Sistema híbrido para la búsqueda de objetos de aprendizaje textuales en repositorios, basado en metadatos y contenido

Germán Augusto Osorio Zuluaga

Universidad Nacional de Colombia

Facultad de Ingeniería

Manizales, Colombia

2022

Hybrid system for searching textual learning objects in repositories, based on metadata and content

Germán Augusto Osorio Zuluaga

Tesis presentada como requisito parcial para optar al título de:

Doctor en Ingeniería

Director:

Ph.D Néstor Darío Duque Méndez

Línea de investigación:

Organizaciones , sistemas y gestión de la tecnología, información, el conocimiento, y la innovación tecnológica

Grupo de Investigación:

GAIA

Universidad Nacional de Colombia

Facultad de Ingeniería

Manizales, Colombia

2022

Dedicatoria

¡A Piedad Eugenia: sin ti difícilmente lo hubiera logrado!

"Scio enim cui credidi et certus sum quia potens est depositum meum servare in illum diem"

II Tim 1,12b

Agradecimientos

¡A Dios Uno y Trino por “quien vivimos, nos movemos y existimos”!

y sus instrumentos:

- Mi familia, quien moldeó mi ser y ha sido apoyo afectivo y efectivo en todos los momentos.
- Mi amada esposa Piedad Eugenia: amorosa, tierna y paciente compañera de camino.
- El profesor y amigo Néstor Darío Duque, por su confianza, paciencia y asesoría, todo el tiempo.
- Sebastián y Viviana, por su amor, ánimo y colaboración permanentes.
- El profesor William Sarache, por permitirme asistir a su curso, el cual fue fundamental en mi proceso doctoral.
- El estimado profesor Mauricio Orozco por su disponibilidad, apoyo e invaluable colaboración.
- Guillermo Jiménez (q.e.p.d), por su cercanía y apoyo en todo momento. Más que compañero, se comportó como un padre, pendiente del desarrollo de este proceso de formación.
- A todas y todos (son incontables) los que de una manera u otra me ayudaron a llevar a feliz término este proceso, ¡muchas gracias! ¡Dios les pague!

Contents

Resumen

El vertiginoso avance de la Web ha promovido el desarrollo de la educación a distancia. En este sentido, numerosas organizaciones ofrecen y comparten recursos educativos en diferentes formatos a los alumnos a nivel local y global. Algunos de estos recursos se denominan objetos de aprendizaje (LO) y generalmente se almacenan en repositorios. Además, LO se pueden etiquetar a través de metadatos para facilitar su búsqueda y recuperación. Estas actividades se basan principalmente en metadatos. En otros contextos web, se utilizan búsquedas de texto completo.

Además, según la revisión de la literatura realizada para apoyar esta investigación, las búsquedas de metadatos y texto completo en repositorios presentan varias problemáticas que persisten y producen una baja precisión en los resultados de búsqueda de objetos de aprendizaje en repositorios.

Por ello, para intentar superar este problema planteado anteriormente, ha ido ganando importancia el uso de métodos híbridos, en los que se integran varios métodos para conseguir mejores resultados de búsqueda. A partir de la investigación realizada para el desarrollo de esta tesis, fue posible demostrar que, al integrar el texto completo y los metadatos en las búsquedas de objetos de aprendizaje en un sistema híbrido, se logran mejoras significativas en los resultados de búsqueda.

Adicionalmente, el modelo híbrido propuesto para la búsqueda de objetos de aprendizaje en repositorios puede ser implementado en otros contextos de isoformas, como gestores bibliográficos. En este sentido, puede convertirse en una herramienta adicional que ayude a los investigadores a explorar su documentación, que, con el tiempo, crece en gran medida.

Palabras clave: Construcción de bases de datos de prueba, metadatos, texto completo, recuperación de información, búsqueda de objetos de aprendizaje.

Abstract

The vertiginous advance of the Web has promoted the development of distance education. In this sense, numerous organizations offer and share educational resources in different formats to learners locally and globally. Some of these resources are called learning objects (LO) and are usually stored in repositories. Additionally, LO can be tagged through metadata to facilitate their search and retrieval. These activities are mainly based on metadata. In other web contexts, full-text searches are used.

Furthermore, according to the literature review carried out to support this research, metadata and full-text searches in repositories present several problematics that persist and produce a low precision in the search results of learning objects in repositories.

Therefore, to try to overcome this problem raised above, the use of hybrid methods has been gaining importance, in which several methods are integrated to achieve better search results. Based on the research carried out for the development of this thesis, it was possible to demonstrate that, by integrating the full-text and the metadata in the searches for learning objects in a hybrid system, significant improvements are achieved in the search results. Additionally, the hybrid model proposed for the search for learning objects in repositories can be implemented in other isoform contexts, such as bibliographic managers. In this sense, it can become an additional tool that helps researchers to explore their documentation, which, over time, grows to a great extent.

Keywords: Dataset construction, metadata, full-text, information retrieval, learning object search.

Introduction

The vertiginous advance of the Web has promoted the development of distance education [81,154]. In this sense, a large number of organizations offer and share educational resources in different formats to learners locally and globally. Some of these resources are called learning objects (LO) and are usually stored in repositories. Additionally, LO can be tagged through metadata to facilitate their search and retrieval [154]. Accordingly, the metadata provides structured descriptions to the LO [81]. In the same way, several repositories can be organized as federations, which offer a unified approach for representing these repositories through a hierarchical system that centralizes educational resources in a single portal, increasing their visibility and facilitating the uniform administration of applications to discover and access the contents of the LO available in a group of repositories [47].

In addition, in order to reuse and share LO between repositories, standardized protocols have been defined to catalog them [81, 172]. Among the most important ones are the IEEE-LOM, Dublin Core [172], Can Core and OBAA. Each one specifies the syntax and semantics of the attributes needed to describe a LO [47]. For example, the IEEE-LOM standard consists of nine categories and about seventy descriptive elements [61].

On the other hand, within the architecture of the repositories, a component that is responsible for the search and recovery of LO is necessary. In this sense, to obtain information on the Internet there are several general-purpose search engines, in which keywords are entered and they return, as a result, web pages containing the user-entered keywords. The paradigms used by these search engines are not the most suitable for the recovery of LO [23] because they have two main drawbacks: the first one is that this keyword approach requires the advance indexing of the content of the learning objects, not only the textual ones but also the multimedia objects; the second one is that learning objects are not semantically related to the subject of learning [81]. These situations produce many irrelevant results for the learner [59].

Furthermore, the search and retrieval of learning objects in repositories are mainly based on metadata [47]. In other web contexts, full text searches are used. In the first one, the search is done from the metadata, in which the users pre-select and search the individual topics of a source

of information, such as author, title, and subject; the search engine finds matches between the terms of the query with the terms of structured metadata and generates the results. In the second, the full text search, the system finds matches between the terms of the query with the terms in the individual documents of a repository and classifies the results algorithmically [20].

Moreover, according to the literature review carried out to support this research, metadata and full text searches in repositories present several problematic aspects such as permanent growth in the number of learning objects in repositories [7, 21, 54, 109, 112, 125]; poor quality of metadata [18, 19, 21, 101, 104, 105, 109, 153]; low precision in search results from metadata [54, 125, 151]; potential learning objects without metadata that allow their discovery and use [21, 37]; restriction of semantic relationships in repositories [62, 143]; and weakness of the full text search [20, 47, 58]. Based on the above, it can be stated that, in general, a low precision persists in the search results of learning objects in repositories.

Therefore, to try to overcome this problem raised above, the use of hybrid methods has been gaining importance, in which several methods are integrated in order to achieve better search results, [14, 70, 81, 179]. In this respect, the object of this thesis is addressed.

Then, the following elements that structure this research arise, such as the question, the hypothesis and the general and specific objectives, which are defined below.

Research question

What elements should be considered to structure a hybrid system oriented to the search of textual learning objects in repositories, based on metadata and content, that improves the indicators of precision and recall of the results obtained in systems based only on metadata?

Hypothesis

A hybrid system for the search of textual learning objects in repositories, based on metadata and content, improves the indicators of precision and recall of the results obtained in systems based only on metadata.

Objectives

General objective

Propose and validate a hybrid system for the search of textual learning objects in repositories, based on metadata and content, to improve the indicators of precision and recall of the results obtained in systems based only on metadata.

Specific objectives

1. *Characterize and define the conceptual elements that support the indexing of textual objects for metadata-based searches and those based on content, used in online service environments.*

2. *Theoretically, conceptualize the fundamentals and methodological elements that support the design of hybrid computational systems, as well as the practical considerations of their implementation, to integrate models and search techniques based on metadata and the content of textual LO in repositories.*

3. *Conceptually, propose a hybrid search system for textual learning objects in repositories, based on metadata and content.*

4. *Design a prototype of the proposed hybrid system to validate the proposal.*

5. *Empirically compare the results obtained by the prototype hybrid search system with those based on search systems supported by metadata.*

Document organization

This thesis is organized as follows: in Chapter 1, this research's general theoretical framework is presented. In Chapter 2 there is a bibliographic review of state of art on the thesis subject. These chapters are aimed at fulfilling specific objectives 1 and 2. In Chapter 3 the implementation of an experimentation dataset in metadata and full text of documents is proposed. Chapter 4 presents the experimental evaluation of models on metadata and full text on the dataset proposed in the previous chapter. In chapter 5 the integrated experimental results on metadata and full text are shown, and a hybrid learning object search model is proposed. These last three chapters point to the fulfillment of objectives 3, 4, and 5. Finally, in Chapter 6 the conclusions are presented.

CHAPTER 1

Theoretical framework

1.1. Introduction

This chapter presents the general theoretical framework that supports the research, especially on the topics of information retrieval, repositories, and learning objects. Also, for organizational considerations of the thesis, required concepts were included in the other chapters.

1.2. Information Retrieval

Information retrieval (IR) is a broad area of IT, focused mainly on providing users with easy access to information of interest, in aspects such as representation, storage, organization and access to information about elements such as documents, web pages, online catalogs, structured and semi-structured records, and multimedia objects. In terms of scope, the area has grown well beyond its first goals of indexing texts and finding useful documents in a collection. Today, the research includes modeling, web searches, text classification, system architecture, user interfaces, data visualization, filtering, and languages [15]. It emerges as a knowledge discipline in the 1950s [34, 140].

It is important to clarify that, from the field of Computer Science, the concepts of Data Recovery and Information Recovery have their differences, which are shown in the Table 1.1 [159].

In the theoretical and applied IR research, converge in a multi and interdisciplinary way, to a lesser or greater extent, the experience of fields such as computer science, information science, documentation, linguistics, artificial intelligence, engineering, cognitive psychology, and librarianship, among other disciplines [64, 129].

Table 1.1: Differences between data and information retrieval [159]

Characteristic	Data Recovery	Information Retrieval
Matching	Exact match	Partial match, best match
Inference	Deduction	Induction
Model	Deterministic	Probabilistic
Classification	Monothetic	Polithetic
Query language	Artificial	Natural
Query specification	Complete	Incomplete
Items wanted	Matching	Relevant
Response error	Sensitive	Insensitive

1.2.1. The concept of IR

Although there is no unified definition of IR, in the review conducted by [64], the following characteristics of IR are emphasized:

1. It is a complex process that consists of a variety of components that have structural and functional characteristics, and that interact with other components.
2. It is a dynamic iterative process.
3. It is related to the satisfaction of the information needs of users.
4. It always happens in some kind of environment.

For purposes of this thesis, we will use the definition of [88]: *"IR consists of finding material, usually documents; of unstructured nature, usually text; that satisfies a need for information from within large collections, normally stored in computers"*. In this sense, IR systems have to deal with incomplete or subspecified information in the form of queries issued by users. The IR systems that receive such queries need to fill in the user's query gaps [131].

1.2.2. History of IR

To address the issue of the history of IR, prehistory or the previous events that preceded it and five periods that have determined its development as a knowledge discipline are presented below.

1.2.2.1. Prehistory: mechanical and electromechanical devices

The long history of IR does not begin with the Internet [132]. For thousands of years, humanity has understood the importance of archiving and finding information [140]. This is how for more than 5000 years the human being has organized information for its later recovery and use. In its most usual forms, clay tablets, hieroglyphics, papyrus scrolls, and books have been compiled, stored, organized and indexed. In this sense and in order to maintain them, special buildings called

libraries were built. Since the volume of information finally grew greatly, it became necessary to build methods and techniques for their conservation and identification of each one, in order to recover them in a correct and fast way. But, it was only with the scientific development along with the industrial revolutions of the eighteenth and nineteenth centuries, that they pushed for these techniques to satisfy the information needs that were increasingly punctual [15, 34].

For this purpose, the works were indexed using cataloging schemes. Eliot and Rose affirm that this approach has millennia: they pointed out that Callimachus, a Greek poet of the 3rd century BC, was the first person known to create a library catalog. Already, in a more recent era, Rudolph presented a US patent in 1891, for a machine composed of catalog cards linked together, which could be rolled up beyond a viewing window that allowed a quick manual scanning of the catalog. Soper registered a patent for a device in 1918, where catalog cards with holes, related to categories, were lined up in front of each other to determine if there were entries in a collection with a particular combination of categories. If you could see the light through the arrangement of cards, there was a coincidence [132].

Likewise, mechanical devices were designed that looked for in a catalog for a particular entrance. It seems that the first person that built this system was Emanuel Goldberg who addressed this problem in the 20s and 30s of the last century. According to a biography of Goldberg written by Buckland, three prototypes of the machine were built. Mooers described the investigations of Davis and Draeger in 1935, in the search with microfilm. The culmination of this approach seems to be Shaw's quick selector, with which the search was reported on a 2000-foot film reel [132].

Subsequently, other mechanical technologies were examined. Luhn, for example, made a selector using punched cards, light and photocells. The prototypes of this system were completed in 1950 and were demonstrated in 1951. A key feature of this system was that a consecutive sequence of characters could be paired within a larger chain. The system searched for 600 cards per minute [132].

1.2.2.2. Period I: start of the use of the computer in IR

With the invention of computers, which occurs in the context of the Second World War, it became possible to store large amounts of information. Therefore, the search for useful information became a necessity [140]. This is how the first computerized search systems were built in the late 1940s [132].

After the war, the great powers devoted considerable sums of money to scientific research [34]. Thanks to this, there was an exponential growth of the available information and emerges, as an underlying difficulty, the adequate management of these large volumes of information. In this sense, Vannevar Bush, scientific advisor to the US presidents Roosevelt and Truman, writes:

> *In view of current concerns, the problem is not so much that excessive publications are made as they have far exceeded our present capacity to make real use of them (...) Professionally our methods of transmitting and reviewing the results of the scientific research are several generations old and, for now, totally inadequate in its purpose ... Mendel's concept of the laws of genetics was lost to the world for a generation*

> *because its publication did not reach the few who were able to capture it and extend it, and this kind of catastrophe is repeating itself with us* [30, 34].

In this sense, two main factors contributed to the beginning of IR as a field of study in the 1950s. First, many documents that were not available to a large community during the Second World War were released. The second factor was related to the intensive work on computers, a device that was considered as a perfect tool to organize, index and retrieve documents [64]. Also, the term "information retrieval" was introduced by Calvin Mooers in 1950, who supplemented its definition one year later [129].

Several works arose in the mid-1950s that developed on the basic idea of searching for text with a computer. One of the most influential methods was described by H. P. Luhn in 1957, in which (in few words) he proposed to use words as units of indexing for documents and to measure the superposition of words as a criterion of recovery [140].

In that same year, 1957, at the Cranfield Institute of Technology and other associated entities, tests began that marked the beginning of the recovery of information as an empirical discipline. These tests made a strong influence on the evolution of the discipline [34]. With them, an evaluation methodology was developed that is still in use by the IR systems nowadays [140].

1.2.2.3. Period II: 1960s decade

In this period, a wide variety of activities were carried out to determine if it was possible to improve IR systems with computers. A prominent figure in this period was Gerard Salton, who formed and led a large research group on IR at the American universities of Harvard and Cornell. The group produced numerous technical reports, establishing ideas and concepts that are still important areas of research today. One of these areas is the formalization of algorithms to classify documents in relation to a query. Of particular importance was an approach in which documents and queries were visualized as vectors within an n-dimensional space, initially proposed by Switzer, and later, the similarity between a document and the query vector, to be measured at through the cosine of the angle between the two vectors, suggested by Salton. Other improvements in IR examined in this period include the grouping of documents with similar content [132].

1.2.2.4. Period III: 1970s decade

One of the key developments of this period was the weighting of the frequency of terms (TF) of Luhn (based on the occurrence of words within a document), complemented by the work of Sparck Jones on the occurrence of words in the documents of a collection [132]. Likewise, Salton synthesized the results of his group's work on vectors to produce the vector space model [128,132].

Additionally, an alternative means of modeling IR systems involved expanding the idea of Maron et al. [89] to use probability theory. Robertson defined the principle of probability classification, which determined how to best classify documents based on probabilistic measures with respect to the defined evaluation measures [119, 132].

1.2.2.5. Period IV: the decade of the 80s and mid-90s of the last century

Based on the developments of the 1970s, there were variations of the weighting schemes [127,132] and the formal recovery models were extended. The original probabilistic model did not include weights and several researchers worked to incorporate them in an effective manner. Among other achievements, this work finally led to the BM25 classification function [121,132]. There were also advances in the basic vector space model and probably the best known one is the latent semantic indexation (LSI), where the dimensionality of the vector space of a collection of documents was reduced by the decomposition into singular values [42,132]. Similarly, another significant advance of this period was the creation of new stemming algorithms, the process of matching words to their lexical variants, which, although they were known since 1960, had an important improvement with the contribution of Porter [114] and other authors, which are still used today [132].

Another representative event of the period was the launching of the TREC (Text Retrieval Conference) in 1992, an initiative of Voorhees and Harman, as an annual exercise in which a large number of international research groups collaborate to build test datasets larger than those that existed before [132, 140, 166]. With the large collections of text available under TREC, many old techniques were modified, new ones were developed, and are still being developed for effective recovery [140].

1.2.2.6. Period V: the mid-1990s to today

This period has been marked by the beginnings of the web and its vertiginous development. This is how web search engines began to emerge at the end of 1993, to cope with the exponential growth in the number of web pages, which was already beginning to be evident [132].

Two important advances were made in order to identify the best pages. One was the analysis of links and the other, the search for anchor text, that is, the search for both the content of a web page and the text of the links to which that page points. Both developments were related to previous work on the use of citation data for search and bibliometric analysis, and using "spreading activation" search in hypertext networks. The anchor text was recognized from the beginning as a valuable source of information and its use as a key feature of the Google search engine from its initial development, along with the use of the method of linkage analysis PageRank [27] and HITS that was developed at the same time by Kleinberg [72, 132].

Other lines of research that have been presented in this last period have been the automated explotation of information extracted from search engine registers, the introduction of a probabilistic approach using language models, and brief queries, which have little linguistic structure [132].

1.2.3. The problem of IR

The primary objective of an IR system is to recover all the documents that are relevant to the user's query while recovering only a few non-relevant documents, whenever possible. The notion of relevance is of central importance in the IR [15].

1.2.4. The process of IR

A model is an abstract representation of processes or phenomena of reality. Regarding the IR, a model conceptually represents the processes of the IR as shown in Figure 1.1. That is, for a search given in a collection, the model will try to retrieve documents that are relevant and to what extent. In this sense, the IR models must find a representation for the documents, the queries, and a function of relevance or similarity, that allows calculating the most relevant documents for the user and their possible classification [32].

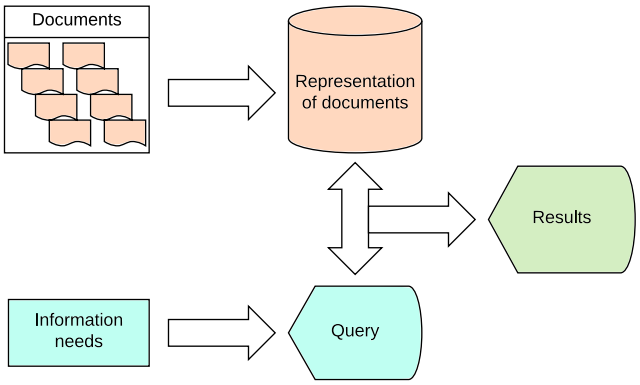


Figure 1.1: The process of IR [32]

1.2.5. Taxonomy of IR

IR models are mainly based on text, that is, they use the text of the documents to classify them with respect to the query. However, on the web, it is also necessary to use the information in the link structure to get a good classification. On the other hand, multimedia documents - images, audio or video - are encoded differently from text documents; therefore, they are classified differently or simply recovered without classification. Given these characteristics, three types of more general IR models are distinguished: text-based, link-based, and those based on multimedia objects [15] as shown in Figure 1.2.

Regarding with text-based models, a distinction is made between those based on unstructured text and those that take into account the structure. In the first category, the three classic models are: the Boolean model, which is based on the theory of sets; the vector model that is an algebraic model; and the probabilistic one, which is based on the probability theory [15].

Over the years, other recovery models based on the classics have been proposed. In this sense, those based on set theory have the fuzzy, the extended boolean, and the sets. Regarding alternative algebraic models, we have the generalized vector, latent semantic indexing, and those of neural networks. Also, among the alternatives to the probabilistic model are the BM25, divergence of randomness and the language models [15].

On the other hand, semi-structured text retrieval models consider approaches such as proximal nodes and those based on XML. In the same way, concerning the web, we have the PageRank and

Hubs & Authorities models. Finally, the multimedia IR models (image, audio, and video) are very different from those of text retrieval.

In the following sections, we will delve a little deeper into the classic models.

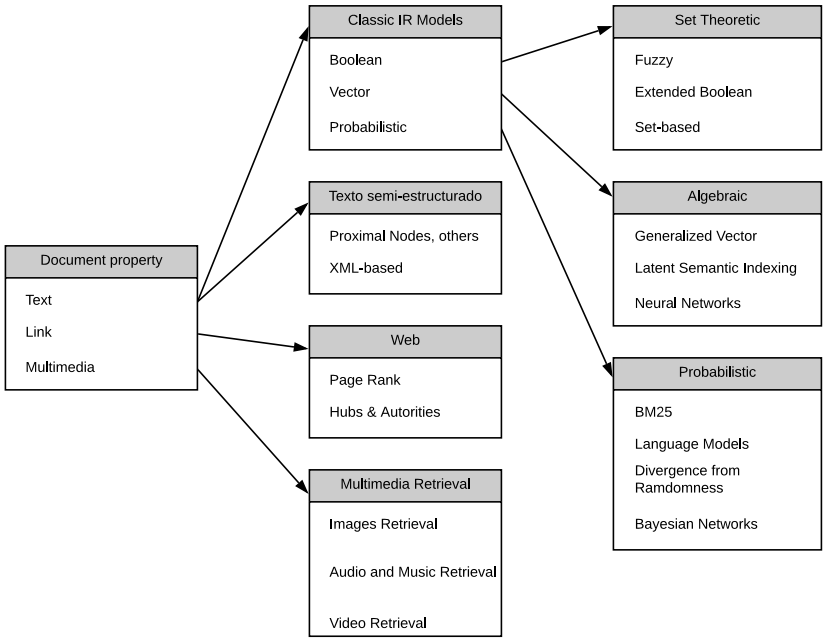


Figure 1.2: Taxonomy of IR [15]

1.2.6. Indexing and classic models of IR

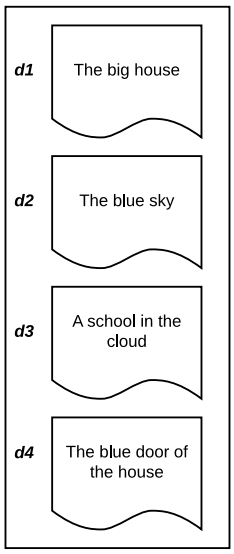
1.2.6.1. Indexing

In most practical cases of IR, the use of an index is required. According to this, an index is a structure of data built on the text to accelerate searches [15]. The process of constructing the index is called indexing [88, 135]. The process or machine that runs it is called an indexer [88].

There are several indexing techniques, the best known being inverted files or indexes, suffix trees and signature files [86]. In this sense, the inverted index technique has been the most used in traditional IR systems and in other alternative systems, based on the classification of documents according to word frequency [15] as well as by modern search engines [39]. On the other hand, suffix trees are very powerful full text recovery indexes, but, at the same time, harder to maintain than inverted indices; they are appropriate for the search of phrases and, therefore, suitable for languages such as Chinese, Japanese and Korean, which are difficult to divide into meaningful words. Additionally, signature files were long considered as an important competitor of inverted indexes, especially when disk and memory resources were important; and provide efficient support for boolean queries by quickly deleting documents that do not match the query; however, it can not

be easily extended to support phrase queries and classified recovery and, in addition, false matches in recovery have been reported [31].

The inverted index is a word-oriented mechanism for indexing a collection of text, in order to speed up the search task. It is composed of two elements: vocabulary (also called dictionary or lexicon) and occurrences. The vocabulary is the set of all the different words in the text; for each word of the vocabulary, the index stores the documents that contain that word; as sugested by its name. In this sense, the simplest way to represent the documents that contain each word of the vocabulary is the term-document matrix, as shown in Figure 1.3. In addition, it is the most used index so far and also the oldest one [15].



Vocabulary	*ni*
A	2
house	2
big	1
the	5
sky	1
blue	2
school	1
in	1
cloud	1
door	1
of	1

d1	*d2*	*d3*	*d4*
-	-	1	-
1	-	-	1
1	-	-	-
1	1	1	2
-	1	-	-
-	1	-	1
-	-	1	-
-	-	1	-
-	-	1	-
-	-	-	1
-	-	-	1

Figure 1.3: Matrix term-document in a basic inverted index

1.2.6.2. Boolean model

The Boolean recovery model is a model of IR in which we can raise any query in the form of a Boolean expression of terms, that is, in which the terms are combined with the operators AND, OR and NOT. The model sees each document as a set of words [88]. In the original model, the results obtained are not classified by relevance, all have the same relevance [156]; that is, there is no indication about which terms are more important than others (the weights are binary 0 or 1) [79].

1.2.6.3. Vector model

The vector model was defined by Salton [128]; it consists in modeling documents and queries as vector terms [32]. This model was developed to handle text retrieval from large databases where the text is heterogeneous and the vocabulary varies. The underlying formal mathematical model is

a vector space model that defines unique vectors for each term (word or concept) and document, and the queries are carried out comparing the representation of the query to the representation of each document in the vector space. The query-document similarities are based on concepts or similar semantic contents [90].

As an example, consider a collection of documents t1, t2, and t3 that only contain the terms *house*, *car*, and *tree*. And let's define how:

$t1 =$ "*tree car*"

$t2 =$ "*house tree*"

$t3 =$ "*house tree car*"

and the query about the collection:

$q =$ "*house car*"

Since there are three words in the collection, the vector space will be three-dimensional and its graphic representation is shown in Figure 1.4:

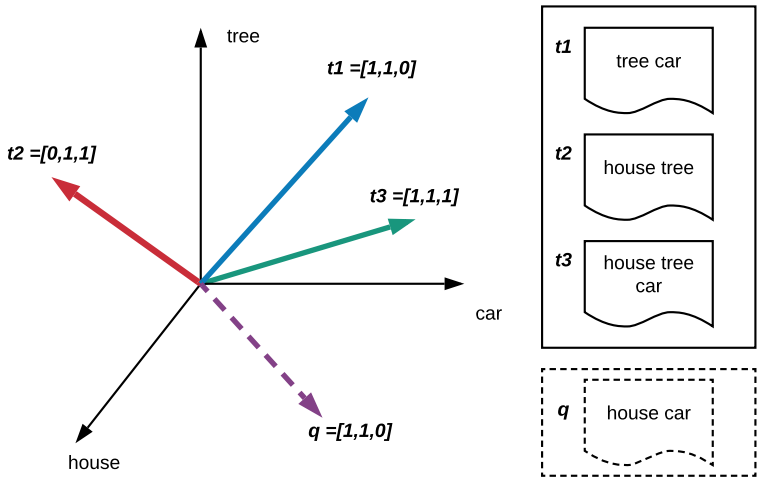


Figure 1.4: Example of a three-dimensional vector space model

The vector model determines the similarity between the documents and the query, through its proximity to vectors in the space [32].

1.2.6.4. Probabilistic model

It was proposed by Robertson and Spark-Jones [120] as a solution to the problem of the IR based on the probabilistic framework [15]. He argues that given that the IR process presents a certain degree of uncertainty, probability theory is an appropriate means to model it. It is based on the assumption that each query has an ideal set of relevant documents. This model seeks to calculate the probability that, given a document d and a query q the document d is relevant to this query [32].

Using a probabilistic model, the obvious ranking in which to present documents to the user is to classify documents by their estimated probability of relevance concerning the need for information [88]. This is the basis of the Probability Ranking Principle (PRP) proposed by [159] that establishes that if the result of an IR system is a list of documents ordered from higher to lower probability of relevance, then the effectiveness of the system will be the maximum possible [32].

1.2.6.5. Hybrid models

Hybrid models combine the strengths of algorithms and models mentioned above (and their derivations), in order to overcome some of the deficiencies and problems they present individually [63]. They are the most used and useful in practice, as they compensate for the limitations that each model presents when used in isolation [32]. The subject is delved a little deeper in Chapter 5.

1.2.7. IR Assessment

Evaluating an IR system is measuring how well the system meets the information needs of the users. Without an adequate evaluation, there is no way to establish how well the system is working, nor can objectively compare its recovery quality to that of other systems [15].

In this regard, the evaluation of IR systems is carried out from two aspects, effectiveness and efficiency. The first, effectiveness, concerns the ability of a system to retrieve information about a certain query while dismissing the useless one. The second, efficiency, refers to machine-related aspects such as the time spent obtaining the response or the amount of memory required to carry out this work, etc. In this sense, effectiveness-related indicators are the most commonly used to evaluate and compare IR systems [32] and used in this thesis.

A key aspect when dealing with the effectiveness of an IR system is relevance. In a preliminary way, relevance can be considered as a measure of the pertinency of a document retrieved by the system to resolve the need for user information. In this context, two metrics are widely used by the IR community, precision and recall [32]. Founded in them, metrics based on interpolation of the points that make up the curve recall-precision and calculating average values are used. [15, 32].

Taking this into account, it elaborates a little deeper into them below.

1.2.7.1. Precision and Recall

Precision refers to the fraction of documents retrieved (set A) that are relevant in a query made by the user. On the other hand, recall corresponds to the fraction of the relevant documents (set R) that have been recovered [15]; see Figure 1.5.

Mathematically, *precision* is given by [15]:

$$p = \frac{|R \bigcap A|}{|A|} \tag{1.1}$$

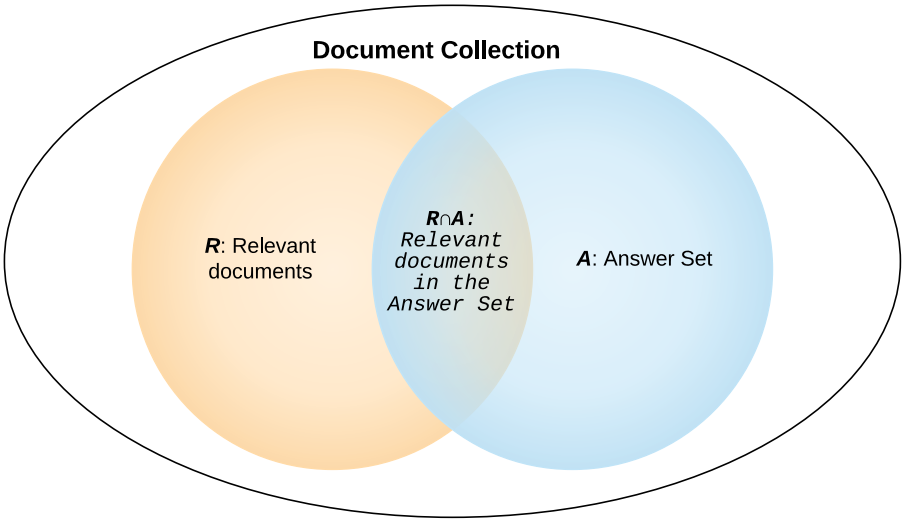


Figure 1.5: Precision and recall for a given information request [15]

where A corresponds to the set of recovered documents and R is the set of relevant documents.

Similarly, the *recall* metric is given by:

$$r = \frac{|R \bigcap A|}{|R|} \tag{1.2}$$

1.2.7.2. Precision-recall curve

In general, information retrieval systems produce an orderly output of documents, in decreasing order of relevance concerning the query. The values of recall and precision are calculated for each position of the output. These values are usually represented graphically, obtaining the so-called precision-recall curves. To do this, recall is represented on the abscissa axis and precision on the ordinate axis [32].

1.2.7.3. Metrics using interpolation

Two of those metrics are interpolated precision and average interpolated precision.

The *interpolated precision* for a certain level of recall is defined as:

$$P_i(R) = max\, P' : R' \geq R \wedge (R', P') \,\varepsilon\, T, \tag{1.3}$$

where $P_i(R)$ is the *precision interpolated* to a certain level of recall R; *P'* means precision; *R'* means recall; and T is the table with the pairs of values (*R', P'*) of recall and precision obtained for the different positions. Based on this metric, the *eleven-point interpoled precision* curve can

be obtained that summarizes the efficiency of a system concerning a query given the eleven interpolated precision values obtained at levels of recall that go from 0 to 1 in steps of 0.1 (Figure 1.6). Additionally, the *eleven-point interpolated average precision* metric of several queries is obtained by calculating the arithmetic mean of the interpolated precision of all the queries for each level of recall (Figure 1.7) [32].

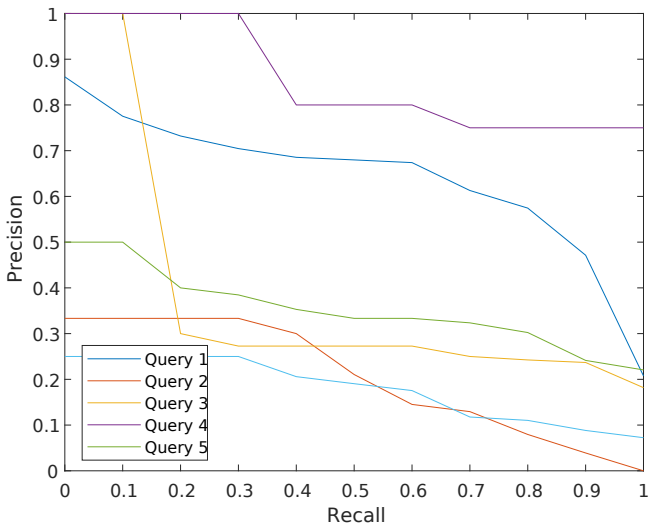


Figure 1.6: 11-point r-p curve for five queries

1.2.7.4. Metrics using mean values

Average precision and mean average precision (MAP) are, today, standard metrics in IR literature and widely used to evaluate IR systems [15, 32]. Average precision is defined as:

$$AP(q) = \frac{\sum_{i=1}^{|Rec|} P_i \times rel_i}{|R^*|} \tag{1.4}$$

where $|Rec|$ is the total number of documents retrieved; P_i is the precision obtained at position i of the ordered response; rel_i is a binary function that is worth one if the document that occupies position i is relevant and zero if not; finally, $|R^*|$ is the number of relevant documents in the set Rec.

On the other hand, the average mean precision is calculated with:

$$MAP = \frac{\sum_{q=1}^{Q|} AP_q}{Q} \tag{1.5}$$

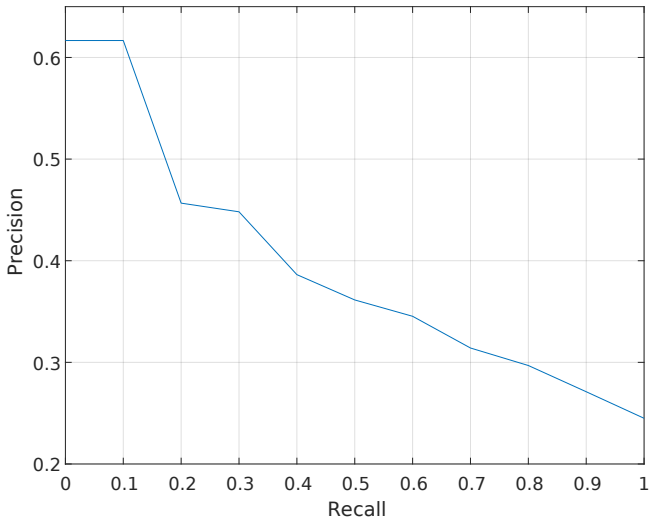


Figure 1.7: 11-point interpolated average precision for the five queries

where Q is the total number of queries made to the system, and AP_q is the value of the mean precision obtained in each of the q queries made.

1.3. Learning objects, repositories and metadata

1.3.1. Metadata

García Aretio, cited by [12], defines metadata as *«a detailed structure of the text, which describes attributes, properties, and characteristics, distributed in different fields that clearly identify the object so that it can be found, assembled and used»*. They contain information about the main objective of the LO, its target group or groups, the designer, date of creation and modification, size, type and use [16]. Through them, you have a first approach to the object, quickly knowing its main characteristics. They are especially useful in resources that are not textual and in which their content cannot be indexed by automatic systems, for example, multimedia or audio [85]. Metadata have the following characteristics [130]:

- "They say something" about the object, in a general sense.
- Physically they are external to the resource itself: they are contained in another file or they are obtained from a different service.
- They use a technical format for their expression and their exchange, generally defined languages on XML.
- They use a series of descriptors, fields or standard elements to achieve a certain degree of interoperability between different systems.

Among the most commonly used metadata standards are Dublin Core (1995), LOM (2002), ADL SCORM (2004). There are also adaptations of these standards called application profiles, such as CanCore (Canada), UK LOM Core (United Kingdom), Vetadata (Australia) [12], OBAA (Brazil) [123], MEN-LOM (Colombia) [95].

The Dublin Core (DC) metadata standard, proposed by the Dublin Core Metadata Initiative (DCMI) in the 1990s, provides a small set of terms to describe any type of physical and web resources. DC is a lightweight standard with only 15 core elements, all optional and repeatable. DC is often considered a minimum common space for the exchange of metadata between different communities [181].

LOM is the acronym for "Learning Object Metadata". Technically, its name is IEEE 1484.12.1-2002. LOM is a Descriptive Metadata Scheme (DMS) associated with digital educational resources. This standard was published in 2002 by the International Committee IEEE-LTSC-LOM (IEEE, Learning Technology Standards Committee - Learning Objects Metadata Working Group) based on the technical specifications of the ARIADNE metadata (Alliance of Remote Instructional Authoring and Distribution Networks for Europe), IMS (Instructional Management System) and Dublin Core. Its objective is to provide a common framework at an international level to promote the distribution and exchange of digital educational resources. The descriptors of version 1.0 of the LOM are divided into nine categories that group 68 metadata [141]. Table 1.2 provides a brief description of each category.

SCORM is a technical standard created and developed by ADL. This standard supports the following keys as high-level requirements: availability, adaptability, economy, durability, interoperability, and reuse. It also includes a high-level set of fundamental characteristics and content standards for e-learning, technologies and related services. The standard uses XML to encode a file that describes the components and resources [1].

1.3.2. Learning object

A learning object (LO) is a *"digital material with different granularity, which can be used for educational purposes from an intentionality defined implicitly or explicitly, for educational purposes and containing metadata that allows its description and recovery, which facilitates its reuse and adaptation to different environments"* [47] as illustrated in Figure 1.8. The term was coined in 1992 by Wayne Hodgins, a futuristic expert and e-learning strategist at Autodesk Inc [134].

The learning objects are oriented to computer-based instruction, learning or teaching. They are not a technology, more properly, they are a philosophy, which is based on the current of computer science known as object-orientation. To retrieve the materials from the databases, each object has to be labeled with metadata defined as data on data [85].

To consider them as such, Polsani [113] proposes that LO must meet the following functional requirements:

- *Accessibility:* the LO must be labeled with metadata such that it can be stored and referenced in a database

Table 1.2: Description of the categories LOM

Category	**Description**	**Sub-elements**
1. General	The general information that describes the object of learning as a whole	9
2. Lifecycle	Characteristics related to the history and present state of the object of learning and those that have affected this object during its evolution	6
3. Meta-metadata	Group information about the same metadata, not about the learning object that is being described.	10
4. Technical	Groups the requirements and technical characteristics of the learning object	11
5. Educational	Conditions of educational use of the resource	11
6. Rights	Terms of use for the exploitation of the resource	3
7. Relation	Defines the relationship of the described resource with other learning objects	7
8. Annotation	Comments on the educational use of the learning object	3
9. Clasification	Thematic description of the resource in some classification system	8

- *Reuse:* once created, an LO should work in different instructional contexts
- *Interoperability:* the LO must be independent of both the media and knowledge management systems.

1.3.3. Digital libraries and repositories

The repositories of learning objects (ROA) form a centerpiece of the technology of digital education systems [55]. These repositories are web systems that store, classify and distribute educational resources in the form of LO [56]. In addition, they provide multiple structures and services such as search, navigation, and personalization; and they are usually constructed having a community of target users with specific interests [76], as shown in Figure 1.9.

There are local repositories that contain their own LO and remote repositories that are those accessed through a network [123]. They can store the metadata only or also the metadata along with the associated educational resources [54]. Among some of the best known ROA, we can highlight MERLOT, ADD, ARIADNE or Connexions [56].

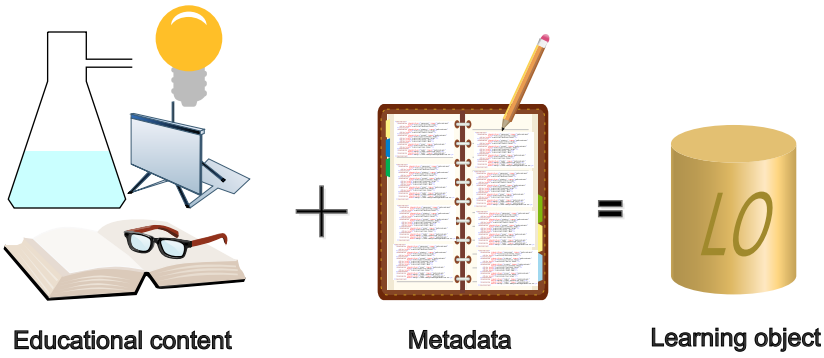


Figure 1.8: Learning object

1.3.4. Federation of ROA

In order to centralize searches on distributed sites, the repositories of digital learning objects are united in repository federations [123]. These federations are an environment that consists of parallel instances of grouped existing repositories that behave as if they were a single repository. The desire to federate repositories arises as a result of the understanding that no single digital library houses all artifacts that are relevant to a specific domain, community or application [158].

Among some of the most well-known federations are the Federation of Repositories Educa Brazil - FEB, ARIADNE, SMETE, GLOBE, LA FLOR, AGREGA and FROAC [47].

1.4. Chapter conclusions

A review of the issues that conceptually frame the proposal of a hybrid system for searching textual learning objects in repositories, based on metadata and content was carried out: information retrieval and learning objects. In the first topic, information retrieval, the background, the problem, the process, the taxonomy, the indexing, the evaluation, and the automatic expansion of queries were reviewed. For the second topic, the concepts of metadata, learning objects, repositories, and repository federations were reviewed.

The review in this chapter was aimed at meeting Objective 1: characterize and define the conceptual elements that support the indexing of textual objects for metadata-based searches and those based on content, used in online service environments.

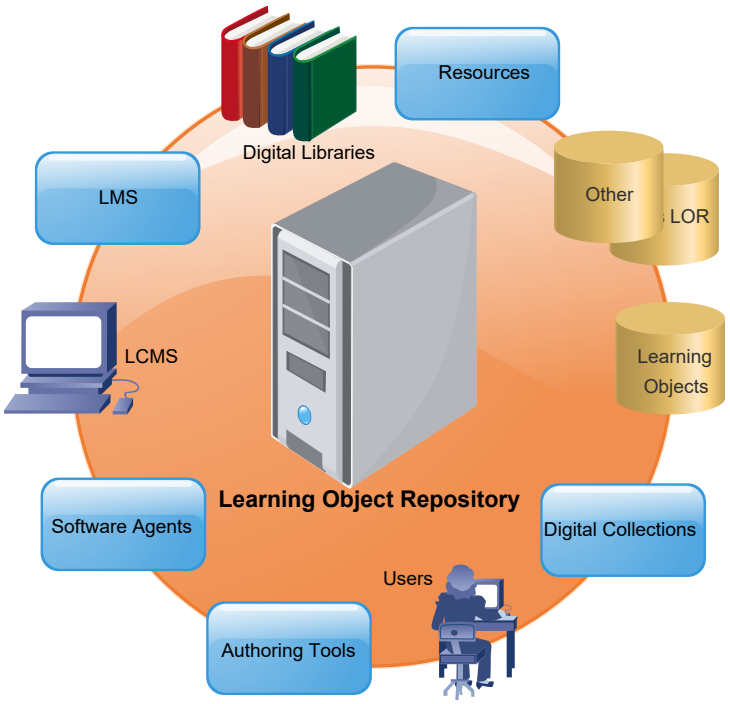


Figure 1.9: Repository of LO

CHAPTER 2

A Survey of Information Retrieval of Learning Objects

2.1. Introduction

The search and retrieval of learning objects in repositories are mainly based on metadata [47]. In other web contexts, full text searches are used. In the first one, the search is done from the metadata, in which several topics of relevance are pre-selected, like author, title, and subject; the search engine returns the results that match the terms of the query with the terms of the structured metadata. In the second, the full text search, the results come from an algorithm that matches the terms of the query with the words in the individual documents of a repository [20].

In the same context, there are also hybrid methods which have been gaining preponderance, in which better results are obtained combining two or more techniques [81], [70], [179], [14].

This chaper is an extended and updated version of [108].

Lastly, this chapter presents a literature review from the last fifteen years (since 2006) related to searching and selecting learning objects in repositories. To achieve this, the following section presents the methodology. In Section 2.3, the literature on the subject is reviewed and characterized. In Section 2.4, the discussion is presented. Finally, in Section 2.5, the conclusions of this chapter are presented.

2.2. Paper selection methodology

For the identification of the papers, the following steps of the methodologies proposed in [155] and [49] were followed.

- Identifying main concepts.
- Listing related terms.
- Determination of the search equation.
- Establishing inclusion and exclusion criteria.
- Selection of papers.

It is an iterative process and is not strictly sequential [96]. In fact, in the different stages, it was necessary to go back several times.

Additionally, for the purposes of conducting the bibliographic search, initially, five databases were considered: Web of Science, Scopus, Dimensions, Google Scholar and Microsoft Academic [98]. WoS is a web portal that provides access to multiple databases for 256 disciplines and around 90 million records. For its part, the Scopus website allows access to around 69 million records, also in various bibliographic databases of various disciplines. Dimensions can be considered as an alternative to WoS and Scopus, allowing access to 102 million publications . On the other hand, Google Scholar and Microsoft Academic index a more significant number of bibliographic records, but not all peer-reviewed [98]. The first three services above index peer-reviewed publications and can be considered as metasearch engines, since, through them, bibliographic data from publishers and databases such as IEEE, Elsevier, Science Direct, Springer, ACM, etc. can be accessed.

Based on the above, WoS and Scopus were used for the review. Dimensions was dropped because it does not allow wildcards or filter by title only. Google Scholar was used to standardize the number of citations per publication and the authors' H-index, as explained later in Section 2.2.5.

The process carried out is explained below.

2.2.1. Identifying main concepts

The summary phrase [91] for this review was "Search and selection of learning objects in repositories" [108]. In this sense, the search equation was included terms that encompass these concepts, as follows:

- *TITLE (("search engine" OR "information retrieval") AND "learning object")*

Few results were obtained. Then, the next phase of the methodology was continued.

2.2.2. Listing related terms

Next, terms such as knowledge object, educational resources, educative materials, educative content, teaching materials, and e-learning were included as synonyms for the main concepts [155]. Similarly, synonyms for information retrieval and search engine were added, such as extract, select, discover, identify, rank, classify, classify, deliver, and index [108].

2.2.3. Determination of the search equation

Taking into account the terms defined in the subsections 2.2.1 and 2.2.1, logical operators that relate concepts [49] and the wildcard symbol "*" [155], the following search equation was reached [108]:

- *TITLE ((search* OR retriev* OR extract* OR select* OR discov* OR identif* OR rank* OR classif* OR deliver* OR index*) AND ("learning object*" OR "knowledge object*" OR elearning OR e-learning OR "educat* resourc*" OR "educat* mater*" OR "educat* content*" OR "teach* mater*"))*

2.2.4. Establishing inclusion and exclusion criteria

For the bibliographic databases used, all types of documents that matched the search equation and had been published in the last fifteen years were included, that is, since 2006 .

Finally, according to the inclusion criteria, 1002 Scopus papers and 286 WoS papers were obtained.

2.2.5. Selection of papers

For the survey, the following sort criteria were applied for the 1288 articles [108]:

1. For all retrieved and relevant papers, eighty percent of the most cited papers per year (number of citations divided by the number of years of publication), sorted from highest to lowest. Then, sorted from highest to lowest, the highest H index of the authors of each paper. Finally, sorted from highest to lowest, the year of publication.
2. Sixteen percent of the papers of the last three years (most recent papers), not including conference proceedings, in ascending order according to the quartile of the journal (from Q1 to Q4). Then, sorted by the impact factor / CiteScore of the journal. Finally, from highest to lowest, the highest H index of the authors of each paper.
3. Four percent of conference proceedings of the last three years, sorted from highest to lowest, the highest H index of the authors of each paper.
4. Other pertinent articles that fulfilled one of the previous characteristics, although they did not comply with the search equation, which were found as bibliographic references in previously reviewed articles.
5. In any case, at least a third of the papers must have been published in the last five years.

Furthermore, since two bibliographic databases that differ in the procedure were used to define the number of citations per paper, Google Scholar was used as a standardization instrument. From there, the number of citations per paper was obtained [108].

Based on the ranking obtained, the review was performed. In this way, 170 papers were reviewed in-depth, and sixty were selected for this survey (Figure 2.1).

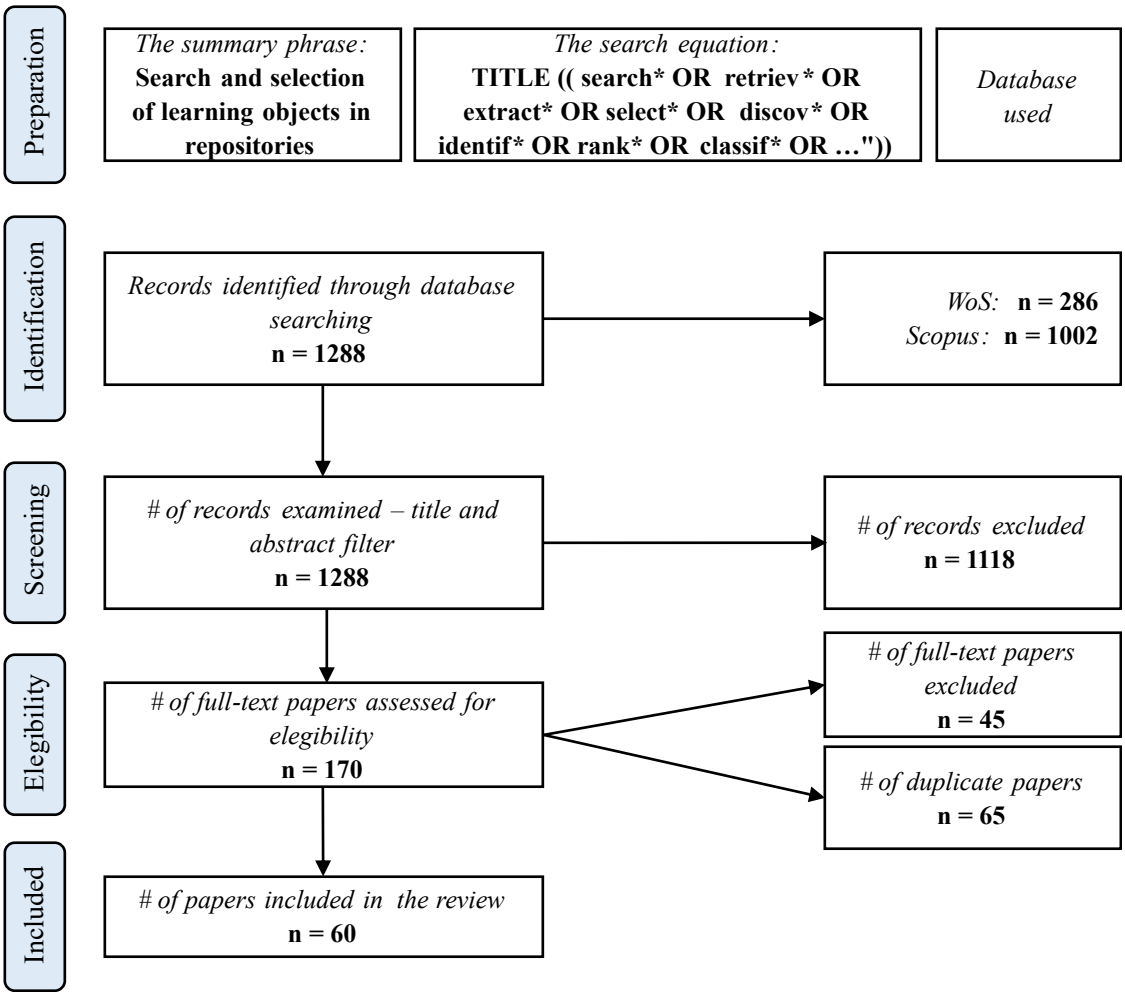


Figure 2.1: Flowchart of the systematic review

2.3. Survey: search and selection of LO

Based on the review of the papers, three strategies were characterized for the search and selection of OAs in repositories.

- Classic search with keywords in the metadata.
- Search based on content.
- Search based on hybrid methods.

For each category, papers are presented in chronological order from the oldest to the most recent (Table 2.1).

Table 2.1: List of papers on search and selection of learning objects in repositories

Paper	Category	Ct.	Ct/ye	Experim./Evaluation	Methods, Techniques, Tools
Ochoa 2006 [102]	Metadata	88	5.9		Fuzzy Logic; Graph Theory
Keleberda 2006 [69]		19	1.3	OnToIDF was implemented	IDEF5 method; OWL; OntoClean methodology; ontologies
Lee 2006 [80]		18	1.2	Two datasets in the Java language domain	Ontology, Automatic Query Expansion (AQE)
Ochoa 2008 [103]		127	9.8	10 users, 10 lessons	Learning to Rank (Ranknet)
Ternier 2008 [154]		68	5.2	European repositories federation	XML-based query engine, keyword-based search, Apache Lucene
Ghebghoub 2008 [53]		16	1.2		Jena Framework; OWL; SPARQL; ontologie
Wang S 2008 [168]		51	3.9	A software system LOSON developed	Ontology
Biletskiy 2009 [23]		74	6.2	50 LOM documents, two learner profiles	Ontologies
Bianchi 2009 [22]		22	1.8	An ontology was built from 7 complementary ones	Ontologies
Martinez-Zaina 2009 [92]		9	0.8	A group of students in a course for 4 months	AHA! architecture; Nearest Neighbor; SEDHI; W4H semantics
Yen 2010 [171]		76	6.9	21,000 LO, 3 topics, 40 users	PageRank, time series, social evaluation mechanism, cosine similarity
Barcelos 2011 [18]		29	2.9	A Java-based prototype, mediated by the JADE	Multi-Agent Systems (MAS), Distributed AI, JADE framework, Java
Yoosooka 2011 [173]		11	1.1	A prototype was developed and evaluated experimentally with 60 learners	DBpedia; Dublin Core; Linked Data; Moodle; OWL; PHP language; Protégé; RDF; XML Declarative Description; SCORM; SPARQL; SWI-Prolog; ontology, IMS Learner Information Package
Vian 2011 [160]		11	1.1	A prototype which was tested on two repositories	AQE; CORDRA; IA; JADE, Jena and Protégé frameworks; Java; MAS; TF-IDF; ontologies
Hsu 2012 [60]		44	4.9	A Java-based prototype, 125 LO, 217 relations between LO, 9 rules	Ontology-based reasoning, rule-based inference, Java

Table 2.1: Continuation

Paper	Category	Ct.	Ct/ye	Experim./Evaluation	Methods, Techniques, Tools
Lopez 2012 [84]		38	4.2	Two Machine Learnig (ML) algorithms tested	C4.5 algorithm; ML MLkNN and RAKEL algorithms; Support Vector Machines (SVM); collaborative tagging; multi-label classification
Ke 2013 [67]		6	0.8	A prototype tested on 3,164 records from 8 learning activities, 813 LO, 4 experts and 41 students	Multi-criteria Decision-making (MCDM); ELECTRE method; center average defuzzifier
Rocha-Campos 2013 [122]		12	1.5	A prototype on two dissimilar repositories and different metadata standards	Ontology, AQE, MAS
Achour 2013 [3]		14	1.8	A prototype was developed and the ontologies were created	Protégé and Jena frameworks; OWL; SPARQL; ontologies; Java J2EE
Yigit 2014 [172]		29	4.1	The SDUNESA software, 12 instructors, 7 criteria	MCDM , AJAX, XML and SOA Web Services, Analytical Hierarchy Process method (AHP)
Sabitha 2016 [126]		22	4.4	Four learning styles, 1,026 LO, 35 metadata attributes	Fuzzy C-Means clustering
Koutsomitropoulos 2017 [75]		7	1.8	Integration with the eClass LMS	Ontologies, AQE
Barbagallo 2017 [17]		16	4.0	13 health professionals, validated in the osteoporosis domain	Ontologies, ALCHEMY NLP tool, MOODLE LMS, MySQL
Koutsomitropoulos 2018 [74]		18	6.0	The system was implemented in a university to attend online courses in mathematics and medicine	AQE; WebProtégé; ontologies; thesaurus; Simple Knowledge Organization System (SKOS)
Sucunuta 2019 [149]		3	1.5	A prototype was implemented in Python	Latent Dirichlet Allocation (LDA); Python; Scrapy framework, cosine similarity
Koutsomitropoulos 2020 [73]		3	3.0	Two datasets (of 1M and 155963 records) harvested from PubMed and MERLOT.	AQE; Doc2Vec; thesaurus; ontologies
Aguilar 2020 [5]		6	6.0	Datasets of learning objects, scientific articles and patents, among others	Best Matching 25 (BM25), the Latent Semantic Analysis (LSA), Doc2Vec, and the Latent Dirichlet Allocation (LDA)
Hassan 2011 [59]	Full-Text	28	2.8	Dataset in computer science domain, 14 topics	Classifiers: Naive Bayes, SVM, and support vector regression

Table 2.1: Continuation

Paper	Category	Ct.	Ct/ye	Experim./Evaluation	Methods, Techniques, Tools
Abolkasim 2016 [2]		3	0.6	51 videos, 2,949 user comments used, and 1,223 unique entities of semantic annotations extracted	Java; Jena framework; SPARQL; Youtube; ontologies; semantic annotation
Zeng 2017 [176]		11	2.8	7,510 diabetic questions, 144 diabetic patient educational materials	Topic modeling with LDA; semantic group-based model; TF-IDF; Vector Space Model (VSM); Cytoscape; Apache Tika
Rahman 2017, 2018 [117] [116]		39	9.8	36 first-year undergraduate students	Fuzzy classification, C4.5 algorithm, Google search
Gasparetti 2018 [52]		49	16.3	Dataset of three collections of different domains	C4.5 algorithm, multilayer perceptron neural network, Naive Bayes classifier, Tagme annotation tool, Wikipedia like weak ontology, Feature Information Gain to feature selection
Rosewelt 2020 [124]		8	8.0	Sentiment analysis dataset from GitHub for performing text analysis and the amazon product review dataset	Embedded feature selection and Fuzzy Decision Tree-based CNN
Molavi 2020 [97]		2	2.0	123 OEA transcripts of Coursera and Khan Academy, educational Youtube dataset	LDA, C_v Coherence
Wang T 2007 [169]	Hybrid	98	7.0	Java Learning Object Ontology (JLOO): 158 concepts, 94 relations	11-point average precision; AQE; Mean Absolute Error; SCORM; TF-IDF
Bolettieri 2007 [25]		34	2.4	The retrieval user interface prototypes were built up from J2EE Web Application	Apache Poi; Java J2EE; MPEG-7; XML database; video OCR; speech recognition; cut detection
Cernea 2008 [38]		39	3.0		Latent Semantic Indexing, collaborative tagging, folksonomy, Pearson Correlation Coefficients
Lee 2008 [81]		128	9.8	Two datasets, six topics, JLOO	Ontologies, AQE, ambiguity elimination function
Zhuhadar 2008 [178]		44	3.4	A training set, 10 concepts, 28 subconcepts and 2,812 documents	Ontology, clustering, cosine similarity, TF-IDF, semantic web
Lemnitzer 2008 [83]		21	1.6	A query has been run on the document set with and without the ontology	WordNet; EuroWordNet, ontology; cross-lingual retrieval, CLaRK System

Table 2.1: Continuation

Paper	Category	Ct.	Ct/ye	Experim./Evaluation	Methods, Techniques, Tools
Zhuhadar 2008a [177]		24	1.8	2,812 documents, from different areas of knowledge, 10 user profiles	Entropy measure; Top-n recall and precision measure; clustering, ontology
Shih 2008 [138]		18	1.4	A prototype with a grid composed of 33 computers interconnected in 4 sites, 12 teachers	Data Grids, ontology, Globus Toolkit
Shih 2009a [139]		26	2.2	A metropolitan-scale data grid testbed. 9,600,000 documents were indexed.	Data Grids, ontology, Globus Toolkit
Khribi 2009 [70]		485	40.4	Data file obtained from 11,542 sessions	Cosine similarity, Apache Nutch search engine, association rules, collaborative filtering; content based filtering
Shih 2009 [137]		41	3.4	Web-based prototype, 20 students	Apache Lucene, expert system shell DRAMA, AQE, ontologies, inverted file indexing, Ontology Building Algorithm, cosine similarity
Zhuhadar 2009 [180]		18	1.5	A semantic search engine with 10 learner profiles with three sizes of queries (1, 2, and 3 keywords)	Apache Nutch; Information Filtering; TF-IDF; cosine similarity; entropy and Top-n recall and precision measures; clustering, ontology
Ahmed-Ouamer 2010 [6]		25	2.3	A semantic search engine with 10 learner profiles with three sizes of queries (1, 2, and 3 keywords)	VSM, ontology, cosine measure, Jena framework, inference engine
Zhuhadar 2010 [179]		25	2.3	The HyperManyMedia search engine	Hand-made ontology, VSM, cosine similarity, K-way clustering, Bisecting K-Means, recommendation as Rule-based
Nasraoui 2010 [99]		22	2.0	7,424 documents (4,888 in English and 2,536 in Spanish)	Ontologies, Protégé framework; entropy measure; graph-partitioning-based clustering; k-way clustering; bisecting k-means
Smine 2013 (2011, 2012) [144] [143] [142]		31	2.8	The SRIDOP system	Java, Apache Lucene, Rocchio's algorithm, SRIDoP, TF-IDF, WOLF dictionary, precision, recall and F-score measures, semantic annotation
Atkinson 2014 [14]		20	2.9	Web-based prototype, 3,300 LO, 500 web documents, 16 teachers	ML methods: Named-Entity Recognition, Naive Bayes classifier; NLP; AQE with WordNet, LSA, Formal Concept Analysis, VSM

Table 2.1: Continuation

Paper	Category	Ct.	Ct/ye	Experim./Evaluation	Methods, Techniques, Tools
Pérez-Rodríguez 2016 [111]		15	3.0	Several workshops and 65 participants	NLP, ML, Wikipedia, Bags-of-Concepts (BoC)
Ramírez-Arellano 2017 [118]		3	0.8	84 students were randomly split into four groups	Jaccard similarity; cosine similarity
Kalyanaraman 2019 [65]				Two different topics, 394 LO	Particle Swarm Optimization; Fuzzy c-mean; K-means
Do 2020 [46]		3	3.0	97 students and 275 queries	Ontologies
Das 2020 [40]		1	1.0	Data obtained from the Ekhool LMS	PCA; Rider Optimization Algorithm; fuzzy logic classifier; t-Distributed Stochastic Neighbour Embedding
Mannar Mannan 2021 [87]				Performance was compared to search engines Bing, Google, and Yahoo with 9 top documents of the results.	Ontologies; Protégé Framework; WordNet; Wu-Palmer similarity; AQE

2.3.1. Classic search with keywords in the metadata

In [102], the authors propose, at the conceptual level, a series of metrics for the recommendation and recovery of learning objects through the use of Contextual Attention Metadata (CAM), to be collected (and stored) from user interactions with the system. In this sense, they present metrics at the level of the ranking of link analysis and similarity for the personalized and contextual recommendation.

Keleberda et al. in [69] present an approach to build ontologies of learning objects and of the student, adjusted to the IEEE LOM standard, using the OntoClean methodology and the IDEF5 method. These ontologies are mapped to OWL and are created to help in the personalized search for educational material. Additionally, they indicate that a software application they called OnToIDF was implemented, aimed at facilitating the implementation of the approaches proposed for the construction of ontologies.

Lee et al., in [80], present an approach semantic-aware, based on ontologies for the retrieval of learning objects. In this regard, they provide a semantic inference engine that connects the user's query with the metadata of the learning object, through a query expansion algorithm. To carry it out, build the user's intention tree based on an ontology of the query domain. From the concepts located at the top of this tree, the query terms are expanded. Additionally, an experiment was conducted with two sets of data in the Java programming language domain. The authors state that their model achieves a significant improvement in precision and recall metrics.

Ochoa and Duval in [103] aim to improve the present status of the search for learning objects. To do this, they review the literature and make a theoretical proposal to search for learning objects based on a classification by relevance. In the review they describe the different relevance metrics and identify three current approaches showing, mainly, their disadvantages: first, the ranking based on human review presents the difficulties of being a very expensive manual process and static in time, which does not adapt to the different user requirements. The second, the ranking based on text similarity, has the advantage that it is easily calculated but due to the low amount of text in the metadata of the learning objects they lead to low performance in the results. The third, the ranking based on user profile, in which the topics of the user's profile are compared with the classification of the learning object; its outstanding disadvantage is that it does not integrate well with the user workflow. Hence, the authors suggest three characteristics that a new generation of search mechanisms for learning objects should meet, such as: taking into account the information generated by the human user; calculating its value automatically; and does not requiring the conscious intervention of the user.

In addition to the review of the literature, Ochoa and Duval [103] propose LearnRank, a metric based on different relevance metrics. For its implementation, the RankNet algorithm was selected, which uses a neural network to learn the optimal ranking based on the original metrics. Furthermore, in order to evaluate the proposal, an experiment was carried out in which ten users participated and ten lessons were created on topics in the computer area. The tests performed showed a significant increase in the performance of the ranking compared with the reference rank.

In another article by [154], the authors describe ProLearn Query Language (PLQL): a query language for learning objects in repositories. In its definition, an exact search has been combined

through the metadata of the learning objects using XML-based query engines and an approximate search through keyword-based searches using information retrieval engines such as Lucene. In this sense, the exact search is executed first and then pruned with the approximate search. In addition, PLQL in combination with a simple query interface (SQI) and an application profile of LOM learning object metadata from a European repositories federation, called Learning Resource Exchange (LRE), performs federated searches in all those repositories.

In [53] an approach is presented to semantically index learning objects through a LOM ontology and, at the same time, a tool is used to describe and search for learning resources in a repository. With the LOM ontology, built through OWL with a multilingual approach (French and English), the concepts and relationships of learning objects are represented. Finally, for the search of educational resources SPARQL queries are used.

Wang [168], in his article, proposes a pedagogical organizational structure of learning objects based on ontologies in order to facilitate the process of knowledge sharing. For this purpose, a small scale ontology was implemented and, based on it, a software called LOSON was developed to access learning objects through their semantic relationships, as well as through keywords and metadata. Finally, the author concludes that the search for learning objects through ontologies is more efficient compared to the search based on metadata. However, he points out that ontologies do not replace metadata, but rather that it adds semantic relationships helping the learner in his learning process.

Biletskiy et al., in [23], propose a personalized search approach for educational Web resources making use of the student profile and the descriptions of the learning objects, based on the IEEE LOM and IMS LIP standards. To carry it out, they propose to develop ontologies of the student and the LO. In addition, the proposed solution allows the student's feedback on the suitability of the educational resources recovered in the personalized search. Finally, for the validation of the proposed approach, a prototype of the learning object search and retrieval system was implemented which provides the student with the twenty best-ranked learning objects that, according to the authors, demonstrate the validity of the proposal.

In [22], it's presented the design, implementation, and testing of a semantic infrastructure, based on the web and an educational ontology, used to support the annotation and recovery of learning objects in the domain of aquatic environments and their resources, at European level. The ontology was built from seven others, from the same domain that covered complementary topics, to be used in a web portal.

In the work of Martinez Zaina in [92], it's proposed the architecture of a system that creates e-learning scenarios dynamically, according to the profile of the learner. In this regard, the profile of the learner is constructed from the analysis of the preferences data of their contextual information in their interactions. Also, the system consists of three layers. The first is the e-learning scenarios layer; the second, the evaluation profile layer; and the last one, the monitoring layer. On the other hand, the recovery of learning objects is made in three moments. In the first moment, the learning objects are recovered according to the apprentice's query, using the Title, Description, and Keywords fields of the LOM standard, belonging to the General category. Then, in the second moment, the results of the previous step are filtered based on the student's learning profile and the fields of the Educational category (LOM standard) of the learning objects. Thirdly, in the last

moment, with these results, the e-learning scenario is composed. Lastly, an experiment was carried out with a group of students in a course, observing their participation in four months, using this architecture. The authors did not present the results of this experiment.

Yen et al. [171] propose a process flow to help users retrieve learning objects in federated repositories (under the SCORM and CORDRA specifications) that the authors call "Guided Search" based on ranking metrics and a more efficient search algorithm. To carry it out, three steps were proposed. First, the use of weighting metrics of learning objects based on time series and a social assessment mechanism inspired by Web 2.0 and social networks. Then, the ranking metrics were used to retrieve the learning objects in a specific order according to the users' query. Finally, the tool called «Search Guider» helps users to recover relevant learning objects according to their needs.

Based on the process flow proposed by [171], on a developed system called «MINE Registry» that stores and shares around 21,000 learning objects, an experiment is conducted in three stages. In the first stage, the overall performance of the MINE Registry is evaluated, in which high values of the precision measure were not obtained, but on the recall measures were. In the second one, the performance is compared with three other known methods (grid, ontological, and inference network approaches) and close results were obtained. Finally, the guided search is evaluated in which shorter times for users were achieved.

In the article by Barcelos et al. [18] is presented the AgCAT system, based on agents, that allows the federated cataloging of learning objects. This system is part of the MILOS infrastructure that gives computational support to the Brazilian OBAA learning object metadata standard. In this sense, the system architecture and the prototype implemented are described. This prototype was developed in JAVA, mediated by the JADE platform. Additionally, the system has an agent called Finder that allows learning objects to be recovered in the local repository and in other remote repositories, which are part of the federated system, based on metadata and logical expressions. To achieve the federated search, it has another agent called InterLibrarian that implements the federation of catalogs and interacts with other InterLibrarian agents of the other repositories. On the other hand, tests were conducted with the prototype, especially in the validation of the search functions and the agent in charge of the catalog. Finally, the authors concluded that it is possible to build a federated metadata search service, based entirely on agents.

Yoosooka et al. en [173] propose an approach for the recovery of learning objects via Linked Open Data. In this sense, this approach allows the adaptive selection of learning objects in local and external repositories, according to the learning styles of the learner, and, besides, discover additional related learning objects from the LOD cloud. Also, the architecture of the framework consists of four complementary models. The first, the learner model, stores learner profiles, based on the IMS LIP (Learner Information Package) standard. The second is the domain model, which contains ontologies that describe the knowledge structures of the domain based on the ACM 2008 computer curriculum. The third, the adaptation model, which stores the rules of dynamic selection of learning objects, using declarative descriptions XML. The last one, the model of learning resources, stores the learning objects that comply with the SCORM standard, which is associated with the model of the domain through ontologies. Finally, a prototype was developed, which was evaluated experimentally with sixty undergraduate learners. The results of the experiment showed that there were positive effects in terms of student satisfaction.

In [160], the authors propose a multi-agent model, supported in ontologies, to index and retrieve learning objects stored in different repositories with different metadata standards. The objective of this model is to promote the reuse of learning objects and improve the metrics in the retrieval of these educational resources. For this purpose, the model is composed of six main components: the repositories, the indexing agent, the search agent, the metadata mapping, the domain ontologies, and the search service interface. Domain ontologies are used to expand the keywords of the user's query. Also, a prototype was built, implemented in Java and JADE, Protégé, and Jena frameworks to evaluate the model, which was tested on two real repositories and an ontology in the domain of computer security. According to the authors, the tests show that the model can improve the precision, recall, and performance of learning object retrieval, and also the reuse and exchange of learning objects between repositories.

In the work of Hsu [60], a Multi-layered Semantic LOM Framework (MSLF) is proposed to integrate Semantic Web technologies in LOM, in order to overcome the weakness of LOM with respect to its lack of semantic metadata. This framework was used to implement an intelligent prototype LOM, to find relevant learning objects in a repository, called LOFinder; which consists of four main components, namely: the LOM base, the knowledge base, the search agent, and the inference agent. In addition, LOFinder integrates three learning object recovery approaches such as LOM metadata, ontology-based reasoning, and rule-based inference.

López et al. [84] propose the use of collaborative tagging of learning objects, using Web 2.0 techniques that allow the user to interact with others to exchange content. With this, they seek to overcome the difficulty of extracting semantic information from learning objects in a repository. To carry it out, they propose the use of multiple labels that are previously classified using machine learning methods. In this regard, they tested two algorithms: multi-label k-nearest neighbor (MLkNN) and random k-labelsets (RAKEL). The latter achieved better performance and showed that it can be used to improve the classification of learning objects in types of queries based on the content of the metadata.

In [67], Ke et al. propose a platform for the optimal selection of learning objects from an electronic portfolio for learners. For this purpose, it uses classic IR techniques to extract key concepts of relevant information to manage the learning activity and candidate learning objects. Then, through a context-based utility model and a multi-criteria decision analysis determines the optimal order of learning objects. Regarding the context module, it collects information at runtime of the student's learning activities from the log file. On the other hand, a prototype was tested in a university on 3164 records of relevant data obtained from eight events of a learning project, carried out by four experts in a domain and forty-one students, and 813 learning objects. Finally, the authors affirm that the experimental results demonstrate the effectiveness of providing knowledge for decision making in the choice of resources for their learning process.

Rocha Campos et al. in [122], propose a model for the indexing, search and retrieval of learning objects in different repositories and different metadata standards, supported in a multi-agent system and ontologies, which provides the user with a classification of the results based on their profile and also prioritized by areas of knowledge. In this sense, the model is composed of the Searcher Agent, the Profile Agent, the Indexer Agent, the Database Profile, and the Ontology. The Searcher Agent acts as the coordinator of the entire system; the Profile Agent enriches the process with personalized search terms according to the user, supported by specific domain ontologies, using

the available knowledge bases; and the Indexer Agent does its work in heterogeneous repositories and different standards and normalizes them. Finally, the model was validated with a prototype on two dissimilar repositories and different metadata standards showing, according to the authors, improvements in coverage and recovery of learning objects more adapted to the context of the user.

Achour et al. [3] propose a model for the search, access, and reuse of learning objects, based on ontologies that also allows multilingual searches (Arabic, French, and English) of complete courses or fragments of courses. In this sense, a prototype was implemented in a learning portal in the Computer Science domain and the topic of OOP in Java. With respect to the model, it is based on three ontologies and a LOM application profile. The first ontology refers to the domain or more general field of learning, organized into subdomains; the second, describes the semantic contents of each of the topics within a subdomain; and the third, based on the LOM standard, is used to describe the different types of educational resources and the smallest units available in the repository. Finally, all the ontologies were created using Protégé and the OWL language and the prototype was developed in J2EE, SPARQL, and Jena technologies.

In the work of Yigit et al. [172], SDUNESA repository software is presented to store, share and select learning objects. It was implemented with Web 2.0 technologies such as XML, AJAX, and SOA Web Services. Furthermore, the analytical hierarchy process (AHP) was used for the selection of the learning objects, which is part of the multi-criteria decision-making (MCDM) methods. In this sense, for the definition of the AHP criteria for the selection of learning objects, twelve instructors who work in the computer engineering department of a university were interviewed. In addition, seven qualitative and quantitative criteria were defined (with their respective sub-criteria) such as the type of learning resource, format, difficulty, level of interactivity, semantic density, the expected role of the end user, and structure. In this study, no evaluation and comparison tests are done with other methods.

Sabitha et al. [126] propose a data mining approach by clustering, according to the attributes of the metadata and the learning styles of the students, for the particularized delivery of learning and knowledge objects to learners. To achieve this, the learning objects are mapped into four dimensions of learning styles (participation, processing, presentation, and organization) and then grouped by fuzzy c-mean clustering. The authors conclude that this approach achieves a more personalized and authentic learning experience.

In the paper of [17], the ELSE system (E-Learning for the Semantic ECM) is presented, which integrates semantic search methodologies and e-learning technologies, which allows the creation of customized courses according to the student's requirements and preferences. It is based on a reference ontology that contains the concepts and relationships of the particular domain, selected by a panel of experts in that domain. In addition, students can choose a combination of the inductive vs. deductive and sequential vs. global approaches for the course; and also specify their training needs starting from the ontology. In this sense, the semantic similarity method SemSim [50] is used. It also allows integration with the MOODLE platform. Finally, it was validated in the osteoporosis domain and in general, the judgment was positive both in terms of usability and personalization.

In the work by Koutsomitropoulos et al. [75], it is mentioned that instructors and students are faced with two major problems when using digital educational resources: the first, is the difficulty to discover and retrieve material complementary to the courses; and in the same line, the second,

the excessive manual work in the queries and the selection of educational material, based on the results. Therefore, they propose a framework and a service to face these problems, making use of the expansion of queries based on keywords that describe a particular course. This service is composed of three main components: the development of a thesaurus under ontologies; a subsystem of management of learning objects; and finally, a semantic middleware to evaluate the semantic relevance between the keywords and the thesaurus, to perform the query expansion and conduct the federated search in remote repositories. In this sense, the authors suggest that the expansion of the query contributes to an improved recovery. Finally, they state that a prototype of the proposed system is in operation for university's online courses.

In [74], the authors propose and introduce the notion of a semantic-aware learning object ontology repository, which aims to help its publication, discovery, and reuse. In order to do this, they propose improving and maintaining the metadata of educational resources through of ontologies in specific domains. Also, linking this repository to thesauri and other semantically linked learning objects. To this end, they designed ontologies of learning objects based on the IEEE LOM standard, integrated to the SKOS standard (Simple Knowledge Organization System). Additionally, the keywords of the queries are automatically expanded through the use of terminological thesauri. These expanded queries are addressed to the local repository and other external repositories in a federated manner. Finally, the proposed system was implemented in a university to attend online courses in mathematics and medicine.

Sucunuta et al. [149], present an experimental proposal to identify and recommend learning objects of the same subject, based on the extraction of metadata from the repositories. The proposal is presented in two parts: the first, the extraction of data from the repository; and the second, the recommendation process. Regarding the recommendation process, it was implemented in Python, using the Latent Dirichlet Analysis generative model library and cosine similarity measure. Additionally, resources were filtered with the best user ratings.

In [73], a method was proposed to help curators and teachers to annotate open educational resources by themes, using ontologies and vector-based terminology learning (Doc2Vec) in a combined manner. The model has been tested in the field of medical literature using two datasets harvested from PubMed and MERLOT.

The authors of [5] conducted an evaluation of four feature extraction techniques in the educational domain on information retrieval and recommendation systems. The techniques used were: the Best Matching 25 (BM25), the Latent Semantic Analysis (LSA), Doc2Vec, and the Latent Dirichlet Allocation (LDA). To carry it out, datasets of learning objects, scientific articles and patents, among others, were used. The results obtained varied according to the use case, without there being a predominant technique in all cases.

2.3.2. Full-text search

In the paper of [59], the possibility of automatically identifying educational resources was evaluated, for which experiments were carried out with a dataset constructed on fourteen topics of computer science and manually annotated. In these experiments three classifiers were used: Naive

Bayes, support vector machines and support vector regression. In this way, the authors conclude that the educational value of a learning object can be automatically assigned with high precision.

Abolkasim et al., in [2], propose a computational model to measure diversity (according to the Stirling framework) in terms of variety, balance, and disparity, to analyze the social cloud, in this case, comments, user profiles, and other metadata of YouTube videos. The above, to identify and rank the most appropriate videos for learners. To this end, it uses semantic annotations of user comments in videos against an ontology, of a specific domain, to facilitate their classification according to diversity. Also, to test the proposed model, a set of videos about job interviews was used. In this sense, fifty-one videos, 2949 user comments were used, and 1223 unique entities of semantic annotations were extracted. The Body Language ontology was used to assist in the process of semantic extraction and the calculation of the dimensions of diversity.

In the work of [176], a study is presented in which three information retrieval algorithms are compared for the recommendation of educational materials about diabetes for questions asked by patients. In this sense, the authors point out the importance of satisfying the information needs of patients, in order to facilitate self-management and care of their disease. To carry it out, they assessed the algorithms of vector space modeling (as baseline), Latent Dirichlet Allocation, and semantic group-based model on educational materials from the Mayo Clinic database and questions obtained from patients in the TuDiabetes web forum. In addition, for its evaluation, the precision metric of the top-ranked documents was used. According to this, it was determined that: the vocabulary of the language used in the educational materials is different from that one used in the forum of questions; the topic modeling-based model had better performance and has the potential to accurately recommend educational materials; and, finally, this one can mitigate the difference of vocabularies between the educational materials and the questions.

Rahman and Abdullah, in [116, 117], in their article, propose a framework for the use of learners, within an institutional instructional environment, and that adds to Google Search an ability to filter their results based on their academic background, the behavior of learning when they use the search engine, and the behavior patterns of other students. To achieve this, the framework makes use of dynamic student profiles and a grouping algorithm. In addition, this proposal seeks to overcome the difficulty of generic search engines in the sense that they do not take into account the differences in the learning profiles of users and that, according to a reported study, only 8% of the results were of educational resources according to the learner's query. On the other hand, the proposed framework consists of two modules: one, the dynamic profile of students created from their academic record; the other, the re-rankig of the results of Google Search based on the profile of the learners. According to that, students are classified as beginner, intermediate or master through the C4.5 algorithm of the decision tree. Furthermore, the framework was tested through a prototype in 36 first-year undergraduate students at the University of Malaya, showing that the application was able to customize Google Search results according to the particular needs of the students.

Gasparetti et at. [52] propose an approach for the identification of prerequisites of textual learning objects, through machine learning. To carry it out, the learning objects are tokenized, their terms labeled and the semantic relationships between the terms are extracted. This last task is carried out with the use of Wikipedia, which is considered a weak ontology, using the Tagme annotation tool. Next, the recognition of requirements is performed using automatic learning classification

algorithms such as C4.5 decision tree, multilayer perceptron neural network, and Naive Bayes. Finally, to evaluate the accuracy of the approach, experiments are conducted on real online courses in different domains.

Rosewelt and Renjit [124] propose an e-learning content recommendation system using embedded feature selection and fuzzy decision tree-based CNN. It was tested using a text analysis GitHub dataset, and the Amazon review dataset was used for the system training process. According to the authors, the proposed system identifies the relevant contents and recommends them to the learners to improve their learning capacity.

The authors of [97] propose an automatic model for the extraction of topics focused on open educational resources (OEAs) on the subject of data science. For this, they compiled the transcripts of 123 OEAs on three skills required in this subject on the educational platforms of Coursera and Khan Academy, and then they identified the topics that need to be covered in each skill using LDA. Subsequently, they tested the model using an educational YouTube dataset, obtaining an F1-score of 79%.

2.3.3. Search based on hybrid methods

In the article of [169], a personalized adaptive hybrid system to recommend learning objects from repositories on the Internet and compatible with SCORM, is proposed. It is supported by ontologies and based on preference and correlation, to classify the degree of relevance according to the intention and choice of the learner. In the first stage, the automatic expansion of the learner's query is conducted, supported by ontologies, with which learning objects are discovered. Subsequently, on the previous results, it applies two classification algorithms: the first, based on the learner's preference and profile, and the second, based on the suggestions of the neighbors through a correlation-based algorithm. Both results are integrated into a single that is presented to the learner. Finally, the learner can return two types of feedback, content, and preference.

In the work of Bolettieri et al. [25], a digital repository architecture is presented that allows the reusing of multimedia learning objects, for which, it automatically extracts semantic metadata, in LOM and MPEG-7 standards, exploiting existing open source technologies such as video OCR, speech recognition, cut detection and visual descriptors MPEG-7, among others. In this regard, the system allows you to recover multimedia objects in various formats (pdf/word documents, web pages, PowerPoint presentations, audio/video documents, etc.), combining full text recovery based on text automatically extracted from audios and videos, of metadata and MPEG-7 visual descriptors.

Cernea et al. [38], in their paper, propose an architecture called SOAF (for its first letters in Spanish of "semantics of learning objects based on folksonomies") for the semantic indexing of LOs in repositories. In this sense, the metadata used in indexing, through Latent Semantic Indexing, are obtained through three sources: the automatic semantic indexing based on the low-level characteristics of the learning objects; the descriptors supplied by the authors; and the collaborative annotations of the learners. In addition, with respect to tags and before being incorporated them into metadata, they are processed through a collaborative filter based on user profiles and linked to an ontology of a specific field.

Lee et al. [81] propose an approach based on ontologies for the semantic-aware retrieval of learning objects, which has two characteristics: the first, an automatic expansion algorithm based on ontologies; and the second, an ambiguity elimination function to adjust the unsuitable query terms. With the proposed model, the authors point out that two drawbacks of traditional information retrieval technology based on keywords of the Salton vector space model are overcome. The first drawback is the need to index the content in advance, which may fail in the case of learning objects given the broad context in which they are immersed, which may contain multimedia elements difficult to include in the index. The other one has to do with the presence of learning objects, not semantically related, that have common keywords. In addition, for the recovery of learning objects in the repository, indexed in a standard way, they are searched in the same way as in the ontology assistant system.

To corroborate the model proposed by Lee et al. [81] experiments were conducted on the automatic expansion algorithm and the semantic-aware learning object retrieval, compared to the performance of traditional keyword-based recovery techniques. The performance was measured based on three metrics used by the information retrieval community: precision, recall, and F-measure. In all the results, the proposed approach overcame the traditional one, based on keywords.

In the paper by Zhuhadar and Nasraoui [178], the authors present an e-Learning platform for the personalized recovery of learning objects that make use of the standards of the Semantic Web to represent the contents and profiles of users as ontologies and re-ranking the search results based on how the terms are assigned to these ontologies. To achieve this, they propose a three-layer architecture: semantics, algorithms, and personalized interface. In this sense, in the first layer, a semantic profile of the student is elaborated based on his search history; queries are constructed with keywords related to the student profile, the courses and the most significant terms of each concept; and a taxonomy of all the documents of the repository is extracted, clustering them in categories finer than those given by the colleges. Then, for a given query of a student, documents similar to the terms of the query are retrieved; these results are re-ranked according to the semantic profile of the student and the most similar clusters of concepts. Finally, the authors state that their results show the effectiveness of the re-ranking of search results based on the semantic profile of the student.

Lemnitzer et al. [83] present the LT4EL project, a semantic search engine, integrated into an LMS that allows the interlingual recovery of learning objects, in order to improve accessibility to these resources and find learning objects in languages other than the user's. For this, an ontology was built in a specific domain (computation), with lexicons in eight languages (English, German, Dutch, Polish, Portuguese, Romanian, Bulgarian and Czech) and around 1300 concepts, based on lexical elements of all the project languages. The EuroWordNet project was used to build the multilingual lexicon. Additionally, the information recovery is carried out in two steps: initially, based on the user's search terms, the elements of the ontology that coincide with these terms are presented so that the user can choose the concepts most appropriate to their interests; in the second step, the search is done with the elements of the ontologies selected by the user. If there is no match of terms with the lexicons and the ontology, the search is done by keywords and full text on the same learning objects.

Zhuhadar et al. in [177], a personalized search approach is presented in an e-Learning platform that takes advantage of the semantic web to represent the content of learning objects and user profiles,

which aims to address two weaknesses that arise when learners look for educational resources. On the one hand, improve the accuracy of search results. On the other hand, it takes into account the user's profile in the results of these searches. To achieve this, the framework consists of four phases: the creation of the e-learning domain, starting with the information of the concepts and subconcepts given by the college; the generation of the semantic profile of the learner based on their history of learning objects consulted over a period of time; the clustering of documents to discover more refined concepts; and the evaluation of the results based on the profile of the learner and the group closest to this profile. The system was tested experimentally with the construction of a corpus of 2812 documents and ten user profiles, from different areas of knowledge. Finally, the authors manage to experimentally prove that the inclusion of the user context, through the profile, improve the indicators of accuracy and coverage.

In the work of Shih et al. [138], the authors present an approach to recover learning objects compatible with SCORM in data grid environments, trying to minimize the time of processing queries and the transmission of content when learning objects are recovered. To do this, they propose the use of a centralized index, called "Taxonomic Indexing Trees (TI-trees)", which is built based on a Vector Space Model in which the metadata, content, and structure of the object are included. For the structure, learning objects are classified based on the underlying taxonomic scheme of the system "Dewey Decimal Classification (DDC)", which is widely used in the classification of books in libraries. Additionally, based on grid monitoring tools, the estimated time of transmission of the learning object is calculated, in order for the user to consider whether he recovers it or not. Finally, a prototype was built, at the level of the metropolitan area, with a grid composed of thirty-three computers interconnected in four sites, which was put to the test by twelve primary school teachers who answered a survey. In this regard, the authors found that they were satisfied with the response time of the system.

In the paper of [139], the same authors of [138] propose an extended approach to the one presented there. First, the index based on taxonomy extends to an index based on ontology. Secondly, in this work, the main objective is aimed at improving the precision of the searches instead of the transmission speed. Finally, they use different similarity functions. In this sense, here they use a combined similarity function between full text and metadata. All this supported in a new architecture that allows the data grid to recover learning objects in a more precise way.

In [137], the Shih and Tseng paper, a ubiquitous learning information retrieval system context-based is proposed, founded on instructional strategies and goals. The proposed system is composed of four components: the user interface for the input of the query and the detection of the context, mainly related to the student's location; the expansion of the query by the inference of rules; the recovery of content; and the construction of ontology and the generation of rules. In this context, for the search and classification of teaching contents, a combined technique of similarity based on keywords (full text through the cosine function) and on metadata (based on the number of matching attributes) was used. In addition, to evaluate the system, a prototype was developed that was tested in a primary school and three experiments were conducted with 20 of its students. Finally, the authors present three results: the first, the proposed system can improve the recovery performance based on the context (in a ubiquitous learning scenario); the second, according to a survey conducted, shows that it is feasible for teachers to help the system generate a simple ontology based on a predefined course scheme; and the third, the results show that expanded

queries work better than the original.

In the work by Khribi et al. [70], a personalized online recommender system for students is proposed, which does not require explicit feedback. The recommended learning objects are calculated based on the student's recent browsing history, as well as the content of the learning object and the exploitation of similarities and differences between the preferences of the students. For the recommendation of the learning objects, a hybrid system is used, where the results of two recommendation approaches are combined and integrated: one, a cascade collaborative filtering, increased by recommendation by content of the learning object; and another, a recommendation approach based on weighted content and collaborative filtering based on the learner's profile. Both techniques are executed separately and the results are integrated into a single recommendation set. Equally important, to improve the recommendation based on the content, made through the Apache Nutch search engine, the content of the LOM educational metadata that provides additional information of the learning objects are added, automatically, to the native index generated by the search engine (inverted index). The results are sorted through the cosine similarity (TF-IDF vector).

In the work of [180], the authors present a system of personalized search for learning objects, enriched semantically. To carry it out, they propose a four-layer architecture: semantic representation of knowledge, algorithms, personalization interface and dual representation of the user's semantic profile. Initially, the repository is hierarchically encoded as a structured ontology on tree OWL, where the learning objects are the leaves, and in which, the taxonomy provided by the educational institution is combined with a taxonomy extracted from the documents themselves through clustering. Subsequently, one of these clusters, the most semantically related to the user's profile, is added to the same profile to complement it. In addition, for a given query from a user, the results are reclassified based on their profile. On the other hand, the user's profile is modified dynamically when their information interests change in time. Finally, the system was tested experimentally with ten users, showing improvement in the indicators of precision and recall; and it was implemented on a university web platform used by its students.

Ahmed-Ouamer et al. [6] present an ontology-based information retrieval approach called OBIREX (Ontology-Based Information Retrieval for E-learning of Computer Science). For indexing, the descriptive words taken directly from the learning objects are not used, but concepts extracted from the ontology. The above, complemented with resources obtained through an inference engine on the semantic links of the learning objects recovered. In addition, tests were conducted in the domain of computer science on twenty concepts. Finally, the authors affirm that this approach allows for recovering relevant learning objects that are not recovered with traditional techniques.

Nasraoui et al. [99], in their paper, propose and evaluate the university repository of learning objects HyperManyMedia that supplies, manages and collects data to adapt to the user's search profiles, taking advantage of the standards of the Semantic Web, that allows representing the content of learning objects and user profiles. According to this, the architecture allows the generic search, the metadata search, and the semantic search. To achieve the latter, ontologies were built based on three sources: the first, the information of the courses as concepts and sub-concepts, which were provided by the colleges; the second, the semantic profiles of the users; and the third, a taxonomy extracted from the documents themselves, by unsupervised clustering techniques that, for each group, obtained the most descriptive terms, which were added to the ontology. In addition, when a search is performed on the system, the results initially obtained are reclassified based on

the user's profile. On the other hand, the experiments carried out showed, according to the authors, that the semantic profile of the user can improve the precision and recall metrics, by reclassifying the results based on past searches of the students.

In [144] (and with variations in [142, 143]), the authors propose a model oriented to the automatic annotation of learning objects with semantic metadata. This model is composed of two parts. The first one is responsible for the semantic annotation of learning objects according to their semantic categories, that is, definitions, examples, exercises, among others. The second, based on the result of the previous part and the textual content of the learning objects, it creates an inverted semantic index. On the other hand, for the annotation process, a technique based on a set of heuristic rules of contextual exploration is used to extract the linguistic structures that define the categories of learning objects. In this regard, an annotation engine for learning objects was implemented for the French language. Additionally, the index was implemented using the Apache Lucene platform, which contains the semantic categories and full text content of the learning objects. Finally, the proposed system was tested with a corpus of 1000 documents in French language and different file formats, fifteen different subjects, and twenty-five queries for each category. In this respect, the authors affirm that the results are promising, taking into account that accuracy exceeds 85% in most categories.

Atkinson et. al [14] propose an approach in which they make use of several paradigms for the semantically guided extraction, indexing, and search of educational metadata found on the Web, in order to identify educational resources and thus help teachers in this task. In this sense, the proposed model incorporates automatic learning techniques, formal analysis of concepts and natural language processing algorithms. To validate the model, a Web-based prototype was implemented and 500 documents were extracted with which three types of experiments were carried out: one, parameters setting; another, classification accuracy; and the third, quality of the extracted metadata. For the latter experiment, sixteen secondary school teachers participated. The authors mention that promising results were obtained and pointed out that the semantically guided metadata extraction can improve access and use of educational resources present on the web.

In the work of Perez-Rodriguez et al. [111], a state-of-the-art exploratory search engine is presented, according to the authors, that indexes educational resources, harvested on high-quality websites, based on the concepts it deals with, supported by a taxonomy of concepts obtained from Wikipedia. In this sense, the architecture of the system is made up of four main components: an information extraction module that harvests educational resources from web pages; a semantic annotation module based on Wikipedia; an indexing module that relates concepts with educational resources and; a query module that allows carrying out explorations, recommendations and, suggestions of alternative educational resources. In addition, among other contributions of the work, the authors point out that the proposed approach shows the viability of using Wikipedia for the construction of information retrieval systems of educational resources, in an exploratory manner. Finally, as of the date of this paper, the search engine is still operational (http://www.itec-sde.net/).

In the article by Ramirez-Arellano [118], the Management System for Merging Learning Objects (msMLO) is presented that retrieves, in a combined way, educational resources based on student learning styles and terms-based queries. To do this, the system first retrieves the learning objects based on the terms of the query. Then, it classifies them based on the student learning style, taking the ten best classified as a source for the merging process, in which a hierarchy of concepts is

constructed to avoid duplication in the subjects. Concerning the classification of learning objects, it uses in a combined way, Jacquard's similarity measures and another one, for the student's learning styles; and a third, the similarity of cosine, to calculate the relevance of learning objects based on the terms of the query. Besides, the system consists of six modules: query (which is the coordinator module), personalized retrieval, content generation, preview, retrieval, and storage. Finally, to evaluate the system, two experiments were carried out with 500 learning objects and eighty-four students, divided into four groups. Based on the results, the authors conclude that msMLO is a promising approach that provides useful educational resources to students based on their learning style, improves overall learning performance, and reduces the number of learning objects reviewed.

In [65], the authors present a distributed model, oriented to the clustering of educational resources based on the dominant cognitive content of each of them, to improve the usability of learning objects. The cognitive skills of the learners considered were the memory, concentration, perception, and logical thinking. Also, the model includes three main processes: the mapping of cognitive skills versus learning objects, the annotation and pre-processing of learning materials, and the grouping of learning objects for storage. In this third process, the K-media and Fuzzy C-media clustering techniques were tested in a hybrid manner with the Particle Swarm Optimization (PSO), the latter as a mechanism to find the centroid of each cluster. Finally, experiments were conducted with 384 learning objects, on a specific course, in which accuracy, recall, and F-score were evaluated. With them, it was possible to verify a higher performance of the K-media clustering combined with PSO.

Do et al. [46] present an engine for intelligent search in the math domain for high school based on an ontology called Search-Onto. The system was tested with ninety-seven students. They conducted 275 queries on definitions of mathematics, properties and rules, exercise kinds, and solution methods, obtaining an average precision of 74.2%. The results were double-checked.

In [40], a model is proposed that provides dynamic and continuous recommendations on an LMS based on intelligent techniques. For the construction of the model, data were obtained from the same LMS and combined techniques were used in the different stages of the process such as PCA, t-Distributed Stochastic Neighbor Embedding (t-SNE), fuzzy logical classifier, and the Rider Optimization Algorithm. The model was compared with five other state-of-the-art techniques and obtained better results considering various performance measures.

The authors of [87] propose a method to rank documents based on the conceptual content of the query words. In this sense, the query terms are extended using WordNet. Likewise, the similarity of the queries is calculated through the Wu-and-Palmer semantic distance using WordNet as a knowledge base as well.

2.4. Discussion

Based on a quantitative analysis of the review in this thesis (an expanded and updated version of [108]), several aspects are highlighted, as shown below.

- Searches based on metadata represent most of the papers reviewed (twenty-seven papers),

followed by hybrid search (twenty-five papers) and, to a lesser extent (eight papers), full-text search (see figures 2.2 and 2.3, and Table 2.2).

- Performing the same citation-based analysis, the hybrid search takes precedence (see Figure 2.4), and full-text searches represent a meager percentage of the total. This situation could indicate that hybrid search is a trend in retrieving information from learning objects without metadata search losing its preponderance.

Table 2.2: List of papers per journal

Journal	**# Art.**
Educational Technology & Society	5
Interdisciplinary Journal of E-Learning and Learning Objects	3
Expert Systems with Applications	2
IEEE Transactions on Learning Technologies	2
Journal of Intelligent and Fuzzy Systems	2
ACM Transactions on Speech and Language Processing	1
Applied Intelligence	1
Computations	1
Computers & Education	1
Cybernetics and Information Technologies	1
D-Lib Magazine	1
Education and Information Technologies	1
Future Generation Computer Systems	1
IEEE Access	1
IET Software	1
IFLA Journal	1
Interactive Learning Environments	1
Int. Journal of Knowledge-Based and Intelligent Eng. Systems	1
Int. Journal of Metadata, Semantics and Ontologies	1
Journal of Digital Information Management	1
Journal of Educational Computing Research	1
Journal of E-Learning and Knowledge Society	1
Journal of Emerging Technologies in Web Intelligence	1
Journal of Enterprise Information Management	1
Journal of Medical Internet Research	1
Journal of Universal Computer Science	1
Mathematical Problems in Engineering	1
Science of Computer Programming	1
Telematics and Informatics	1

- The fifth part of the articles reviewed (20%) refers to the use of automatic query expansion to increase the recall, preserving the precision [75].

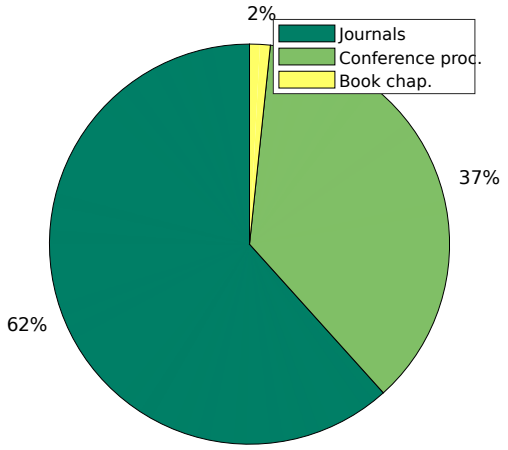


Figure 2.2: Types of papers in the review

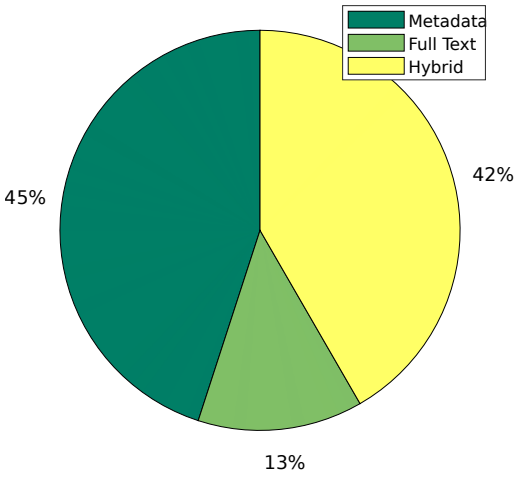


Figure 2.3: Papers by search category

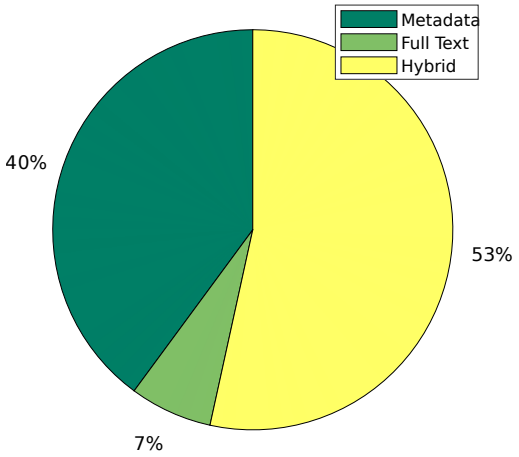


Figure 2.4: Citations by search category

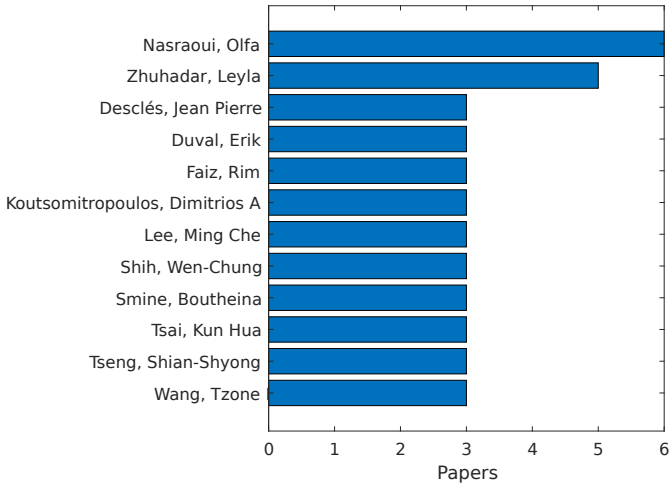


Figure 2.5: Authors with three or more papers in the review

- In 50% of the papers (60% in the classic search category and 56% in hybrid methods) suggest the use of ontologies to integrate semantic search for specific domains, since current learning object standards lack semantic metadata [60, 62]. In this sense, for more general fields, it begins to experiment with so-called weak ontologies such as Wikipedia [52].
- Other methods and techniques used in the works were: data mining, machine learning, fuzzy logic, natural language processing, and neural networks.
- Expansion of the research scope not only to learning objects but also to open educational resources.
- Concerning the technological infrastructure for the development of information retrieval systems, it was found that several papers use applications of the Apache Software Foundation such as Lucene, Jena, Tika, Poi, Nutch is reported. Furthermore, programming languages like Java, and Python to a lesser extent.
- The papers were written by 145 authors (Figure 2.5) from thirty countries on all continents (Figure 2.6) and seventy-three institutions (Figure 2.7).

2.5. Chapter conclusions

This chapter surveyed papers on searching for learning objects in repositories. To select the documents to be reviewed, a methodology that incorporated the use of Google Scholar was used as a tool to standard the number of citations per paper for different bibliographic databases. This web search engine also allowed to know the H index of authors most of the time. Also, the methodology used to select the papers of the review was adequate since it allowed a good approach to the topic.

Additionally, even though the search for LO through metadata still dominates, the hybrid search is incrementally attracting the interest of the academic community, as shown by the number of

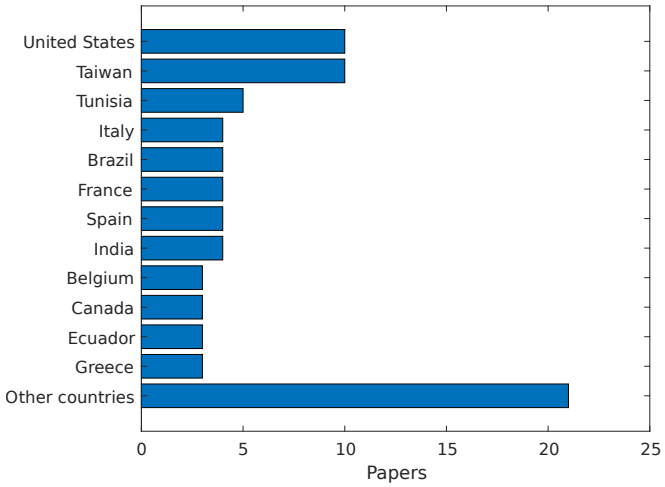


Figure 2.6: Top countries in the review

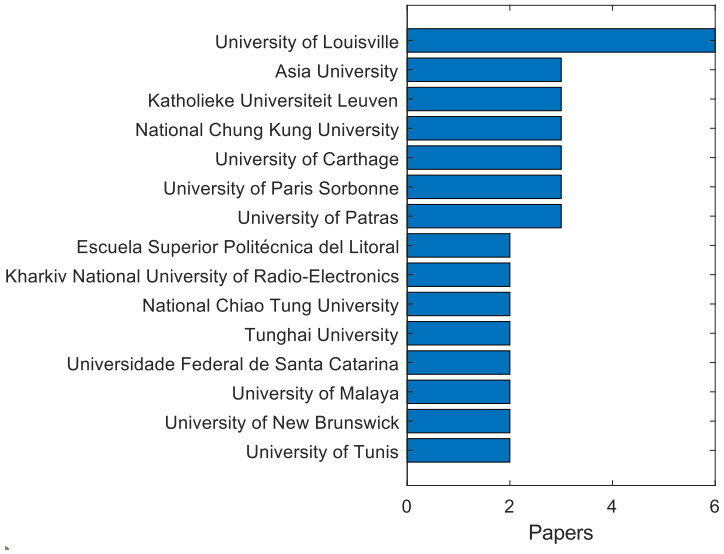


Figure 2.7: Institutions with two or more papers in the review

citations. Likewise, it was verified that for specific domains use ontologies as a way to improve search results. In addition to the above, several works propose the use of the user profile as a method to achieve more personalized results. Finally, also in several papers, the integration of the automatic expansion of queries with searches is proposed to improve recall.

Finally, this survey was also aimed at meeting Objective 1: Caracterize and define the conceptual elements that support the indexing of textual objects for metadata-based searches and those based on content, used in online service environments.

CHAPTER 3

Collaborative construction of metadata and full text dataset

3.1. Introduction

The collections of test data in IR are the result of early experimentation initiated in the 50s by Cyril Cleverdon, which culminated in the so-called Cranfield experiments that provided a basis for the evaluation of IR systems. At that time, Cleverdon obtained a grant from the National Science Foundation to compare different indexation systems. The project was known as Cranfield 1, required the manual indexing of 18000 aeronautical engineering articles in each indexing system and the evaluation of the results of 1200 search questions [15].

These experiments introduced a paradigm (Cranfield) for the evaluation of the system that has been the dominant experimental IR model for several decades and is the model used in evaluation efforts such as TREC[1], CLEF[2], and NTCIR[3], among others. This paradigm raises three main simplifying assumptions. The first assumption is that relevance can be approximated by thematic similarity. This assumption has several implications, for example: that all relevant documents are equally desirable, that the relevance of a document is independent of the relevance of any other document, and that the need for user information is static. The second assumption is that a single set of judgments of a subject is representative of the user population. The final assumption is that the lists of the relevant documents for each topic are complete (all relevant documents are known). The vast majority of test collection experiments since then have also assumed that relevance is a binary selection, although the original Cranfield experiments used a five-point relevance scale [165].

These collections must be composed of three essential elements: a set of documents, a set of queries, and a set of relevant judgments that indicate the relevant documents concerning each

[1]Text REtrieval Conference

[2]Cross-Language IR and Evaluation Forum

[3]NII-NACSIS Test Collection for IR Systems

query [32]. In this sense, the most recognized collections are Cranfield, TREC, INEX[4], Reuters, OHSUMED, NewsGroup, NTCIR, and CLEF [15, 165].

The TREC conference cycle began in 1992 to encourage research in information retrieval, offering large test collections and a forum for researchers to discuss their work on a common problem [164]. It carries out a series of experiments in the IR, in which multiple search engines run on the same topics and documents, but generate different classifications of the estimated level of relevance. It is organized by the National Institute of Standards and Technology (NIST) and sponsored by the Office of Information Technology of the Defense Advanced Research Projects Agency (DARPA), which has been developing for several years [68].

One of the problems in the current initiatives for the construction of TREC-style test collections is the cost related to the evaluation of relevance: the evaluation requires resources in terms of infrastructure, organization, time, money, and does not scale easily [11].

The IR community continues to use, in general, the evaluation model initiated by the pioneering work of Thorne, Cleverdon, and Gull in the 1950s and consolidated by collections of Cranfield's Cleverdon of the 1960s [131].

Respecting the essential bases of the Cranfield paradigm, variations in the way the collections are constructed have been presented. Some of these works are described below.

In the work of [148], the Open Mind Initiative (OMI) is proposed as a framework for large-scale collaborative efforts in the construction of the basic components of "smart" systems that address common sense reasoning, the understanding of documents and language, the recognition of characters and speech, among others. This initiative allows specialists in a domain to contribute algorithms; to tool developers, provide software infrastructure and tools; and to non-specialized "e-citizens", provide data and test information to large databases. It is based on the principles of the Free Software and Open Source movements [44].

The same author, in a subsequent work and based on OMI, proposes the collection of data open on the Web, in which any user of the web can contribute "informal" data to large databases, it presents several challenges that require new approaches in human interface design, algorithmic machine learning and collaborative infrastructure. It proposes a collaborative architecture for the construction of these data collections. The key components of this architecture have been implemented and tested as part of the Open Mind Initiative. It suggests that collaborative efforts can be extended to a large number of collaborators (potentially anyone on the web), whose technical experience may be low (more than the ability to point and click). He mentions that traditional Open Source projects release software, OMI releases both software and data [147].

Also inspired by OMI, [161] presents a new interactive system in the form of a game (ESP Game) with a unique feature: people who play, tag the images for the information collector. The tags generated by the game can be useful for a variety of applications. They estimate that 5000 people playing the game for twenty-four hours a day would allow you to tag all images indexed by Google in a matter of weeks. This is surprising because 5000 is not a large number: the most popular games on the web have more than 5000 players at any given time. They point out that their main contribution comes from how the labeling problem is attacked. Instead of developing a

[4]INitiative for the Evaluation of XML Retrieval

complicated algorithm, they have shown that it is conceivable that a large-scale problem can be solved with a method that uses people who play on the Web. In summary, the authors propose turning a tedious work into something that people want to do and that perhaps other problems can be attacked similarly.

Along the same lines of the previous work, [162] proposes another computer game (Verbosity) with which data is collected for a "common sense" database. They point out that several efforts have been devoted to collecting common-sense knowledge to make computer programs more intelligent, without being able to accumulate enough data, since the process of manually introducing these facts is tedious, raising their transformation to a friendly game. Verbosity is an example of an emerging class of games similar to ESP Game that can be considered "human algorithms": human beings act as processing nodes for problems that computers cannot solve.

Von Ahn in [163] presents an evolution of the use of CAPTCHA[5], exploring whether the human effort to decipher characters can be channeled towards a useful purpose: to help digitize old printed material by asking users to decipher the scanned words of the books that the optical recognition of computerized characters did not recognize. As a result, it was shown that this method could transcribe the text with an accuracy that exceeds 99%, matching what professional human transcriptionists guarantee. This device has been implemented in more than 40000 websites and has transcribed more than 440 million words. They suggest that the results presented are part of a proof of concept of a more general idea: wasted human processing power can be harnessed to solve problems that computers cannot solve. Some have referred to this idea like "human computing."

In the work of [145] it is mentioned that human linguistic annotation is crucial for many natural language processing tasks, but it is expensive and time-consuming. For this reason, the use of the Amazon Mechanical Turk (AMT) system was explored, a significantly cheaper and faster method to collect annotations from a broad base of non-expert collaborators paid over the Internet. Five linguistic annotation tasks were investigated. It was found that in the five tasks, a high concordance of the annotations of the non-experts of AMT and standard labels of existing gold collections, which were provided by expert labelers, was achieved. They were able to conclude that many of the large labeling tasks can be designed and conducted with this method effectively, at a cost that is a fraction of the usual. In a detailed study of agreement between experts and non-experts for a task, it was found that an average of four labels of non-experts per item is required in order to emulate the quality of the label at the expert level.

In [66], a method is presented for the collection of relevance evaluations through a collective effort of human evaluators who, as a group, are involved in a social game. The game offers incentives for evaluators to follow a predefined review procedure and establishes provisions for quality control of collected relevance judgments. To validate the approach, a pilot study was carried out in a book corpus. The study demonstrates that it is feasible to achieve a productive environment in which individuals and teams are encouraged to participate in the thorough evaluation of the relevant documents.

[5]CAPTCHA is a generalized security measure on the Web that prevents automated programs from abusing online services, consulting human beings to carry out a task that computers cannot perform, such as deciphering distorted characters.

As mentioned in [9], crowdsourcing[6] is perfect for relevance assessments since they are small tasks, so they do not need to be divided and have been used successfully. This paper presents the results of a series of experiments using the Spanish language part of CLEF, using crowdsourcing platforms and showing that they work for languages other than English, at least in the case of Spanish.

Along the same lines of the previous work, [10] explores the design and execution of relevance judgments using AMT as a crowdsourcing platform, introducing a methodology for the evaluation of relevance through this work scheme and the results of a series of experiments using TREC 8 with a specific budget. They indicate that one of the most critical aspects of conducting evaluations through crowdsourcing is to design the experiment carefully and proposes a four-stage methodology: data preparation, interface design, worker filtering, and task scheduling. As a finding, it was found that workers (AMT) are as good as the TREC experts. The results showed that most of the experiment design must be in the user interface as in the instructions and reinforce the conception regarding the advantages that crowdsourcing has, in the case of relevance assessments, which are generally not difficult but they are tedious and of considerable volume.

Similarly, [11] reports on the first attempts to combine crowdsourcing and TREC, seeking to validate the use of the latter in the evaluation of relevance. It was found that the agreement between each worker (AMT) and the original TREC evaluator is not very high when measured individually, but increases when workers are grouped. In some cases, they were as accurate as of the original evaluating experts, if not more. Besides, the tasks are completed in days and, in some cases, in hours. It also seems useful for researchers who wish to build (in "home") their own evaluation collections. In conclusion, supported by the experimental results, it is stated that crowdsourcing is a cheap, fast, and reliable alternative for the evaluation of relevance.

Another aspect to consider has to do with the needs in specific domains. In the case of the authors, test collections of textual digital objects containing the associated metadata and the full-text were required. In the exploration of the literature carried out, no such collections were found. Hence, the need to build a collection of test data with these characteristics. This paper proposes a methodology for its construction.

This chapter is an extended and updated version of [107].

Lastly, this chapter is organized as follows. First, in Section 3.3, the application of the methodology to construct a collection of test data is described. Subsequently, in Section 3.4, the results are discussed. Finally, Section 3.5 presents the conclusions and proposals for future work.

3.2. Methodology

As seen in Sections 3.1, the Cranfield paradigm is still in force for the construction of test data collections in IR. According to the review of the literature consulted, no other paradigm has been found. Only variations in the mechanisms of human intervention that determine the relevance or

[6] In crowdsourcing, large works are fragmented into many small tasks that are then outsourced directly to individual workers, through a public call through the Internet.

not of a recovered data have been proposed. In this sense and mainly influenced by the Open Mind Initiative, crowdsourcing and works of Luis von Ahn, this article proposes a methodology for the collaborative construction of test data collections that contain metadata and the full text of objects digital that compose it.

In this methodology, unlike crowdsourcing, there was no hiring of workers. The influence of the latter is basically in the sense that non-experts in IR participate. They are on specific topics of a discipline.

It should be noted that the collection of test data to be constructed will be used in experiments with search systems for learning objects (LO) in repositories. Given the difficulty of achieving the conjunction of experts on specific topics and the objects associated with those topics in a repository, and that also have the metadata and the full text, the use of isomorphic systems such as the computer tools of bibliographic management. These tools currently handle the metadata and the full text of the documents and allow searches in them. They are already in everyday use among researchers.

Learning objects in repositories are considered to be isomorphic systems with academic documents, since several of the metadata associated with the General category in the LOM standard [61] that mainly describe the LO, correspond to those that describe an article or scientific document. In this sense, the title and the keywords match one by one, and the description can be assimilated to the summary of the scientific paper.

The stages of the proposed methodology are shown in Figure 3.1. It is important to note that the stages do not develop strictly sequentially. It may be the case that several times, it is necessary to return to previous steps. Next, the description of each stage.

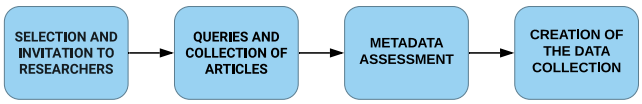


Figure 3.1: Proposed methodology for the creation of the test data collection.

Selection and invitation to researchers: For the construction of the collection, the participation of researchers at the master's or doctoral level is required, who has advanced a research topic and have the scientific documents that support it in digital format. The project is explained and invited to participate in it.

Queries and collection of articles: Each researcher who has accepted the invitation must provide all the scientific documents of their research. The papers must then be imported to the chosen bibliographic manager. Subsequently, you are asked to construct two, three, or four-word queries associated with your research topic, which should be entered into the full text query option of the bibliographic manager. For the results of each specific query, the researcher must determine which documents are relevant and which are not. These documents, already classified for each query, are extracted and compiled in separate folders.

When the information of all researchers is collected, the corresponding files are collected in a root folder that will contain subfolders, named with the words of each query, their associated

documents. Non-relevant documents of all subjects and researchers are grouped into a single subfolder. If for a given query, there are less than five relevant documents, the query, and its results are excluded [164].

Metadata assessment: Bibliographic managers allow you to enter the metadata associated with each document. Many of them have two mechanisms to do it: one, automatically, with algorithms that try to extract the text of the metadata from the same document; the second, with the manual registration, conducted by the researchers, although not all are careful and thorough in their registration. In both cases, 100% accuracy is not achieved in the metadata record for all documents. Therefore, it is necessary to manually review each and every one of the metadata associated with the compiled articles.

Creation of the data collection: When all metadata have been reviewed, an algorithm is created that creates a relational database that contains the metadata, the link to the file that contains the full plain text of the document, the queries, and their associated relevant objects.

This methodology can be used for the creation of test collections of digital educational resources, of which the metadata corresponding to the title, description (or abstract), and keywords are available, and the document with the textual content. This is the case of public or institutional repositories that use learning object standards. Furthermore, for personal repositories with documents referenced through BiBTex and other bibliographic formats.

Based on the proposed methodology, the following section describes its application to create a test collection for experimental purposes.

3.3. Application of the proposed methodology

The bibliographic manager used for the development of the experiment was Mendeley[7]. The considerations that took into account were: despite being proprietary software, it has a free version with almost the same features as the paid version; it is available on the most common operating system platforms; it is easy to install; it presents a friendly interface and has an excellent full text search tool within its repository.

3.3.1. Selection and invitation to researchers

Fourteen researchers were contacted at the master's or doctoral level, in process or already completed, and who had an already advanced research topic. The project was presented, and they agreed to participate. They provided the digital archives with the academic papers that supported their research.

The general research topics were: big data, electrical engineering, functional food, marketing, metabolomics, operation management, and physics with queries in English; e-learning with queries in English, Portuguese and Spanish; logistics and administration theory with queries in English and Spanish.

[7]http://www.mendeley.com

3.3.2. Queries and collection of articles

The first thing that was done at this stage was to configure the Mendeley application to automatically generate the BibTeX file for each collection that contains the main metadata of the document. In addition to saving all documents in a single folder and rename all files with the following format: the last name of the first author, year and the title of the article.

The researcher was then asked to ask questions about his research topic with two, three, and four words. They were written in the Mendeley query panel. For each query, a collection was created whose name corresponded to the keywords of the query. Additionally, a collection called *not relevant* was created. Based on the results obtained, the relevant documents were taken for the query given to the corresponding collection. The same was done with the papers that the researcher considered were not very relevant for the given query and his research topic. All this procedure was conducted for all queries.

The language of the query was taken into account. That is, only those results in the language itself were considered relevant. In many cases, the abstract and keywords were in two languages and for inquiries, for example, in English, results were obtained from articles in Spanish for those above. These were labeled as not relevant.

In the end, all the PDF files were copied to the folder where Mendeley stores all the documents and also the BibTeX files associated with the queries made. With an algorithm implemented in Python[8], the BibTeX files were taken with the results of the queries, and the files associated with them were extracted.

At this stage, the *big data* issue was discarded since the researcher had few documents and the results of the consultations only showed at most four relevant results. Under the same criteria, fourteen other queries in *e-learning*, *management theory*, *metabolomics*, and *physics* were not taken into account. Of the discarded queries, there were all of the Portuguese language (Table 3.1).

3.3.3. Metadata assessment

In a Mendeley account, all documents that were compiled in the previous step were imported. This application automatically creates metadata that describes each stored document. Although the accuracy with which this task is performed is good, it is not 100%. Intervention is required to adjust and ensure that the metadata correctly describes the associated documents.

Based on the names of the BibTeX files, collections with the names of the knowledge areas were created. Within each of them, subcollections were added with the names of the two, three, and four-word queries as appropriate and also the subcollection of non-relevant documents.

In each subcollection, the BibTeX file associated with each query was added with their respective metadata of the relevant documents and also the non-relevant ones. It is important to remember that the Mendeley application, when you upload a BibTeX file, creates an entry for each document described in the file, searches its repository and if it has the corresponding PDF file, it associates

[8]https://www.python.org

Table 3.1: Queries generated by the researchers.

Subject	**Language**	**# docs**	**# queries**	**# words**		
				2	**3**	**4**
E-Learning	en es	433	15	3	7	5
Electrical engineering	en	305	6	3	2	1
Functional food	en	19	3	1	2	0
Logistic	en	45	1	1	0	0
Management theory	en es	78	3	2	1	0
Marketing	en	110	2	1	1	0
Metabolomics	en	119	5	2	2	1
Operation management	en	543	1	1	0	0
Physics	en	24	2	1	0	1
		1676	***38***	***15***	***15***	***8***

it with the uploaded metadata. Again, with this automatic work, 100% results are not achieved, so it is necessary to manually review and search the PDF file of each entry and associate it with the corresponding metadata.

After performing this task, entries without the corresponding PDF file were deleted. After consolidating all the documents, the metadata of each entry was manually contrasted with the content of the associated PDF document.

3.3.4. Creation of the data collection

Starting from the documents and metadata registered in Mendeley, again all the documents were extracted from its internal repository and the associated metadata that were registered in the BibTeX file.

Each of the BibTeX records from Mendeley contains a file tag with the absolute path of the PDF file address on disk, referenced by the record. This path is absolute concerning the system where Mendeley's data comes from, so, from it, the name of the file was obtained and searched the file system where the data collection was generated to identify the document. Also, since the name of the file is in a BibTeX file encoded in LaTeX, the tildes and special characters are written in LaTeX as well, the accents were removed before searching for the file in the system.

As a final result, the collection of test data constructed was made up of 1676 full text documents: in Spanish (9%) and English (91%); their associated metadata; thirty-eight queries of two (39.47%), three (39.47%) and four words (21.06%); and with their respective relevant judgments.

3.4. Discussion and results

From this process, a collection of test data could be constructed with 1676 digital objects, each with full text and associated metadata. Most digital objects are in English; the others in Spanish. Thirty-eight queries were defined: fifteen of two words, fifteen of three words and eight of four words. For each one, the relevant documents were labeled, according to the experts' criteria.

In an area of investigation of information retrieval algorithms, to the extent that more data is available in the test collections, better results can be achieved. In this sense, this collection can be considered adequate for research purposes.

Another aspect to keep in mind is that the collection can be increased with new objects and queries. In such a case, it should be verified that old queries should be evaluated with the new documents, labeling those that are relevant if any. The same procedure should be done with further queries on the initial documents.

On the other hand, in the topics of e-learning and management theory, it was possible to obtain documents in two languages, which enables the collection to conduct experimentation in the retrieval of multilingual information.

Finally, it should be noted that the algorithm developed in this article to create the dataset only extracts markup information from BibTeX files. Nevertheless, the methods presented here are also applicable to other contexts where metadata standards are used. For this, the algorithm used must be adjusted.

3.5. Chapter conclusions

It was possible to verify that the proposed methodology is suitable for the construction of test collections such as the one proposed. It can be easily applicable in other contexts, autonomously, and tools that are available to most researchers. Similarly, compared to other methodologies, the cost of development is low because volunteers are used who are themselves, experts in the areas and provide the required information. Therefore, it would facilitate research groups to build their own test collections. Also, it would open spaces for collaboration between the different groups who can exchange and integrate their datasets.

As future work, the implementation of an open access web API is considered, that automates many of the tasks presented here and that facilitates the integration of several collections.

Lastly, it should be noted that the construction of the dataset is intended to test the model proposed in this thesis and thus, comply with Objective 4: design a prototype of the proposed hybrid system to validate the proposal.

CHAPTER 4

Experimental evaluation of two models of information retrieval of learning objects with metadata and full-text

4.1. Introduction

This chapter proposes to perform a comparative evaluation of the efficiency of classic information retrieval systems in repositories through metadata (MD) and full-text (FT) searches. Specifically, the classic Vector Space Model (VSM) and Latent Semantic Analysis (LSA) are used with Term frequency – Inverse document frequency weighting (TF-IDF). In this regard, LSA was chosen for two reasons. First, it is based on the same term-document matrix as the classic VSM model, allowing the comparison of its effectiveness in the same terms. Second, LSA reduces or eliminates the effects of polysemy and synonymy [13, 48, 82, 157], problems mentioned above.

In order to carry out this experimentation, the data set proposed in the previous chapter was used. In this sense, an a previous work by the author [106] the learning objects contained in "Federación de Objetos de Aprendizaje Colombia" (FROAC) [150, 152] were used for experimentation. This time it was decided to build a new data set for two reasons. First, many of the learning objects in this federation have very fine granularity. Often, the metadata associated with the description was more widespread than the content of the learning object itself. The second was to increase the number of domains in the experimentation. On the other hand, within the framework of the projects of the GAIA research group, this thesis will be able to improve the searches within the FROAC.

The remainder of this chapter is organized as follows. In Section 4.2, the conceptual elements that support the work carried out are presented. Next, in Section 4.3, the related research works are

discussed. In Section 4.4, the materials and methods are given. In Section 4.5, the results and analysis are presented. Finally, in Section 4.6, the conclusions of this work are shown.

On the other hand, this chapter is an extended and updated version of [106].

4.2. Background

This section presents conceptual aspects of the information retrieval models VSM and LSA, TF-IDF weighting, semantic search, and automatic query modification.

4.2.1. Classic vector model

Creating a vector space model, the term-document matrix must first be constructed. Rows are made up of terms, which are individual components that make up a document. These are usually single words but can also be phrases or concepts depending on the application. The columns of the input matrix are made up of documents, which are of predetermined text size, such as paragraphs, collection of paragraphs, sentences, book chapters, books, and so on, again depending on the application. A collection of documents composed of n documents and m terms can be represented as an $\boldsymbol{A}$ $m \times n$ term-document matrix. Very often $m >> n$; however, there are cases where it is the reverse and $n >> m$, for example, when the collection of documents is from the Internet [90].

Each non-zero element $A_{i,j}$ of $\boldsymbol{A}$, corresponds to the frequency of the i-th term in the j-th document. Typically, matrix $\boldsymbol{A}$ is considered sparse because it contains many more zero-valued entries than non-zero values. Each document in a collection tends only use a subset of terms from the set [90].

4.2.1.1. Weighting scheme of the input matrix

To achieve better results in the information retrieval process, the term-document matrix in the vector model can be weighted, taking into account two aspects [32]:

- *The frequency of the terms in the document.* It is assumed that the more a term is repeated in a document, the more important it is.
- *The specificity of the term in the collection.* Not all terms are equally common, so it makes sense to give more weight to the rarer ones since they have great discriminating power.

In this sense, the TF-IDF model is one of the best-known information retrieval schemes that consider these aspects. This model comprises the term frequency (TF) and the inverse document frequency (IDF) [32].

The formulation of TF is defined in the same terms raised in the Section 4.2.1:

$$tf_{i,j} = f_{i,j} \tag{4.1}$$

where $f_{i,j}$ is the frequency of term i-th in document j-th.

For this thesis, the TF log normalization variant is used to smooth its magnitude and avoid strong oscillations that produce biases [15, 88]:

$$tf_{i,j} = 1 + logf_{i,j} \tag{4.2}$$

In turn, IDF is defined as:

$$tf_i = log\frac{N}{n_i} \tag{4.3}$$

where N corresponds to the total number of documents in the collection and n_i is the number of documents in which the term i-th appears.

Thus, the TF-IDF weighting scheme of the input matrix is defined as [15, 88]:

$$w_{i,j} = \begin{cases} 1 + logf_{i,j} \times log\frac{N}{n_i} & \text{if } f_{i,j}>0 \\ 0 & \text{otherwise} \end{cases} \tag{4.4}$$

4.2.2. Latent Semantic Analysis

LSA is a theory and a method to extract and represent the contextual meaning of words by statistical calculations applied to a wide corpus of texts [78].

It is based on the concept of VSM, an approach that uses linear algebra for effective automatic information retrieval. It was developed to handle the retrieval of text from large databases where the text is heterogeneous, and the vocabulary varies. The underlying formal mathematical model of the VSM defines unique vectors for each term (word or concept) and document, and the queries are carried out comparing the representation of the query to the representation of each document in the vector space. The query-document similarities are based on similar concepts or semantic content [90].

The premise of the method is that information about contexts in which a particular word appears or does not appear provides a set of constraints that determines the similarity of the meanings of the words to each other. LSA has shown that it addresses the problems of polysemy and synonymy quite well, which is important in relation to the problem of feature localization since users (in this case, developers of software systems) often construct queries without knowing precisely the vocabulary of the target system. It is very suitable to deal with this situation since it does not use predefined grammar or vocabulary. The meanings of words are derived from their use, rather than from a dictionary or thesaurus, which is an advantage over a traditional natural language approach, where a subset of grammar and an English dictionary must be developed [115]. When speaking about feature or concept localization, we refer to identifying an initial location in the source code that implements a given functionality in a software system [45].

Based on empirical evidence, LSA is known to produce measures of word-word relationships, word-passages of text, and passages of text-passages of text, which correlate well with various human cognitive phenomena involving association or semantic similarity. These correlations show a strong similarity between LSA results and the representation of meaning that people reflect from what they have read and heard. As a practical consequence of this correspondence, LSA allows to carry out similarity judgments of meanings between words, very close to those made by humans, and objectively predict the global similarity based on words between text passages [78].

The objective of LSA is to identify the semantic dimensions hidden in text data and then use the mapping of the original words of these semantic dimensions to obtain a better measure of similarity between documents and queries [94]. LSA is an evolution of a bibliographic information retrieval technique called Latent Semantic Indexing (LSI) [42]. It is described mathematically below.

4.2.2.1. Decomposition of the input matrix into orthogonal components

Once the input matrix $\boldsymbol{A}$ is created, it is transformed into a vector space of term and document by orthogonal decompositions to take advantage of vector truncation. Transforming a matrix by using orthogonal decomposition (or as a product of orthogonal matrices) preserves specific properties of the matrix, including the norms, or vector lengths or distances, of the row and column vectors that make up the $\boldsymbol{A}\, m \times n$ term-document matrix [90].

There are several methods to decompose $\boldsymbol{A}$ into orthogonal components. The most widely used method for LSA is singular value decomposition (SVD) for several reasons. First, the SVD decomposes $\boldsymbol{A}$ into orthogonal components that represent both the terms and the documents. Second, SVD sufficiently captures the underlying semantic structure of a collection and allows adjusting the representation of terms and documents in a vector space by choosing the number of dimensions. Finally, SVD is manageable for large data sets [90].

The SVD of an $\boldsymbol{A}\, m \times n$ matrix is defined as follows:

$$\mathcal{A} = U\Sigma V^T \tag{4.5}$$

where $\boldsymbol{U}$ is an orthogonal matrix, $\boldsymbol{V}$ is another orthogonal matrix, and Σ is a diagonal matrix with all other positions being zeros. The first r columns of the orthogonal matrix U contain r orthonormal eigenvectors associated with the r non-zero eigenvalues of $\boldsymbol{A}\boldsymbol{A}^T$. The first r columns of the orthogonal matrix $\boldsymbol{V}$ contain r orthonormal eigenvectors associated with the r nonzero eigenvalues of $\boldsymbol{A}^T\boldsymbol{A}$. The first r entries on the diagonal of Σ are the nonnegative square roots of the non-zero eigenvalues of $\boldsymbol{A}\boldsymbol{A}^T$ and $\boldsymbol{A}^T\boldsymbol{A}$.

The rows of the matrix $\boldsymbol{U}$ are the vectors of the terms and are called left singular vectors. The rows of V are the document vectors and called right singular vectors [90].

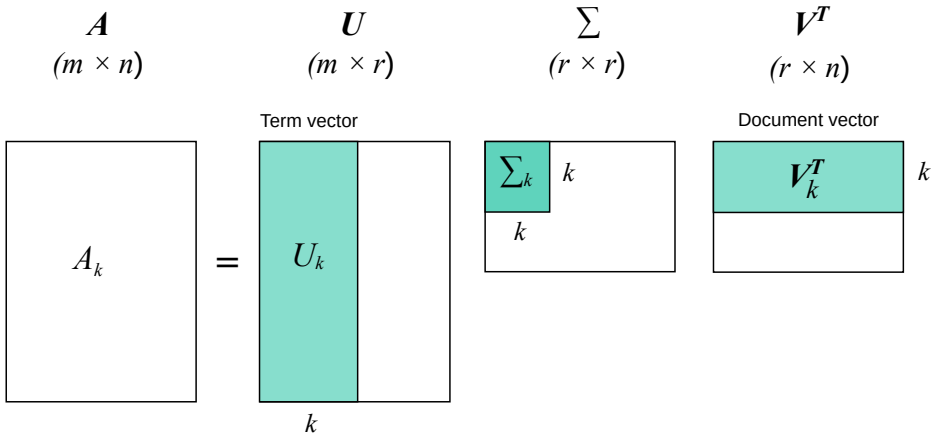


Figure 4.1: Truncated Singular Value Decomposition diagram.

4.2.2.2. Truncated orthogonal values

Given that $\boldsymbol{A}$ can be written as the sum of matrices of rank 1: $\sum u_i \sigma_i v_i^T$, r can be reduced to k to create $\boldsymbol{A_k} = \sum u_i \sigma_i v_i^T$. The matrix $\boldsymbol{A_k}$ is the best or closest approximation to rank k (the distance is minimized) of the original matrix $\boldsymbol{A}$. The matrix $\boldsymbol{A_k}$ ($\boldsymbol{A_k} = U_k \Sigma V_k^T$) is created by ignoring or zeroing all the elements, except the first k elements or columns of the vector of terms in U, the first k eigenvalues in Σ, and the first k elements or columns of the document vector in $\boldsymbol{V}$. To reduce the dimension from r to k, atypical information and variability in the use of terms, referred to as "noise", which is associated with the collection of documents, are removed. By truncating the SVD and creating $\boldsymbol{A_k}$ is how the essential underlying semantic structure of terms and documents is captured. Terms similar in meaning are "close" to each other in the k-dimensional vector space, even if they never co-occur in a document, and documents similar in conceptual meaning are close to each other, even if they do not share common terms.This k-dimensional vector space is the foundation that the LSA exploits [90].

In Figure 4.1, it can see the diagram of this transformation.

The best selection of k remains an open question. In practice, the choice of k depends on empirical trials, which have shown that, in large databases, the optimal choice for the number of dimensions is in the range from 100 to 300 [90].

4.2.2.3. Semantic search

When the user enters the search terms in the search engine, it can be considered that he has given a vector containing zeros and frequencies of terms corresponding to those terms specified in the query, defined by:

$$\mathbf{q}^T = (q_1, q_2, \ldots, q_m) \in \mathbb{R}^{1 \times m} \tag{4.6}$$

where

$$q_i = \begin{cases} 1 & \text{if the term } T_i \text{ appears,} \\ 0 & \text{otherwise} \end{cases}$$

Once the pseudo-document (query vector) is formed, it is projected in the term-document space, and a similarity measure is used to determine which terms and documents are closest to the query. The cosine similarity measure is commonly used: the cosine of the angle between the vector of the query or pseudo-document and it is computed for each of the documents or terms, as follows [57]:

$$cos\varphi_j = \frac{\mathbf{q}^{\mathbf{T}}\mathbf{d}_j}{\|\mathbf{q}\| \left\|\mathbf{d}_j\right\|} \tag{4.7}$$

where $\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_j, \ldots, \mathbf{d}_n$ correspond to the columns of the matrix $A_\mathbf{k}$.

The cosines associated with each document are arranged in descending order; thus, the document with the highest cosine in the query will be the first. Only documents that are above a certain threshold are listed [90].

4.2.3. Modification of queries

One of the most critical information retrieval (IR) problems is to formulate the query that adequately reflects the need for information, which presents two difficulties. The first, the inherent difficulty of users to specify their information needs. The second, the difficulty in adapting the query to the specific domain. Due to the above, it is frequently required to reformulate and refine the query to obtain results appropriate to the information need. In this sense, techniques have been developed to allow the user to reformulate the query manually or automatically. These techniques can be classified into two main approaches [32]:

- *Query feedback using user relevance criteria.* As its name implies, the user interacts with the system to refine the query.
- *Query expansion.* For the most part, these techniques consist of incorporating new terms into the original query. The aim is to improve precision and recall results, although by their very nature further enhances recall.

Concerning query expansion, manual dictionaries and thesauri can be used as well as automatic thesauri. A thesaurus is a classification system that contains terms and phrases organized into classes based on the relationships between them. Additionally, a *thesaurus of terms* is a matrix that measures relationships between the terms of a collection of documents. The construction of the matrix can be done by applying different techniques, although most of them provide a symmetric matrix. Some of the techniques for the automatic creation of a thesaurus of terms are the following [32]:

- Thesauri built considering co-occurrence values.

- Thesaurus of similarity.
- Thesauri built from the association of terms and phrases.
- Thesauri of infrequent terms.

By far, the most widely used approach is the first one, that of co-occurrence. The underlying idea is that if two terms frequently appear in the same contexts, they are somehow related. In this sense, a "fast" way to obtain the co-occurrence matrix is by multiplying the term-document matrix (described in subsection 4.2.1) by its transpose [32]. It is the one used in this thesis because of what is stated in the literature.

On the other hand, to expand the original query with the new terms extracted from the thesaurus, the first terms that are best related to all the terms of the original query are searched. Empirically, it has been determined that between twenty-five and fifty terms in the expanded query usually provide good results [32]. These terms are calculated analogously to how it is described in Subsection 4.2.2.3.

4.3. Related work

The LSA model is used in a wide range of applications, as shown in the following.

In the work of [136], a hybrid bibliometric method is proposed that combines citation network analysis and text mining on a field of knowledge. With the first, the main streams were identified; with the second, using LSA, the subgroups of topics in each of the streams were identified. In this same sense, in [71], a new modeling topic method called Word2vec-based LSA (W2V-LSA), oriented to the bibliometric research of trends applied to the blockchain domain, is proposed. It uses LSA, Word2vec, and Spherical k-means clustering.

In [51], LSA is used to score sets of divergent thinking responses, reducing the bias noted in previous studies by eliminating stopwords and corrections conducted through simulation.

In [33], a new multilingual summarization algorithm is proposed based on combining frequent itemset mining and LSA methods, which allows to overcome the individual drawbacks and take the best of each of them.

In the work of [133], a method is proposed to determine the similarity between altered states of consciousness, caused by psychoactive substances, and the states that occur in high/low lucidity dreams, based on the semantic similarity, calculated with LSA, among a large number of subjective reports about these substances.

In [93], an automatic system, based on LSA, is proposed for summarizing legal text that generates short summaries keeping the important ideas of the original document to support the work of the lawyers.

In [170], LSA was used with the cosine metric to quantify the state level of creativity in analogical reasoning in a group of participants in an experiment in the domain of psychology.

One of them is content-based image retrieval (CBIR). In this sense, the work of [146] shows that LSA effectively combines visual and textual information (annotations to medical images). Compared to other state-of-the-art CBIR techniques, on the same dataset, its performance is also superior.

Another application is in software engineering, where there is a very active community using this technique. To reconstruct the traceability links between the software artifacts, the work of [35] proposes improvements in indexing, leaving only nouns in the text of said artifacts. On the other hand, in a large empirical study, the authors of [41] analyze information retrieval techniques for tagging source code, including LSA. Other works in this line are those of [8, 26, 167, 174].

Furthermore, this technique is used in [100] to automatically extract ontologies in documents and collections of them. A similar work is that of [14], where LSA is used to automatically extract metadata of learning objects from the Web to identify them for educational uses. Similarly, in [28], the authors use LSA along with other Natural Language Processing techniques for automatic semantic annotation.

In the work of [13], three techniques are analyzed in the tasks of information retrieval on the medical bibliography database and other health databases: VSM, its variant, LSA, and Formal Concept Analysis (FCA).

4.4. Materials and methods

This section describes an experimental process for evaluating and comparing the effectiveness of classical vector and LSA models of IR on metadata and full-text content in a dataset. For this, IR precision and coverage metrics were used, and some of their derivatives, such as the precision-coverage curve and metrics of mean values. Both methods were evaluated with twenty-eight queries on the metadata and full-text of the dataset documents. Additionally, the same models were evaluated with the modified queries through automatic query expansion (AQE).

4.4.1. Dataset

The dataset used for these experiments was the one proposed and implemented in Chapter 4 of this thesis, which has been published as [107]. After the review and debugging process, the dataset was made up of 1435 documents: 1387 in English and forty-eight in Spanish. Subsequently, the queries and documents in Spanish were discarded because it was considered that the number of documents was not sufficient for the tests. Therefore, only the documents and queries in English were used for the experiments. In this sense, the dataset contains has twenty-eight queries: ten of two words, thirteen of three words, and five of four words as shown in Table 4.1.

Table 4.1: Two, three, and four word queries in English that were considered for the experiments.

Subject	**Query**
eLearning	information retrieval
	information retrieval learning object
	learning object
	learning object metadata
	learning object quality
	learning object recommendation
	learning object recommendation system
	learning object repository
	learning object search
	learning object search engine
Electric Atmospheric Discharges	return stroke
	return stroke model
Electrical engineering	high voltage mosfet configuration
	high voltage switch
	mosfet stack
	power electronic switch
Functional foods	microencapsulation probiotic
	microencapsulation probiotic spray
	probiotic viability encapsulation
Logistic	humanitarian logistic
Marketing	entrepreneurial marketing
	entrepreneurial marketing network
Metabolomics	cell culture
	cell culture metabolomics
	metabolomics pesticide
	metabolomics pesticide organochlorine
	metabolomics pesticide organochlorine endosulfan
Operations management	manufacturing strategy

In this regard, each document in the dataset has its respective metadata, which is stored in the *paper* table of the dataset database, and its textual content in a separate plain text file, which was extracted via Apache Tika[1] version 1.24.1. The database was implemented in PostgreSQL[2] and consists of three tables as shown in Figure 4.2. Apart from the *paper* table mentioned above, the *query* table contains the queries about the dataset; and the *querypaper* table contains all relevant documents from the dataset for a given query.

[1]https://tika.apache.org/
[2]https://www.postgresql.org/

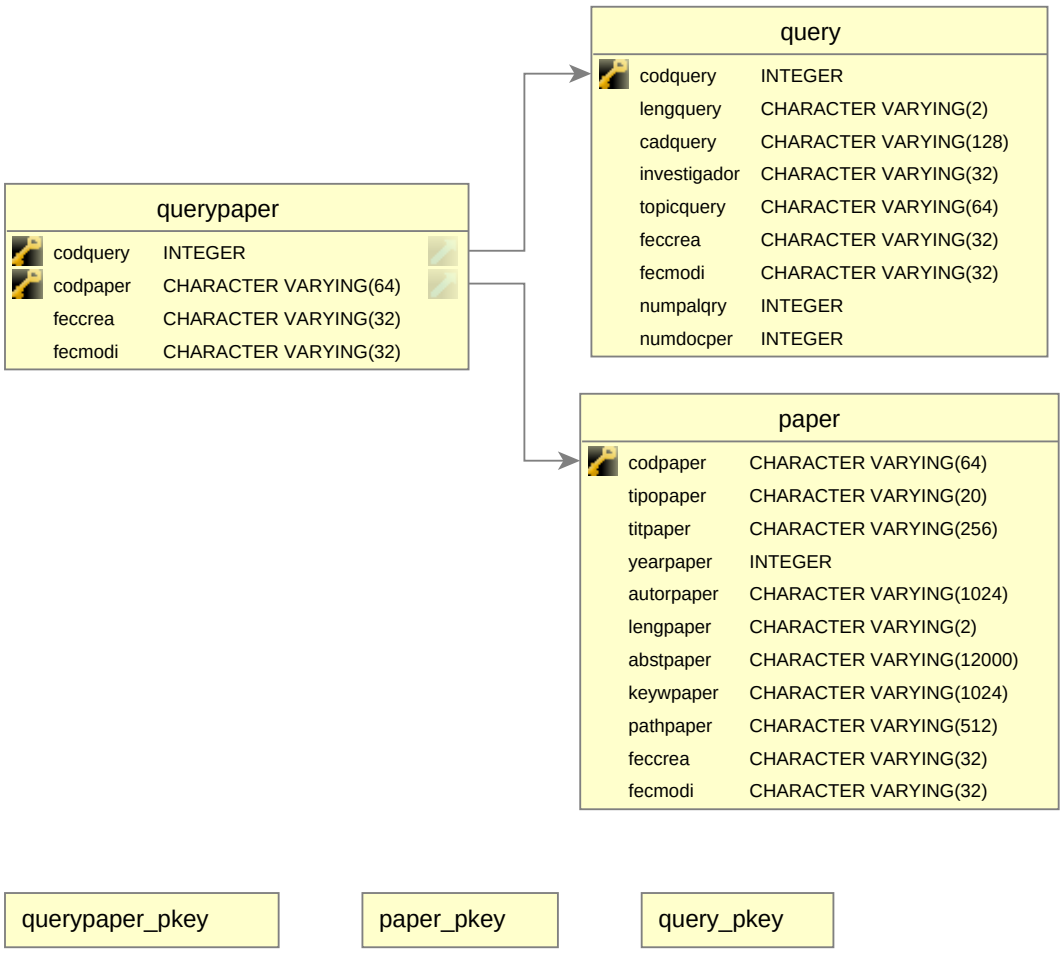


Figure 4.2: Entity–relationship model of the dataset

4.4.2. Preprocessing

The first step of this stage consisted of reviewing the textual content of the metadata and the text of each document's flat file. LaTeX codes of special characters were found in the metadata text, as they were extracted from BibTeX files, automatically generated by Mendeley[3]. Therefore, it was necessary to replace them with the corresponding character. On the other hand, it was detected that the conversion of dataset documents to plain text files using Apache Tika, most of them in pdf format, was not 100% accurate: some terms were found made up of two or more words together; words divided into two-character syllables; words split by a hyphen, generated by moving to another line of the original text; numerical values; terms of a single character, among others. In this sense, based on a tokenization process (through an Apache Lucene[4] package), two corpora were created: the first, with the text of the metadata associated with the title, keywords, and abstract together; the second, with the terms of the plain text files. In both, they only included alphabetic terms of three or more characters.

Subsequently, the stopwords were removed from the two corpora, for which the NLTK[5] list was used. Then each term was reduced to its common base form by a stemming process. The tool used was the Apache Lucene package. Later, based on the corpora, two preliminary dictionaries were constructed. With them, the term-document matrices of the dataset were built.

Finally, starting from the matrices mentioned above, the terms that had a frequency less than six in all the English documents of the dataset were detected and excluded from the dictionaries. With this debugging, the dictionary generated from the metadata text was made up of 7923 terms and plain text files, with 28793 terms.

4.4.3. Algorithms

The experiments were conducted with three main algorithms: construction of the index matrices, construction of thesauri, and query of documents with or without AQE. They were implemented in Java[6]. Additionally, in the first algorithm, it was necessary to use a third-party service and one of its components. The notations used in the algorithms are shown in Table 4.2.

The first algorithm (Algorithm 1) builds the term-document index matrix, starting from the metadata's text (title, keywords, and description) or each plaintext of all dataset documents. It is a different matrix in each case. Besides, starting from the aforementioned matrix, it builds the truncated matrix using LSA, which was developed with JAMA[7] (a package for Java).

For the determination of k (within the step 26) and by analogy with LSA with the Principal Component Analysis method (for data dimension reduction), in which eigenvectors and eigenvalues are calculated to work on a subspace of the original matrix [43], one of the criteria was used to determine the number of components that consists of selecting the first eigenvalues ordered from

[3] http://mendeley.com/
[4] https://lucene.apache.org/
[5] https://www.nltk.org/
[6] https://www.java.com
[7] https://math.nist.gov/javanumerics/jama/

Table 4.2: List of notations used in the algorithms

Notation	**Description**
m	Number of dictionary terms
n	Number of documents in the dataset in the given language
$M \in \mathbb{R}^{m \times 1}$	Set of dataset terms in the given language
$S \in \mathbb{R}^{m \times 1}$	Set of terms for a dataset document
$N_j, \forall\ j = 1 \ldots n$	Term set of $document_j$
tit, keyW, desc	Texts of title, keywords, and description (abstract) of the metadata for each document in the dataset.
$A \in \mathbb{R}^{m \times n}$	Term frequency matrix
$T \in \mathbb{R}^{m \times m}$	Thesaurus matrix
k	LSA truncation value
$A_k \in \mathbb{R}^{m \times n}$	Truncated term frequency matrix (LSA)
usrQuery	User query text
$\mathbf{q} \in \mathbb{R}^{1 \times m}$	Terms frequency vector of the user query
$\mathbf{d_j} \in \mathbb{R}^{m \times 1}$	j-th column of $\boldsymbol{A}$ or A_k
$\mathbf{t_j} \in \mathbb{R}^{m \times 1}$	j-th column of $\boldsymbol{T}$
$\mathbf{cos} \in \mathbb{R}^{1 \times n}$	Cosine vector

highest to lowest until the remaining ones have approximately the same value. The aim is to find an "elbow" in the graph, that is, a point from which the eigenvalues are roughly equal [110]. Besides, it should be in the range of 50 to 1000, which according to Landauer [77], are optimal dimensions for most languages.

Algorithm 1 Index matrix construction

Input: Dataset: metadata, plain text files, isMetada (or not)
Output: A, A_K

1: $M \leftarrow \emptyset$
2: $j \leftarrow 0$
3: **for all** DocOfDataset **do**
4: **if** isMetadata **then**
5: **read** *tit, keyW, desc*
6: $concat \leftarrow tit + keyW + desc$
7: **else**
8: **read** *textOfFile*
9: $concat \leftarrow textOfFile$
10: **end if**
11: $S \leftarrow$ **tokenizer**$(concat)$
12: $S \leftarrow$ **removeStopWords**(S)
13: $S \leftarrow$ **stemming**(S)
14: $N \leftarrow S$
15: $M \leftarrow M \cup S$
16: $p \leftarrow 0$
17: **for all** N **do**
18: $w =$ **getTermCod**(M, N_p)
19: $A_{w,j} \leftarrow A_{w,j} + 1$
20: $p \leftarrow p + 1$
21: **end for**
22: $j \leftarrow j + 1$
23: **end for**
24: $A \leftarrow$ **TF-IDFweighting**(A)
25: $[U, s, V] \leftarrow$ **lsa**(A)
26: **k** $\leftarrow$ **define**$(\mathbf{s})$
27: $A_k \leftarrow U_k \times s_k \times V_k^t$
28: **return** A, A_k

The second algorithm (Algorithm 2) builds the thesaurus starting from the matrix term-document. Matrix $\boldsymbol{T}$ is constructed by simply multiplying matrix $\boldsymbol{A}$ by its transpose. It was implemented in Java, for which the JAMA package was also used.

The last algorithm (Algorithm 3) retrieves the documents with the highest semantic similarity with a given query. To do this, it takes the index term-document matrix $\boldsymbol{A}$ (or A_k) and the text of the user's query, which is transformed into a vector. According to the literature [32], this vector of the original query is expanded to another of fifty terms (lines 13 to 15 of the algorithm), using the

Algorithm 2 Thesaurus matrix construction

Input: A
Output: T
1: $T \leftarrow A \times A^t$
2: $T \leftarrow$ **normalize**(T)
3: **return** T

thesaurus generated with the Algorithm 2. With this vector, the dot product is applied with all the columns of the matrix. These results are sorted from highest to lowest, keeping only those that exceed a given threshold, which indicate the most relevant documents for the query.

The algorithm can also be executed directly with the original query of the user or expanded through the use of the thesaurus $\boldsymbol{T}$. In the latter case, a procedure similar to that mentioned in the previous paragraph is followed: with the original query, the dot product is made, but, now, on the thesaurus $\boldsymbol{T}$, and the first results are taken, typically between twenty-five and fifty [32], which will correspond to the terms of the expanded query. With these new weighted terms, the index matrix is queried.

4.4.4. Experiments

The experiments consisted of carrying out queries on the MD and FT in the dataset, using four different models to evaluate their effectiveness:

1. Classic vector model with TF-IDF weighting.
2. Classic vector model TF-IDF weighted with AQE.
3. Latent Semantic Analysis.
4. Latent Semantic Analysis with AQE.

The tests were conducted on a PC with a 2 GHz Intel® Core i7-4510U processor and 12 GB of RAM, Ubuntu 18.04 operating system. Algorithm 1 worked well with the $\boldsymbol{A}$ matrices for MD and FT and the A_k array for MD. The process crashed out of memory when a singular value decomposition test was performed with the dataset plaintext files containing about 110000 different dictionary terms and nearly 1400 documents. For this reason, a third-party service was sought to build the A_k matrices. The service used was Google Colab[8], in which LSA was implemented in

[8]https://colab.research.google.com/

Algorithm 3 Document retrieval

Input: $A, A_K, T, usrQuery, cosThreshold, withAQE$ (or not)
Output: **cos**, retrieved documents

$termQry \leftarrow$ **tokenizer**$(usrQuery)$
$termQry \leftarrow$ **removeStopWords**$(termQry)$
$termQry \leftarrow$ **stemming**$(termQry)$
$\mathbf{q} \leftarrow$ **buildVec**$(termQry)$
if *withAQE* **then**
  for *j=0 to n-1* **do**
    $c_j \leftarrow \langle \mathbf{q^t}, \mathbf{t_j} \rangle$
    $cos_j \leftarrow \frac{c_j}{\|q\| \|t_j\|}$
    $termPos_j \leftarrow j$
  end for
  $[\mathbf{cos}, \mathbf{termPos}] \leftarrow$ **sort**$(\mathbf{cos}, \mathbf{termPos}, 'descend')$
  $\mathbf{q} \leftarrow [\,]$
  for *i=0 to 49* **do**
    $q_j \leftarrow termPos_j$
  end for
end if
for *j=0 to n-1* **do**
  $c_j \leftarrow \langle \mathbf{q^t}, \mathbf{d_j} \rangle$
  $cos_j \leftarrow \frac{c_j}{\|q\| \|d_j\|}$
  $docPos_j \leftarrow j$
end for
$[\mathbf{cos}, \mathbf{docPos}] \leftarrow$ **sort**$(\mathbf{cos}, \mathbf{docPos}, 'descend')$
for *i=0 to n-1* **do**
  if $cos_i > cosThreshold$ **then**
    print cos_i, doc_{docPos_i}
  else
    break
  end if
end for

Table 4.3: The number of terms according to the type of processing.

Type	Number of terms
Full-text	28793
Metadata	7923

Python[9], and the A_k matrices were generated. The value of k was determined graphically (Figure 4.3): 109 for full-text; and 121 for metadata. The other two algorithms worked properly.

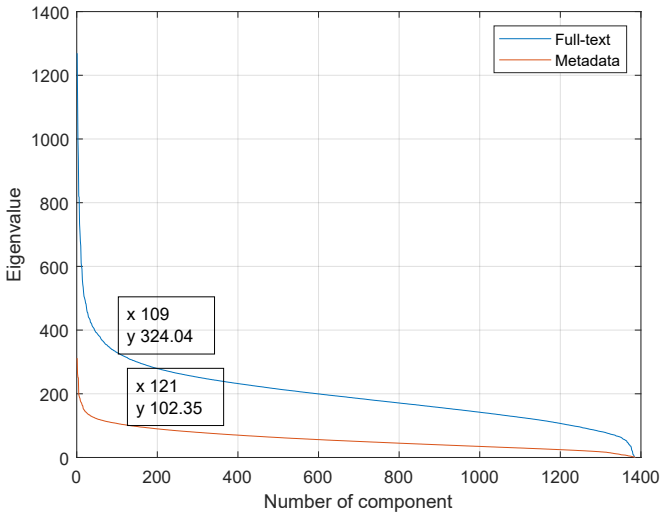


Figure 4.3: Estimation of k for LSA truncation

Regarding the tests with the query algorithm (Algorithm 3), two cosine thresholds were tested, above which documents were retrieved: zero, to retrieve all the calculated results; and another, greater than zero, to improve the precision metric. These thresholds were calculated empirically. Also, for AQE queries, the first fifty terms of the expanded query were taken.

Besides, the thesaurus used was constructed from the metadata term-document matrix. Two reasons were taken into account to build it from the metadata: the first, the metadata text was entered by the users and reviewed in the dataset construction process; therefore, the expected error percentage is minimal; the second, the number of terms corresponds to almost a quarter of those generated with the plain text files (Table 4.3), accordingly, its size is thirteen times smaller than if it were constructed with the full-text terms.

4.5. Results and analysis

The VSM and LSA models' performance, with or without AQE, were evaluated in the dataset presented in subsection 4.4.1. In this vein, two aspects were considered for the analysis of results:

[9]https://www.python.org/

the analysis with classic IR metrics; and statistical analysis. They are presented in the following subsections.

4.5.1. Analysis with IR metrics

The metrics used here were precision and recall, and its derived metrics: average precision, mean average precision (MAP), precision-recall curve, and eleven-point interpolated precision curve. These metrics are described in Chapter 1.

4.5.1.1. Classic vector model with TF-IDF weighting

As shown in figures 4.4 and 4.5, with high precision values, the metadata search has better performance than the full-text search; the situation changes for high recall values. Additionally, for this dataset, queries with fewer words obtained better results in this and the rest of the experiments, contrary to expected. A possible explanation for this result may lie in how the dataset was built: the researchers who participated in its construction contributed keywords from queries related to their research project's main topic, which is usually described in fewer words and are very recurring in the documents. On the other hand, metadata searches in this model obtained better overall performance than full-text searches, as shown in tables 4.5 and 4.4. In this sense, the performance in precision was superior by 5.86% when considering all the results; and 6.01% better when results were filtered with a threshold.

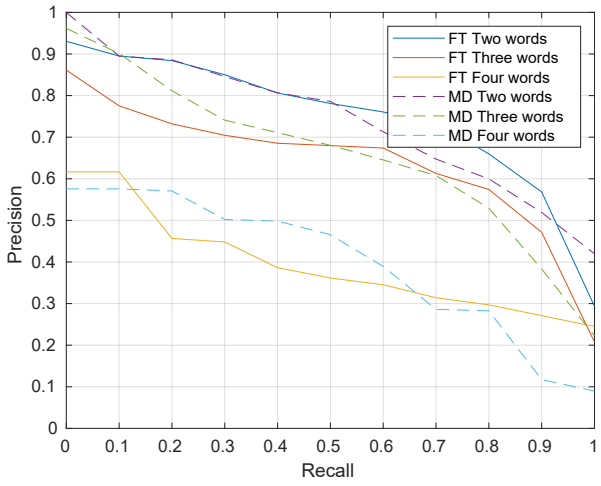


Figure 4.4: Precision-recall curve for two, three, and four words of VSM

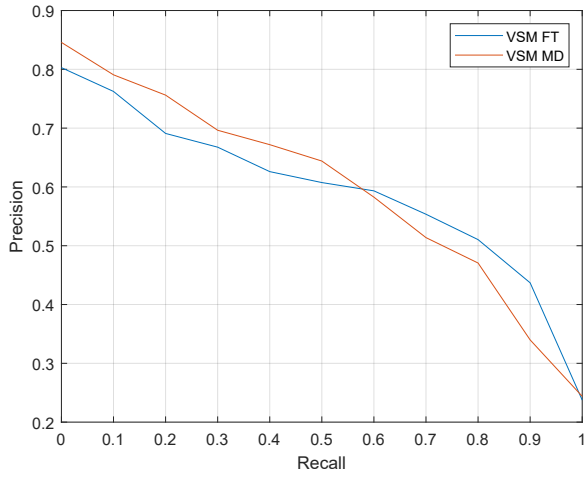


Figure 4.5: Precision-recall curve for metadata and full-text of VSM

4.5.1.2. Classic vector model TF-IDF weighted with AQE

In this test, when adding AQE to the VSM, the overall performance improved as shown by figures 4.6 and 4.7. Likewise, adding AQE to the model made the precision values tighter between FT and MD (tables 4.5 and 4.4).

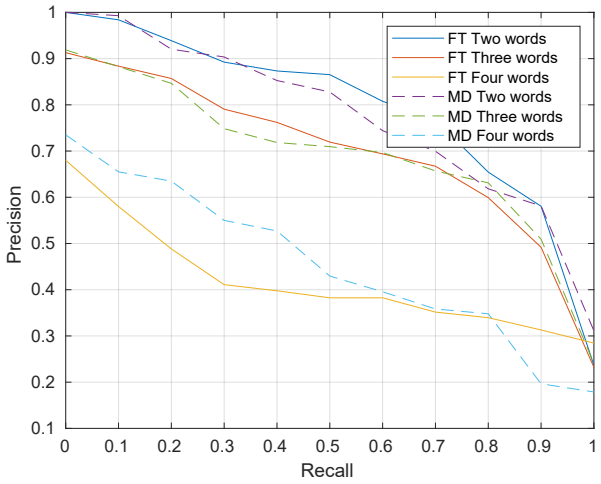


Figure 4.6: Precision-recall curve for two, three, and four words of VSM with AQE

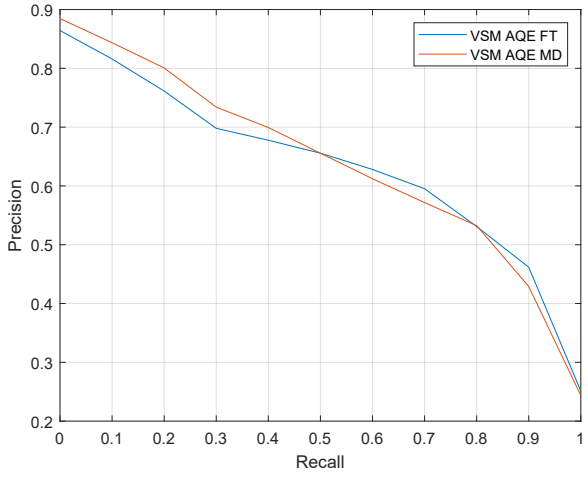


Figure 4.7: Precision-recall curve for metadata and full-text of VSM with AQE

4.5.1.3. Latent Semantic Analysis

Figures 4.8 and 4.9 show the behavior of this model. In general terms, the searches with metadata performed better than with full-text, which can be better appreciated in Figure 4.9. Furthermore, the MAP in metadata searches was very high for low recall values.

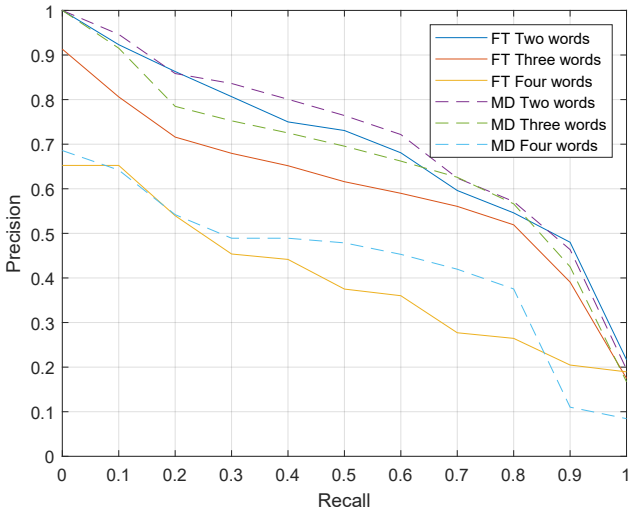


Figure 4.8: Precision-recall curve for two, three, and four words of LSA model

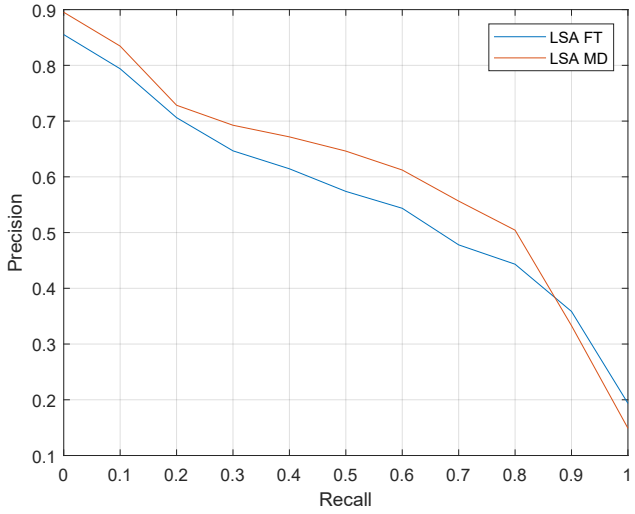


Figure 4.9: Precision-recall curve for metadata and full-text of LSA model

4.5.1.4. Latent Semantic Analysis with AQE

The behavior of the model in this test is similar to that of the VSM with AQE, but with lower performance (figures 4.10 and 4.11). Similarly, again, metadata searches perform better for high precision values and lower for high recall values, the highest for all the models tested.

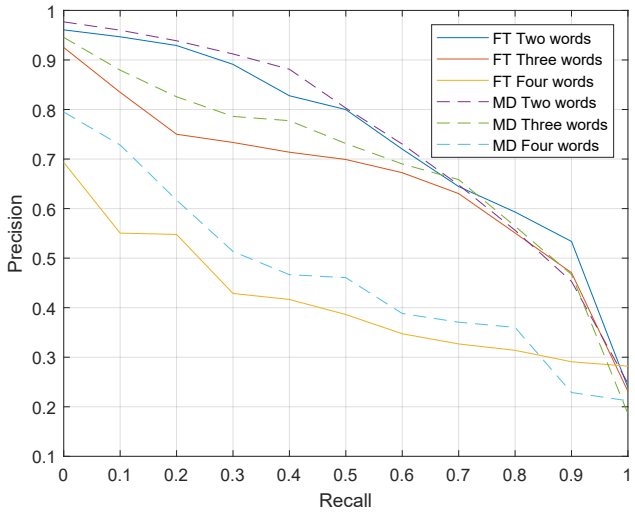


Figure 4.10: Precision-recall curve for two, three, and four words of LSA model with AQE

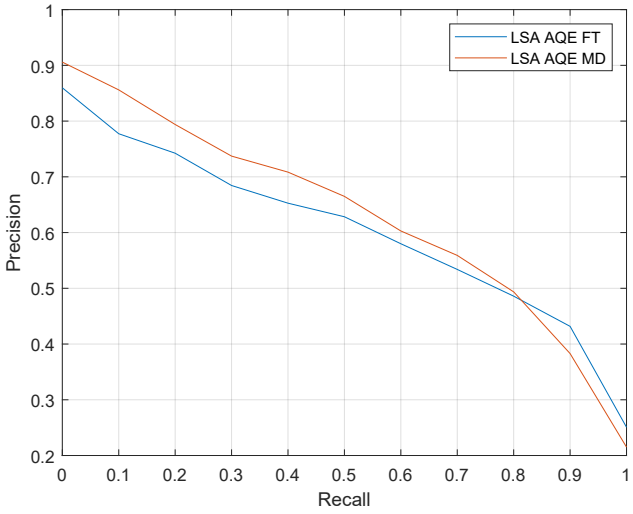


Figure 4.11: Precision-recall curve for metadata and full-text of LSA model with AQE

4.5.1.5. Global analysis

In the tests conducted on this dataset, it was confirmed that VSM with TF-IDF weighting, as a classic model, in itself, is a good IR model (Table 4.5 and Figure 4.12). In fact, its performance was only surpassed by the models in along with AQE, with which the best results were obtained, when included all data (without threshold), applied to both the metadata and the full-text of the dataset documents. Likewise, VSM and LSA models with FT obtained the lowest results.

Regarding the results filtered with a threshold (Table 4.4), the LSA AQE with MD model slightly outperforms the VSM models with AQE. Also, regarding the LSA model's use, parameterized according to the literature's suggestions, no improvements were obtained in the results. Only in one of the global experiments did it exceed the average performance from VSM in the equivalent tests. Already taken specifically, in the three and four-word tests, the LSA model obtained comparable results in two of its modalities (Table 4.4) when they were considered with a threshold. On the other hand, it should be noted that the LSA model tested with MD for high precision (and low recall) values outperformed the rest of the tests (Figure Figure 4.12).

Another aspect to consider in this analysis is the text's pre-processing, especially that of the plain text of the files, which may have slightly influenced in the tests carried out with full-text. In this sense, the package that converts files from PDF or other formats to plain text does so by bringing all its content such as section names, split words (separated by a hyphen) by line breaks, references, annexes, and information exogenous to the subject of the document, among others.

Finally, overall, the application of AQE to the tested models allowed appreciable improvements in the results obtained, which agrees with the academic literature. Thus, in VSM, the increases in mean average precision were from 4 to 12%. In the LSA model, the increase were from 2 to 8%.

Table 4.4: Mean average precision for all models with threshold, sorted by MAP

Model	Type	Cosine Threshold	Two words	Three words	Four words	All queries
LSA AQE	MD	0.130	0.896	0.730	**0.487**	**0.746**
VSM AQE	FT	0.120	**0.914**	0.731	0.424	0.742
VSM AQE	MD	0.130	0.902	0.718	0.463	0.738
LSA	MD	0.200	0.843	**0.750**	0.465	0.732
LSA AQE	FT	0.135	0.874	0.736	0.421	0.729
VSM	MD	0.145	0.824	0.722	0.423	0.705
LSA	FT	0.080	0.803	0.673	0.456	0.681
VSM	FT	0.055	**0.800**	**0.655**	**0.421**	**0.665**

Table 4.5: Mean average precision for all models without threshold, sorted by MAP

Model	Type	Two words	Three words	Four words	All queries
VSM AQE	FT	**0.804**	**0.685**	0.399	**0.677**
VSM AQE	MD	0.790	0.681	0.439	0.676
LSA AQE	MD	0.766	0.685	**0.445**	0.671
VSM	MD	0.751	0.671	0.393	0.650
LSA	MD	0.721	0.669	0.420	0.643
LSA AQE	FT	0.747	0.649	0.386	0.637
VSM	FT	0.743	0.609	**0.370**	0.614
LSA	FT	**0.699**	**0.586**	0.377	**0.589**

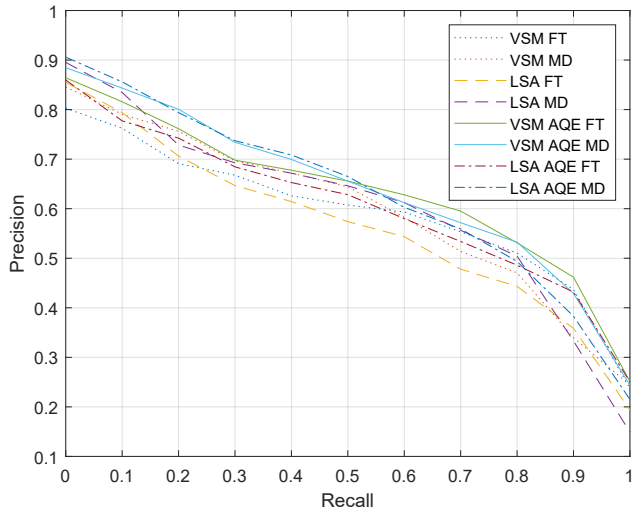


Figure 4.12: Precision-recall curve for metadata and full-text of all models

Table 4.6: Normality test data of models

Model	Type	All data		With threshold	
		W	*p*-value	W	*p*-value
VSM	FT	0.90553	< 2.2e-16	0.78295	< 2.2e-16
VSM	MD	0.86666	< 2.2e-16	0.67084	< 2.2e-16
VSM AQE	FT	0.87035	< 2.2e-16	0.79481	< 2.2e-16
VSM AQE	MD	0.89218	< 2.2e-16	0.84792	< 2.2e-16
LSA	FT	0.92058	< 2.2e-16	0.7876	< 2.2e-16
LSA	MD	0.86281	< 2.2e-16	0.70104	< 2.2e-16
LSA AQE	FT	0.88080	< 2.2e-16	0.72948	< 2.2e-16
LSA AQE	MD	0.85785	< 2.2e-16	0.64439	< 2.2e-16

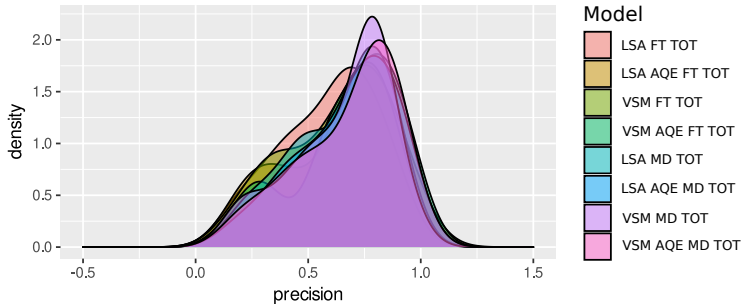


Figure 4.13: Data distribution density

4.5.2. Statistic analysis

To determine if the previous subsection's results apply to the entire population and not only to the dataset, a statistical analysis was carried out, as shown below.

4.5.2.1. Descriptive analysis

Initially, a descriptive analysis of the data is carried out, showing the distribution of the precision of each method, using a density curve. Additionally, a Shapiro-Wilk test is applied to determine if there is normality in the distribution of each method. Based on the calculations, it was determined that the *p*-value is less than 0.05 in all cases (Table 4.6), entering the rejection zone of the null hypothesis. Therefore, the data do not present a normal distribution.

Figure 4.13 shows that the data have a bias towards the left and that the dispersion of the range in precision is different in all models; therefore, it is observed that there is no homogeneity in the variance.

Since the data do not present a normal distribution, non-parametric tests compare the medians of the model's precision.

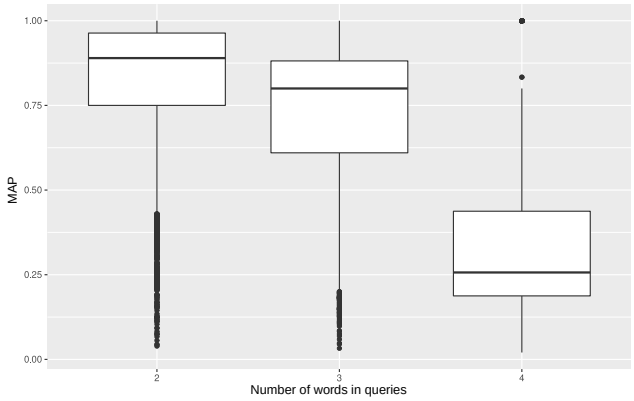


Figure 4.14: Data distribution with two, three, and four words in the queries

Table 4.7: Kruskal-Wallis test of models by the number of words of the queries

Chi-squared	**Degrees of freedom**	***p*-value**
106513	37620	< 2.2e-16

4.5.2.2. Analysis of dispersion by number of words in the queries

A box plot is made to visualize the distribution of the data when two, three, or four words are used in the queries (Figure 4.14). The figure shows that the search with two words presents a higher median in precision, as also determined in Section 4.5.1.

Next, a Kruskal-Wallis test is performed to determine if there are differences between medians when using two, three, or four words. It is found that if there are significant differences between the medians of the precision (Table 4.7). Then, a Post-Hoc pairwise-Wilcoxon analysis is performed for multiple comparisons, resulting in significant differences between all medians as shown in Table 4.8.

To continue with the study, the data from the two-word queries were chosen to compare the medians of the precision since they represented 35.7% of the total of the tests and obtained a higher precision in the experiments.

4.5.2.3. Difference between medians

To determine if there are significant differences between the medians of precision in the different models, a non-parametric Kruskal-Wallis test is performed. In this sense, the data are divided into

Table 4.8: Multiple comparisons (Wilcoxon test)

	Two words	**Three words**
Three words	< 2e-16	-
Four words	< 2e-16	< 2e-16

Table 4.9: Kruskal-Wallis test of models with threshold

Chi-squared	**Degrees of freedom**	***p*-value**
683.42	604	0.01352

Table 4.10: The pairwise Wilcoxon test of models with threshold

	LSA FT	LSA FT AQE	VSM FT	VSM FT AQE	LSA MD	LSA MD AQE	VSM MD	VSM MD AQE
LSA FT	-	**4.6e-11**	**0.0051**	**< 2e-16**	**2.0e-13**	**2.0e-13**	**3.2e-15**	**1.6e-09**
LSA FT AQE	**4.6e-11**	-	**9.8e-07**	0.8074	0.3376	**3.4e-05**	**0.0422**	0.7539
VSM FT	**0.0051**	**9.8e-07**	-	**1.3e-11**	**2.8e-06**	**2.7e-13**	**9.8e-07**	**4.7e-05**
VSM FT AQE	**< 2e-16**	0.8074	**1.3e-11**	-	0.0713	**6.6e-05**	**0.0247**	0.1483
LSA MD	**2.0e-13**	0.3376	**2.8e-06**	0.0713	-	**9.8e-07**	0.8074	0.8460
LSA MD AQE	**< 2e-16**	**3.4e-05**	**2.7e-13**	**6.6e-05**	**9.8e-07**	-	**7.2e-08**	**7.6e-06**
VSM MD	**3.2e-15**	**0.0422**	**9.8e-07**	**0.0247**	0.8074	**7.2e-08**	-	0.7539
VSM MD AQE	**1.6e-09**	0.7539	**4.7e-05**	0.1483	0.8460	**7.6e-06**	0.7539	-

two groups: the models that filtered the results with a threshold (UMB); and the same models with all results (TOT). In this respect, the null hypothesis states that all the models' medians are equal; and the alternative hypothesis states that at least one model has a significantly different median. To validate which model is different, a post-hoc analysis of multiple comparisons is proposed for non-parametric data. In this case, the pairwise Wilcoxon test was used.

- *Comparison of medians with filtered data with a threshold (UMB)*

The results of the Kruskal Wallis test are shown in the Table 4.9. Because p-value is less than 0.05, there are significant differences between the medians.

Table 4.10 shows the results of the multiple comparisons test. Values in bold indicate when there is a significant difference between models.

The median distribution of the data with the threshold filter is visualized using the box plot (Figure 4.15).

- *Comparison of medians with the total data (TOT).*

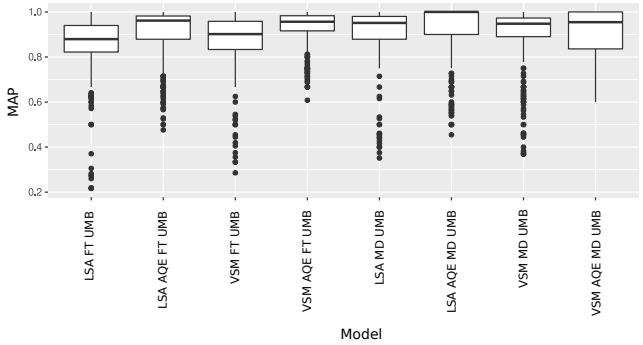


Figure 4.15: The median distribution of two-words models with threshold

Table 4.11: Kruskal-Wallis test of models without threshold (all data)

Chi-squared	**Degrees of freedom**	***p*-value**
2499.8	2362	0.02407

Table 4.12: The pairwise Wilcoxon test of models without threshold

	LSA FT	LSA FT AQE	VSM FT	VSM FT AQE	LSA MD	LSA MD AQE	VSM MD	VSM MD AQE
LSA FT	-	**6.0e-08**	**0.00062**	**2.1e-13**	**4.4e-08**	**5.7e-12**	**5.7e-12**	**2.1e-13**
LSA FT AQE	**6.0e-08**	-	0.05576	0.06464	0.56473	**0.02846**	**0.22622**	0.05301
VSM FT	**0.00062**	0.05576	-	**3.3e-05**	**0.02315**	**0.00017**	**0.00045**	**3.3e-05**
VSM FT AQE	**2.1e-13**	0.06464	**3.3e-05**	-	0.31154	0.54411	0.81738	0.69472
LSA MD	**4.4e-08**	0.56473	**0.02315**	0.31154	-	0.09271	0.59839	0.12552
LSA MD AQE	**5.7e-12**	**0.02846**	**0.00017**	0.54411	0.09271	-	0.28208	0.86159
VSM MD	**5.7e-12**	**0.22622**	**0.00045**	0.81738	0.59839	0.28208	-	0.50350
VSM MD AQE	**2.1e-13**	0.05301	**3.3e-05**	0.69472	0.12552	0.86159	0.50350	-

The medians comparison analyzes were replicated for the total data (without threshold), showing significant differences between the medians since the *p*-value was less than 0.05 (Table 4.11).

The results of the models' comparisons are shown in Table 4.12, and values in bold indicate when there is a significant difference between models.

The data is visualized using a box plot (Figure 4.16).

4.5.3. Joint analysis

Considering the results obtained in the subsections 4.5.1 and 4.5.2, it was evaluated which model would be the most appropriate to experiment with MD and FT searches in an integrated way; and, thus, propose a hybrid system. Thus, as it was possible to establish in Subsection 4.5.1, the models that used AQE obtained better results. Then, supported by the statistical results (4.5.2), it was proceeded to determine the model to be used between VSM and LSA, along with AQE.

Firstly, starting from the comparison of the filtered models with a threshold (Tabla 4.10), it was possible to determine that there is a significant difference (*p*-value 7.6e-6) between LSA AQE

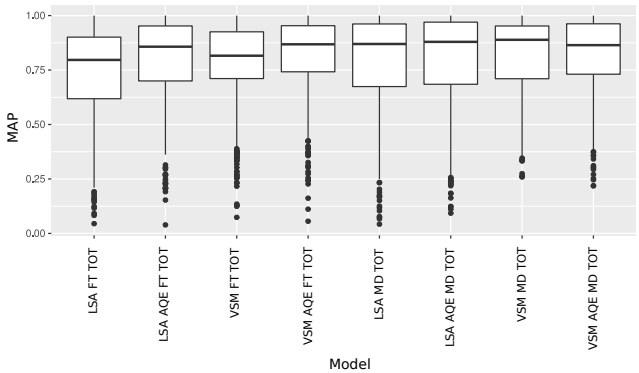


Figure 4.16: The median distribution of two-words models, all data

and VSM AQE when they are executed with MD; not like this, when they are executed with FT (p-value 0.8074). In this sense, the MAP values were, respectively, 0.746 and 0.738.

Secondly, when these same models without threshold were compared (all data, Table 4.12), with MD and FT, it is established that, in both cases, there is no significant difference between them (with p-values of 0.86159 and 0.5301, respectively).

Based on the above and taking into account that in three of the four comparisons of the models with AQE, no significant differences were found and that in the test in which there is a difference, the MAP values are very tight, the use of either model could be considered, based on statistical tests. Then, for the choice of the model, an additional aspect was considered: the use of resources. In this regard, and as mentioned in Subsection 4.4.4, to test the LSA model, it was necessary to use a third-party service (Google Colab), with all that implies. Therefore, it was decided to use VSM AQE for the final experimentation, presented in the next chapter.

4.6. Chapter conclusions

In this chapter, two information retrieval models (classic VSM and LSA) were evaluated on a dataset that contained the metadata and the full-text of the documents. These models were also tested with and without AQE. In this sense, it was found that, in general, the best results were achieved when metadata was used along with AQE.

Additionally, contrary to the expectations, the LSA model used in this dataset did not show improvements in search performance with respect to the classic VSM. Thus, it will be necessary to carry out other tests, modifying the algorithm's execution parameters to obtain better results.

On the other hand, it will be necessary for future works to go a little further in preprocessing the full-text of plain files to correct the small deficiencies that occur when the text is automatically extracted.

Finally, carrying out these experiments sought conceptual and practical elements to propose a hybrid search system for textual learning objects in repositories, based on metadata and content, which corresponds to Objective 3.

CHAPTER 5

A proposal for a hybrid system for searching textual learning objects

5.1. Introduction

In the previous chapter, experiments were carried out to determine the individual (or isolated) behavior of the information retrieval models on the metadata and the full-text of the learning objects. This chapter will be analyzed in an integrated way so that, based on the theoretical foundations of hybrid systems, the literature review, and the results, propose a hybrid system, the objective of this thesis.

5.2. Hybrid models fundamentals

Hybrid models combine the strengths of algorithms and models mentioned in Section 1.2.6 (and their derivations), in order to overcome some of the deficiencies and problems they present individually [63]. They are the most used and useful in practice, as they compensate for the limitations that each model presents when used in isolation [32].

As reviewed, there is not much literature on the theoretical conceptualization of hybridization processes. In this sense, Burke's work [29] on the hybridization of the recommender systems has become a seminal paper. Several manuals, articles, and thesis [4, 24, 36, 63, 175], in this field of IR, take it as a base paper when dealing with the topic of hybrid systems. Based on this, for the purposes of this thesis, the taxonomy proposed by this author of the different combination methods employed is used, as follows:

- Weighted.
- Switching.
- Mixed.
- Feature combination.
- Cascade.
- Feature augmentation.
- Meta-level.

The *weighted model* combines the results of two or more IR systems by calculating weighted sums of scores.

The *switching model* changes between several recommender systems as required.

In the *mixed model*, the results of several recommender systems are presented together at the same time.

In the *feature combination model*, the characteristics of different data sources are integrated into a single recommender system.

In the case of the *cascade model*, the results of a recommender system are refined by another.

In the *feature augmentation model*, the results of a recommender system are used as an input feature of another.

In a *meta-level model*, a recommender system builds a model that is used as the input of another. It differs from the feature augmentation model in that the first model is used to generate some features as inputs for the second one; in this, it is the whole model that becomes input for the second.

On the other hand, Aggarwal in [4] (taking, in turn, [63] as a reference) adds two other levels of aggregation to the taxonomy proposed by Burke, as follows (Figure 5.1):

- Monolithic design.
- Ensemble design.
- Mixed systems.

In the case of *monolithic design*, an integrated system is created by using various types of data from various sources.

In the *ensemble design*, the results of the individual algorithms are integrated into a single, more robust output. In turn, this design can be designed in *parallel* (Figure 5.2) or *sequentially* (Figure 5.3). In the first design, the systems work independently of each other and, in the end, the individual results are combined. In the second one, the result of one system is used as an input of another.

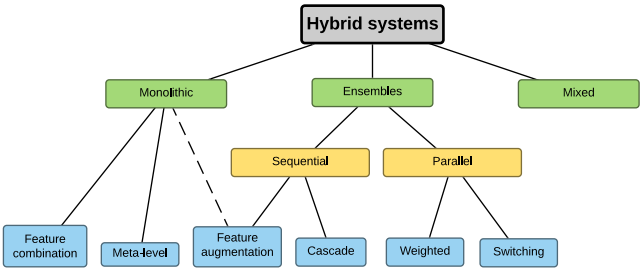


Figure 5.1: Aggregate taxonomy of hybrid systems [4, 63]

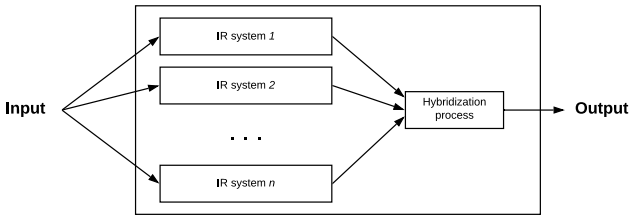


Figure 5.2: Parallel design [4, 63]

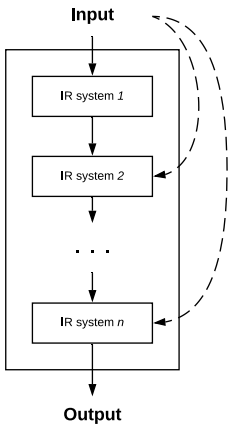


Figure 5.3: Sequential design [4, 63]

The *mixed systems* use several recommendation systems, and their results are presented together.

5.3. A hybrid system proposal

Based on the bibliographic review (Chapter 2) and to propose a hybrid LO search system that integrates MD and FT, the following main aspects were extracted:

- The use of AQE.

- Exploit existing open technologies for their implementation.
- Take into account the domains present in the repository.
- The complementary use of techniques improves the accuracy of search results.
- The use of unsupervised techniques that facilitate searching across multiple domains.

Based on these characteristics, the proposed model is shown in Figure 5.4 and is explained below .

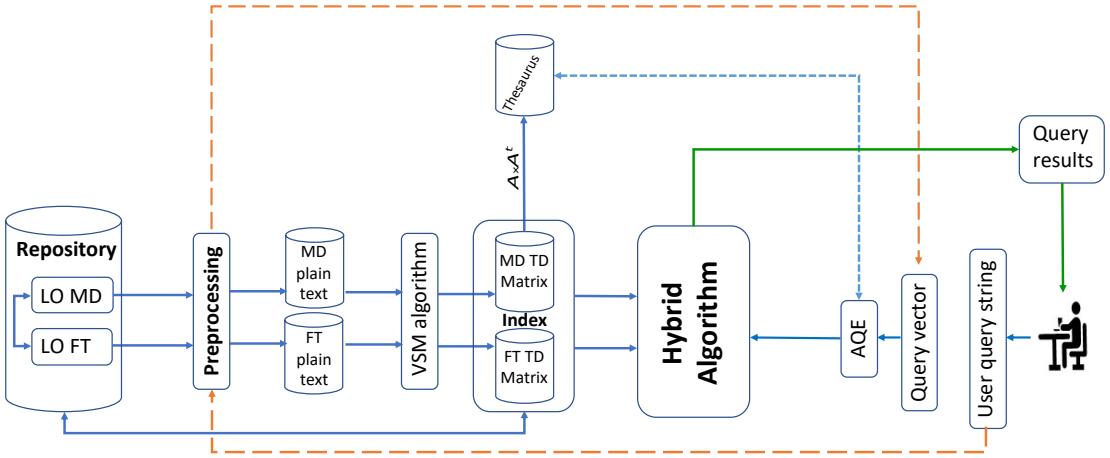


Figure 5.4: Proposed hybrid model system

The *repository* concept in this figure refers to the same one explained in the 5.4 subsection. But, besides, in a broad way, it can be extended to other platforms that use metadata associated with some documents, such as bibliography management systems.

In the *preprocessing*, four processes are conducted: the extraction of the text from the metadata and the full-text of the documents, the tokenization of the texts, the suppression of stopwords, and the stemming. In the first process, the text is extracted from the metadata associated with the title, description, and keywords. Also, the plain text of all the documents in the repository that are in any format that complies with the MIME[1] conventions is extracted. In the second, the text is segmented into terms or concepts. In the third, words that by themselves have no meaning, are very common, or are used as linking words are eliminated. Finally, in stemming, the words are reduced to their roots.

In the above process, two word bags are generated: metadata and full-text. With them, the two term-document index matrices are generated. Besides, the thesaurus to be used in the system is

[1] http://www.utoronto.ca/webdocs/HTMLdocs/Book/book-3ed/appb/mimetype.html

also generated from the document term matrix of the metadata. This thesaurus is constructed from the transposition of the aforementioned matrix.

Additionally, the system viewed from the user starts with the *query text*, which is preprocessed with the same algorithm used with the metadata and full-text in the repository, which generates the *query vector*. With this vector, the query is expanded to fifty words, according to the suggestion of the literature [32], generating a new extended query vector used by the *hybrid algorithm* to calculate the *query results* that the user receives.

Finally, it remains to be determined which hybrid algorithm to use. For this, the results obtained in Chapter 5 are analyzed in the following section in an integrated manner.

5.4. Prototype implementation

To build the dataset, described in Chapter 4, and to evaluate the proposed model, the Java library that we have called *condor-IR* was implemented. It consists of two packages: *clasesIR* and *condorIR*. The first contains the classes that describe the dataset tables and the connection to the database management system. The second contains all the methods used in the implementation of the algorithms. Additionally, for the development of *condor-IR*, tools from the Apache Lucene and Apache Tika frameworks, the Snowball stemmer library, and the PostgreSQL JDBC driver were used (Figure 5.5). The class diagram is presented in Figure 5.6, which was generated through the Netbeans[2] *easyUML* plugin.

Next, based on the condor-IR library, the applications that generate the index files of metadata and full-text were implemented, as well as the corresponding thesauri, except for the index files generated using LSA. These index files were generated through Google Colab for the reasons explained in Subsection 4.4.4.

Finally, using classes and methods from the condor-IR library, the *queryIndiceAQE* application was implemented to experiment with the proposed hybrid system (Figure 5.7).

5.5. Integrated analysis of experiments with full-text and metadata

In Chapter 4, the experiments carried out to observe the behavior of two IR models in FT and MD were analyzed. Now, in this section, the results are evaluated, integrating them according to the suggestions of the literature presented in the subsection 5.2. Specifically, VSM results are evaluated with AQE filtered with a threshold and total (no threshold).

[2]https://netbeans.apache.org/

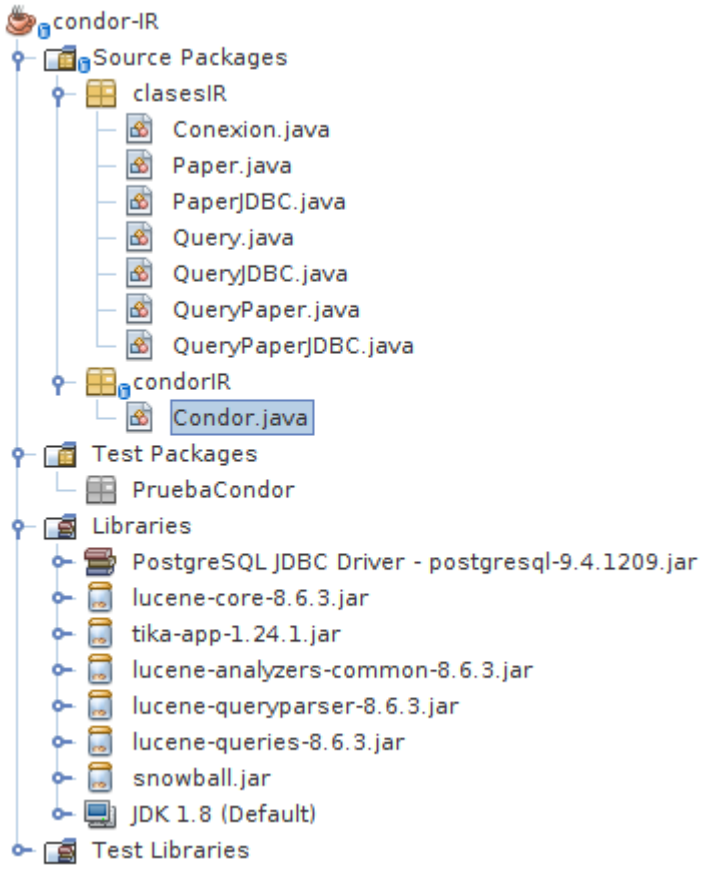


Figure 5.5: condor-IR library structure

5.5.1. Initial data exploration

First, a Venn diagram was made with the total data (without threshold) with MD and FT, which are shown in Figure 5.8 and then, the same is done with the filtered data with threshold, as shown in Figure 5.9.

As shown in both figures, the recovery rate of relevant common results of both models is high: 98,06% for all data and 71,31% for filtered data. Therefore, it is not feasible to use the hybrid cascade or meta-level models presented in the section 5.2. Also, neither features are added or combined, nor are the results presented together. Therefore, only the parallel ensemble models would remain: weighted and switching.

Concerning the latter, a characteristic that can serve as a trigger for the model is not perceived. In this order of ideas, it only remains to use the weighted model.

Two weighted models are tested in the following subsections.

5.5.2. Testing with a weighted hybrid model based on union the data

In the first instance and to test a weighted hybrid model with the union of the data of the FT and MD queries results, the Table 5.1 was defined, both for all the data as for those filtered by a threshold. To do this, weights were defined that sum to one for the intersection data. Besides, for FT and MD complement data, which have a single value, a subtle weight was defined (which could not bias the data greatly) close to one that would reflect, in some way, the weight stated in the intersection.

Table 5.1: Values of weights used with FT and MD data in union hybrid model

Weighting type	**For intersection**		**For FT and MD complements**	
	FT	**MD**	**FT**	**MD**
1	0.5	0.5	1.00	1.00
2	0.2	0.8	0.97	1.03
3	0.4	0.6	0.99	1.01
4	0.6	0.4	1.01	0.99
5	0.8	0.2	1.03	0.97

With the proposed weighting, the results are shown in Figure 5.10 and in tables 5.2 and 5.3 were obtained. In this regard, a decrease in MAP is observed when the filtered data are considered and behavior very similar to the unweighted models when all the data is considered.

Table 5.2: MAP for all data (without threshold) for union data

Type	**Two words**	**Three words**	**Four words**	**All**
VSM AQE FT	0.804	0.685	0.399	0.677
VSM AQE MD	0.790	0.681	0.439	0.676
4	**0.811**	**0.698**	0.438	**0.692**
1	0.807	0.696	**0.442**	0.690
3	0.800	0.695	0.437	0.687
5	0.811	0.696	**0.410**	0.686
2	**0.793**	**0.685**	0.435	**0.679**

Taking into account these results, it can be concluded that when considering the union of the data, there is no improvement in the MAP of the models. So in the next subsection, it will test the models with only the intersection of the data.

5.5.3. Testing with a weighted hybrid model with intersected data

The total and threshold filtered MD and FT data were tested with different weights, as shown in Table 5.4.

Table 5.3: MAP for filtered data (with threshold) for union data

Type	Two words	Three words	Four words	All
VSM AQE FT	0.914	0.731	0.424	0.742
VSM AQE MD	0.902	0.718	0.463	0.738
1	0.897	0.727	0.459	**0.740**
4	0.901	**0.722**	**0.458**	0.739
3	0.901	0.719	0.454	0.737
2	**0.902**	**0.711**	0.452	0.733
5	**0.897**	0.718	**0.429**	**0.730**

Table 5.4: Values of weights used with FT and MD data for intersection data

Weighting type	FT	MD
1	0.5	0.5
2	0.2	0.8
3	0.4	0.6
4	0.6	0.4
5	0.8	0.2

With these weights, the results shown in Figure 5.11 and the tables 5.5 and 5.6 are presented.

As can be seen in the two tables, all combinations of weights exceed the individual MAP values of MD and FT: 0.677 (FT) and 0.676 (MD) for all data (Table 4.5); and 0.742 (FT) and 0.738 (MD) for the threshold data (Table 4.4). Furthermore, in both cases, the *0.6FT + 0.4MD* weighting achieved a higher MAP.

They are then statistically evaluated if these MAP improvements can be considered significant.

The first consideration that took into account is that the data of the evaluated models do not have a normal distribution (Subsection 4.5.2.1). Therefore, nonparametric tests were used for the significance tests. Then, the Kruskal-Wallis test was used (Table 5.7). As the p-value is less than 0.05, the null hypothesis is rejected. Therefore, there is a difference in at least one of the medians of the models. Next, it was determined between which models were differences using a post hoc analysis of multiple pairwise Wilcoxon comparisons (Table 5.8). This table shows that all the hybrid models with intersected data have a significant difference from the FT and MD models of VSM AQE.

Based on the above, it can be concluded that the improvements in the hybrid models apply to the entire population. In this vein, the improvements achieved in the results were of the order of 3.40% when considering all the results (without threshold) and 4.18% with filtered data (with threshold). It is worth noting that these improvements occur on a model, in itself, very effective.

Table 5.5: MAP for all data (without threshold) for intersected data

Type	Two words	Three words	Four words	All
VSM AQE FT	0.804	0.685	0.399	0.677
VSM AQE MD	0.790	0.681	0.439	0.676
4	0.821	**0.704**	0.446	**0.700**
1	0.817	0.702	**0.449**	0.698
3	0.811	0.700	0.444	0.694
5	**0.822**	0.701	**0.419**	0.694
2	**0.804**	**0.690**	0.441	**0.686**

Table 5.6: MAP for filtered data (with threshold) for intersected data

Type	Two words	Three words	Four words	All
VSM AQE FT	0.914	0.731	0.424	0.742
VSM AQE MD	0.902	0.718	0.463	0.738
4	0.936	**0.762**	0.476	**0.773**
1	0.937	0.759	**0.477**	0.772
3	0.937	0.756	0.471	0.770
2	**0.938**	**0.746**	0.468	0.765
5	**0.929**	0.760	**0.443**	**0.764**

5.6. Chapter conclusions

The concepts that support the construction of hybrid information retrieval systems were reviewed. Based on this review, those of Chapters 1 and 2, and the results of the experiments in Chapter 4, a model for finding learning objects was proposed. Then, two new experiments were carried out to test the proposed system.

Next, the results of the experiments with IR metrics were evaluated, which showed that in one of them, improvements were achieved. So, for that experiment, the results were statistically evaluated, showing that they are significant. These improvements were of the order of 3 to 4%.

In this way, it was possible to demonstrate that the hypothesis stated for this thesis was true: a hybrid system for the search of textual learning objects in repositories, based on metadata and content, improves the indicators of precision and recall of the results obtained in systems based only on metadata.

On the other hand, this chapter aimed to fulfill objectives 3, 4, and 5.

Table 5.7: Kruskal-Wallis test of hybrid models of intersection with threshold

Chi-squared	**Degrees of freedom**	***p*-value**
1028.9	440	< 2.2e-16

Table 5.8: The pairwise Wilcoxon test of hybrid models of intersection with threshold

	VSM FT AQE	VSM MD AQE	Type 1	Type 2	Type 3	Type 4	Type 5
VSM FT AQE	-	0.22244	**6.0e-05**	**2.2e-05**	**6.0e-05**	**0.00014**	**0.00041**
VSM MD AQE	0.22244	-	**2.8e-06**	**2.8e-06**	**2.8e-06**	**4.2e-06**	**2.9e-05**
Type 1	**6.0e-05**	**2.8e-06**	-	0.77654	0.93851	0.90264	0.67482
Type 2	**2.2e-05**	**2.8e-06**	0.77654	-	0.77976	0.76623	0.44392
Type 3	**6.0e-05**	**2.8e-06**	0.93851	0.77976	-	0.89411	0.63140
Type 4	**0.00014**	**4.2e-06**	0.90264	0.76623	0.89411	-	0.77654
Type 5	**0.00041**	**2.9e-05**	0.67482	0.44392	0.63140	0.77654	-

Figure 5.6: Class diagram of *condor-IR*

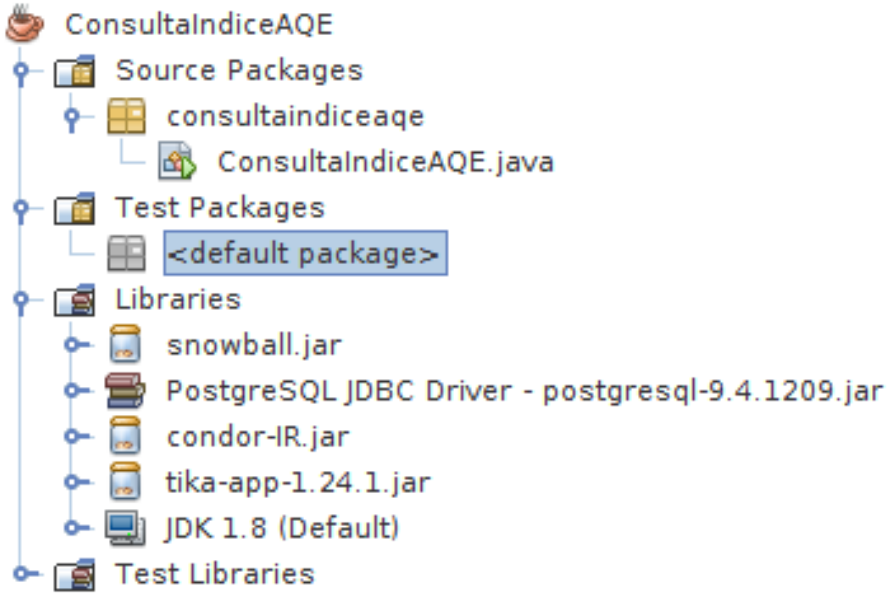


Figure 5.7: consultaIndiceAQE app structure

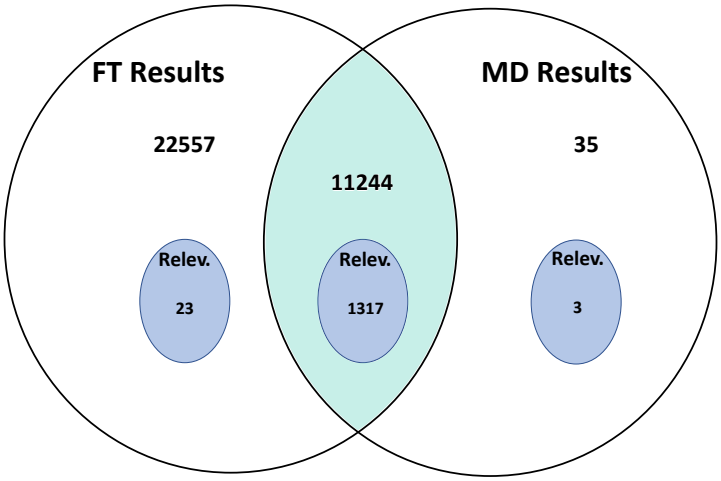


Figure 5.8: Union and intersection of the data for all results (no threshold).

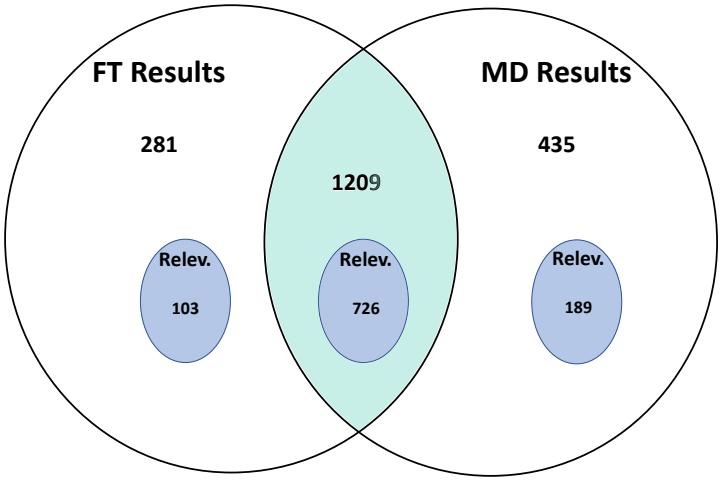


Figure 5.9: Union and intersection of filtered data with a threshold.

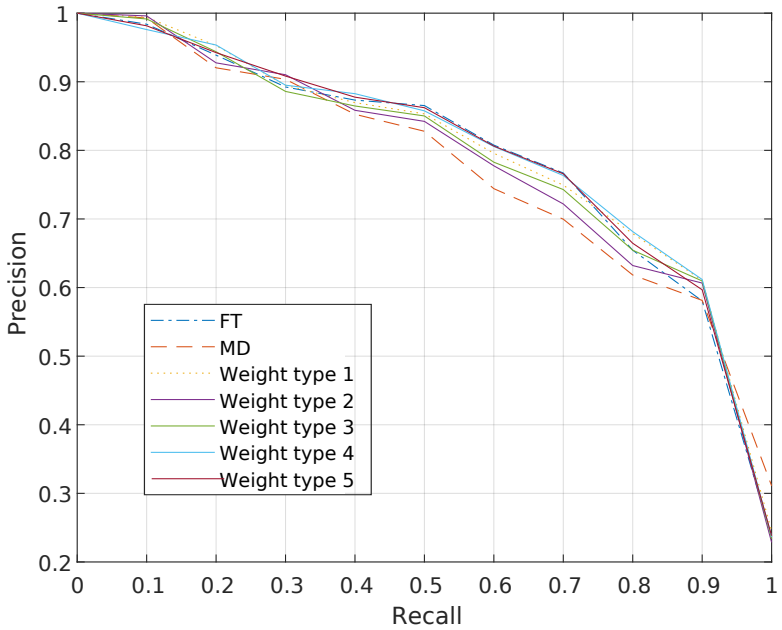


Figure 5.10: Mean Average Precision with all types of weights for union data

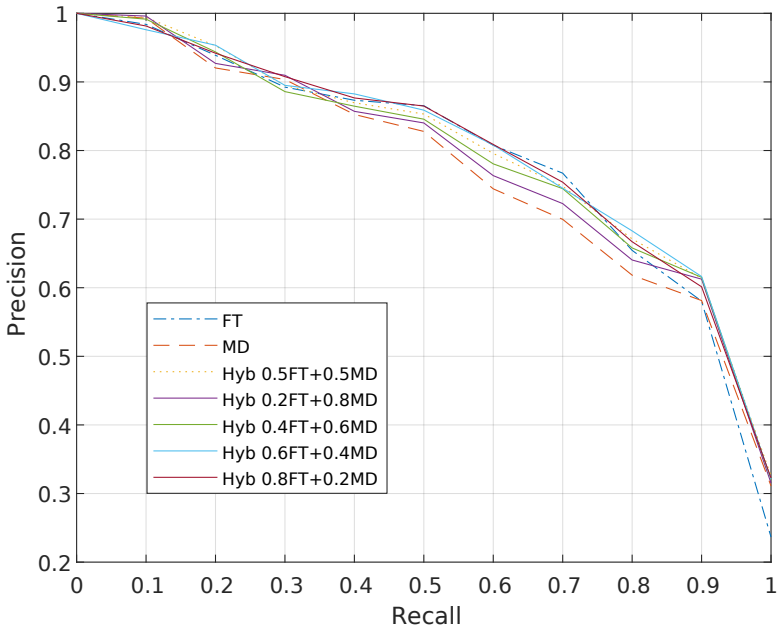


Figure 5.11: Mean Average Precision with all types of weights for intersected data

CHAPTER 6

Concluding remarks

6.1. Conclusions

Based on the research carried out for the development of this thesis, it was possible to demonstrate that, by integrating the full-text and the metadata in the searches for learning objects in a hybrid system, significant improvements are achieved in the search results. Although, the search for learning objects in repositories is still largely done through metadata. However, search systems that incorporate other elements into metadata have been on the rise, such as the user's profile, search history, the use of ontologies in specific domains, among others. In this sense, this proposal goes in the same direction.

Additionally, the hybrid model proposed for the search for learning objects in repositories can be implemented in other isoform contexts such as bibliographic managers. In this sense, it can become an additional tool that helps researchers to explore their own documentation, which, over time, grows to a great extent.

On the other hand, for the systematic review of the literature for this thesis, several bibliographic databases were consulted, which use different ways of measuring citations. Then, it was necessary to devise a methodology to standardize and prioritize the documents to be examined. In this context, the use of Google Scholar allowed to standardize the number of citations and take information from the authors, which were used to rank the papers to be reviewed.

Likewise, the dataset built based on a proposed methodology fulfilled what was expected and allowed to carry out the necessary experiments for the development of the research. In addition, it can be used for research in natural language processing and other information retrieval methods.

Lastly, this research is not the point of arrival but the starting point for new research in this field.

6.2. Main contributions

Below are those that are considered the main contributions of this thesis.

6.2.1. The proposed model

Given the growing number of learning objects in the repositories, a significant improvement in the precision and recall metrics such as those shown by the proposed model, is presented as an opportunity to improve the relevance in the search and retrieval of educational resources in the repositories in which it is implemented. In addition, the use of the system can be easily extended to other platforms in which metadata and textual content are used, such as bibliographic databases, commercial portals, institutional and personal repositories, among others.

6.2.2. Use of standardization tools for systematic review

One difficulty when conducting a systematic literature review is organizing, in a standard way, the information obtained from different bibliographic sources. In this sense, the methodology proposed and used in the review of state of the art (Chapter 2) allowed organizing and ranking the information for review, which can replicate in other investigations.

6.2.3. Methodology for collaborative dataset construction

A methodology was proposed for the collaborative construction of the dataset, with the participation of volunteer researchers, easy to develop and at a low cost. This methodology could be tested in the construction of the dataset used in this investigation.

6.2.4. A dataset

The dataset detailed in Chapter 3 and used in chapters 4 and 5 opens the door to new research and spaces for collaboration with other research groups. Also, it is scalable.

6.2.5. The condor-IR library

As mentioned in subsection 4.4.3, most of the algorithms used in the creation of the dataset and in the experimentation (chapters 3 and 4) were developed in Java, and the source code was stored in the package that we have called *condor-IR* version 1.0. We hope that this tool will be the starting point of a research process that will have an academic impact on the courses that are guided by members of our research group.

6.3. Future work

- Development of an application that implements the proposed model.
- Propose new tools and methods for the preprocessing of the full-text of plain files to correct the small deficiencies that occur when the text is automatically extracted.
- Propose a methodology for evaluating the new queries and documents that will integrate into the dataset.
- Propose a project to increase the number of queries and documents of the current dataset, especially in Spanish and Portuguese languages.
- In LSA, it is proposed a method to adjust the parameters for better results.
- The implementation of an open access web API is considered, that automates many of the tasks presented for dataset construction and that facilitates the integration of several collections.
- Evaluate the possibility of the experimental use of the dataset in bibliometrics.
- Propose an indexing and search application in institutional repositories. Similarly, for personal repositories.
- Implement this search system in FROAC for academic purposes.

6.4. Publications and participation in events and research projects

The participations and names of the events are written in the original language.

6.4.1. Papers

- Osorio-Zuluaga, G. A., & Duque-Méndez, N. D. (2015). Recuperación de objetos de aprendizaje en repositorios: Una aplicación con búsqueda semántica. Revista Gerencia Tecnológica Informática, 14(40), 43–54.
- Osorio-Zuluaga, G. A., & Duque-Méndez, N. D. (2016). Collaborative construction of metadata and full-text dataset. Proceedings - 2016 11th Latin American Conference on Learning Objects and Technology, LACLO 2016.
- Osorio-Zuluaga, G. A., & Duque-Méndez, N. D. (2018). Search and selection of learning objects in repositories : a review. XIII Conferência Latino-Americana de Tecnologias de Aprendizagem - LACLO 2018, 1–8. https://doi.org/10.1109/LACLO.2018.00090

6.4.2. Participation in academic dissemination events and presentation of articles for publication

- II Seminario Internacional para el Análisis de Redes Sociales de Colombia. 2013. Barranquilla, Colombia
- 11th Latin American Conference on Learning Objects and Technology - LACLO 2016. San Carlos, Costa Rica.
- XIII Conferência Latino-americana de Tecnologias de Aprendizagem - LACLO 2018. Sao Paulo, Brasil
- Coloquio Internacional: los retos de la investigación a la solución de problemas sociales. 2019. Manizales, Colombia

6.4.3. Another co-authored investigative production

- Networking en pequeña empresa: una revisión bibliográfica utilizando la teoría de grafos
- 2015 GRSME in Chicago: A review of Entrepreneurial Marketing using Tree of Science
- Metabolomics and pesticides: systematic literature review using graph theory for analysis of references

6.4.4. Participation in research projects

- Red iberoamericana de apoyo a los procesos de enseñanza-aprendizaje de competencias profesionales a través de entornos ubicuos y colaborativos (U-CSCL).

6.4.5. Program committee member (peer evaluator)

- El Congreso Internacional de Ambientes Virtuales de Aprendizaje Adaptativos y Accesibles - CAVA: 2012, 2013, 2015, 2016 and 2017.
- Latin American Conference on Learning Objects and Technologies - LACLO: 2014, 2019, 2020, and 2021.
- Conferencia Colombiana en Gestión de Sistemas de Información y de TIC: GSTIC 2013.
- 15 Congreso Colombiano de Computación.

Bibliography

[1] Manal Abdullah and Nashwa Abdel Aziz Ali. E-learning Standards. In *Communication, Management and Information Technology:*, pages 639–648. CRC Press, Leiden, The Netherlands, 2017.

[2] Entisar Abolkasim, Lydia Lau, and Vania Dimitrova. A Semantic-Driven Model for Ranking Digital Learning Objects Based on Diversity in the User Comments. In *European Conference on Technology Enhanced Learning EC-TEL 2016*, pages 3–15, Lyon, France, 2016. Springer Cham.

[3] Hadhemi Achour and Maroua Zouari. Multilingual Learning Objects Indexing and Retrieving Based on Ontologies. In *2013 World Congress on Computer and Information Technology (WCCIT)*, Sousse, Tunisia, 2013. IEEE Xplore.

[4] Charu C Aggarwal. *Recommender Systems: The Textbook*. Springer, 2016.

[5] Jose Aguilar, Camilo Salazar, Henry Velasco, Julian Monsalve-Pulido, and Edwin Montoya. Comparison and evaluation of different methods for the feature extraction from educational contents. *Computations*, 8(2):1–20, 2020.

[6] Rachid Ahmed-Ouamer and Arezki Hammache. Ontology-based information retrieval for e-learning of computer science. In *2010 International Conference on Machine and Web Intelligence, ICMWI 2010*, pages 250–257, Algiers, Algeria, 2010. IEEE.

[7] Ali Alharbi, Frans Henskens, and Michael Hannaford. Student-Centered Learning Objects to Support the Self-Regulated Learning of Computer Science. *Creative Education*, 03(26):773–783, 2012.

[8] Nasir Ali, Yann-Gaël Guéhénueuc, and Giuliano Antoniol. Trustrace: Mining Software Repositories to Improve the Accuracy of Requirement Traceability Links. *IEEE Transactions on Software Engineering*, 39(5):725–741, 2013.

[9] Omar Alonso and Ricardo Baeza-Yates. An Analysis of Crowdsourcing Relevance Assessments in Spanish. In *Actas del I Congreso Español de Recuperación de Información, CERI 2010*, pages 243–250, 2010.

[10] Omar Alonso and Ricardo Baeza-Yates. Design and implementation of relevance assessments using crowdsourcing. In *Advances in information retrieval*, pages 153–164. Springer Berlin Heidelberg, 2011.

[11] Omar Alonso and Stefano Mizzaro. Using crowdsourcing for TREC relevance assessment. *Information Processing and Management*, 48(6):1053–1066, 2012.

[12] Gustavo Javier Astudillo. Análisis del estado del arte de los objetos de aprendizaje. Revisión de su definición y sus posibilidades. Technical report, Universidad Nacional de La Plata, 2011.

[13] Ch Aswani Kumar, M Radvansky, and J Annapurna. Analysis of a vector space model, latent semantic indexing and formal concept analysis for information retrieval. *Cybernetics and Information Technologies*, 12(1):34–48, 2012.

[14] John Atkinson, Andrea Gonzalez, Mauricio Munoz, and Hernán Astudillo. Web metadata extraction and semantic indexing for learning objects extraction. *Applied Intelligence*, 41(2):649–664, 2014.

[15] Ricardo Baeza-Yates and Berthier Ribeiro-Neto. *Modern Information retrieval - the concepts and technology behind search*. Addison Wesley, Essex, 2011.

[16] Miri Barak and Shani Ziv. Wandering: A Web-based platform for the creation of location-based interactive learning objects. *Computers and Education*, 62:159–170, mar 2013.

[17] A Barbagallo and Anna Formica. ELSE: an ontology-based system integrating semantic search and e-learning technologies. *Interactive Learning Environments*, 25(5):650–666, 2017.

[18] Carla Fillmann Barcelos, João Carlos Gluz, and Rosa Maria Vicari. An Agent-based Federated Learning Object Search Service. *Interdisciplinary Journal of E-Learning and Learning Objects*, 7, 2011.

[19] Jeffrey Beall. Metadata and Data Quality Problems in the Digital Library. *Journal of Digital Information*, 6(3), 2006.

[20] Jeffrey Beall. The Weaknesses of Full-Text Searching. *The Journal of Academic Librarianship*, 34(5):438–444, sep 2008.

[21] Carlos Becerra, Hernán Astudillo, and Marcelo Mendoza. Improving Learning Objects Recommendation Processes by Using Domain Description Models. *LACLO*, 3(1), 2012.

[22] Stefano Bianchi, Christian Mastrodonato, Gianni Vercelli, and Giuliano Vivanet. Use of ontologies to annotate and retrieve educational contents: The AquaRing approach. *Journal of E-Learning and Knowledge Society*, 5(1):211–220, 2009.

[23] Yevgen Biletskiy, Hamidreza Baghi, Igor Keleberda, and Michael Fleming. An adjustable personalization of search and delivery of learning objects to learners. *Expert Systems with Applications*, 36(5):9113–9120, 2009.

[24] J. Bobadilla, F. Ortega, A. Hernando, and A. Gutiérrez. Recommender systems survey. *Knowledge-Based Systems*, 46:109–132, 2013.

[25] Paolo Bolettieri, Fabrizio Falchi, Claudio Gennaro, and Fausto Rabitti. Automatic metadata extraction and indexing for reusing e-learning multimedia objects. In *The ACM International Multimedia Conference and Exhibition*, pages 21–28, 2007.

[26] Markus Borg, Per Runeson, and Anders Ardo. Recovering from a decade: a systematic mapping of information retrieval approaches to software traceability. *Empirical Software Engineering*, 19(6):1565–1616, 2014.

[27] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems*, 30(1/7):107–117, 1998.

[28] Mihaela M. Brut, Florence Sedes, and Stefan D. Dumitrescu. A semantic-oriented approach for organizing and developing annotation for E-learning. *IEEE Transactions on Learning Technologies*, 4(3):239–248, 2011.

[29] Robin Burke. Hybrid Recommender Systems : Survey and Experiments. *User Modeling and UserAdapted Interaction*, 12(4):331–370, 2002.

[30] Vannevar Bush. As we May Think. *The atlantic monthly*, 176(1):101–108, 1945.

[31] Stefan Buttcher, Charles L. A. Clarke, and Gordon V. Cormack. *Information Retrieval: Implementing and Evaluating Search Engines*. MIT Press, Cambridge, Massachusetts, 2010.

[32] Fidel Cacheda Seijo, Juan Manuel Fernández Luna, and Juan Francisoc Huete Guadix. *Recuperación de Información: Un enfoque práctico y disciplinar*. RA-MA S.A. Editorial y Publicaciones, 2011.

[33] Luca Cagliero, Paolo Garza, and Elena Baralis. ELSA: A Multilingual Document Summarization Algorithm Based on Frequent Itemsets and Latent Semantic Analysis. *ACM Transactions on Information Systems*, 37(2):1–33, mar 2019.

[34] Alberto Camaraza Monserrate. Recuperación de información: reflexiones epistémicas de una ciencia en su estado embrionario. *Acimed*, 13(6):1–26, 2005.

[35] Giovanni Capobianco, Andrea De Lucia, Rocco Oliveto, Annibale Panichella, and Sebastiano Panichella. Improving IR-based traceability recovery via noun-based indexing of software artifacts. *Software-Evolution and Process*, 25(7):743–762, jul 2013.

[36] Jorge Castro Gallardo. *Novel Models in Recommender Systems and Group Recommender Systems for Improving Recommendations*. PhD thesis, Universidad de Granada, 2018.

[37] Cristian Cechinel, Sandro Silva Da Camargo, Xavier Ochoa, Salvador Sánchez Alonso, and Miguel Ángel Sicilia. Populating learning object repositories with hidden internal quality information. In *CEUR Workshop Proceedings*, volume 896, pages 11–22, 2012.

[38] Doina Ana Cernea, Esther Del Moral, and Jose E Labra Gayo. SOAF: Semantic Indexing System Based on Collaborative Tagging. *Interdisciplinary Journal of E-Learning and Learning Objects*, 4(1):137–150, 2008.

[39] W. Bruce Croft, Donald Metzler, and Trevor Strohman. *Search Engines: Information retrieval in practice*. 2015.

[40] Anupam Das and Mohammad Al Akour. Intelligent Recommendation System for E-Learning using Membership Optimized Fuzzy Logic Classifier. In *2020 IEEE Pune Section International Conference, PuneCon 2020*, pages 1–10, 2020.

[41] Andrea De Lucia, Massimiliano Di Penta, Rocco Oliveto, Annibale Panichella, and Sebastiano Panichella. Labeling source code with information retrieval methods: an empirical study. *Empirical Software Engineering*, pages 1–38, 2013.

[42] Scott Deerwester, Susan Dumais, George W Furnas, Thomas Landauer, and Richard Harshman. Indexing by Latent Semantic Analysis. *Journal of the American Society for Information Science*, 41(6):391–407, 1990.

[43] Luis Guillermo Díaz M. *Estadística Multivariada: Inferencia y Métodos*. Universidad Nacional de Colombia, Bogotá, 2007.

[44] Chris DiBona, Sam Ockman, and Mark Stone. *OpenSources: Voices from the open source revolution*. O'Reilly, 1999.

[45] Bogdan Dit, Meghan Revelle, Malcom Gethers, and Denys Poshyvanyk. Feature location in source code: a taxonomy and survey. *Journal of Software-Evolution and Process*, 25(1):53–95, 2013.

[46] Nhon V. Do, Hien D. Nguyen, and Long N. Hoang. Some techniques for intelligent searching on ontology-based knowledge domain in e-learning. In *IC3K 2020 - Proceedings of the 12th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management*, volume 2, pages 313–320, 2020.

[47] Néstor Darío Duque-Méndez, Demetrio Arturo Ovalle, and Julián Moreno Cadavid. *Objetos de Aprendizaje, Repositorios y Federaciones: Conocimiento para Todos*. Universidad Nacional de Colombia, Manizales, 2014.

[48] Nicholas Evangelopoulos, Xiaoni Zhang, and Victor R. Prybutok. Latent Semantic Analysis : Five methodological recommendations. *European Journal of Information Systems*, 21(1):70–86, 2012.

[49] Tránsito Ferreras-Fernández, Helena Martín-Rodero, Francisco J. García-Peñalvo, and José A. Merlo-Vega. The systematic review of literature in LIS: An approach. *Proceedings of the Fourth International Conference on Technological Ecosystems for Enhancing Multiculturality - TEEM '16*, pages 291–296, 2016.

[50] Anna Formica, Michele Missikoff, Elaheh Pourabbas, and Francesco Taglino. Semantic search for matching user requests with profiled enterprises. *Computers in Industry*, 64(3):191–202, 2013.

[51] Boris Forthmann, Oluwatosin Oyebade, Adebusola Ojo, Fritz Günther, and Heinz Holling. Application of Latent Semantic Analysis to Divergent Thinking is Biased by Elaboration. *The Journal of Creative Behavior*, 53(4):559–575, dec 2019.

[52] Fabio Gasparetti, Carlo De Medio, Carla Limongelli, Filippo Sciarrone, and Marco Temperini. Prerequisites between learning objects: Automatic extraction based on a machine learning approach. *Telematics and Informatics*, 35(3):595–610, 2018.

[53] Ouafia Ghebghoub, Marie Hélène Abel, and Claude Moulin. Learning Object Indexing Tool Based on a LOM Ontology. In *IEEE International Conference on Advanced Learning Technologies*, pages 576–578, Santander, Spain, 2008. IEEE Computer Society.

[54] Ana Belén Gil, Fernando De la Prieta, and Vivian F López. Hybrid Multiagent System for Automatic Object Learning Classification. In E S Corchado Rodriguez, editor, *Hybrid Artificial Intelligency Systems*, pages 61–68. Springer, San Sebastián, Spain, 2010.

[55] João Carlos Gluz, Ederson Luis Silveira, Luiz Rodrigo Jardim da Silva, and Jorge Luis Victoria Barbosa. Towards a semantic repository for learning objects: Design and evaluation of core services. *Journal of Universal Computer Science*, 22(1):16–36, 2016.

[56] A. Gordillo, E. Barra, and J. Quemada. A Hybrid Recommendation Model for Learning Object Repositories. *IEEE Latin America Transactions*, 15(3):462–473, 2017.

[57] Juan-Miguel Gracia. Álgebra Lineal tras los buscadores de Internet. Technical report, 2002.

[58] Hany M Harb, Khaled Fouad, and Nagdy M. Nagdy. Semantic Retrieval Approach for Web Documents. *International Journal of Advanced Computer Science and Applications*, 2(9):67–76, 2011.

[59] Samer Hassan and Rada Mihalcea. Learning to identify educational materials. *ACM Transactions on Speech and Language Processing*, 8(2):1–18, 2008.

[60] I-Ching Hsu. Intelligent Discovery for Learning Objects Using Semantic Web Technologies. *Educational Technology and Society*, 15(1):298–312, 2012.

[61] IEEE. Standard for Learning Object Metadata. Technical report, Institute of Electrical and Electronics Engineers, New York, 2002.

[62] Amirah Ismail and Mike Joy. Semantic Searches for Extracting Similarities in a Content Management System. In *IEEE International Conference on Semantic Technology and Information Retrieval*, number June, pages 113–118, Putrajaya (Malaysia), 2011. IEEE.

[63] Dietmar Jannach, Markus Zanker, Alexander Felfernig, and Gerhard Friedrich. *Recommender Systems An Introduction*. Cambridge University Press, New York, 2011.

[64] Yuri Kagolovsky and Jochen R. Moehr. Terminological Problems in Information Retrieval. *Journal of Medical Systems*, 27(5):399–408, 2003.

[65] P Kalyanaraman and S Margret Anouncia. Nature inspired clustering and indexing of learning objects based on learners cognitive skills. *International Journal of Knowledge-Based and Intelligent Engineering Systems*, 23(1):41–53, 2019.

[66] Gabriella Kazai, Natasa Milic-Frayling, and Jamie Costello. Towards Methods for the Collective Gathering and Quality Control of Relevance Assessments. In *Proceedings of the 32Nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '09, pages 452–459, New York, NY, USA, 2009. ACM.

[67] Chih-Kun Ke, Kai-Ping Liu, and Wen-Chin Chen. Building a Smart E-Portfolio Platform for Optimal E-Learning Objects Acquisition. *Mathematical Problems in Engineering*, 2013.

[68] Sabrina Keenan, Alan F. Smeaton, and Gary Keogh. The effect of pool depth on system evaluation in TREC. *Journal of the American Society for Information Science and Technology*, 52(7):570–574, 2001.

[69] Igor Keleberda, Victoria Repka, and Yevgen Biletskiy. Building learner's ontologies to assist personalized search of learning objects. In *International conference on electronic commerce*, pages 569–573. ACM, 2006.

[70] Mohamed Koutheaïr Khribi, Mohamed Jemni, and Olfa Nasraoui. Automatic Recommendations for E-Learning Personalization Based on Web Usage Mining Techniques and Information Retrieval. *Educational Technology and Society*, 12(4):30–42, 2009.

[71] Suhyeon Kim, Haecheong Park, and Junghye Lee. Word2vec-based latent semantic analysis (W2V-LSA) for topic modeling: A study on blockchain technology trend analysis. *Expert Systems with Applications*, 152:113401, aug 2020.

[72] Jon M. Kleinberg. Authoritative Sources in a Hyperlinked Environment. *Journal of the ACM*, 46(May 1997):668–677, 1999.

[73] Dimitrios A Koutsomitropoulos, Andreas D Andriopoulos, and Spiridon D Likothanassis. Semantic Classification and Indexing of Open Educational Resources with Word Embeddings and Ontologies. *Cybernetics and Information Technologies*, 20(5):95–116, 2020.

[74] Dimitrios A Koutsomitropoulos and Georgia Solomou. A learning object ontology repository to support annotation and discovery of educational resources using semantic thesauri. *IFLA Journal*, 44(1):4–22, 2018.

[75] Dimitrios A Koutsomitropoulos, Georgia Solomou, and Katerina Kalou. Federated semantic search using terminological thesauri for learning object discovery. *Journal of Enterprise Information Management*, 30(5), 2017.

[76] Alberto H F Laender, Marcos André Gonçalves, Ricardo G. Cota, Anderson A Ferreira, Rodrygo L T Santos, and Allan J C Silva. Keeping a Digital Library Clean : New Solutions to Old Problems. In *Eighth ACM symposium on Document engineering*, pages 257–262, New York, 2008. ACM.

[77] Thomas Landauer and Susan Dumais. Latent semantic analysis. *Scholarpedia*, 3(11):4356, 2008.

[78] Thomas Landauer, Peter W. Foltz, and Darrell Laham. An introduction to latent semantic analysis. *Discourse Processes*, 25(2-3):259–284, jan 1998.

[79] Arash Habibi Lashkari, Fereshteh Mahdavi, and Vahid Ghomi. A Boolean Model in Information Retrieval for Search Engines. In *2009 International Conference on Information Management and Engineering*, pages 385–389, 2009.

[80] Ming Che Lee, Kun Hua Tsai, and Tzone I Wang. An Ontological Approach for Semantic-Aware Learning Object Retrieval. In *IEEE International Conference on Advanced Learning Technologies (ICALT'06)*, pages 208–210, Kerkrade, The Netherlands, 2006. IEEE Xplore.

[81] Ming Che Lee, Kun Hua Tsai, and Tzone I Wang. A practical ontology query expansion algorithm for semantic-aware learning objects retrieval. *Computers and Education*, 50(4):1240–1257, may 2008.

[82] Sangno Lee, Jaeki Song, and Yongjin Kim. An Empirical Comparison of Four Text Mining Methods. *Journal of Computer Information Systems*, pages 1–10, 2010.

[83] Lothar Lemnitzer, Kiril Simov, Petya Osenova, Eelco Mossel, and Paola Monachesi. Using a domain-ontology and semantic search in an e-learning environment. In *Innovative Techniques in Instruction Technology, E-Learning, E-Assessment, and Education*, pages 279–284. Springer, 2008.

[84] Vivian F López, Fernando De La Prieta, Mitsunori Ogihara, and Ding Ding Wong. A model for multi-label classification and ranking of learning objects. *Expert Systems with Applications*, 39:8878–8884, 2012.

[85] Clara López Guzmán. *Los Repositorios de Objetos de Aprendizaje como soporte a un entorno e-learning*. PhD thesis, Universidad de Salamanca, 2005.

[86] Zohair Malki. Comprehensive Study and Comparison of Information Retrieval Indexing Techniques. *International Journal of Advanced Computer Science and Applications*, 7(1):132–140, 2016.

[87] J. Mannar Mannan, K. Sindhanai Selvan, and R. Mohemmed Yousuf. Independent document ranking for E-learning using semantic-based document term classification. *Journal of Intelligent and Fuzzy Systems*, 40(1):893–905, 2021.

[88] Christopher D Manning, Prabhakar Raghavan, and Hinrich Schutze. *An Introduction to Information Retrieval*. Cambridge University Press, Cambridge, England, 2009.

[89] M. E. Maron and J. L. Kuhns. On Relevance, Probabilistic Indexing and Information Retrieval. *Journal of the ACM*, 7(3):216–244, 1960.

[90] Dian I Martin and Michael W Berry. Mathematical Foundations Behind Latent Semantic Analysis. In *Handbook of Latent Semantic Analysis*, chapter 2, pages 35–55. RoutLedge, New York, 2007.

[91] Luis Javier Martínez Rodríguez. *Cómo buscar y usar información científica: Guía para estudiantes universitarios 2013*. Universidad de Cantabria, Santander, España, 2013.

[92] Luciana A Martinez Zaina and Graça Bressan. Learning Objects Retrieval From Contextual Analysis of User Preferences To Enhance E- Learning Personalization. In *IADIS International Conference WWW/Internet*, pages 237–244, Algarve, Portugal, 2009.

[93] Kaiz Merchant and Yash Pande. NLP Based Latent Semantic Analysis for Legal Text Summarization. In *2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pages 1803–1807. IEEE, sep 2018.

[94] Rada Mihalcea and Dragomir Radev. *Graph-Based Natural Language Processing and Information Retrieval*. Cambridge University Press, New York, 2011.

[95] Colombia Ministerio de Educación Nacional. *Recursos Educativos Digitales Abiertos - Colombia*. MEN, Bogotá, 2012.

[96] David Moher, Alessandro Liberati, Jennifer Tetzlaff, Douglas G Altman, and Group PRISMA. Preferred Reporting Items for Systematic Reviews and Meta-Analyses: The PRISMA Statement. *Annals of Internal Medicine*, 151(4):264–269, 2009.

[97] Mohammadreza Molavi, Mohammadreza Tavakoli, and Gábor Kismihók. Extracting topics from open educational resources. In *European Conference on Technology Enhanced Learning*, volume 12315 LNCS, pages 455–460. Springer Cham, 2020.

[98] José A. Moral-Muñoz, Enrique Herrera-Viedma, Antonio Santisteban-Espejo, and Manuel J. Cobo. Software tools for conducting bibliometric analysis in science: An up-to-date review. *Profesional de la Informacion*, 29(1):1–20, 2020.

[99] Olfa Nasraoui and Leyla Zhuhadar. Improving recall and precision of a personalized semantic search engine for E-learning. In *4th International Conference on Digital Society, ICDS 2010*, volume 4, pages 216–221. IEEE Computer Society, 2010.

[100] Andreia Dal Ponte Novelli and José Maria Parente De Oliveira. Simple Method for Ontology Automatic Extraction from Documents. *International Journal of Advanced Computer Science and Applications*, 3(12):44–51, 2012.

[101] Xavier Ochoa and Erik Duval. Towards Automatic Evaluation of Learning Object Metadata Quality. In *Advances in Conceptual Modeling-Theory and Practice*, pages 372–381. Springer Berlin Heidelberg, 2006.

[102] Xavier Ochoa and Erik Duval. Use of contextualized attention metadata for ranking and recommending learning objects. In *The 1st international workshop on Contextualized attention metadata collecting managing and exploiting of rich usage information CAMA 06*, pages 9–16, 2006.

[103] Xavier Ochoa and Erik Duval. Relevance Ranking Metrics for Learning Objects. *IEEE Transactions on Learning Technologies*, 1(1):34–48, 2008.

[104] Xavier Ochoa and Erik Duval. Quantitative Analysis of Learning Object Repositories. *IEEE Transactions on Learning Technologies*, 2(3):226–238, jul 2009.

[105] Xavier Ochoa, Joris Klerkx, Bram Vandeputte, and Erik Duval. On the Use of Learning Object Metadata: The GLOBE Experience. In *Lecture Notes in Computer Science*, pages 271–284. Springer-Verlag Berlin Heidelberg, 2011.

[106] Germán A Osorio-Zuluaga and Néstor Darío Duque-Méndez. Recuperación de objetos de aprendizaje en repositorios: Una aplicación con búsqueda semántica. *Revista Gerencia Tecnológica Informática*, 14(40):43–54, 2015.

[107] Germán A Osorio-Zuluaga and Néstor Darío Duque-Méndez. Collaborative construction of metadata and full-text dataset. In *Proceedings - 2016 11th Latin American Conference on Learning Objects and Technology, LACLO 2016*, San Carlos, Costa Rica, 2016. IEEE Xplore.

[108] Germán A. Osorio-Zuluaga and Néstor Darío Duque-Méndez. Search and selection of learning objects in repositories : a review. In *XIII Conferência Latino-americana de Tecnologias de Aprendizagem - LACLO 2018*, pages 1–8, Sao Paulo, 2018. IEEE Xplore.

[109] Nikolaos Palavitsinis, Nikos Manouselis, and Salvador Sanchez-Alonso. Metadata quality in Learning Object Repositories: A case study. *Electronic Library*, 32(1), 2014.

[110] Daniel Peña. *Análisis de datos multivariados*. McGraw Hill, Madrid, 2002.

[111] Roberto Pérez-Rodríguez, Luis Anido-Rifón, Miguel Gómez-Carballa, and Marcos Mouriño-García. Architecture of a concept-based information retrieval system for educational resources. *Science of Computer Programming*, 129:72–91, 2016.

[112] Laura Plaza Morales. *Uso de Grafos Semánticos en la Generación Automática de Resúmenes y Estudio de su Aplicación en Distintos Dominios: Biomedicina , Periodismo y Turismo*. PhD thesis, Universidad Complutense de Madrid, 2011.

[113] Pithamber R Polsani. Use and Abuse of Reusable Learning Objects. *Journal of Digital Information*, 3(4), 2003.

[114] Martin F Porter. An algorithm for suffix stripping, 1980.

[115] Denys Poshyvanyk, Yann-Gael Guéhéneuc, Andrian Marcus, Giuliano Antoniol, and Vaclav Rajlich. Feature Location Using Probabilistic Ranking of Methods Based on Execution Scenarios and Information Retrieval. *IEEE Transactions on Software Engineering*, 33(6):420–432, 2007.

[116] Mohammad Mustaneer Rahman and Nor Aniza Abdullah. A personalized group-based recommendation approach for web search in E-learning. *IEEE Access*, 6:34166–34178, 2018.

[117] Mohammad Mustaneer Rahman, Nor Aniza Abdullah, and Fnu Aurangozeb. A Framework for Designing a Personalised Web-based Search Assistant Tool for eLearning. In *International Conference on Information and Communication Technology (ICoICT)*, Malacca, Malaysia, 2017. IEEE Xplore.

[118] Aldo Ramirez-Arellano, Juan Bory-Reyes, and Luis Manuel Hernández-Simón. Learning Object Assembly Based on Learning Styles. *Journal of Educational Computing Research*, 55(6):757–788, 2017.

[119] Stephen Robertson. The probability ranking principle in IR. *Journal of documentation*, 33(4):281–286, 1977.

[120] Stephen Robertson and K Sparck Jones. Relevance Weighting of Search Terms. *Journal of the American Society for Information Science*, 27(3):129–146, 1976.

[121] Stephen Robertson, S Walker, S Jones, M M Hancock-Beaulieu, and M Gatford. Okapi at TREC-3. In *Overview of the Third Text REtrieval Conference (TREC-3)*, pages 109–126. 1995.

[122] Ronaldo Lima Rocha Campos, Rafaela Lunardi Comarella, and Ricardo Azumbuja Silveira. Multiagent Based Recommendation System Model for Indexing and Retrieving Learning Objects. In *PAAMS 2013: International Conference on Practical Applications of Agents and Multi-Agent Systems*, pages 328–339. Springer Berlin Heidelberg, 2013.

[123] Paula Andrea Rodríguez M, Gustavo Isaza, and Néstor Darío Duque-Méndez. Búsqueda personalizada en Repositorios de Objetos de Aprendizaje a partir del perfil del estudiante. *Avances Investigación en Ingeniería*, 9(1):73–83, 2012.

[124] L. Antony Rosewelt and J. Arokia Renjit. A content recommendation system for effective e-learning using embedded feature selection and fuzzy DT based CNN. *Journal of Intelligent and Fuzzy Systems*, 39(1):795–808, 2020.

[125] A Sai Sabitha and Deepti Mehrotra. A push strategy for delivering of Learning Objects using meta data based association analysis (FP-Tree). In *2013 International Conference on Computer Communication and Informatics*, pages 1–4. IEEE, 2013.

[126] A Sai Sabitha, Deepti Mehrotra, and Abhay Bansal. Delivery of learning knowledge objects using fuzzy clustering. *Education and Information Technologies*, 21(5):1329–1349, 2016.

[127] Gerard Salton and Christopher Buckley. Term-weighting approaches in automatic text retrieval. *Information Processing and Management*, 24(5):513–523, 1988.

[128] Gerard Salton, A Wong, and C S Yang. A Vector Space Model for Automatic Indexing. *Magazine Communications of the ACM*, 18(11):613–620, 1975.

[129] José Antonio Salvador Oliván and Rosario Arquero Avilés. Una aproximación al concepto de recuperación de información en el marco de la ciencia de la documentación. *Investigación Bibliotecológica*, 20(41):13–43, 2006.

[130] Salvador Sánchez-Alonso, Ramón Ovelar, and Miguel Ángel Sicilia. Estándares de e-learning. In Ana Landeta Etxeberría, editor, *Buenas Prácticas de e-learning*. 2007.

[131] Mark Sanderson. Test Collection Based Evaluation of Information Retrieval Systems. *Foundations and Trends in Information Retrieval*, 4(4):247–375, 2010.

[132] Mark Sanderson and W. Bruce Croft. The history of information retrieval research. *Proceedings of the IEEE*, 100(Special Centennial Issue):1444–1451, 2012.

[133] Camila Sanz, Federico Zamberlan, Earth Erowid, Fire Erowid, and Enzo Tagliazucchi. The Experience Elicited by Hallucinogens Presents the Highest Similarity to Dreaming within a Large Database of Psychoactive Substance Reports. *Frontiers in Neuroscience*, 12, jan 2018.

[134] Robert R Saum. An Abridged History of Learning Objects. In *Learning Objects for Instruction: Design and Evaluation*. Idea Group Inc (IGI), 2007.

[135] Najmus Saher Shah. Review of Indexing Techniques Applied in Information Retrieval. *Pakistan Journal of Engineering and Applied Sciences*, 5(1):27–47, 2015.

[136] Chien-wen Shen and Jung-tsung Ho. Technology-enhanced learning in higher education: A bibliometric analysis with latent semantic approach. *Computers in Human Behavior*, 104:106177, mar 2020.

[137] Wen-Chung Shih and Shian-Shyong Tseng. A Knowledge-based Approach to Retrieving Teaching Materials for Context-aware Learning. *Educational Technology and Society*, 12(1):82–106, 2009.

[138] Wen-Chung Shih, Shian-Shyong Tseng, and Chao-Tung Yang. Using taxonomic indexing trees to efficiently retrieve SCORM-compliant documents in e-learning grids. *Educational Technology and Society*, 11(2):206–226, 2008.

[139] Wen-Chung Shih, Chao-Tung Yang, and Shian-Shyong Tseng. Ontology-based content organization and retrieval for SCORM-compliant teaching materials in data grids. *Future Generation Computer Systems*, 25(6):687–694, jun 2009.

[140] Amit Singhal. Modern Information Retrieval: A Brief Overview. *Bulletin of the Ieee Computer Society Technical Committee on Data Engineering*, 24(4):35–43, 2001.

[141] Hamid Slimani, Nour-eddine El Faddouli, Samir Bennani, and N Amrous. Models of Digital Educational Resources Indexing and Dynamic User Profile Evolution. *International Journal of Emerging Technologies in Learning (iJET)*, 11(1):26–32, 2016.

[142] Boutheina Smine, Rim Faiz, and Jean Pierre Desclés. A semantic annotation model for indexing and retrieving learning objects. *Journal of Digital Information Management*, 9(4):159–166, 2011.

[143] Boutheina Smine, Rim Faiz, and Jean Pierre Desclés. Extracting relevant learning objects using a semantic annotation method. In *IEEE International Conference on Education and e-Learning Innovations*, pages 1–6, Sousse, Tunisia, jul 2012. IEEE.

[144] Boutheina Smine, Rim Faiz, and Jean Pierre Desclés. Relevant learning objects extraction based on semantic annotation. *International Journal of Metadata, Semantics and Ontologies*, 8(1):13–27, 2013.

[145] Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Y Ng. Cheap and fast - but is it good? Evaluation non-expert annotiations for natural language tasks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 254–263, 2008.

[146] Spyridon Stathopoulos and Theodore Kalamboukis. Applying latent semantic analysis to large-scale medical image databases. *Computerized Medical Imaging and Graphics*, 39:27–34, 2015.

[147] David G Stork. An architecture supporting the collection and monitoring of data openly contributed over the World Wide Web. In *Tenth IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, 2001. WET ICE 2001*, number June, pages 380–385, 2001.

[148] David G Stork, Sand Hill Road, and Menlo Park. Character and Document Research in the Open Mind Initiative. In *Fifth International Conference on Document Analysis and Recognition*, pages 1–12, 1999.

[149] Manuel Sucunuta, Guido Riofrio, and Edmundo Tovar. Information Retrieval Model for Open Educational Resources. In *IEEE Global Engineering Education Conference (EDUCON)*, pages 1255–1261, Dubai, 2019. IEEE.

[150] Valentina Tabares Morales, Néstor Duque Méndez, Paula Rodríguez Marín, Julián Moreno Cadavid, and Demetrio Ovalle Carranza. FROAC: Una Iniciativa Colombiana para la Integración de Repositorios de Objetos de Aprendizaje. *Campus Virtuales*, 4(1):108–117, 2015.

[151] Valentina Tabares Morales, Néstor Darío Duque-Méndez, and Julián Moreno. Evaluación experimental de la calidad en la recuperación de objetos de aprendizaje desde repositorios remotos. In *Congreso de Ambientes Virtuales Adaptativos CAVA 2011*, Bogotá, 2011.

[152] Valentina Tabares Morales, Néstor Darío Duque-Méndez, Julián Moreno, and Demetrio Ovalle. FROAC - Federación de Objetos de Aprendizaje Colombia. In *Novena Conferencia Latinoamericana de Objetos y Tecnologías de Aprendizaje*, Manizales, 2014. Universidad Nacional.

[153] Alice Tani, Leonardo Candela, and Donatella Castelli. Dealing with metadata quality: The legacy of digital library efforts. *Information Processing and Management*, 49(6):1194–1205, nov 2013.

[154] Stefaan Ternier, David Massart, Alessandro Campi, Sam Guinea, Stefano Ceri, and Erik Duval. Interoperability for Searching Learning Object Repositories. The ProLearn Query Language. *D-Lib Magazine*, 14(1/2), 2008.

[155] The Ohio State University. *Choosing and Using Sources: A Guide to Academic Research*. The Ohio State University, Columbus, 2016.

[156] Gabriel H Tolosa and Fernando R A Bordignon. *Introducción a la Recuperación de Información: conceptos, modelos y algoritmos*. Universidad Nacional de Luján, Buenos Aires, 2008.

[157] Juan C. Valle-Lisboa and Eduardo Mizraji. The uncovering of hidden structures by Latent Semantic Analysis. *Information Sciences*, 177(19):4122–4147, oct 2007.

[158] Herbert Van de Sompel, Ryan Chute, and Patrick Hochstenbach. The aDORe federation architecture: digital repositories at scale. *International Journal on Digital Libraries*, 9(2):83–100, 2008.

[159] Cornelis Joost; Van Rijsbergen. *Information Retrieval*. University of Glasgow, second edition, 1979.

[160] Jonas Vian, Ronaldo Lima Rocha Campos, Cecilia Estela Giuffra Palomino, and Ricardo Azambuja Silveira. A multiagent model for searching learning objects in heterogeneous set of repositories. In *The 2011 11th IEEE International Conference on Advanced Learning Technologies, ICALT 2011*, pages 48–52, Athens, GA, USA, 2011. IEEE Xplore.

[161] Luis von Ahn and Laura Dabbish. Labeling images with a computer game. In *Proceedings of the 2004 conference on Human factors in computing systems - CHI '04*, pages 319–326, 2004.

[162] Luis von Ahn, Mihir Kedia, and Manuel Blum. Verbosity: a game for collecting commonsense facts. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 75–78, 2006.

[163] Luis von Ahn, Benjamin Maurer, Colin McMillen, David Abraham, and Manuel Blum. reCAPTCHA: human-based character recognition via Web security measures. *Science*, 321(5895):1465–1468, 2008.

[164] Ellen M Voorhees. Variations in Relevance Judgements and the Measurement of Retrieval Effectiveness. *Proceedings SIGIR'98*, 36:315–323, 1998.

[165] Ellen M Voorhees. The philosophy of information retrieval evaluation. In *Evaluation of cross-language information retrieval systems*, pages 355–37. Springer Berlin Heidelberg, 2002.

[166] Ellen M Voorhees and Donna K Harman. *TREC: Experiment and Evaluation in Information Retrieval.* MIT Press, Cambridge, 2005.

[167] G Alan Wang, Jian Jiao, Alan S Abrahams, Weiguo Fan, and Zhongju Zhang. ExpertRank: A topic-aware expert finding algorithm for online knowledge communities. *Decision Support Systems*, 54:1442–1451, feb 2013.

[168] Shouhong Wang. Ontology of Learning Objects Repository for Pedagogical Knowledge Sharing. *Interdisciplinary Journal of E-Learning and Learning Objects*, 4:1–12, 2008.

[169] Tzone I Wang, Kun Hua Tsai, Ming Che Lee, and Ti Kai Chiu. Personalized learning objects recommendation based on the semantic-aware discovery and the learner preference pattern. *Educational Technology and Society*, 10(3):84–105, 2007.

[170] Adam B. Weinberger, Hari Iyer, and Adam E. Green. Conscious Augmentation of Creative State Enhances "Real" Creativity in Open-Ended Analogical Reasoning. *PLOS ONE*, 11(3):e0150773, mar 2016.

[171] Neil Y Yen, Timothy K Shih, Louis R Chao, and Qun Jin. Ranking Metrics and Search Guidance for Learning Object Repository. *IEEE Transactions on Learning Technologies*, 3(3):250–264, 2010.

[172] Tuncay Yigit, Ali Hakan Isik, and Murat Ince. Web-based learning object selection software using analytical hierarchy process. *IET Software*, 8(4):174–183, 2014.

[173] Burasakorn Yoosooka and Vilas Wuwongse. Linked Open Data for Learning Object Discovery: Adaptive e-Learning Systems. In *International Conference on Intelligent Networking and Collaborative Systems*, pages 60–67. Ieee, nov 2011.

[174] Sima Zamani, Sai Peck Lee, Ramin Shokripour, and John Anvik. A noun-based approach to feature location using time-aware term-weighting. *Information and Software Technology*, 56(8):991–1011, aug 2014.

[175] Markus Zanker and Markus Jessenitschnig. Case-studies on exploiting explicit customer requirements in recommender systems. *User Modeling and User-Adapted Interaction*, 19(1-2 SPEC. ISS.):133–166, 2009.

[176] Yuqun Zeng, Xusheng Liu, Yanshan Wang, Feichen Shen, Sijia Liu, Majid Rastegar-Mojarad, Liwei Wang, and Hongfang Liu. Recommending Education Materials for Diabetic Questions Using Information Retrieval Approaches. *Journal of Medical Internet Research*, 19(10):e342, 2017.

[177] Leyla Zhuhadar and Olfa Nasraoui. Personalized cluster-based semantically enriched web search for e-learning. In *The 2nd international workshop on Ontologies and nformation systems for the semantic web - ONISW '08*, pages 105–112, Napa Valley, USA, 2008. ACM.

[178] Leyla Zhuhadar and Olfa Nasraoui. Semantic information retrieval for personalized e-learning. In *IEEE International Conference on Tools with Artificial Intelligence, ICTAI*, volume 1, pages 364–368, 2008.

[179] Leyla Zhuhadar and Olfa Nasraoui. A hybrid recommender system guided by semantic user profiles for search in the e-learning domain. *Journal of Emerging Technologies in Web Intelligence*, 2(4):272–281, 2010.

[180] Leyla Zhuhadar, Olfa Nasraoui, Robert Wyatt, and Elizabeth Romero. Automated discovery, categorization and retrieval of personalized semantically enriched E-learning resources. In *IEEE International Conference on Semantic Computing*, pages 414–419, Berkeley, USA, 2009.

[181] Claus Zinn, Thorsten Trippel, Steve Kaminski, and Emanuel Dima. Crosswalking from CMDI to Dublin Core and MARC 21. In *The International Conference on Language Resources and Evaluation*, pages 2489–2495, Portorož, Slovenia, 2016.