*, Springer, 2010, pp. 183–196.

[9] F. Paulsson, A. Naeve, *Establishing technical quality criteria for Learning Objects, Media technology and graphic arts*, School of computer science and communication, Royal institute of technology (KTH), 2006.

[10] J. Serrano-Guerrero, F. P. Romero, J. A. Olivas, J. de la Mata, *Budi: Architecture for fuzzy search in documental repositories*, Mathware & Soft Computing 16 (1) (2009) 71–85.

[11] W. Meng, C. Yu, K.-L. Liu, *Building efficient and effective metasearch engines*, ACM Computing Surveys (CSUR) 34 (1) (2002) 48–89.

[12] Y. Lu, W. Meng, L. Shu, C. Yu, K.-L. Liu, *Evaluation of result merging strategies for metasearch engines*, in: Web Information Systems Engineering–WISE 2005, Springer, 2005, pp. 53–66.

[13] D. F. Barrero, M. D. R-Moreno, O. García, A. Moreno, Searchy: *A metasearch engine for heterogeneous sources in distributed environments*, in: International Conference on Dublin Core and Metadata Applications, 2005, pp. pp–235.

[14] S. Ternier, K. Verbert, G. Parra, B. Vandeputte, J. Klerkx, E. Duval, V. Ordoez, X. Ochoa, *The ariadne infrastructure for managing and storing metadata*, Internet Computing, IEEE 13 (4) (2009) 18–25.

[15] S. Zhou, M. Xu, J. Guan, Lesson: *A system for lecture notes searching and sharing over internet*, Journal of Systems and Software 83 (10) (2010) 1851–1863.

[16] P. Dinis, A. R. da Silva, *Application scenarios for the learning objects pool*, Journal of Universal Computer Science 15 (7) (2009) 1455–1471.

[17] X. Ochoa, E. Duval, *Relevance ranking metrics for learning objects, Learning Technologies, IEEE Transactions* on 1 (1) (2008) 34–48.

[18] X. Ochoa, E. Duval, *Use of contextualized attention metadata for ranking and recommending learning objects*, in: Proceedings of the 1st international workshop on Contextualized attention metadata: collecting, managing and exploiting of rich usage information, ACM, 2006, pp. 9–16.

[19] T. K. Shih, *Repository and search based on distance learning standards,* in: IT in Medicine & Education, 2009. ITIME'09. IEEE International Symposium on, Vol. 1, IEEE, 2009, pp. 3–3.

[20] B. Simon, D. Massart, F. van Assche, S. Ternier, E. Duval, S. Brantner, D. Olmedilla, Z. Miklós, *A simple query interface for interoperable learning repositories*, in: Proceedings of the 1st Workshop on Interoperability of Web-based Educational Systems, 2005, pp. 11–18.

[21] X. Ochoa, Learnometrics: *Metrics for learning objects*, in: Proceedings of the 1st International Conference on Learning Analytics and Knowledge, ACM, 2011, pp. 1–8.

[22] R. K. Belew, *Finding out about: a cognitive perspective on search engine technology and the WWW*, Vol. 1, Cambridge University Press, 2000.

[23] V. Menendez, A. Zapata, C. Vidal, A. Segura, M. Prieto, *An approach to metadata generation for learning objects,* in: Knowledge Management, Information Systems, E-Learning, and Sustainability Research, Springer, 2010, pp. 190–195.

[24] J. S.-G. E. Rivas-Sanchez, V.H. Menendez-Dominguez, *An approach for merging learning objects based on metadata semantic relationships,* The 2nd International Conference on Technology Enhanced Learning, Quality of Teaching and Reforming Education: Learning Technologies, Quality of Education, Educational Systems, Evaluation, Pedagogies (TECH- EDUCATION 2011.

[25] V. H. E. Rivas-Sanchez, J. Serrano-Guerrero, A metasearch architecture for learning objects merging., SPDECE 1 (1) (2011) 177–180.

[26] M. E. Prieto, V. H. Menéndez, A. A. Segura, C. L. Vidal, *A recommender system architecture for instructional engineering*, in: Emerging Technolo- gies and Information Systems for the Knowledge Society, Springer, 2008, pp. 314–321.

[27] M. Stefaner, E. Dalla Vecchia, M. Condotta, M. Wolpers, M. Specht, S. Apelt, E. Duval, *Mace-enriching architectural learning objects for experience multiplication*, in: Creating New Learning Experiences on a Global Scale, Springer, 2007, pp. 322–336.

[28] M. Burns, G. P. Schell, *Merlot: a repository of e-learning objects for higher education*, E-service Journal 1 (2) (2002) 53–64.

[29] D. Crockford, *Introducing json*, Available: json.org

[30] J. J. Garrett, et al., *Ajax: A new approach to web applications* (2005).

[31] O. A. Initiative, et al., *The open archives initiative protocol for metadata harvesting*, Protocol version 2.

Elio Rivas Sánchez is a PhD student in the Security and Systems Lab at Mondragon University, Mondragon, Spain. His research focuses on Safety-Critical Computer Systems. He holds a Master in Advanced Computing Technology from the Universidad de Castilla –La Mancha, Ciudad Real, Spain (2011), and the B.Sc. in System Engineering from National University of Engineering (UNI), Managua, Nicaragua (2005)