View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

A Taxonomy of Information Retrieval Models and Tools

Gerardo Canfora and Luigi Cerulo

RCOST – Research Centre on Software Technology, University of Sannio, Benevento, Italy

Information retrieval is attracting significant attention due to the exponential growth of the amount of information available in digital format. The proliferation of information retrieval objects, including algorithms, methods, technologies, and tools, makes it difficult to assess their capabilities and features and to understand the relationships that exist among them. In addition, the terminology is often confusing and misleading, as different terms are used to denote the same, or similar, tasks.

This paper proposes a taxonomy of information retrieval models and tools and provides precise definitions for the key terms. The taxonomy consists of superimposing two views: vertical taxonomy, that classifies IR models with respect to a set of basic features, and horizontal taxonomy, which classifies IR systems and services with respect to the tasks they support.

The aim is to provide a framework for classifying existing information retrieval models and tools and a solid point to assess future developments in the field.

Keywords: information retrieval, taxonomy, tools, models.

1. Introduction

In recent years information retrieval has become an important subject of much research, because the amount of information available in digital formats has grown exponentially and the need for retrieving relevant information has assumed a crucial importance. The World Wide Web and the Digital Libraries have shown to a large audience the importance of effective mechanisms and tools to retrieve documents from a very large document collection based on user information needs.

Information Retrieval (IR) is the scientific discipline that deals with the analysis, design and implementation of computerized systems that address the representation, organization of, and access to large amounts of heterogeneous information encoded in digital format [58].

In this paper we focus on text document retrieval, in which the information is represented by text documents. Therefore, for the purposes of this paper, the terms information and documents are used interchangeably. Text document retrieval is the most traditional subfield of IR; however, IR comprises other subfields, such as image retrieval, speech retrieval, information generation, query answering, and text summarization, that we do not cover in this paper.

A key feature of a text IR systems is retrieving the documents that can satisfy the information needs of a user from a large collection of documents. Such systems, especially in the context of the web, are usually known as search engines, so that in the rest of the paper we will consider search engine as a synonym of information retrieval system. IR systems prepare the collection of documents for retrieval through an indexing step. User information needs are usually represented by keywords or phrases, which are themselves indexed, although more complex representation languages are available. This representation, which causes inevitably a loss of information, is usually known as query. Indexing can assume different forms according to the model adopted to represent both the documents in the collection and the user information needs. Many current IR systems exploit ranked IR methods, i.e. they rank the documents in the collection based on a measure of their relevance with respect to the user information needs as represented by a query.

The proliferation of information retrieval algorithms, methods, technologies, and tools, is

making it more difficult to assess the features and the characteristics of each IR aspect and to understand the relationships that exist among them. The terminology is often confusing; for example, terms such as crawling, indexing, spidering, are often used to denote similar tasks, with no clear distinction of the differences.

In this paper we propose a classification of IR models and tools and provide definitions for the key terms. The classification consists of superimposing two views: one for the IR models and one for the IR objects, either tools or services. A vertical taxonomy classifies IR models with respect to a set of basic features, and a horizontal taxonomy classifies IR objects with respect to their tasks, form, and context. The vertical taxonomy is built by exploding two basic features of any IR model: the representation, that is the model adopted to represent both the documents and the user queries; and the reasoning, which refers to the framework adopted to resolve a representation similarity problem. The horizontal taxonomy is derived from an analysis of the application areas of IR.

1.1. Related Works

In the literature, several studies have been proposed that outline classifications of IR models and tools. However, most of these studies do not cover the entire spectrum of IR objects; the reasons can be found either in the age of the papers or in the specific objectives of the studies. For example, in 1984 Smith and Warner [69] published a document representation taxonomy with the aim of relating new research works to previous works and to suggest new areas of research. Nowadays, this taxonomy is largely incomplete, because it does not consider, for example, the representation of structured documents. In 1987 Belkin and Croft [20] published a classification of the most important retrieval techniques in which no reference is made to the relevance feedback model, because, as the authors explicitly state, relevance feedback is not considered a retrieval technique, rather a help to refine the retrieval model.

In a more recent work, Paijmans [54] made an interesting analysis of the most important retrieval models. The approach adopted to construct a taxonomy of IR models consists of identifying a generic model that forms a basis for a variety of more specific models. Paijmans identified the vector document model as the basis for building the classification and showed how the vector model can subsume other popular models. Whilst this constitutes a concise style of classification, it is unable to classify IR techniques that are not derived from the vector based model, such as the logic-based techniques.

Our approach is different, as we start from a classification of the basic features of IR models and proceed with a classification of the objects produced in the various fields of information retrieval in terms of tools and services. The flexibility of this faceted view is evident when we consider that different information retrieval objects can be based on the same information retrieval model, and the same information retrieval model can be exploited to implement different information retrieval objects. For example, the classic vector model, generally presented as a retrieval technique, can be used for building information filtering and document clustering tools, too. The latter are different information retrieval objects that exploit the same information retrieval model.

1.2. Content and Structure of the Paper

There are two main viewpoints that characterize information retrieval: we call these two viewpoints information retrieval objects and information retrieval models. The former is generally an artifact that exists in the form of a tool or a service and responds to the "what" question; the latter is a set of theories on which the information retrieval object is based and respond to the "how" question. The two aspects are related, as one object can be based on more than one model and one model can be the basis for more than one object. On this framework we have built a horizontal taxonomy and a vertical taxonomy. The horizontal taxonomy refers to IR objects, while the vertical one considers IR models.

The remainder of the paper is organized as follows. Sections 2 and 3 introduce the vertical and the horizontal taxonomies, together with examples of their application. Section 4 superimposes the vertical and horizontal taxonomies and shows how this can be used to obtain a mapping of the object's features on the underlying models.

2. Vertical Taxonomy

Modeling the process of information retrieval is complex, because many parts are, by their nature, vague and difficult to formalize. The human component assumes an important role and many concepts, such as relevance and information needs, are subjective. Therefore, information retrieval models can be very complex and, consequently, their classification can be hard. However, in the definition of any IR model we can identify some common aspects. Generally, the first step is the **representation** of documents and information needs. From these representations a **reasoning** strategy is defined that solves a representation similarity problem to compute the relevance of documents with respect to queries. Various strategies have been introduced with the aim of improving the retrieval process: we classify these methodologies under the reasoning component.

Representation and Reasoning can be used to characterize an information retrieval model. For example, in [52] an information retrieval model is characterized as a quadruple $\{D, Q, F, R(q,d)\}$ where:

- D is a set of logical views for the documents in the collection (Representation component);
- Q is a set of logical views for the user information needs (Representation component);
- F is a framework for modeling document representation, queries and their relationships (Reasoning component);
- R(q,d) is a ranking function which associates a real number with a query $q \in Q$ and a document $d \in D$ (Reasoning component).

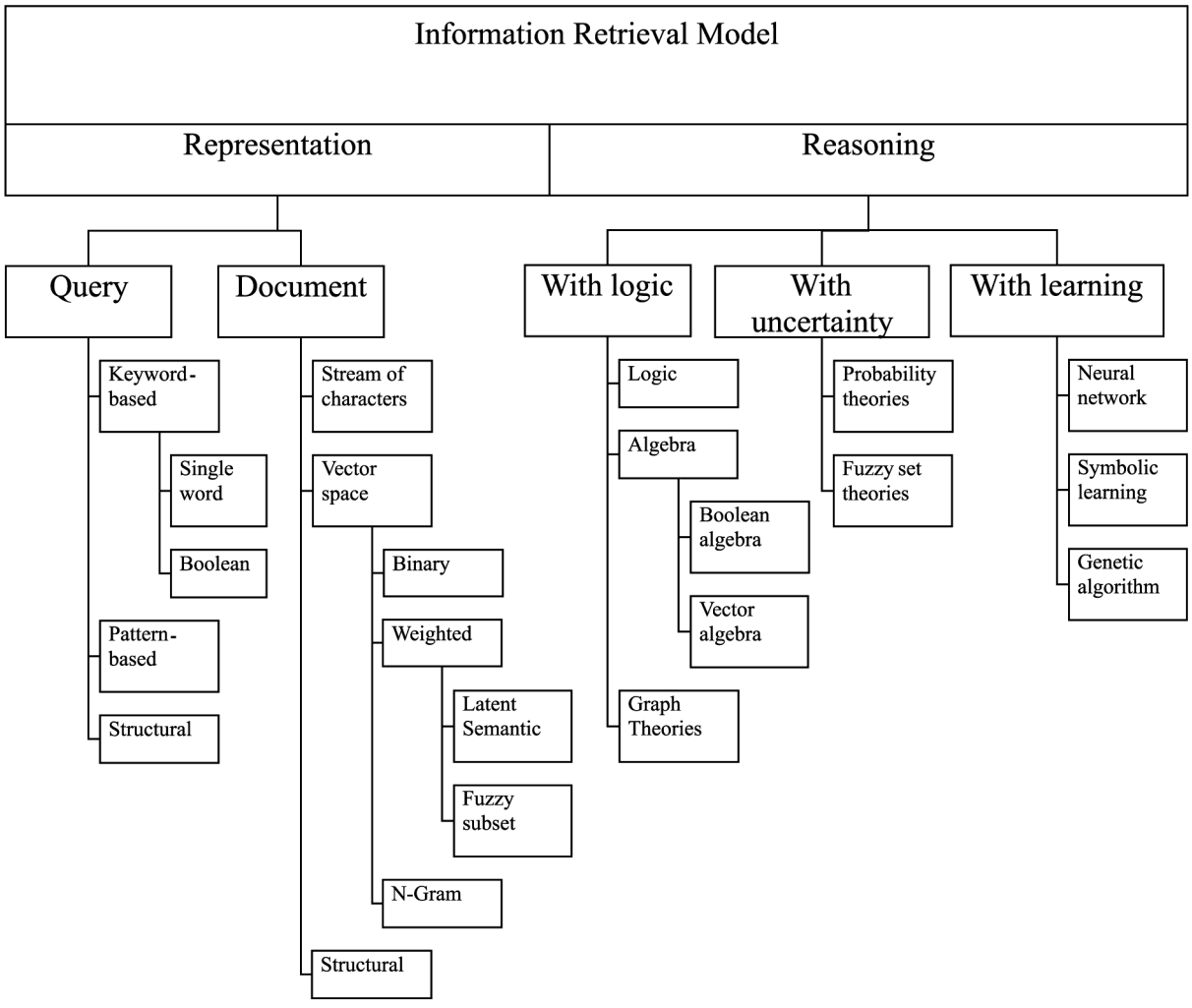

Fig. 1. Vertical taxonomy.

An information retrieval model can be modeled as a couple <Rp, Rs> where Rp is the representation model of documents and queries, and Rs is a framework for modeling the relationship between document and query representations, which is the reasoning strategy. Every component can be divided into subcomponents and for every subcomponent we can build a tree of possible approaches and solutions presented in the literature, as shown in Fig. 1.

Defining the approaches used for each component identifies an IR model. For example, the couple <Rp, Rs>:

Rp(query) = { keyword-based }

Rp(document) = { weighted vector }

Rs(with logic) = { vector algebra }

identifies the well-known vector model, as we will discuss later. We will now go into each of these components.

2.1. Representation

A fundamental component of an IR system is the representation of the information itself: information can be processed if it is represented in some way.

In text information retrieval, representation means representing documents and queries. A document is the representation of the information the author wished to encode; it is the unity of information that can be retrieved by an IR system. Queries are the representation of information needs of a user.

Any text can be characterized by using four attributes: syntax, structure, semantics, and style. A text has a given syntax and a structure, which are usually dictated by the application or by the person who created it. Text also has a semantics, specified by the author of the document. Additionally, a document may have a presentation style associated with it, which specifies how it should be displayed or printed. In many approaches to text representation the style is coupled with the document syntax and structure (see for example the LaTeX document preparation system [40]). Modern representations, such as XML [80], separate the representation of syntax and structures, which are defined either by a DTD or an XSD, and style, which is captured by XSL.

Whilst documents are characterized by syntax, structure, semantics and style, the structure and semantics of text are generally sufficient to characterize queries.

Query Representation

A query is the representation of a user information needs. The user information needs is originated by a problem that the user should resolve; it is implicit in the user mind and its purpose is the necessity to bridge a knowledge gap. An information need can be of three types [50]: known item information need, conscious information need, and confused information need. The first is when users search or verify the existence of documents they know. The second is when users search for documents they do not know, but regard a subject they know. The third is when users know neither the documents nor the subject. The following classes of query representations can be identified:

- *Keyword-based.* This is the simplest form for a query. It is composed by keywords and the documents containing such keywords are searched for. Keyword-based queries are popular, because they are intuitive and easy to express. Usually, a keyword query is a single word, but, in general, it can be a more complex combination of (Boolean) operations applied to several words.
  - — *Single word.* It is the most elementary query that can be formulated in a text retrieval system. Depending on the reasoning component, the result of a single word query is generally the set of documents containing at least one occurrence of the searched word.
  - — *Boolean.* It is the oldest and still widely used form of combining the keywords in a query. A Boolean query is an expression whose elements are keywords, Boolean operators and a precedence notation. In addition to classical Boolean operators, several new operators have been proposed, such as: the NEAR operator, which allows context search capabilities and the fuzzy Boolean operator, which relaxes the meaning of canonical AND and OR.
- *Pattern-based.* It is a more specific query formulation, which allows the specification

of text having some properties. A pattern is a set of syntactic features that must occur in a text segment. The segments satisfying the pattern specification are said to match the pattern.

- *Structural*. Structural queries are a mechanism to improve the retrieval quality of structured information. This mechanism is generally built on top of the basic queries with the addition of structural constrains expressed using containment, proximity, or other restrictions on the structural elements in the documents. Structural queries can be categorized into three main categories: fixed structure, hypertext, and hierarchical structure. The first is the simplest form and, for this reason, it is more restrictive. The documents are divided into a set of fields each of which contains some text. A fixed structural query restricts the search to text contained in certain document fields. The hypertext is probably the most flexible form of structuring. It is a directed graph where the nodes hold some text and the links represent connections between the nodes. However, it is not possible to query the hypertext structural connectivity, but only the text content of the nodes. This transforms the retrieval activity into a navigational activity (browsing task). The hierarchical structure is an intermediate structuring model and represents a natural decomposition for many text collections (books, articles, structural programs etc.). For example, XML is the most prominent structural representation model and the XPath [81] is a query language for addressing pieces of content in the hierarchical structure.

Document Representation

A document is a retrievable element of the document space of an information retrieval system. It can be considered as the minimal resource that an information retrieval system can retrieve. Historically, documents have been represented by a set of terms called keywords, which are usually extracted from the text or inserted by the author. The following are the most significant types of document representation:

- *Stream of characters*. Text is represented as a stream of characters and no interpretation is made on its structure or semantic content.
- *Vector space*. The basic principle of this text representation model is to consider that each document is described by a vector of components that are representative of the semantic content of the document. Traditional vector space approaches use a set of keywords, called index terms, but other types of representative components, such as n-grams, are used. An index term is a word whose semantics helps in identifying the documents main themes. Of course, not all terms of a document are useful for describing the document content. In fact, there are index terms which are vaguer than others. Deciding the importance of terms is not a trivial task. In a large collection of documents a word which appears in each document is useless as an index term, because it does not discriminate between documents. On the other hand, a term that appears in one document will likely describe the content of this document ([45], [83]). Vector representations can be further categorized a s follows.
  - — *Binary*. The text document is represented as a binary vector of terms. Each element of the vector represents a term and its value is '1' if the term appears in the document, '0' otherwise.
  - — *Weighted*. In this case element values are real numbers between 0 and 1, called term weights, and represent the affinity of the term with respect to the document. A widespread method to compute the term weights exploits two factors [58]: Term Frequency (TF) and Inverse Document Frequency (IDF). The first provides a measure of how well the term describes the document contents (intra-cluster similarity); the second measures how well the term can discriminate documents among the collections cluster dissimilarity). A well-known term weighting scheme, valid for generic collections, is the product between the TF and IDF factors. Several variations are described by Salton and Buckley [66].
    - ∗ *Latent semantic*. In the traditional vector space approach each document is represented by a vector of n components, where n is the number of terms occurring in the collection (dimension

of the document space). Latent Semantic Indexing (LSI) [27] reduces the dimension of the document space by capturing term-to-term statistical relationships. The document space is then represented by a new coordinate system of dimension k < n, called k-space (or LSI space), in which each of the k dimension is a derived concept often called *LSI factor* or *LSI feature*. LSI features are identified by using a method for matrix decomposition called *Singular Value Decomposition* (SVD). The derived concepts may be thought of as artificial concepts; they represent extracted common meaning components of many different words and documents.

  - *Fuzzy subset.* Fuzzy set theories deal with the representation of classes whose boundaries are non-well defined. Each element of the class is associated with a membership function that defines the membership degree of the element in the class. In many fuzzy representation approaches the TF-IDF function of the weighted vector model is used as the fuzzy membership function ([35], [37]).

— *N-Gram.* The n-gram approach is in some respects an evolution of vector space approaches. In the traditional vector space approaches the dimensions of the document space for a given collection of documents are the words (or sometimes phrases) that occur in the collection. By contrast, in the n-gram approach, the dimensions of the document space are n-grams: strings of n consecutive characters extracted from the text without considering word lengths, and even word boundaries. Hence, the n-gram is a remarkably pure statistical approach, one that measures the statistical properties of strings of text in the given collection and does not consider the vocabulary, lexical, or semantic properties of the natural language in which the documents are written. The n-gram length (n) and the method for extracting n-grams from documents vary from one author to another. In [22] Damashek uses n-grams of length 5 and 6 for clustering text by language and topic. He uses a sliding window approach in which n-grams are obtained by moving a window of n characters through a document or a query, one character at a time. Some authors [82] also use n-grams that cross word boundaries, i.e., that start within one word, end in another word, and include the space characters that separate consecutive words.

- *Structural*. Structural documents, similarly to structural queries, are a mechanism to improve the retrieval quality. The main idea is to enrich documents with additional information that allow a computer to make part of the semantic content explicit. XML is the most prominent standard for modeling these aspects of information.

2.2. Reasoning

With the term reasoning we refer to the set of methods, models, and technologies used to match document and query representations in a retrieval task. Strictly related with the reasoning component is the concept of relevance. The primary goal of an information retrieval system is to retrieve the documents relevant to a query. The reasoning component defines the framework to measure the relevance between documents and queries using their representations.

A key question to address in order to understand the reasoning component of an IR system is to find a precise definition for relevance. This is still an open problem within the IR community; the literature reports different definitions, but a widespread definition is [67]:

> *Relevance is the (A) of a (B) existing between a (C) and a (D) as determined by an (E).*
>
> *Where:*
>
> *(A). measure, estimate, judgment...*
>
> *(B). utility, matching, satisfaction...*
>
> *(C). document, document representation, information provided...*
>
> *(D). question, question representation, information need...*
>
> *(E). request, intermediary, export...*

An attempt to clarify this definition has been proposed by Mizzaro [51]. Starting from an accurate analysis of the interactions between the users and the system, the paper identifies various types of relevance on which it is possible to define an order relation.

An information retrieval reasoning strategy can be one (or any combination) of: reasoning with logic, reasoning with uncertainty, and reasoning with learning. A reasoning with logic approach deals especially with models developed as logical-mathematical theories. A reasoning with uncertainty approach comes useful whenever the system is unable to assess the truth of all the aspects of the environment in which it operates. In these cases its behavior is affected by uncertainty. This is due to many reasons: it does not understand the environment properties; there are many variables to process and not enough time available, etc. Reasoning with learning approaches apply with inductive machine learning techniques. Machine learning is concerned with systems that learn from experience. In a classical system, the system designer inserts all the knowledge. Whenever the designer does not possess complete knowledge of the system's application domain, a learning mechanism is the only way to acquiring new knowledge. Learning mechanisms are used both for fulfilling an objective or to improve it. In IR the primary goal is to improve retrieval effectiveness, for example, in terms of precision and recall.

Most of the classical information retrieval models deal with the reasoning with logic and reasoning with uncertainty strategies. In the first, for example, fall methods based on first order logic ([47], [8], [6]), and methods based on Boolean and vector algebra ([74], [64], [25], [78], [77]). In the second fall methods in which the vagueness and uncertainty aspects of IR are treated in terms of probabilistic and fuzzy set approaches. Since many information retrieval aspects are affected by vagueness and uncertainty, many reasoning processes based on uncertainty have been proposed ([59], [13], [14], [76], [10], [53], [49], [48], [63], [70]). Machine learning techniques gained a growing popularity in the past ten years ([23], [16], [43]).

Recently, several novel approaches have been proposed, based on either graph theory ([12], [24], [33], [55]) or formal ontology [31].

Reasoning with Logic

- *Logic*. The logical approach to information retrieval can be formulated in terms of the logical formula $P(d \rightarrow n)$, where the arrow is the conditional connective formalized by a logic to be chosen and P is the predicate: "the representation of document d is relevant to the representation of information need n". The central problem is selecting the right implication connective, i.e. selecting the logic whose implication connective best mirrors relevance. An overview of the role of logic information retrieval is reported in [68].
- *Algebra*. Algebra calculus is the most common approach. Under this item we include the reasoning strategies which are based on a set of operations defined in an algebraic field.
  - — *Boolean algebra*. In the conventional Boolean algebra reasoning strategy the query Boolean expression is computed to verify whether a document either satisfies a query (is relevant) or does not satisfy it (is non-relevant). No ranking is possible, and this is a significant limitation. A number of extended Boolean models have been developed to provide ranked output. These extended Boolean models employ extended Boolean operators (also called soft Boolean operators) [42].
  - — *Vector algebra*. Using a weighting scheme for document and query representations the vector algebra approach computes a numeric similarity between the query and each document. The documents can then be ranked according to how similar they are to the query. The usual similarity measure exploited in document vector space is the inner product between the query vector and a given document vector [65]. If both vectors have been cosine normalized, then the inner product represents the cosine of the angle between the two vectors; hence this similarity measure is often called *cosine similarity*. Other well-known variants of similarity functions are: Dice's coefficient and Jaccard's coefficient [58].
- *Graph theories*. Graph theories deal with structures formed by vertices and edges. The

application of graphs algorithms to information retrieval becomes more interesting with the advent of the web. Web resources can be well modelled with a graph structure in which documents represent vertices and hyperlinks represent edges. In [24] a Maximum Flow method is introduced to identify web communities. Previous graph-based approaches were applied to bibliographic documents and were principally based on bibliometric methods such as co citation and bibliographic coupling. Some of these are used in the web context, too. Such algorithm includes: PageRank algorithm [12] on which the Google [104] web search engine is based, HITS algorithm [33], and SAE algorithm [55].

Reasoning with Uncertainty

- *Probability theories*. Probabilistic theories were introduced by Robertson and Sparck Jones [59]. The fundamental reasoning approach is based on the following assumption: given a user query and a document in the collection, the probabilistic reasoning process tries to estimate the probability that the user will find the document interesting. There exist some alternative approaches based on Bayesian networks. In particular, the inference network [71] model has been used in the INQUERY system [13], while reference [57] introduces a generalization called belief network.
- *Fuzzy set theories*. Fuzzy IR models have been defined to overcome the limitations of the crisp Boolean IR models, in particular to manage the vagueness and incompleteness of users in query formulation. Fuzzy extended Boolean models are a superstructure of the Boolean model by means of which existing Boolean IR systems can be extended without redesigning them completely. The standard Boolean models apply an exact match between the query and the document representations, and then partition the document base into two sets: the retrieved documents and the rejected ones. As a consequence of this crisp behavior, they are liable to reject useful items as a result of too restrictive queries, and to retrieve useless material in reply to excessively general queries. Thus, softening the retrieval activity to rank the retrieved items in decreasing order of relevance to a user query can greatly improve the effectiveness of such systems. This objective can be reached by extending the Boolean mode in several ways [35]. In the fuzzy extensions of document representations the aim is to provide more specific and exhaustive representations of the documents information content, in order to reduce the imprecision and incompleteness of the Boolean indexing. For example, a document can be represented as a fuzzy set of terms. In the fuzzy generalization of the Boolean query language the objective must have a more expressive query language, in order to capture the vagueness of the user needs as well as to simplify the user system interaction. Various approaches have been proposed. One of these introduces soft connectives of selection criteria [11], characterized by a parametric behavior which can be set between the two extremes "AND" and "OR". In other approaches, the Boolean query language has been generalized by defining aggregation operators as linguistic quantifiers, such as "at least k" or "about k".

Reasoning with Learning

Several authors have proposed the use of machine learning approach in IR. The most frequently used techniques include [16]: multiple layered and feed-forward neural networks such as back propagation networks [62], symbolic and inductive learning algorithms such as ID3 [56] and ID5R [72], and evolution-based algorithms such as genetic algorithms [34].

- *Neural networks*. Neural network computing seems to fit well with conventional retrieval models such as the vector space model and the probabilistic model. One of the first applications in IR comes from Belew [7]. He developed a three-layer neural network of authors, index terms, and documents. The system used relevance feedback from its user to change its representation of authors, index terms, and documents over time. An evolution of this application has been introduced by Kwok [39], who uses a modified Hebbian learning rule to reformulate probabilistic information retrieval. In other applications the

Neural Network approach has been used for more specific tasks. For example, in [44], a Kohonen's self-organizing feature map was applied to construct a self organizing representation of the semantic relationships between documents. A Neural Network document clustering algorithms was developed in [46]. The Hopfield neural network's parallel relaxation method was used in [17] for concept-based document retrieval and exploration.

- *Symbolic learning*. In IR the use of symbolic learning is more limited with respect to other learning techniques. In [9] a symbolic learning technique is used for automatic text classification. The symbolic learning process represents the numeric classification results in terms of IF-THEN rules. In [26] a regression method and ID3 were used to implement a feature-based indexing technique. In [18] ID3 and the incremental ID5R algorithm were adopted for information retrieval. Both algorithms were able to use user-supplied samples of desired documents to construct decision trees of important keywords which could represent the user's query.
- *Genetic algorithms.* Several genetic algorithms implementations have been developed in the context of IR. [29] presents a genetic algorithm-based approach to document indexing, in which competing document descriptions (binary vector of term) are associated with a document and altered over time by using genetic mutation and crossover operators. In this design, a keyword represents a gene (bit pattern), a document which is a vector of keywords (bit string) represents individuals, and a collection of documents, initially judged relevant by a user, represents the initial population. Based on a Jaccard's matching function, the initial population evolves through generations and eventually converges to an optimal, improved population. In [30] a similar approach is adopted for document clustering.

2.3. An Example

As an example of application of the vertical taxonomy, we have taken some relevant works from the IR models field and tried to classify

Information Retrieval Model	References	Vertical Taxonomy													
		Representation						Reasoning							
		Query			Document			With Logic			With Uncertainty		With Learning		
		Keyword-based	Pattern-based	Structural	Stream of characters	Vector space	Structural	Logic	Algebra	Graph theories	Probability theories	Fuzzy set theories	Neural network	Symbolic learning	Genetic algorithms
Pattern match	[4], [5], [79]		X		X				X						
Vector	[74], [64], [25], [78], [77]	X				X			X						
Probabilistic	[59], [13], [14], [76], [19]	X				X					X				
Fuzzy	[10], [48], [63], [70]	X				X						X			
Fuzzy with learning	[53]	X				X						X	X		
Fuzzy Thesaurus	[49]					X		X				X			
Genetic	[36], [29], [30]	X				X			X						X
Neural Network	[75], [2], [41]	X				X			X				X		
Probabilistic Neural Network	[39]	X				X					X		X		
Neural Network clustering	[46]					X	X		X				X		
Symbolic learning	[9], [18], [26]	X				X				X				X	
Browsing	[1], [21], [15]						X			X					
Rule-based and logic	[47], [8], [6]	X			X			X							

Table 1. Vertical taxonomy of a set of Information Retrieval Models.

them using the vertical taxonomy. We identify each information retrieval model in relation to the representation and reasoning components described above. This is shown in Tab. 1. A notable aspect is that many models contain the weighted vector as a representation component; this is why Paijmans [54] introduced the vector document model.

3. Horizontal Taxonomy

The vertical taxonomy alone is not sufficient to take into account all the objects that have been produced under the IR umbrella. Users do not interact with a model, but generally they use a software tool that is able to solve an information retrieval problem. This calls for the introduction of a further dimension, a new viewpoint that we call horizontal taxonomy. Through the horizontal taxonomy we classify information retrieval objects. An information retrieval object is an artifact that solves a more or less general IR problem. An information retrieval object is identified by three components, as illustrated in Fig. 2: Tasks, Form, and Context.

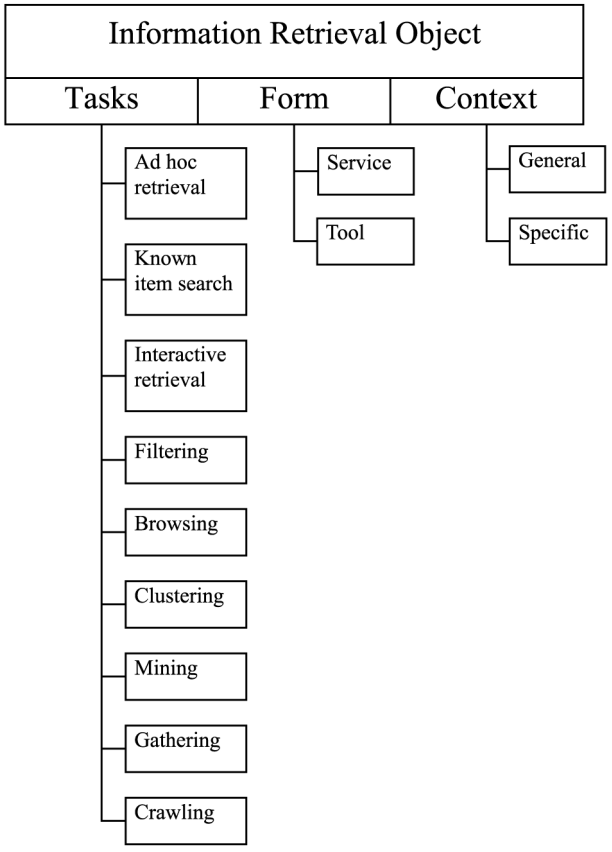

Fig. 2. Horizontal taxonomy.

3.1. Tasks

Information retrieval tasks are concerned with a particular aspect of information retrieval derived from a user point of view and should not be confused with the tasks in an information retrieval process, such as query formulation, query expansion, comparison, ranking, document presentation. An information retrieval object can support one or more tasks and a task can be stand-alone or it can be integrated in a process to perform a larger task. We have identified the following tasks: ad hoc retrieval, known item search, interactive retrieval, filtering, browsing, clustering, mining, gathering and crawling. Sometime they are known by different names because they are inherited from various research areas.

Ad Hoc Retrieval

An ad hoc retrieval task is characterized by an arbitrary subject of the search and a short duration [73]. It is typically performed by a researcher doing a literature search in a library. In this environment the retrieval system knows the set of documents to be searched, but cannot anticipate the particular topic that will be investigated [73]. A retrieval system's response to an ad hoc search is generally a list of documents ranked by decreasing similarity to the query. The internet search engines are examples of information retrieval objects from which one can perform ad hoc search.

Known Item Search

A known item search is similar to an ad hoc search, but the target of the search is a particular document (or a small set of documents) that the searcher knows to exist in the collection and wants to find it [73]. An information retrieval object that performs this task usually implements a precise query language (for example, structural query language) with which a searcher can reach parts of a document with known structure and semantics. For example, in the library environment, a researcher that will retrieve all articles by an author.

Interactive Retrieval

A user's judgment of the usefulness of a document may vary during an information seeking activity [38]; this can be captured by the system through an interactive information retrieval task. During the interactive task the system attempts to perceive how the user interacts with it and, as a consequence, it can modify the current search strategy [60]. Classical relevance feedback approaches [61] can be seen as early techniques for interactive retrieval; the user interaction is captured as yes/no judgment of documents relevance. The system uses these judgments to expand and/or reweigh the query [32].

Filtering

Also known as selective dissemination of information, or text routing, filtering combines aspects of text retrieval and text categorization. Like text categorization, a text filtering system processes documents in real time and assigns them to zero or more classes. However, like text retrieval, each class is typically associated with the information needs of one or a small group of users. Each user, or user group, can typically add, remove, or modify the queries, or profiles, according to their needs. Examples include: NewsSieve [100] a client/server USENET news filtering system that can be used in a desktop environment, NewsWeeder [87] an experimental USENET news filtering service, and SIFT the Stanford Information Filtering Tool [86], which includes two selective dissemination services, one for computer science technical reports and one for USENET news articles.

Browsing

When users are not interested in posing a specific query to the system, but they invest some time in exploring the document space, looking for interesting references, then they are browsing the space, instead of searching. There are three types of browsing, namely, flat, structure-guided and hypertext. In flat browsing the idea is that the user explores a document space which has a flat organization; for example, files in a directory. In structure-guided browsing the user is generally guided by a hierarchical structure in which documents are organized in categories and subcategories. The hypertext model introduces a navigational structure which allows a user to browse text in a non sequential manner. The web is the most well know example of hypertext structure.

Clustering

The term emerges from the statistics community, where it is well known as classification analysis and discriminant analysis [3]. In the artificial intelligence community, the task is often called concept learning. Clustering is the automatic recognition and the generation of categories of entities that can be text documents. It is usually based on some similarity measure between documents, as well as an explicit or implicit definition of what distinguishing characteristic should the groups of documents have. It is generally used to improve the retrieval process, because the search can be restricted on a set of interested category. In conjunction with clustering is categorizing, which is the recognition and assignment of the document to one or more pre-existing categories. An example of categorization tools is CORA (Computer Science Research Paper Search Engine) [84], an automatic categorizing tool for scientific papers. An example of categorizing service is the Yahoo Directory [99]; in this case the categorization is performed manually, by human experts.

Mining

Mining is the process of automatically extracting key information from text documents. Such information can be: language identification, feature extraction, terminology extraction, predominant themes extraction, abbreviation extraction and relation extraction. LEXA [89] is an example of a corpus processing software, while the IBM text miner [91] is a mining tool integrated with the homonymous text search engine.

Gathering

This is an activity involving pro-active acquisition of information from possibly heterogeneous sources. The metasearch engines exemplify a particular type of gathering task. Metacrawler [92], InFind [116] are some examples. They combine outputs of several search engines and present the results as if produced by a single search engine.

Crawling

Crawling is concerned with the activity of selecting new, or updating the existing, sources of information that will be processed by successive activities, for example mining and/or gathering. It is also known as indexing process and, especially in the Web context, as spidering. Well known examples are: Scooter [94], ArchitextSpider [110], Sidewinder [112], Slurp [102] and Guliver [114]; the spiders of Altavista [93], Excite [109], Infoseek [111], Inktomi [101] and Northernlight [113].

3.2. Form

The form refers to the way in which the object is supplied to the final user. It can be supplied in the form of tool or service. When the object is implemented as a software product, then it is a tool. It exists because, for example, a company has produced it to make business. It can be distributed, installed, sold, etc. When the object exists only in one, or a few instances used to deliver some information retrieval services, then it is a service. Examples are search engines on the web.

3.3. Context

The context of an information retrieval object regards its domain of application. It can be general or specific. A general purpose information retrieval object operates on heterogeneous domains and contents, unlike a context specific system that operates on document collections belonging to a specific domain, such as legal and business documents, technical papers etc. Notable examples are web search engines, where the high heterogeneity of the information calls for a very general purpose approach. Google [104], Altavista [93], and Infoseek [111], are some general purpose engines that currently operate on the web. A specialized retrieval system is one that is developed with a particular application domain in mind. For instance, the LEXIS-NEXIS [119] retrieval system is a specialized retrieval system that provides access to a very large collection of legal and business documents. Similarly, the ResearchIndex service [105] provides free access to a large collection of scientific paper.

3.4. An Example

As we did with the vertical taxonomy, here we apply the horizontal taxonomy to a set of information retrieval objects. We have chosen 31 objects from various sources: research labs, companies, and institutions.

The main classification scheme consists of identifying, for each object, its horizontal components included in Fig. 2.

This is done by analyzing the object as a black box and trying to fetch information about what it does. The result is viewed in the Appendix in which information retrieval objects are listed with some information notes and references. The presence of a cross establishes that the corresponding horizontal component is supported by the information retrieval object.

4. Concluding Remarks

For the purpose of simplicity, we have conducted the classification on two separate paths: a horizontal taxonomy and a vertical taxonomy. In reality, these taxonomies are not disjoint and in this concluding section we show how these two important aspects of information retrieval can be combined. We have already remarked that an information retrieval object can be based on more than one model and an information retrieval model can be the basis for more than one object.

The vertical dimension classifies information retrieval models based on a two components view, namely representation and reasoning. The horizontal dimension classifies information retrieval objects with respect to the application

Information Retrieval Object	Vertical Taxonomy													
	Representation						Reasoning							
	Query			Document			With logic			With uncertainty		With learning		
	Keyword-based	Pattern-based	Structural	Stream of chars	Vector space	Structural	Logic	Algebra	Graph theories	Probability theories	Fuzzy set theories	Neural network	Symbolic learning	Genetic algorithms
CORA	X					X			X					
TACHIR						X			X					
SIFT	X				X			X						
NewsWeeder	X				X			X						
grep		X		X				X						
LEXA		X		X				X						
OCP		X		X				X						
IBM Text miner														
Metacrawler														
Altavista, Scooter														
INQUERY	X				X					X				
SMART	X				X			X						
ILA	X		X			X	X							
WebLearner	X				X					X				
Yahoo directory														
NewsSieve														
Inktomi, Slurp														
Isearch	X		X		X			X						
Google	X				X				X	X	X			
ResearchIndex	X				X	X		X	X					
Glimpse, Agrep	X			X	X			X						
Scatter/Gather	X				X			X	X					
Amalthaea	X				X				X					X
Excite, ArchitextSpider														
Infoseek, Sidewinder														
Northernlight, Guliver														
WEBSOM	X				X			X				X		
Infind														
Lycos														
GeoSearch														
LEXIS-NEXIS														

Table 2. Vertical projections.

areas. Indeed, objects can themselves be classified with respect to the vertical components, namely representation and reasoning. We call this further classification of an IR object the vertical projection of the object; Tab. 2 shows the vertical projection for the IR objects referred to in the Appendix. Note that a few rows in the table are left blank, as we were not able to access the information needed to produce the vertical projections of the related objects.

In recent years, information retrieval has assumed an increasing importance because of the dramatic growth of the amount of information available in digital formats. The proliferation of information retrieval algorithms, methods,

technologies, and tools calls for the definition of basic concepts and terminology; this is useful to assess the features and the characteristics of each IR object and to understand the relationships that exist between the objects. In this paper we have proposed a taxonomy of IR objects, accompanied with definitions for the key terms. This taxonomy is a tentative first step in classifying IR models and tools, since it does not cover all aspects of IR. The market and the development of IR technologies are still evolving and this evolution will make some observations contained in this paper obsolete. As a result, this work will need to be updated incrementally as the technology develops. However, we think that the taxonomy presented in this paper provides a good starting point for such a continuous updating.

One of the main limitations of the taxonomy presented in this paper is the fact that it covers only text information retrieval. Indeed, current information needs require more and more integrated retrieval models and tools that combine the traditional retrieval of text documents with the retrieval of multimedia content, such as images and speech, and even structured data from databases. Therefore, there is room for improvement of the proposed taxonomy and we are currently working on extending it in order to include other important aspects of IR not covered here, primarily the retrieval of multimedia content.

5. Acknowledgment

The work described in this paper has been supported by the EUREKA Project E!2235, IKF – Information and Knowledge Fusion.

References

[1] AGOSTI, M., CRESTATI, F., TACHIR: a Tool for the Automated Construction of Hypertexts in Information Retrieval, *Proceedings of RIAO, Rockfeller University*, (1994), NewYork (USA).

[2] ANANDEEP S., SYCARA, P.K., A Learning Personal Agent for Text Filtering and Notification, *Proceedings of the International Conference of Knowledge Based Systems*, (1996), (`http://www.ri.cmu.edu/pubs/pub_2174.html`).

[3] ANDERBERG, M.R., *Cluster analysis for applications*, Academic Press, NewYork, 1973.

[4] BAEZA-YATES, R., GONNET, G., Efficient text searching of regular expressions, *Proceedings of the 16th International Colloquium on Automata, Languages and Programming*, LNCS **372**, (1989), pp. 46–62, Berlin (Germany).

[5] BAEZA-YATES, R., NAVARRO, G., Fast approximate string matching, *Algorithmica*, **23(2)**, (1999), pp. 127–158.

[6] BEERI, C., KORNATZKY, Y., A logical query language for hypertext systems, *Proceedings of the European Conference on Hypertext*, (1990), pp. 67–80, Versailles, (France).

[7] BELEW, R.K., Adaptative information retrieval, *Proceedings of the 12th Annual International ACM/SIGIR Conference on Research and Development in information Retrieval*, (1989), pp. 11–20, Cambridge (MA).

[8] BERND T., Logic Programs for Intelligent Web Search, *Proceedings of the 11th International Symposium on Methodologies for Intelligent Systems*, (1999), LNAI **1609**, Warsaw, (Poland).

[9] BLOSSEVILLE, M.J., HEBRAIL, G., MONTEIL, M.G., PENOT, N., Automatic document classification: natural language processing, statistical analysis, and expert system techniques used together, *Proceedings of the 15th Annual International ACM/SIGIR Conference on Research and Development in information Retrieval*, (1992), pp. 51–57, Copenhagen (Denmark).

[10] BOOKSTEIN A., Fuzzy request: an approach to weighted Boolean searches, *Journal of the American Society for Information Science*, **31**, (1980), pp. 240–247.

[11] BORDOGNA, G., PASI, G., A Fuzzy Linguistic Approach Generalizing Boolean Information Retrieval; a Model and Its Evaluation, *Journal of the American Society for Information Science*, **44**, (1993), pp. 70–82.

[12] BRIN, S., PAGE, L., MOTWANI, R., WINOGRAD, T., The PageRank Citation Ranking: Bringing Order to the Web, *Technical report*, Stanford University, 1998.

[13] BROGLIO, J., CALLAN, J.P., CROFT, W.B., NACHBAR, D.W., Document retrieval and routing using INQUERY system, *Proceedings of the 3rd Retrieval Conference TREC*, (1995), pp. 29–38, Gaithersburg (Maryland).

[14] CALLAN, J., Document filtering with inference network. *Proceedings of the 19th Annual Int. ACM SIGIR Conference on Research and Development in Information Retrieval*, (1996), pp. 262–269, Zurich (Switzerland).

[15] CHANG, S.J., RICE, R.E., Browsing: a multidimensional framework, *Annual Review of Information Science and Technology*, **28**, (1993), pp. 231–276.

[16] CHEN, H., Machine learning for information retrieval: neural networks, Symbolic learning, and genetic algorithms, *Journal of the American Society for Information Science*, **46(3)**, (1995), pp. 194–216.

[17] CHEN, H. LYNCH, K.J., BASU, K., NG.,D.T., Generating, integrating, and activating thesauri for concept-based document retrieval, *IEEE EXPERT, Special Series on Artificial Intelligence in Text-based Information Systems*, **8(2)**, (1993), pp. 25–34.

[18] CHEN, H., SHE, L., Inductive query by examples (IQBE): A machine learning approach, *Proceedings of the 27th Annual International Conference on System Sciences, Information Sharing and Knowledge Discovery Track*, (1994), Maui (Hawaii).

[19] COOPER, W.S., GEY, F.C., DABNEY, D.P., Probabilistic retrieval based on staged logistic regression, *Proceedings of the 15th Annual Int. ACM SIGIR Conference on Research and Development in Information Retrieval*, (1992), pp. 198–210, Copenhagen (Denmark).

[20] CROFT, W.B., Approaches to intelligent information retrieval, *Information Processing and Management*, **23(4)**, (1987), pp. 249–254.

[21] CUTTING, D.R., PEDERSEN, J.O., KARGER, D., TUKEY, J.W., Scatter/gather: a cluster-based approach to browsing large document collections, *Proceedings of the 15th Annual Int. ACM SIGIR Conference on Research and Development in Information Retrieval*, (1992), pp. 318–329, Copenhagen (Denmark).

[22] DAMASHEK, M., Gauging similarity with n-grams: Language-independent categorization of text, *Science*, **267**, (1995), pp. 843–848.

[23] DOSZKOCS, T.E., REGGIA, J., LIN, X., Connectionist models and information retrieval, *Annual Review of Information Science and Technology*, **25**, (1990), pp. 209–260.

[24] FLAKE, G.W., LAWRENCE, S., GILES, C.L., COETZEE, F.M., Self Organization and Identification of Web Communities, *Journal of the IEEE Computer Society*, **35(3)**, (2002), pp. 66–71.

[25] FOX, E. A., Extending the Boolean and vector space models of information retrieval with P-norm queries and multiple concept types, *PhD thesis*, Cornell University, 1983.

[26] FUHR, N., HARTMANN, S. KNORZ, G., LUSTIG, G., SCHWANTNER, M., TZERAS, K., AIR/X – a rule-based multistage indexing system for large subject fields, *Proceedings of the 8th National Conference on Artificial Intelligence*, (1990), pp. 789–895, Boston (MA).

[27] FURNAS, G. W., DEERWESTER, S., DUMAIS, S. T., LANDAUER, T.K., HARSHMAN, R.A., STREETER, L.A., LOCHBAUM, K.E., Information retrieval using a singular value decomposition model of latent semantic structure, *Proceedings of the 11th Annual Int. ACM SIGIR Conference on Research and Development in Information Retrieval*, (1998), pp. 257–265, Grenoble (France).

[28] GARFIELD, E., *Citation Indexing: Its Theory and Application in Science*, John Wiley & Sons, NewYork, 1979.

[29] GORDON, M., Probabilistic and genetic algorithms for document retrieval, *Comunication of the ACM*, **31(10)**, (1988), pp. 1208–1218.

[30] GORDON, M.D., User-based document clustering by redescribing subject descriptions with a genetic algorithm, *Journal of the American Society for Information Science*, **42(5)**, (1991), pp. 311–322.

[31] GUARINO, N., MASOLO, C., VETERE, G., Ontoseek: Content-Based access to the web, *IEEE Intelligent Systems*, **14(3)**, (1999), pp. 70–80.

[32] HAINES, D., CROFT, W.B., Relevance feedback and inference networks, *Proceedings of the 16th Annual Int. ACM SIGIR Conference on Research and Development in Information Retrieval*, (1993), pp. 2–11, Pittsburgh (USA).

[33] KLEINBERG, J.M., Authoritative Sources in a Hyperlinked Environment, *Proceedings of the 9th Annual Int. ACM SIAM Symposium on Discrete Algorithms*, (1998), pp. 668–677, New York (USA).

[34] KOZA, J.R., *Genetic Programming: On the Programming of Computers by Means of Natural Selection*, The MIT Press, Cambridge, MA, 1992.

[35] KRAFT, D., BUEL, D.A., Fuzzy sets and generalized Boolean retrieval systems, *International Journal of Man-machine Studies*, **19**, (1983), pp. 45–56.

[36] KRAFT, D., PETRY, F.E., BUCKLES, B.P., SADASIVAN, T., The use of genetic programming to build queries for information retrieval, *IEEE Symposium on Evolutionary Computation*, (1994), pp. 468–473, Orlando (USA).

[37] KRAFT, D.H., BORDOGNA, G., PASI, G., Fuzzy set techniques in information retrieval, in J. Bezdek, D. Dubois and H. Prade (eds), *Fuzzy Sets in Approximate Reasoning and Information Systems*, **3(8)**, (1999), pp. 469–510, Kluwer Academic Publishers.

[38] KUHLTHAY, C. C., Inside the search process: Information seeking from the user's perspective, *Journal of the American Society for Information Science*, **42(5)**, (1991), pp. 361–371.

[39] KWOK, K.L., A neural network for probabilistic information retrieval, *Proceedings of the 12th Annual Int. ACM SIGIR Conference on Research and Development in Information Retrieval*, (1989), pp. 202–210, Cambridge (USA).

[40] LAMPORT, L., *LaTeX: A document Preparation System*, User's guide and Reference manual; 2nd edition, Prentice Hall, 1994.

[41] LAYAIDA, R., BOUGHANEM, M. CARON, A., Constructing an information retrieval system with neural networks, *Lecture Notes in Computer Science*, **856**, (1994), pp. 561–570.

[42] LEE, J.H., Properties of extended boolean models in information retrieval, *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, (1994), pp. 182–190.

[43] LEWIS, D.D., Learning in intelligent information retrieval, *Proceedings of the 8th International Workshop on Machine Learning*, (1991), pp. 235–239, Morgan Kaufmann.

[44] LIN, X., SOERGEL, D., MARCHIONINI, G., A self-organizing semantic map for information retrieval, *Proceedings of the 14th Annual Int. ACM SIGIR Conference on Research and Development in Information Retrieval*, (1991), pp. 262–269, Chicago (IL).

[45] LUHN, H.P., A statistical approach to mechanized encoding and searching of library information, *IBM Journal of Research and Development*, **1**, (1957), pp. 309–317.

[46] MACLEOD, K.J., ROBERTSON, W., A neural algorithm for document clustering, *Information Processing & Management*, **27(4)**, (1991), pp. 337–346.

[47] MCCUNE, B., TONG, R., DEAN, J.S., SHAPIRO, D., Rubric: a system for rule-based information retrieval, *IEEE Transaction on Software Engineering*, 1985, **11(9)**.

[48] MIYAMOTO, S., NAKAYAMA, K., Fuzzy information retrieval based on a fuzzy pseudo thesaurus, *IEEE Transactions on Systems and Man Cybernetics*, 1986, **16(2)**, pp. 278–282.

[49] MIYAMOTO, S., TERUHISA, M., KAZUHIKO, N., Generation of a Pseudothesaurus for Information Retrieval base co-occurrences and fuzzy set operations, *IEEE Transaction Systems, Man and Cybernetics*, **13(1)**, (1983), pp. 62–69.

[50] MIZZARO, S., A cognitive analysis of information retrieval, *Proceedings of CoLIS2*, (1996), pp. 233–250, Copenhagen (Denmark).

[51] MIZZARO, S., How many relevancies in information retrieval?, *Interacting with Computers*, **10(3)**, (1998), pp. 305–322.

[52] NAEZA-YATES, R., RIEBEIRO-NETO, B., *Modern Information Retrieval*, Addison Wesley, New York, 1999.

[53] OGAWA, Y., MORITA, T., KOBAYASHI, K., A fuzzy document retrieval system using the keyword connection matrix and a learning method, *Fuzzy Sets and Systems*, **39**, (1991), pp. 163–179.

[54] PAIJMANS, H., Explorations in the document vector model of information retrieval, *Dissertation*, Tilburg University, 1999. `http://pi0959.kub.nl:2080/Paai/Bibliogr/`

[55] PIROLLI, P., PITKOW, J., RAO, R., Silk from Sow's Ear: Extracting Usable Structures from the web, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, (1996), pp. 118–125, New York (USA).

[56] QUINLAN, J.R., Learning efficient classification procedures and their application to chess and games, *Machine Learning, an Artificial Intelligence Approach*, (1983), pp. 463–482, Tioga Publishing company, Palo Alto, CA.

[57] RIBEIRO-NETO, B.A., MUNTZ, R., A Belief network model for IR, *Proceedings of the 19th Annual Int. ACM SIGIR Conference on Research and Development in Information Retrieval*, (1996), pp. 253–260, Zurich (Switzerland).

[58] RIJSBERGEN, C.J., *Information Retrieval*, Butterworths, London, 1979.

[59] ROBERTSON, S.E., SPARCK JONES, K., Relevance weighting of search terms, *Journal of the American Society for Information Sciences*, **27(3)**, (1976), pp. 129–146.

[60] ROBINS, D., Interactive Information Retrieval: Context and Basic Notions, *Information Science*, **3(2)**, (2000), pp. 57–61.

[61] ROCCHIO, J.J., *Relevance Feedback in Information Retrieval*, Prentice Hall, 1971.

[62] RUMELHART, D.E., HINTON, G.E., WILLIAMS, R.J., Learning Internal Representations by Error Propagation, *Parallel Distributed Processing*, (1986), pp. 318–362, The MIT Press, Cambridge, MA.

[63] SACHS W.M., An approach to associative retrieval through the theory of fuzzy sets, *Journal of the American Society for Information Sciences*, (1976), pp. 85–87.

[64] SALTON, G., *The SMART Retrieval System – Experiments in Automatic Document Processing*, Prentice Hall, New York, 1971.

[65] SALTON, G., *Automatic text processing: The transformation, analysis, and retrieval of information by computer*, Addison-Wesley, 1989.

[66] SALTON, G., BUCKLEY C., Term weighting approaches in automatic retrieval, *Information Processing and Management*, **24(5)**, (1988), pp. 513–523.

[67] SARACEVIC, T., RELEVANCE: A Review of and a Framework for the thinking of the notion in information science, *Journal of the American Society for Information Science*, **26(6)**, (1975), pp. 321–343.

[68] SEBASTIANI, F., On the Role of Logic in Information Retrieval, *Information Processing & Management*, **34(1)**, (1998), pp. 1–18.

[69] SMITH, L.C., WARNER, A.J., A taxonomy of representation in information retrieval design, *Journal of Information Science*, **8**, (1984), pp. 113–121.

[70] TAHANI, V.A., A fuzzy model of document retrieval systems, *Information Processing and Management*, **12**, (1976), pp. 177–187.

[71] TURTLE, H., CROFT. W.B., Inference networks for document retrieval, *Proceedings of the 13th Annual Int. ACM SIGIR Conference on Research and Development in Information Retrieval*, (1990), pp. 1–24, Brussels (Belgium).

[72] UTGOFF, P.E., Incremental induction of decision trees, *Machine Learning*, **4**, (1989), pp. 161–186.

[73] VOORHESS, E.M., HARMAN, D., *Overview of TREC 2001*, National Institute of Standards and Technology, 2001.

[74] WALLER, W.G., KRAFT, D.H., A Mathematical Model of a Weighted Boolean Retrieval System, *Information Processing & Management*, **15(5)**, (1979), pp. 235–245.

[75] WILKINSON, R., HINGSTON, P., Using the cosine measure in neural network for document retrieval, *Proceedings of the 14th Annual Int. ACM SIGIR Conference on Research and Development in Information Retrieval*, (1991), pp. 202–210, Chicago (USA).

[76] WONG, S.K.M., YAO, Y.Y., On modeling information retrieval with probabilistic inference, *ACM Transactions on Information Systems*, **13(1)**, (1995), pp. 39–68.

[77] WONG, S.K.M., ZIARKO, W., RAGHAVAN, V.V., WONG, P.C.N., On Extending the Vector Space Model for Boolean Query Processing, *Proceedings of the 9th Annual Int. ACM SIGIR Conference on Research and Development in Information Retrieval*, (1986), pp. 175–185, Pisa (Italy).

[78] WONG, S.K.M., ZIARKO, W., WONG, P.C.N., Generalized vector space model in information retrieval, *Proceedings of the 8th Annual Int. ACM SIGIR Conference on Research and Development in Information Retrieval*, (1985), pp. 18–25, New York (USA).

[79] WU, S., MANBER, U., Agrep: a fast approximate pattern matching tool, *Proceedings of USENIX Technical Conference*, (1992), pp. 153–162, San Francisco (USA).

[80] XML eXtensible Markup Language 1.0 (Second Edition) W3C Recommendation 6 October 2000. `http://www.w3.org/XML/`

[81] XPath XML Path Language 1.0 W3C Recommendation 16 November 1999. `http://www.w3.org/TR/xpath`

[82] YANNAKOUDAKIS, E.J., GOYAL, P., HUGGIL, J.A., The generation and use of text fragments for data compression, *Information Processing and Management*, **18**, (1982), pp. 15–21.

[83] ZIPF, H.P., *Human Behaviour and the Principle of Least Effort*, Addison-Wesley, Cambridge, 1949.

[84] CORA. `http://cora.whizbang.com`

[85] TACHIR. `http://www.dei.unipd.it/~ims/tachir.html`

[86] SIFT. `ftp://db.stanford.edu/pub/sift/sift-1.1-netnews.tar.Z`

[87] NewsWeeder. `http://anther.learning.cs.cmu.edu/ifhome.html`

[88] Grep. `http://www.gnu.org`

[89] LEXA. `http://nora.hd.uib.no/lexainf.html`

[90] OCP. `http://info.ox.ac.uk/ctitext/resguide/resources/o125.html`

[91] IBM text miner. `http://www.ibm.com`

[92] Metacrawler. `http://www.metacrawler.com/`

[93] Altavista. `http://www.altavista.com`

[94] Scooter. `http://www.altavista.com`

[95] INQUERY. `http://www-ciir.cs.umass.edu`

[96] SMART. `ftp://ftp.cs.cornell.edu/pub/smart/`

[97] ILA. Internet Learning Agent. `http://www.cs.washington.edu/homes/map/ila.html`

[98] WebLearner. `http://www.ics.uci.edu/~pazzani/Coldlist.html`

[99] Yahoo Directory. `http://www.yahoo.com`

[100] NewsSieve. `http://www.newssieve.com/`

[101] Inktomi. `http://www.inktomi.com`

[102] Slurp. `http://www.inktomi.com`

[103] Isearch. `http://www.cnidr.org/isearch.html`

[104] Google. `http://www.google.com`

[105] ResearchIndex. `http://www.researchindex.com`

[106] Agrep, Glimpse. `http://glimpse.cs.arizona.edu/`

[107] Scatter/Gather: `http://www.sims.berkeley.edu/~hearst/sg-overview.html`

[108] Amalthaea. `http://lcs.www.media.mit.edu/~moux/papers/PAAM96/PAAM96.html`

[109] Excite. `http://www.excite.com`

[110] ArchitextSpider. `http://www.excite.com`

[111] Infoseek. `http://www.infoseek.com`

[112] Sidewinder. `http://www.infoseek.com`

[113] Northern Light. `http://www.northernlight.com`

[114] Guliver. `http://www.northernlight.com`

[115] WEBSOM. `http://websom.hut.fi/websom`

[116] Infind. `http://www.infind.com`

[117] Lycos. `http://www.lycos.com`

[118] GeoSearch. `http://www.northernlight.com`

[119] LEXIS-NEXIS. `http://www.lexis-nexis.com`

Appendix: Horizontal Taxonomy of a Set of Information Retrieval Objects

Information Retrieval Object	Description	Ref	Tasks									Form		Context	
			Ad hoc retrieval	Known item search	Interactive retrieval	Filtering	Browsing	Clustering	Mining	Gathering	Crawling	Service	Tool	General purpose	Specific context
CORA	Computer Science Research Paper Search Engine. It provides access through a topic hierarchy and Boolean query mechanism. Using text classification technique each paper is automatically placed into a topic leaf through a learning mechanism.	[84]	X	X				X				X			X
TACHIR	TACHIR stands for "a Tool for the Automatic Construction of Hypertexts for Information Retrieval". It builds a hypertext for Information Retrieval starting from a document collection.	[85]					X	X					X		
SIFT	The Stanford Information Filtering Tool was developed by Tak Yan and Hector Garcia-Molina. It is currently being used to filter USENET articles. A user submits a profile giving keywords in articles they would like to read. When a new article comes in, it is matched against all the profiles	[86]	X			X							X	X	
NewsWeeder	A system created by Ken Lang at Carnegie Mellon for filtering USENET articles. This program examines how a user rates articles, and attempts to predict the rating for new articles. It also uses information gain and singular value decomposition to choose terms and reduce the feature space for classification.	[87]	X		X	X							X	X	
grep	GREP, Global Regular Expression Print. It comes from the ed command to print all lines matching a certain pattern: g/re/p where "re" is a "regular expression".	[88]	X										X	X	
LEXA	Corpus Processing Software. It allows automatic lemmatization of any input ASCII texts, creation of frequency lists of the types and tokens occurring in any loaded text, and generation of lexical density tables.	[89]							X				X		
OCP	The Oxford Concordance Program (OCP) is a tool for generating concordances, word lists, and indexes from texts in any language or alphabet. OCP operates on an ASCII file of the text and up to 8 characters may be defined to represent one letter.	[90]							X				X		
IBM Text miner	A general purpose Information Retrieval System that additionally offers specific task feature such as clustering, filtering, mining, gathering and crawling.	[91]	X			X		X	X	X	X		X	X	
Metacrawler	Web search engine that takes results from a variety of other search engines and merges them.	[92]	X							X		X		X	
Altavista, Scooter	A general purpose retrieval system. It is available as a web search engine service and a software tool. Famous for its fast crawling and indexing engine.	[93] [94]	X							X	X	X	X	X	
INQUERY	Product of the Center for Intelligent Information Retrieval at the University of Massachusetts. INQUERY is based on a probabilistic model that uses a Bayesian network architecture. It has two parts: a document network and a query network. The document net is static for a given collection. To perform retrieval, the system connects these two networks together, and can thus calculate the conditional probability that the information need is satisfied given each document.	[95]	X										X	X	
SMART	SMART was developed by Gerard Salton of Cornell University. It uses a vector space model for representing documents. SMART performs automatic indexing by removing stopwords from a predetermined list, stemming via suffix deletion, and weighting. Given a new query, it converts it to a vector, and then uses a similarity measure to compare it to the documents in the vector space. SMART ranks the documents, and returns the top n, where n is a number determined by the user. SMART can perform relevance feedback based on the result of the retrieval	[96]	X		X								X	X	

Information Retrieval Object	Description	Ref	Tasks									Form		Context	
			Ad hoc retrieval	Known item search	Interactive retrieval	Filtering	Browsing	Clustering	Mining	Gathering	Crawling	Service	Tool	General purpose	Specific context
ILA	ILA, or Internet Learning Agent, was written by Oren Etzioni and Mike Perkowitz at the University of Washington. The ILA analyzes an information source, such as a finger server. It gives a query to the server, and then tries to determine the category of the data that it receives.	[97]						X		X			X		
WebLearner	This project originates from the University of California. It observes the user. The user puts pages on either a hotlist (if good) or coldlist (if bad). The system then tries to analyze those examples to rate future pages.	[98]	X										X	X	
Yahoo Directory	One of the first internet web search engine is based principally on a manual indexing and categorizing process in which every web resource are first evaluated by experts and then inserted in the database.	[99]					X	X				X		X	
NewsSieve	NewsSIEVE is a client server based software used to filter messages from UseNet according to the user interest. The central server searches for appropriate messages which are submitted to the client software	[100]	X			X						X	X		
Inktomi, Slurp	Developed by Infoseek Search Technology. It is also called Ultraseek Server.	[101] [102]	X							X	X	X	X	X	
Isearch	It is an opensource project developed in 1994. Many commercial search engines are based on the source code of this product. It is based on the probabilistic model.	[103]	X										X	X	
Google	It is a web search engine based on page popularity measured in links to it from other pages: high rank if a lot of other pages link to it. It also have automatic fuzzy AND expansion.	[104]	X							X	X	X		X	
ResearchIndex	It is a scientific literature digital library with full source code available. It allows browsing the database using citation links and can show the context of citations to a given paper. ResearchIndex indexes the full-text of the entire articles and citations and allows full boolean, phrase and proximity search.	[105]	X	X			X					X	X		X
Glimpse, Agrep	Indexing and query tool that allows to search through local and/or web file systems. Glimpse supports most of agrep's options (agrep is a powerful version of grep, and it is part of glimpse) including approximate matching (misspelled words), Boolean queries, and even some limited forms of regular expressions.	[106]	X										X	X	
Scatter/Gather	Scatter/Gather interface uses text clustering as a way to group document according to the overall similarities in their content. Scatter/Gather is so named because it allows the user to scatter documents into clusters, or groups, then gather a subset of these groups and re-scatter them to form new groups	[107]	X		X			X					X		
Amalthaea	It is an experimental tool for information discovery and information filtering. The main Idea come from the field of autonomous agents and artificial life. It is principally based on an evolving ecosystem composed of competing and cooperating agents.	[108]	X			X			X	X			X		
Excite, ArchitextSpider	General purpose web search engine.	[109] [110]	X							X	X	X	X	X	
Infoseek, Sidewinder	General purpose web search engine.	[111] [112]	X							X	X	X	X	X	
Northernlight, Guliver	General purpose web search engine with a patented classification and precision relevancy ranking function.	[113] [114]	X							X	X	X	X	X	

Information Retrieval Object	Description	Ref	Tasks									Form		Context	
			Ad hoc retrieval	Known item search	Interactive retrieval	Filtering	Browsing	Clustering	Mining	Gathering	Crawling	Service	Tool	General purpose	Specific context
WEBSOM	WEBSOM is a method for organizing miscellaneous text documents onto meaningful graphical maps for exploration and search. It uses a neural networks approach to find similarities between documents.	[115]						X				X			
Infind	Web search engine that takes results from a variety of other search engines and merges them.	[116]	X							X		X		X	
Lycos	General purpose web search engine.	[117]	X							X	X	X		X	
GeoSearch	GeoSearch is a special purpose search engine to find local web sites with addresses for information about professional services, reviews, local businesses, publishers and products anywhere in the US or Canada. It can be used to search for example hospitals, insurance, restaurants relevant to a zip code area	[118]	X	X								X			X
LEXIS-NEXIS	LexisNexis offers services for legal, business, academic, and government professionals. One of its service is a special purpose online search engine for legal and business documentation.	[119]	X	X								X			X

Received: September, 2002
Revised: January, 2004
Accepted: May, 2004

Contact address:
Gerardo Canfora
Research Centre on Software Technology
Department of Engineering
University of Sannio
Palazzo ex Poste – Via Traiano
82100 Benevento
ITALY
e-mail: gerardo.canfora@unisannio.it

GERARDO CANFORA received the Laurea degree in electronic engineering from the University of Naples, Federico II, Italy, in 1989. He is currently a full professor of computer science at the Faculty of Engineering and the Director of the Research Centre on Software Technology (RCOST) of the University of Sannio in Benevento, Italy. From 1990 to 1991, he was with the Italian National Research Council (CNR). During 1992, he was at the Department of Informatica e Sistemistica of the University of Naples, Federico II, Italy. From 1992 to 1993, he was a visiting researcher at the Centre for Software Maintenance of the University of Durham, UK. In 1993, he joined the Faculty of Engineering of the University of Sannio in Benevento, Italy. He has served on the program committees of a number of international conferences. He was a program co-chair of the 1997 International Workshop on Program Comprehension and of the 2001 International Conference and the General Chair of the 2003 European Conference on Software Maintenance and Reengineering. His research interests include software maintenance, program comprehension, reverse engineering, workflow management, document and knowledge management, and information retrieval. He serves on the Editorial Board of the IEEE Transactions on Software Engineering. He is a member of the IEEE and the IEEE Computer Society.

LUIGI CERULO received the Laurea degree in computer engineering from the University of Sannio, Italy, in 2001. He is currently an assistant researcher at the Research Centre on Software Technology (RCOST) of the University of Sannio in Benevento, Italy. His research interests include information retrieval, fuzzy logic, and visual languages.