Deliverable D16.3: MPA3 Cross-language retrieval and access tools

DOCUMENT IDENTIFIER	PS_WP16UTV6-3_Cross-language_retrieval_and_access_tools
DATE	20/3/2006
ABSTRACT	This investigates the state of the art and tools for cross-lingual information retrieval and discusses their feasibility and practical usability.
KEYWORDS	Cross-language applications, Conceptual retrieval, Natural Language Processing, Information Retrieval, Semantic Web, Ontologies, Metadata Extraction, Database, Categorization.
WORKPACKAGE / TASK	WP 16
AUTHOR, COMPANY	Roberto Basili (UTV) Alessandro Moschitti (UTV) Marco Cammisa (UTV) Emanuele Donati (UTV), Borislav Popov (Ontotext)
NATURE	Report
DISSEMINATION	Public

DOCUMENT HISTORY

Release	Date	Reason of change	Status	Distribution
0.1	2004-05-12	First Draft	Living	Confidential
0.2	2005-03-02	Second Note on the First Draft	Living	Confidential
1.0	2005-07-05	First extended DRAFT	Living	Confidential
1.1	2006-01-15	Sent to the TF6 MAD partners	Living	Confidential
1.3	2006-03-11	First Final version	Living	Confidential
1.4	2006-03-20	Final version	Delivered	Restricted
1.5	2008-03-20	Dissemination status changed to Public	Complete	Public

Contents

Executive Summary

This deliverable is a survey of tools and techniques for cross-linguistic information retrieval and extraction and their applicability as best practices in the area of metadata extraction and retrieval of audiovisual material.

It will describe in detail the state-of-art of methods, technologies and software tools that implement/support cross-lingual forms of retrieval and extraction from textual material. It will include also some general guidelines for the technological solutions best suited for PrestoSpace and recommendation for the definition of one (or more) candidate test-bed(s) in the assessment of the MAD system.

Motivations and Scope

This part of the deliverable will discuss main achievements of state-of-art information access technologies, such as Information Retrieval, Information Extraction and Question Answering, that are considered relevant for the MAD system.

Overview

As globalization is emerging, information access across language boundaries is becoming a critical issue. The World Wide Web has become accessible to more and more countries and technological advances overcome the network, interface and computer system differences which have impeded information access. Consequently, it is now more common for searchers to wish to explore collection of documents that are not written in their native languages. Systems that helps in such kind of tasks have been developed in the Information Retrieval area.

Information Retrieval (IR) is the discipline that deals with retrieval of unstructured data, especially textual documents, in response to a query or topic statement, which may itself be unstructured, e.g., a sentence or even another document, or which may be structured, e.g., a Boolean expression. Documents are the items that we want to retrieve and they are typically described by collections of terms. Usually a document is thought as a piece of text, most likely in a machine readable form, and a term as a word or phrase which helps to describe the document, and which may indeed occur in the document, once or several times. For example, a document might be about car dealers, and could be described by corresponding terms “car”, “price”, “speed”, “paint”, “brake”, “ABS” and so on. Terms can be any feature that helps to describe documents, e.g. “ABS” or the number of car speeds are useful to select certain class of cars.

Such models for the automatic indexing and retrieving of textual documents are based on the assumption that documents and user queries can be represented by the set of their terms. Additionally, weights or probabilities are assigned to such terms to produce a list of answers ranked according to their relevance to the user query.

Broadly, there are two major categories of IR technology and research: statistical and semantic. In statistical approaches, the documents that are retrieved or that are highly ranked are those that match the query most closely in terms of some statistical measures. Semantic or conceptual approaches attempt to implement some degree of syntactic and semantic analysis; in other words, they try to reproduce to some (perhaps modest) degree the understanding of the natural language text that a human user would provide.

Although the IR approaches can be adopted for any individual language they cannot be used to retrieve documents that are expressed in a language different from the query. Thus, beyond merely accepting extended character sets and performing language identification, the information retrieval systems should be able to provide help in search for information across language boundaries.

A Cross Language Information Retrieval (CLIR) system retrieves documents in a language that is different from the query language. A user of a CLIR system can enter a query in one language, but the

returned documents will be in the language of the document collection. One example showing the usefulness of such a system is when a user might have some knowledge of the document language but has difficulty formulating effective queries. These users might very well be able to distinguish good documents form bad documents based on their limited knowledge and could then send the documents that they have judged relevant on to a translation service bureau or machine translation system.

The above example shows that CLIR systems are strongly based on traditionally IR. Indeed, the most part of the work in CLIR relates on the use of traditional IR along with machine translation or other techniques which aim to reduce the problem from multi to mono language retrieval.

However, the word ambiguity problems make more critical the choice of an effective document representation, thus the simple bag-of-words is often not adequate. This has led researchers to explore more advanced document representations based on Natural Language Processing techniques and, more recently, on ontology and *semantic web*.

As the greatest amount of work to date has been devoted to statistical approaches, they will be presented in Section 1. Then a survey of current Cross Language Information Retrieval is presented in Section 2. Conceptual approaches are divided in two main categories: (a) those which, by applying traditional techniques of Natural Language Processing, automatically derive conceptual representations (discussed in Section 3) and (b) those which use metadata or domain specific ontologies to build conceptual structures typical of semantic web (discussed in the Section 4).

1 Pure Statistical Information Retrieval

Statistical approaches fall into a number of categories: Boolean, extended Boolean, vector space, and probabilistic models. Their main idea is that documents and queries are divided into terms. These terms are the population that is counted and measured statistically. To improve the representational power of terms, weights are assigned to them within a given document, i.e., the same term may have a different weight in each distinct document in which it occurs. The weight is usually a measure of how effective the given term is likely to be in distinguishing the given document from other documents in the given collection. Weights can also be assigned to the terms in a query. The weight of a query term is usually a measure of how much importance the term is to be assigned in computation of the similarity of documents to the given query. As with documents, a given term may have a different weight in one query than in another. The work in statistical IR relates to model term weights (or term probabilities) such that the similarity measure between the target query and documents produces the best document rank list according to the user information needs.

1.1 Basic Document Representations

In almost all document representation terms are the words that occur in a given query or collection of documents. The words often undergo pre-processing. They are “stemmed” to extract the “root” of each word. [Porter, 1980] [Porter, 1997]. The objective is to eliminate the variation that arises from the occurrence of different grammatical forms of the same word, e.g., “retrieve,” “retrieved,” “retrieves,” and “retrieval” should all be recognized as forms of the same word. Hence, it should not be necessary for the user who formulates a query to specify every possible form of a word that he believes may occur in the documents for which he/she is searching.

Another common form of preprocessing is the elimination of common words that have little power to discriminate relevant from non-relevant documents, e.g., “the”, “a” or “it”. Hence, IR engines are usually provided with a “stop list” of such “noisy” words. Note that both stemming and stop lists are language-dependent.

Some sophisticated engines also extract “phrases” as terms. A phrase is a combination of adjacent words which may be recognized by frequency of co-occurrence in a given collection or by presence in a phrase dictionary. At the other extreme, some engines break documents and queries into “*n*-grams”, i.e., arbitrary strings of *n* consecutive characters [Damashek et al., 1995]. This may be done, e.g., by moving a “window” of *n* characters in length through a document or query one character at a time.

In other words, the first *n*-gram will consist of the first *n* characters in the document, the 2nd *n*-gram will consist of the 2nd through the (*n*+1)-th character, etc. (Early research used $n = 2$, $n = 3$; recent applications have used values of $n=5$, and $n=6$ but the user is free to use the value of *n* that works best

for his application.) The window can be moved through the entire document, completely ignoring word, phrases, or punctuation boundaries. Alternatively, the window can be constrained by word separators, or by other punctuation characters, e.g., the engine can gather *n*-gram statistics separately for each word. [Zamora et al., 1981] [Suen, 1979].

Once a document representation is chosen, we need to encode it in a numeric format suitable for a computer program. The simplest approach is the Boolean encoding, i.e. 1 if a word is contained in the document and 0 otherwise. This choice impacts on the operations that a user can carry out and implicitly define what has been called as the Boolean retrieval model.

1.2 Boolean Model

In the boolean case, the query is formulated as a boolean combination of terms. A conventional boolean query uses the classical operators AND, OR, and NOT. The query "t1 AND t2" is satisfied by a given document d_1 if and only if d_1 contains both terms t1 and t2. Similarly, the query "t1 OR t2" is satisfied by d_1 if and only if it contains t1 or t2 or both. The query "t1 AND NOT t2" satisfies d_1 if and only if it contains t1 and does not contain t2. More complex boolean queries can be built up out of these operators and evaluated according to the classical rules of boolean algebra.

Such a classical boolean query is either true or false. Correspondingly, a document either satisfies such a query (is "relevant") or does not satisfy it (is non-relevant"). This is a significant limitation [Harman, 1992] since no ranking is possible. Note however that if stemming is employed, a query condition specifying that a document must contain the word "retrieve" will be satisfied by a document that contains any of the forms "retrieve", "retrieves", "retrieved", "retrieval", etc. Several kinds of refinement of this classical boolean query are possible when it is applied to IR.

First, the query may be applied to a specified syntactic component of each document, e.g., the boolean condition may be applied to the title or the abstract rather than to the document as a whole.

Second, it may be specified that the condition must apply to a specified position within a syntactic component, e.g., to the words at the beginning of the title rather than to any part of the title.

Third, an additional boolean operator may be added to the classical set, e.g. a "proximity" operator [Z39.50-1995]. A proximity operator specifies how close two terms must be in the text to satisfy the query condition. In its general form, the proximity operator specifies a unit, e.g., word, sentence, paragraph, etc., and an integer. For example, the proximity operator may be used to specify that two terms must not only both occur in a given document but must be within *n* words of each other; e.g., *n* = 0 may mean that the words must be adjacent. Similarly, the operator may specify that two terms must be within *n* sentences of each other, etc. A proximity operator can be applied to boolean conditions as well as to simple terms, e.g., it might specify that a sentence satisfying one boolean condition must be adjacent to a sentence satisfying some other boolean conditions. A proximity operator may specify order as well as proximity, e.g., not only how close two words must be but in what order they must occur.

Finally, the classical boolean approach does not use term weights. Or, what comes to the same thing, it uses only two weights, zero (a term is absent) and one (a term is present). The classical boolean model can be viewed as a crude way of expressing phrase and thesaurus relationships. For example, t1 AND t2 says that both terms t1 and t2 must be present, a condition that is applicable if the two terms form a phrase. If a proximity operator is employed, the boolean condition can be made to say that t2 must immediately follow t1 in the text, which corresponds still more closely (though still crudely) to the conventional meaning of a "phrase." Similarly, t1 OR t2 says that either t1 or t2 can serve as an index term to relevant documents, i.e., in some sense t1 and t2 are "equivalent". This is roughly speaking what we mean when we assign t1 and t2 to the same class in a thesaurus. In fact, some systems use this viewpoint to generate expanded boolean conditions automatically, e.g., given a set of query terms supplied by the user, a boolean expression is composed by query terms in OR with any stored synonyms and then by putting in AND these clusters together [Anick, 1994].

1.2.1 Advanced Boolean Model

Even with the addition of a proximity operator, boolean conditions remain classical in the sense that they are either true or false. Such an all-or-nothing condition tends to have the effect that either a large number of documents or none at all are retrieved [Harman,1992].

Classical boolean models also tend to produce counter-intuitive results because of this all-or-nothing characteristic, e.g., in response to a multi-term OR, "a document containing all [or many of] the query

terms is not treated better than a document containing one term" [Salton et al.,1988]. Similarly, in response to a multi-term AND, "a document containing all but one query term is treated just as badly as a document containing no query term at all" [Salton et al., 1988]. A number of extended boolean models have been developed to provide ranked output, i.e., provide output such that some documents satisfy the query condition more closely than others [Lee, 1994]. These extended boolean models employ extended boolean operators (also called "soft boolean" operators).

Extended boolean operators make use of the weights assigned to the terms in each document. A classical boolean operator evaluates its arguments to return a value of either true or false. These truth values are often represented numerically by zero (false — or in IR terms "doesn't match given document") and one (true — or in IR terms "matches given document"). An extended Boolean operator evaluates its arguments to a number in the range zero to one, corresponding to the estimated degree to which the given logical expression matches the given document. [Lee, 1994] has examined a number of extended boolean models [Paice, 1984] [Waller et al., 1979] [Zimmerman, 1991] and proved that by certain significant (but not necessarily the only significant) criteria, a model called "p-norm" [Salton et al., CACM 1983] has the most desirable properties.

By "most desirable" is meant that the p-norm model tends to evaluate the degree to which a document matches (satisfies) a query more in accordance with a human user's judgment than the other models. For each of the other models examined, there are cases where the model's evaluation of the degree of query/document match is at variance with a human user's intuition. In each of those cases, the p-norm model's evaluation of match agrees with a human user's intuition.

A refining of degree of user satisfaction has been obtained with the introduction of the Vector Space Model, in which the idea of similarity between document and query replaces the exact matching operation of the Boolean approach.

1.3 Vector Space Model

One common approach to document representation and indexing for statistical purposes is to represent each textual document as a set of terms. This defines a "space" such that each distinct term represents one dimension in that space. Since we are representing each document as a set of terms, we can view this space as a "document space". [Salton, 1983] [Salton, 1989].

We can then assign a numeric weight to each term in a given document, representing an estimate (usually but not necessarily statistical) of the usefulness of the given term as a descriptor of the given document, i.e., an estimate of its usefulness for distinguishing the given document from other documents in the same collection. It should be stressed that a given term may receive a different weight in each document in which it occurs; a term may be a better descriptor of one document than of another. A term that is not in a given document receives a weight of zero in that document.

The weights assigned to the terms in a given document d can then be interpreted as the coordinates of d in the document space; in other words, d is represented as a point in document space. Equivalently, we can interpret d as a vector from the origin of document space to the point defined by d's coordinates.

In document space, each document d is defined by the weights of the terms that represent it. Sometimes, it is desirable to define a "term space" for a given collection. In a term space, each document is a dimension. Each point (or vector) is a term in the given collection. The coordinates of a given term are the weights assigned to it in each document in which it occurs. As before, a term receives a weight of zero for each document in which it does not occur.

We can combine the "document space" and "term space" perspectives by viewing the collection as represented by a document-by-term matrix. Each row of this matrix is a document (in term space). Each column of this matrix is a term (in document space). The element at row i, column j, is the weight of term j in document i.

A query may be specified by the user as a set of terms with accompanying numeric weights. Or a query may be specified in natural language. In the latter case, the query can be processed exactly like a document; indeed, the query might *be* a document, e.g., a sample of the kind of document the user wants to retrieve. A natural language query can receive the usual processing, i.e., removal of "stop" words, stemming, etc., transforming it into a set of terms with accompanying weights. Hence, the query

can always be interpreted as another document in document space. Note that if the query contains terms that are not in the collection, these represent additional dimensions in document space.

More formally, documents are represented in a space D whose dimensions are the features $f_i \in F$ composing the documents $\vec{d} = < w^d_{f_1},...,w^d_{f_n} > \in D$, where $w^d_{f_i}$ is the weight of the feature f_i in the document d represented by the vector $\vec{d}$ and n= $|F|$. Queries are represented in the same document space, as if they were real documents, i.e. $\vec{q} = < w^q_{f_1},...,w^q_{f_n} > \in D$.

1.3.1 Weighting Schemes

An important question is how weights are assigned to terms either in documents or in queries. A variety of weighting schemes have been used [Salton and Buckley, 1988]. Given a large collection, manual assignment of weights is very expensive. The most successful and widely used scheme for automatic generation of weights is the product between the "term frequency" and the "inverse document frequency" scheme, commonly abbreviated tf × idf.

The "term frequency" (*tf*) is the frequency of occurrence of the given term within the given document, thus *tf* is a document-specific statistic. It varies from one document to another, attempting to measure the importance of the term within a given document. By contrast, inverse document frequency (*idf*) is a "global" statistic; *idf* characterizes a given term within an entire collection of documents. It is a measure of how widely the term is distributed over the given collection, and hence of how likely the term is to occur within any given document by chance.

The *idf* is defined as $\log\frac{N}{N_f}$ where N is the number of documents in the collection and N_f is the number of documents that contain the term (feature) f. The fewer the documents containing the given term, the larger the *idf*. If every document in the collection contains the given term, the *idf* is zero. This expresses the commonsense intuition that a term that occurs in every document in a given collection is not likely to be useful for distinguishing relevant from non-relevant documents. Or what is equivalent, a term that occurs in every document in a collection is not likely to be useful for distinguishing documents about one topic from documents about another topic.

To cite a commonly-used example, in a collection of documents about computer science or software, the term "computer" is likely to occur in all or most of the documents, so it won't be very good at discriminating documents relevant to a given query from documents that are non-relevant to the given query. However, the same term might be very good at discriminating a document about computer science from documents that are not about computer science in another collection where computer science documents are rare.

By adopting tf × idf weighting scheme, the component associate with the feature f of the vector $\vec{d}$ is computed by

$$w^d_f = (\log\frac{N}{N_f}) \times \mathrm{tf}^d_f = IDF(f) \times \mathrm{tf}^d_f$$

where tf^d_f is the number of occurrences of the feature f in the document d, N is the number of documents in the collection and N_f is the number of documents where f appears.
Such weight assumes that the best descriptors of a given document will be the terms that occur often in the given document and very rarely in other documents. Similarly, a term that occurs a moderate number of times in a moderate proportion of the documents in the given collection will also be a good descriptor. Hence, the terms that are the best document descriptors in a given collection will be terms that occur with moderate frequency in that collection. The lowest weights will be assigned to terms that occur very infrequently in *any* document (low-frequency documents), and terms that occur in most or all of the documents (high frequency documents).

1.3.2 Similarity Estimation

Once vectors have been computed for the query and for each document in the given collection, e.g., using a weighting scheme like those described above, the next step is to compute a numeric "similarity" between the query and each document.

Documents can be ranked according to how similar they are to a query, i.e., the highest ranking document is the document most similar to the query, etc. While it would be too much to hope that ranking by similarity in document vector space would correspond exactly with human judgment of degree of relevance to the given query, the hope (borne out to some degree in practice) is that the documents with high similarity will include a high proportion of the relevant documents, and that the documents with very low similarity will include very few relevant documents (this assumes that the given collection contains some relevant documents).

Ranking allows the human user to restrict his/her attention to a set of documents of manageable size, e.g., the top 20 documents, etc. The usual similarity measure employed in document vector space is the "inner product" between the query vector and a given document vector [Salton et al., 1983] [Salton, 1989]. The inner product between a query vector and a document vector is computed by multiplying the query vector component (i.e., weight), for each term f, by the corresponding document vector component weight, and summing these products over all f. Hence the inner product is given by:

$$s_{d,q} = \vec{d} \cdot \vec{q} = \sum_f w_f^d \times w_f^q$$

Documents are ranked according to their similarity to a query. If both vectors have been cosine normalized, then this inner product represents the cosine of the angle between the two vectors; hence this similarity measure is often called "cosine similarity":

$$s_{d,q} = \cos(\vec{d}, \vec{q}) = \frac{\vec{d} \cdot \vec{q}}{\left\|\vec{d}\right\| \times \left\|\vec{q}\right\|} = \frac{\sum_f w_f^d \times w_f^q}{\left\|\vec{d}\right\| \times \left\|\vec{q}\right\|}$$

The maximum similarity is one, corresponding to the query and document vectors being identical (the angle between them is zero). The minimum similarity is zero corresponding to the two vectors having no terms in common (angle between them is 90 degrees).

One problem with cosine similarity, noted by both Salton and Lee, is that it tends to produce relatively low similarity values for long documents, especially (as Lee points out) when the document is long because it deals with multiple topics. Lee's solution relates to the merging of the result of a retrieval system using cosine similarity with the result of a retrieval model using term frequency normalization, e.g., maximum normalization.

In other words, Lee supplements cosine similarity rather than replaces it, thereby getting the advantages of two relatively simple similarity measures. The solution of Singhal et al., is to develop improved normalization factors for term weighting, factors that do a better job of normalizing for document length and term frequency during a *single* retrieval run, thereby eliminating the need for the fusion of separate runs.

In an earlier approach [Salton and Buckley, 1988] dealt with the problem of long documents by combining the usual cosine similarity of query and document ("global" similarity) with similarity of the query to parts of the document ("local" similarity). The parts they tried included sentences and paragraphs. In other words, if two documents d_1 and d_2 have comparable similarity to a given query, but d_1 also contains a sentence or paragraph that is particularly similar to the query, then d_1 will be given a higher similarity value than d_2. They have also tried combining multiple local similarity measures, e.g., sentence and paragraph similarity, with the global similarity. However, the Singhal et al. enhanced term *Lnu* weighting scheme by improving document length normalization, and term frequency normalization. They reduced the importance of such local similarity measures.

The inner product and its normalized form, i.e. the cosine similarity, are not the only similarity functions employed to compare a document vector with a topic vector (although they are by far the most widely used). A variety of "distance" functions, and other term matching functions are available. For example, a family of distance metrics [Korfhage, 1997] is given by:

$$s_{d,q} = \sqrt[p]{\sum_f \left(w_f^d \times w_f^q\right)^p}$$

These metrics compute the distance in vector space between vectors $\vec{d}$ and $\vec{q}$ in terms of the components w_f^d of *d*, the components w_f^q of *q*, and a parameter *p* that determines a specific metric from the family. If *p*=1, the metric is the city block distance, i.e., distance measured as number of city blocks from one street intersection (corner) to another in a city where the streets are laid out as a rectangular grid. If *p*=2, the metric is the familiar Euclidean distance, i.e., the straight line distance in the vector space. This is the same metric that is used to normalize the length of a vector. If $p \to \infty$, the metric is the *maximal direction distance*. That is, as *p* tends to infinity, the largest difference $|w_f^d - w_f^q|$ tends to dominate all the others, and the function reduces to the absolute value of this maximum difference. Since each vector component corresponds to one dimension, one direction, in vector space, each difference between a pair of corresponding components is the distance between the vectors in a given direction. The maximal direction distance metric is the distance along the dimension where the vectors are farthest apart.

Apart from such distance metrics, there are a host of similarity formulas that "normalize" by avoiding term frequencies altogether, i.e., functions that only count the number of terms that match and (sometimes) the number of terms that don't match. One such popular function is Dice's coefficient [Van Rijsbergen, 1979]:

$$Dice = \frac{2n}{n_1 + n_2}$$

n is the number of terms common to vectors d_1 and d_2, n_1 is the number of non-zero terms in d_1, and n_2 is the number of non-zero terms in d_2. Note that the denominator here performs a kind of normalization, so that a short document d_1 will get a high score relative to a short topic description d_2 to which it is relevant. A long document d_3 relevant to d_2 will get a lower *Dice* score provided that the additional text in d_3 contains terms that are not in d_1 and also not in the topic description (greater n_1, same *n*). This could happen if d_3 contains long sections not relevant to d_2, It could also happen if d_3 contains additional discussion of the topic described by d_2, but this additional discussion uses terms that were overlooked by the user who specified topic d_2. On the other hand, if d_3 and d_1 contain most of the same topic-relevant terms that d_2 contains, but d_3 just uses them more frequently and uses few additional terms that d_1 doesn't use, then d_3 and d_1 will receive similar *Dice* scores despite their difference in length.

Another common similarity function is Jaccard's coefficient [Van Rijsbergen, 1979]:

$$Jaccard = \frac{n}{N - z}$$

where *n* is the number of terms common to vectors d_1 and d_2, *N* is the total number of distinct terms (not term occurrences) in the vector space and *z* is the number of distinct terms that are neither in d_1 nor in d_2. In other words, *N* - *z* is the total number of distinct terms that occur in d_1 or d_2 or both. Note that the more distinct terms are either in d_1 but not d_2 or vice versa, the lower is the value of the Jaccard function. It doesn't matter whether the mismatch is caused by non-relevance, or difference in document length. On the other hand, it doesn't matter how frequently a mismatching term (or a matching term) occurs in either d_1 or d_2.

1.4 Probabilistic Approach

In our project we adopted the vector space model (VSM) which seems to be simple and robust to the diverse application domains. For sake of completion we introduce the probabilistic model which can be seen as vector space approaches which are better theoretically motivated. Indeed, there is no clear line separating probabilistic from statistical methods of IR since probabilities are often calculated on the basis of statistical evidence. Of course, given a formula based on a probabilistic model, any source of evidence can be used to compute probabilities, but as a practical matter the evidence is usually statistical, e.g., *tf*'s and *idf*'s, used in the vector space models.

1.4.1 Distinctions between probabilistic and vector space models

According to [Cooper et al., 1992], in a thoroughgoing probabilistic design methodology, systematic use of formal probability theory is made to derive the probability of relevance by which the documents are ranked. Such a methodology should be distinguished from other approaches like the "vector space" retrieval model, in which the retrieved items are ranked by a similarity measure (e.g., the cosine function) whose values are not directly interpretable as probabilities.

[Cooper et al., 1992] list four *potential* advantages of a true probabilistic design methodology:

1. It provides theoretical motivations to derive the expectation of the retrieval effectiveness.
2. It allows us to avoid the traditional trial-and-error retrieval experiments which make the final system *less reliable*. These experiments are needed to determine the parameter values of the best figure model. As examples of such trial and error, consider the variety of term weighting schemes that have been tried in different vector space experiments.
3. It provides more powerful statistical indicators of predictivity and goodness than Precision, Recall, etc.
4. Document's probability of relevance for a user can be reported in the ranked output. It would presumably be easier for most users to understand and base their selection upon a probability of relevance than a cosine similarity value.

Nevertheless, probabilistic methods have not yet been as widely used as these advantages would suggest and, when they were used, they achieved retrieval performance (measured by Precision and Recall) comparable to but not clearly superior to non-probabilistic methods [Cooper, 1994]. Cooper identifies various reasons for these shortfalls:

- The formulation of exact statistical assumptions is often an unnecessary theoretical burden. The time and effort spent on throughout analysis would be better spent on ad hoc experimentation using formalisms looser and friendlier than probability theory.
- The estimation procedures used in probabilistic IR are usually based on statistical simplifying assumptions or 'models' of some sort. The retrieval clues that bear on a document's probability of usefulness must somehow be combined into a single relevance probability, and modeling assumptions are needed to accomplish such combination. Typically, the assumptions adopted for the task are crude and at best only approximately true. The introduction of simplifying assumptions known to be less than universally valid surely compromises to some degree the accuracy of the probability estimates that result.
- The assumptions underlying some IR models, most notably the widely used (and misnamed) "Binary Independence" IR model, can lead to logical inconsistencies [Cooper, 1991], [Cooper, 1995]. The successes of probabilistic models, that have been achieved in spite of the inconsistency of the theoretical models, are due to the fact that the actual assumptions used in practice are different than the assumptions of the theoretical models, and stronger than they needed to be.

Although the previous paragraphs highlighted several the different properties, probabilistic models can be seen as VSM models and vice versa as explained by the following points:

- In a probabilistic method, one usually computes the "conditional" probability $P(d|R)$ that a document d is observed on a random basis given event R, that d is relevant to a given query [Salton, 1989] [Van Rijsbergen, 1979]. If, as is typically the case, query and document are represented by sets of terms, then $P(d|R)$ is calculated as a function of the probability of occurrence of these terms in relevant vs. non-relevant documents. Consequently, the term probabilities are analogous to the term weights in the vector space model (and may be calculated using the same statistical measures). A probabilistic formula is used to calculate $P(d|R)$, in place of the vector similarity formula, e.g., cosine similarity, used to calculate relevance ranking in the vector space model. The probability formula depends on the specific model used, and also on the assumptions made about the distribution of terms, e.g., how terms are distributed over documents in the set of relevant documents, and in the set of non-relevant documents.

- More generally, $P(d|R)$ may be computed based on any clue available about the document, e.g., manually assigned index terms (concepts with which the document deals, synonyms, etc.) as well as terms extracted automatically from the actual text of the document. Hence, we want to calculate $P(d|A, B, C, \ldots)$, i.e., the probability that the given document, d, is relevant, given the clues A, B, C, etc. As a further complication, the clues themselves may be viewed as

complex units, e.g., if the presence of term *t* is a clue to the relevance of document *d*, *t* may be viewed as a cluster of related clues, e.g., its frequency in the query, its frequency in the document, its *idf*, synonyms, etc. This has led to the idea of a "staged" computation, in which a probabilistic model is first applied to each composite clue (stage one), and then applied to the combination of these composite clues (stage two) [Cooper et al., 1992].

5.1 The probabilistic model

The probabilistic model was introduced by Robertson and Spärck Jones. It is developed in (Spärck Jones et al., 1998). This model is built upon the user's information need. Hence, the assumption that the document set is divided in two parts: according to a query, a document is relevant or not. Robertson says a document is relevant when the user *likes* it (the L event) and not relevant in case of dislikeness (the $\sim L$ event). A decision rule can be seen as a ranking function derived by the following quantities:

$$\text{score}(d_j) = \frac{P(L|d_j)}{P(\sim L|d_j)}$$

where $P(L|d_j)$ is the probability that the user likes the document d_j and $P(\sim L|d_j)$ is the probability that he does not like it. Then, the Bayes' theorem is applied to rewrite the conditional probabilities:

$$\text{score}(d_j) = \frac{P(d_j|L)P(L)}{P(d_j|\sim L)P(\sim L)}$$

where d_j can be represented by its attributes f_i (e.g. the words it contains). We consider the attributes (features) to be independent events to simplify computations even if it is not the case in reality. Let A_i be the event bound to an *attribute* f_i:

$$\text{score}(d_j) = \frac{\Pi_i P(A_i|L)P(L)}{\Pi_i P(A_i|\sim L)P(\sim L)}$$

This ranking function is transposed in the logarithmic domain removing constants, i.e. $P(L)$ and $P(\sim L)$ for a given query:

$$\text{score}_{\log}(d_j) = \sum_{A_i \in d_j} \frac{P_i(1-\overline{P}_i)}{\overline{P}_i(1-P_i)} = \sum_{A_i \in d_j} \text{weight}(A_i)$$

where P_i is the probability that the attribute A_i is present in the document when it is *liked* by the user, and $\overline{P}_i$ is the probability for the attribute to be present when the document is not liked (i.e. $P(A_i|L) = P_i(1-\overline{P}_i)$). The drawback of this formulation is that the score of a document does not involve *attribute* weights. For instance, word (as attribute) weight can be computed using its concurrencies. This notion was introduced in (Robertson and Walker, 1997) and is called *attribute eliteness*. It is estimated using *Poisson* distributions.

One of the major problem of VSM and probabilistic models is that no match is triggered between words that have different surface form. The next section shows a statistical technique that attempts to deal with such problem by deriving a semantic similarity between different terms.

1.5 Latent Semantic Indexing (LSI)

In the traditional vector space approach to IR described in Section 5.3, a vector "space" is defined for a collection of documents such that each dimension of the space is a term occurring in the collection, and each document is specified as a vector with a coordinate for each term occurring in the given document. The value of each coordinate is a weight assigned to the corresponding term. A weight is intended to be a measure of how important the given term is in characterizing the given document and distinguishing it from the other documents in the given collection.

This approach is an effective first approximation of the statistical properties of the collection, but it is nevertheless an oversimplification. Its major limitation is that it assumes that terms are independent, orthogonal dimensions of the document space. On the contrary, adding a new term to the space, e.g., a term that was previously omitted because it wasn't considered a good discriminator, has no effect whatever on the existing terms defining the space. Adding a new *document* to the collection not only adds new terms to the space but also does affect the weights of the existing terms because it affects their *idf*'s. But this relates to a term-document relationship, not a term-term relationship. Hence, relationships among the terms, e.g., the fact that certain terms are likely to co-occur in documents about a given topic because they all refer to aspects of that topic, are ignored. Similarly (and more subtly), the traditional term vector approach ignores the fact that a term A and a term B may occur in similar contexts in two distinct documents because they are synonyms.

The traditional vector space approach has another drawback that affects some applications. As the number of terms that occur in a collection can be large, the traditional term-based document space has a large number of dimensions. To solve such problems, Latent Semantic Indexing (LSI) [Deerwester et al., 1990] attempts to capture term-term statistical relationships to cluster together terms that express similar information. In LSI, the document space is replaced by a lower dimensional document space called k-space (or LSI space) in which each dimension is a derived concept, a "conceptual index," called an LSI "factor" or "feature." In LSA the individual dimensions of the target k-space are designed to express the principal components of the term-document distributions in the original space. An algebraic technique called Singular Value Decomposition is here applied to this purpose.

The LSI factors are truly independent statistically, i.e., uncorrelated, in a way that terms are not. Hence, LSI factors capture the term-term relationships that ordinary term-based document space does not. Documents are represented by LSI factor vectors in k-space just as they are represented by term vectors in traditional term-based document space. Vector similarity can be calculated in the same way in k-space as in traditional document space. However, documents and queries dealing with the same topic that would be far apart in traditional document space (e.g., because they use different but synonymous terms) may be close together in k-space.

As [Bartell, et al. 1992] explain, the individual words are not adequate discriminators of semantic content. This because the indexing relationship between words and documents is many-to-many: a number of concepts can be indexed by a single term *polysemy*, and a number of terms can index a single concept *synonymy*. Hence, some relevant documents are missed and some irrelevant documents are retrieved.

LSI aims to address these limitations by mapping each document from a vector space representation based on keyword frequency to a vector in a lower dimensional space. Terms are also mapped into vectors in the reduced space. The claim is that the similarity between vectors in the reduced space may be a better retrieval indicator than the similarity measured in the original term space. The main reasons for such a claim is that, in the reduced space, two related documents may be *quite similar* even though they do not share any keywords. This may occur, for example, if the words used in each of the documents co-occur frequently in other documents.

In other words, if document d_1 uses term t_A and document d_2 uses equivalent term t_b, LSI may derive a high similarity between d_1 and d_2 if t_A and t_b co-occur frequently in similar contexts in other documents. For example, consider the words *car, automobile*, driver, and *elephant.* The terms *car* and *automobile* are synonyms, *driver* is a related concept and *elephant* is unrelated. In most retrieval systems, the query *automobiles* is no more likely to retrieve documents about cars than documents about elephants, if the precise term *automobile* was not used in the documents. It would be preferable if a query about *automobiles* also retrieved articles about *cars* and even articles about *drivers* to a lesser extent. The derived LSI *k*-dimensional feature space can represent these useful term relationships.

Roughly speaking, the words *car* and *automobile* occur with many of the same words (e.g., *motor, model, vehicle*, *chassis*, *carmakers*, *sedan engine*, etc.), and they have similar representations in *k*-space. The contexts for *driver* will overlap to a lesser extent, and those for *elephant* will be quite dissimilar. To be fair about it, a good query submitted to a traditional term-based system would use more terms than "automobile", e.g., perhaps some of the other contextual words mentioned in the passage quoted above.

In other words, the traditional term-based vector space model assumes term independence. Since there are strong associations between terms in natural languages, this assumption is never satisfied

though it may be a reasonable first order approximation [Hull, 1994]. LSI attempts to capture some of these semantic term dependencies using an automatic and purely quantitative method, i.e., without syntactic or semantic natural language analysis and without manual human intervention.

LSI accomplishes this by using a method of matrix decomposition called Singular Value Decomposition (SVD). LSI takes the original document-by-term matrix describing the traditional term-based document space as input. It produces as output three new matrices: *T*, *S*, and *D* such that their product $T \times S \times D$ captures this same statistical information in a new coordinate space, *k*-space, where each of the *k* dimensions represents one of the derived LSI features (or concepts or factors). These factors may be thought of as artificial concepts; they represent extracted common meaning components of many different words and documents [Deerwester et al, 1990].

D is a document matrix. Each column of *D* is one of the *k* derived concepts. Each row of *D* is the vector for a given document, specified in terms of the *k* concepts. The matrix element for the *j*-th concept in the *i*-th document represents the strength of association of concept *j* with document *i.* Hence, *D* specifies documents in *k*-space. Similarly, *T* is a term matrix. Each column as before is one of the *k* derived concepts. But in *T*, each row is a vector in *k*-space describing a term in the original collection, i.e. a term in the original term-by-document matrix that characterized the collection. Hence, each term in this matrix is then characterized by a vector of weights indicating its strength of association with each of these underlying concepts [Deerwester et al, 1990]. In other words, each term vector (i.e., row) in *T* is a weighted average of the different meanings of the term [Hull, 1994].

The diagonal elements in the 2nd matrix *S* assign weights (called “singular values”) to the *k* LSI factors according to their significance. This allows the user to have some control over how many dimensions *k*-space is to have. Some of the power of this decomposition comes from the fact that the new factors are presented in order of their importance (as measured by the diagonal of *S*). Therefore, the least important factors can easily be removed by truncating the matrices *T, S*, and *D*, i.e., by deleting some of the rightmost columns of these matrices. The remaining *k* columns are called the LSI factors [Hull, 1994]. Note that *k* is a parameter under the user’s control. Reducing *k* can eliminate “noise”, e.g., “rare and less important usages of certain terms.” However, if the number of dimensions (LSI factors) is too low, important information may be lost. The optimum number of dimensions obviously depends on the collection and the task. One report finds that improvement starts at about 10 or 20 dimensions, peaks between 70 and 100, and then decreases [Berry, 1995]. As the number of LSI factors approaches the number of terms, performance necessarily approaches that of standard vector methods. Another report says that the optimum number of dimensions is usually between 100 and 200.

Projection of a set of documents into k space is optimal in the sense that the projection is guaranteed to have, among all possible projections to a *k*-dimensional space, the lowest possible least square distance to the original documents. In this sense, LSI finds an optimal solution to the problem of dimensionality reduction [Schutze et al, 1997]. The side-effect is that the derived *k* factors are a sort of “artificial” concepts, i.e. no attempt is made to interpret them in simple English. Indeed, in many cases, it may not be possible to summarize these concepts, to explain what each one “means.” What one *can* say is that a given document is heavily weighted with regard to concept 1, doesn't deal at all with concept 2, is lightly weighted with respect to concept 3, etc. For a single document, such a description may have no value at all. But with regards to a second document (also described in terms of weights of the same k concepts) we *can* say how similar the documents are (in *k*-space). Additionally, if one of those “documents” is a query (or a sample document used as a query, or the centroid of a set of sample documents), we can say how close the given document is to the given query, in *k*-space of course.

Following the usual vector space similarity methods, e.g., calculating the cosine similarities, we can rank documents by how similar they are to the query, in *k*-space. Similarity in *k*-space is more statistically meaningful, and therefore, one hopes, more semantically meaningful, than similarity in conventional term space, because the *k* concepts reflect statistical correlations in the document population, while the original terms do not.

Note also that along with the query-document similarities using the *k*-by-document matrix, *D*, we can compute the term-by-*k* matrix, *T*. This allows us to compute term similarities. Presumably, two terms are very similar if they co-occur, i.e., are strongly correlated, with many of the same other terms. Hence, such a similarity can be used to suggest to a user who enters a query, other terms, statistically similar to the terms he used, which could be added to his query. Or the similarities can be used to construct automatically a domain-dependent or collection-dependent thesaurus.

By the way, although each row of matrix *T* is called a “term vector”, the phrase is used quite differently in LSI terminology than in conventional vector space terminology. In the conventional vector space approach, a “term vector” is a vector in *document space* describing a *document* in terms of weights assigned to each term for the given document. In LSI, both terms and documents are described in LSI factor *k*-space. A term vector is a vector describing a given *term* in LSI *k*-space in terms of the weights assigned to the LSI factors for the given term. A document is described in LSI by a *document* vector specifying the weights assigned to the LSI factors for the given document.

Hearst, et al. (in Text Retrieval Conference, TREC 4) point out an additional advantage of LSI with respect to the routing or classification of documents: the routing task can be treated as a problem of machine learning or statistical classification. The training set of judged documents is used to construct a classification rule which predicts the relevance of newly arriving documents. Traditional learning algorithms do not work effectively when applied to the full vector space representation of the document collection due to the scale of the problem. In the vector space model, one dimension is reserved for each unique term in the collection. Standard classification techniques cannot operate in such a high dimensional space, due to insufficient training data and computational restrictions. Therefore, some form of dimensionality reduction must be considered. One approach is to apply Latent Semantic Indexing (LSI) to represent documents by a low-dimensional linear combination of orthogonal indexing variables.

[Berry, 1995] discusses another advantage of LSI, i.e. its robustness to noisy input: as LSI does not depend on literal term matching, it is especially useful when the input text is noisy, as in OCR (optical character recognition), open input, or spelling errors. If there are scanning errors, and a word, e.g. *Dumais* is misspelled as *Dumas*, many of the other words in the document will be spelled correctly. If these correctly spelled context words also occur in documents that contain a correctly spelled version of *Dumais*, then *Dumas* will probably be near *Dumais* in the *k*-dimensional LSI factor space.

On the other hand, LSI has some serious drawbacks too. As [Hull, 1994] points out: while a reduced representation based on a small number of orthogonal variables might appear to cut storage costs substantially (compared to the traditional term-based vector space model), the opposite is actually true. The LSI representation requires storage of a substantially larger set of values. In addition the LSI values are real numbers while the original term frequencies (weights) are integers. This increases the storage costs. Using LSI vectors, we can no longer take advantage of the fact that each term occurs in a limited number of documents, which accounts for the sparse nature of the term by document matrix.

1.6 Finer-Grain Information Retrieval: Information Extraction and Question Answering

An overview of Information Retrieval in these days cannot avoid to mention more recent techniques that have been introduced to improve the IR technology in terms of grain and precision. The outcome of a traditional Information Retrieval processes is a list of documents ranked according their relevance to an initial target query. The atomic unit of the IR system answer is thus still a document, or in some cases document passages that have been decided as specifically relevant to the query. The latter is a case where the grain of the answer is smaller as relevant passages are ususally proper subset of relevant documents. The grain of the answer is a relevant concept in Information Retrieval as it reduce the notion of “*access time to useful information*”, one of the overall major aims of IR.
Along this line of research two major methods for reducing the grain of returned information have been recently studied: *Information Extraction* and *Question Answering*. We will briefly discuss these two notions hereafter, suggest the potential applications and survey some results, without the aim of being exhaustive. The interested reader can make use of the suggested references to get more information and technical details.

Information extraction (IE) systems emerged in the late 1980's and early 1990's [Pazienza,1997], though forerunners date back to the late 1960's [Gaizauskias & Wilks,98]. In contrast to IR, IE systems do not return the subset of documents from the collection deemed to be relevant to a given query. They are usually given a "template" definition of one or more specific concepts and a document collection, and return a set of filled templates. Templates, in the context of IE, are structured data representations, designed to capture attributes of objects and events predictably present in stereotypical occurrences in texts.

For example, in a *plane crash* we would typically expect to find the type of plane, the airline, number of passengers, flight origin and destination, and location and time of a crash. In this case, a template is usually designed to capture this information. The IE system would, given the template and a document collection, seek to fill an instance of the template for each plane crash event it detected.

Information Extraction is thus the process by which an automatic system is able to process documents in a linguistically motivated way and derive a structured representation of (part of) their content. It is to be seen as a process through which a *structure* is derived from unstructured and noisy texts. The IE technology has the key advantage over IR that the search for an event of a type for which a template has been defined can be more easily satisfied.

The IE process is usually applied in a batch fashion (*off-line*) to the document collection and it builds a large corresponding set of filled template as a structured database. The extracted database can be used when the user requirement is to find a similar event. The higher the level of abstraction provided by the template, the easier and more natural will be the interaction of the user with the updated template database. Notice how a template instantiated with details of the new event can be used to derive a series of database queries. They are effectively the same template but with one or more template slots replaced by variables to be instantiated in the search.

Widely explored fields of application range from the recognition and management of terrorist events ([MUC-3/4]), of joint venture news ([MUC-5]) to machine translation of meteorological bulletins, where translation of templates is applied rather than the more complex text translation.

One of the major subtasks of any IE process is Named-Entity recognition, as most of the traditional IE target information make explicit reference to person, locations that are seen as the participants to the event described by templates. For this reason automatic Named-Entity recognition (NERC) is by its own a typical IE task for which a large set of technologies has been defined and applied. Symbolic pattern recognition (e.g. [Appelt et al., 1993] and statistical methods [Bikel et al., 1999] are largely employed for NERC. The research over models of NERC and relation extraction has been recently conveyed on benchmarking and comparative evaluation by the NIST ACE challenge (ACE,2001). The objective of the ACE program is to develop automatic content extraction technology to support the automatic processing of source language data. Possible down-stream processing includes classification, filtering, and selection based on the content of the source data, i.e., based on the meaning conveyed by the language. In the ACE 2005 challenge, five primary recognition tasks have been defined: entities, values, temporal expressions, relations and events.

There are two major limitations in general with IE technology. First, high performance, systems need to be hand-tailored for each new template/domain (21) and this can be expensive. Secondly, template definitions limit the number of "questions" that can be asked. This means that the analyst cannot be expected to ask "ad hoc" questions that a particular topic might involve.

Question answering by computer has long been a preoccupation of the AI community (Simmons, 1965), and in fact has been argued as defining it. However, the recent TREC challenge has sparked tremendous interest in an area that has been almost neglected in the preceding decade.

In the initial TREC QA tasks (TREC-8, 1999), systems are presented with a set of previously unseen questions in English and a large (about 4GB) text collection. The aim is to find text snippets from the text collection (defined as either 50- or 250-byte strings in two test conditions) which answer the question. More recent QA tasks have become more ambitious. Questions may involve multi-component, or “list” answers (“*Which countries did the Pope visit in the 1990’s*?”) which need to be drawn from multiple documents. Not only must information be fused from several documents, but redundant information must be eliminated, since parts of the answer may well occur repeatedly, though expressed differently, in different texts. Thus, multidocument *summarisation* and *fusion* is becoming an integral part of formulating answers to questions.

Like IR systems and unlike IE systems, QA systems support "*ad hoc*" questions in arbitrary domains. Their question support is more sensitive than most IR systems since, at least in principle, they need to distinguish "*Who did Oswald shot*?" from "*Who shot Oswald*?", whereas virtually no IR system would do so. This, from the perspective of the current IR users is an advantage.

It is important to notice though that QA alone may be not effective for users in specific domains/tasks. For example, a journalist armed only with a QA system would be endlessly typing in low-level questions – "*When did the last Concorde crash*?", "*When was the last air-crash in Paris*?", “*How many casualties*

were there in that crash?" – , etc. Some context will always be required for reassurance; and further context may, on balance, prove useful for additional background.

It is thus worth to consider the positive effects of IE on QA processes. The templates in an IE system capture related, useful questions in a domain. Thus, QA systems, as currently conceived, may offer little more benefit to a working journalist, or an intelligence gatherer, than conventional search engines. Two extensions are made possible by a tighter integration between Information Extraction and Question Answering. First, the kind of relational information obtained in IE is critical to capture the "explanatory" knowledge that would improve the "factual" knowledge actually produced by the current QA systems. Second, interaction with the system is important. User sessions may in fact be strongly improved by considering multiple queries within a single session. Chains of questions (i.e. models of persistent queries) can be conceived based on the template information related to the retrieved texts. This would impact critically on usability and natural interactivity of the overall IR system.

1.7 Conclusions

The review of statistical approaches for document retrieval suggests two important considerations. On the on hand, a deep statistical study on the retrieval task has provided many accurate models to optimize the probability to satisfy the user needs. Often such techniques are tuned or modeled for different application domains, e.g. a specific weighting scheme may be more appropriate for a target corpus. On the other hand, the most common adopted document representation is quite basic, i.e. the *bag-of-words*. This may be considered surprising if we note that statistical models can work with any representation whatever its complexity is, i.e. we can design any feature space and then apply the usual IR techniques developed for the simple words.

In the next section, we will show that the apparently the trivial objective of improving a *bag-of-words* representation hide technical problems of complex solutions.

2 Current Cross-Language Information Retrieval

The rapid spread of communication technologies, such as the World Wide Web, and improvements of general information retrieval (IR) techniques have allowed people worldwide to access previously unavailable information. With these advances, however, it has become increasingly clear that there is a growing need to access to information in many languages. Until recently, monolingual IR has been the main research focus of scientists all over the world, and much of what is available has been in English, although English is the native language for only 6% of the world's population (Haddouti, 1999). This has left a need for non-English speakers to access to information in their own language and a global desire to obtain access to information in multiple languages. Oard and Dorr (1996) give several motivations for research into cross-language information retrieval:

a) for collections containing documents in many languages, where query formulation for each language would be extremely inefficient.

b) for documents containing text in more than one language.

c) for users not able to form queries in other languages, but able to make use of documents retrieved in a foreign language that contain images or names not requiring fluency.

In another article Oard (1997) points out that cross-language information retrieval would also be very helpful for those who read and write only one language, but need information that may not be available in that language. For all of these reasons, cross-language information retrieval is an important and rapidly growing area of IR, and as such, it merits exploration.

In this section a general overview of cross-language information retrieval, including its definition, problems involved in creating cross-language systems, basic IR approaches, major work and projects undertaken, and possible directions for future research. This work introduces the major components

of cross-language information retrieval and summarizes the actions that have been taken so far in this area.

2.1 Definitions of Cross-Language Information Retrieval

Before delving into the problems of retrieving information in multiple languages and the basic techniques used to make multiple-language system functional, it is appropriate to present a clear definition and a word about the terminology used in the body of research about the subject. Much of the literature uses the more common term "multilingual information retrieval" (MLIR) to represent any IR subject that deals with more than one language. Hull and Grefenstette (1996, p.484) give five definitions of MLIR:

a) IR in any language other than English.

b) IR on a parallel document collection or on a multilingual document collection where the search space is restricted to the query language.

c) IR on a monolingual document collection that can be queried in multiple languages.

d) IR on a multilingual document collection, where queries can retrieve documents in multiple languages.

e) IR on multilingual documents, i.e. more than one language can be present in the individual documents.

Another definition comes from Oard (1997, p.1), which says that MLIR is, "selection of useful documents from collections that may contain several languages." This broad definition seems to encompass the last three definitions given by Hull and Grefenstette. The Oard's definition relates to "cross-language information retrieval" (CLIR), because it speaks specifically to IR work across languages, while "multilingual information retrieval" covers more concepts, including the first two of Hull and Grefenstette's definitions.

2.2 General Issues with CLIR

In addition to the problems or drawbacks discussed below with regard to the different techniques used to approach the creation of CLIR systems, there are several general issues to be considered:

First, the basic problem of multilingual text access. For many years, computers and Web browsers were only able to present certain character sets to users. Languages using non-Western characters or including accents or other markings, such as umlauts, etc., could not be viewed accurately. This is a basic issue that concerns CLIR because systems must be able to understand and present the characters from the languages with which they propose to work. For an in depth discussion of the progress being made toward multilingual text access, including character sets, user interfaces, HTML, XML, URL/URI, and HTTP, see Haddouti (1999).

Second, another problem facing CLIR is the fact that different languages vary widely in their structure. In monolingual English IR systems, stemming, (Croft, Broglio and Fujii, 1996, p.101) is often used to increase recall performance. However, stemming cannot be easily generalized to all languages. Spanish, for example, has many more forms of each verb than English, and many other languages have other complex structures with regard to decomposing words for stemming. There is also difficulty with normalization, or breaking down compound words in some languages, such as German (Sheridan and Ballerini, 1996), and there are also languages without clear breaks between words, such as Chinese. Clearly a CLIR system must be adapted to the characteristics of whichever languages it will use.

Various solutions have been proposed for both the stemming and decomposition of terms. A few examples include Croft, Broglio and Fujii's, (1996) approach to stemming that uses measures of statistical dependence based on co-occurrence of terms and Sheridan and Ballerini's (1996) use of a dictionary combined with a mechanism to string word meanings together in order to get exact meaning. Croft, Broglio and Fujii also discuss a word segmentation program created by the Center for Intelligent Information Retrieval at the University of Massachusetts, which can rapidly break Chinese text into words. It is important to note that although these solutions have helped performance, none of them has been fully successful, and adjustments are needed.

Third, it is even more difficult with multiple languages for IR systems to choose correct word meaning. When the system takes in a query, there are often several possible meanings for the terms in that

query, and unless there is a mechanism for determining which meaning is appropriate to the query, the system may retrieve irrelevant documents. This problem is accentuated in multiple language searches, where a term may have various meanings in various languages. This ambiguity of terms is one of the greatest problems in CLIR.

Some solutions suggested for the disambiguation of query terms include automatic query enrichment (Ballesteros and Croft, 1997) and allowing for simple translations of small parts of text to allow the user to see results and then choose the correct term meanings (Hayashi, Kikui, and Susaki, 1997).

Next, the problem of choosing at which time applying translation techniques in the system. The majority of work done on CLIR involves translation of the query, either manually, or sometimes, automatically. But as Hull and Grefenstette (1996, p.485) note, "there is no reason in principle why the problem could not be approached using document translation." However, due to difficulties with automatic translation and the labour intensiveness of manual translation, this is likely to be less efficient for most systems at this point.

2.3 Basic Approaches to CLIR

With a general understanding of what CLIR is and general problems it brings with it to the IR field, this section explores several important approaches to CLIR systems. Machine translation, controlled vocabulary and dictionary-based approaches will be discussed as well as latent semantic indexing and corpora-based approaches.

2.3.1 Machine Translation

One approach to CLIR is to use machine translation (MT), which can automatically translate queries or documents. This can be beneficial because the query can be translated from the language of the user to another language for search, and the results can be translated back into the user's language for viewing (Fluhr, 1996). One of the few available examples of research done on MT with specific regard to CLIR comes from Fluhr and Radwan (1993) (as cited in Oard and Dorr, 1996).

Unfortunately, MT systems often make translation errors because of missing information in the term index or ambiguous definitions. When the MT system chooses a wrong definition, results can become irrelevant. Oard and Dorr (1996) note that MT only produces high quality translations for specific domains, possibly because semantic accuracy suffers when insufficient domain knowledge is incorporated into a translation system.

2.3.2 Controlled Vocabulary

The controlled vocabulary approach has long been the most dominant and effective for CLIR. One of the first CLIR system experiments (Salton, 1970) involved use of controlled vocabulary, with surprising results for its relative simplicity. Usually with CLIR a multilingual thesaurus of some sort is created to hold a list of descriptors for each document in a collection and the semantic relations between them, and each term in the thesaurus must be translated for each language involved (Fluhr, 1996). The descriptors can be added to the thesaurus manually or automatically (if the system can learn from previous indexing which terms are likely to be important) (Fluhr). Sheridan and Ballerini (1996) used a multilingual thesaurus and query expansion with the SPIDER system to achieve results with Italian and German that were significant, although poorer than those from using only a monolingual group of documents with monolingual queries.

This method has limitations in that the user must create a query using only vocabulary from the thesaurus, which can lead to an inability to search for certain terms that are not included. In addition, there is a problem when the queries contain terms that contain different concepts in different languages. There is also a limit to the precision that a controlled vocabulary system can achieve because of the limited number of terms in the thesaurus (Fluhr, 1996), and Haddouti (1999) points out that the larger the size of the vocabulary in the thesaurus, the less effective it becomes. Finally, a controlled vocabulary search can be difficult for a user who does not understand the way the system or the thesaurus is constructed, and assignment of index terms and construction of the thesaurus can be labor-intensive (Oard, 1997).

2.3.3 Dictionary-Based Approaches

This approach uses combinations of monolingual or bilingual dictionaries to provide something similar to a thesaurus (Oard and Dorr, 1996), which provides a platform for developing multilingual systems. Hull and Grefenstette (1996) used a bilingual dictionary for their CLIR experiments with French and English queries.

The dictionary-based approach typically suffers from the problems of ambiguity and a limited scope, omitting technical terminology (Haddouti, 1999), and Hull and Grefenstette (1996) noted that they had to cull out large amounts of information from the dictionary that would be harmful to the system's performance for their research.

2.3.4 Latent Semantic Indexing

Another way to approach CLIR is through latent semantic indexing (LSI), which makes comparisons between sets of semantically related words (Fluhr, 1996). "In LSI the principal components are thought to represent important conceptual distinctions" (Oard and Dorr, 1996, p.21). This allows for retrieval that works better with actual word concept relations. Because this approach orders documents by how closely related they are semantically, LSI can also help to limit or clear up ambiguity problems (Davis and Dunning, 1995). Landauer and Littman (1991) (as cited in Oard and Dorr, 1996) were some of the first to work on CLIR using LSI, and their study depicts a basic approach to its use. Berry and Young (1995) were able to use LSI with more success when they used more finely grained training data.

2.3.5 Corpora-Based Approaches

According to Haddouti (1999, p.4), the corpora-based technique, "analyzes large collections of existing texts and automatically extracts the information needed." Oard and Dorr (1996, p.18) describe this approach as a type of automatic thesaurus building where information about the relationships between terms is obtained, "from observed statistics of term usage". Lin and Chen (1996) have used this approach in their research on machine learning and multilingual thesaurus construction. However, this technique ideally requires large collections to be made, and such collections are scarce.

It is important to note that different aspects of these approaches can be combined in an attempt to create superior CLIR systems. For example, the EMIR (European Multilingual Information Retrieval) project, which lasted from 1991 to 1994, combines MT and other IR methods, such as statistical models for weighting query-document intersections, as well as normalization of terms, grammatical tagging and a reformulation system aimed at disambiguation (Fluhr, 1996). Many approaches incorporate statistical or term vector translation techniques as well, mapping sets of TF-IDF term weights between languages (Oard and Dorr, 1996).

2.4 Major Projects

Over the past ten years or so, multiple projects have been started to explore CLIR issues, and the subject is gaining popularity at IR conferences. One of the projects, EMIR, has led to the commercial product, SPIRIT (Syntactic and Probabilistic System for Indexing and Retrieving Textual Information), which allows users to input natural language queries and retrieve documents in English, French, German and Russian (Haddouti, 1999). Another project, MULINEX (Multilingual Indexing, Navigation and Editing Extensions for the World-Wide Web), allows searches to be filtered by language and subject area and uses automatic translation to help users understand foreign documents (Haddouti). CANAL (Catalogue with Multilingual Natural Language Access/Linguistic Server) and TRANSLIB (Tools for Accessing Multilingual Library Catalogs) are both tools supporting multilingual access to library catalogs (Oard, 1997). CANAL analyzes queries syntactically and semantically using recognition of compound words and translation of key words in other languages, and TRANSLIB uses MT and corpora, such as thesauri and dictionaries to give access to English, Greek, and Spanish documents. Finally, TwentyOne, a European Union project, is a tool for the dissemination of multimedia information that supports cross-language queries and partial translation of retrieved documents (Haddouti). These are only a few of the larger projects that have been undertaken worldwide.

With regard to conferences, CLIR has grown in popularity to the point where a rapidly growing track at the TREC meetings has spun off into its own set of conferences, known as CLEF (Cross-Language

Evaluation Forum) (Braschler, Peters, Schauble, 2000). CLEF was launched in 2000 to deal specifically with CLIR issues, and continues to be held today.

2.5 Directions of Future Research

As the field expands rapidly, research will continue to confront the issues specific to the basic approaches to CLIR. Resolving disambiguity and addressing normalization issues between languages will be important. The refinement and growth of multilingual thesauri, bilingual dictionaries, and corpora for retrieval will also continue. Hopefully, MT will be improved upon, and more use will be made of relevance feedback, as it provides excellent opportunities for increased performance in CLIR systems by lessening term ambiguity. It is also possible to combine different approaches, as has been shown, and new combinations or variations thereof may increase system performance as well.

It is important to note that while databases such as Lexis/Nexis and Dialog have begun to incorporate CLIR into their systems, the results are still quite limited (Oard, 1997). For search engines like Google, Yahoo, and Altavista, a user can perform an advanced search in a specific language or in several languages, but the results often return in English. When the results come back in other languages like Arabic, the pages are only readable by a non-Arabic speaker if the web page already contains other languages that the user can read. It is a signal of progress that there is movement toward multilingual search capabilities in these popular products, but it is clear that much improvement must be made for CLIR tools to reach the same levels of performance as regular IR systems.

2.6 Conclusion

The need for IR systems capable of handling cross-language issues is increasing as the world becomes more connected by technology. This section has given a general overview of the rapidly expanding work in the field of cross-language information retrieval by exploring its purpose, difficulties, basic tools, major works and future research goals. On this line next sections describe advanced conceptual document representations which aim to provide CLIR with the adequate tools to achieve high accuracy.

3 Conceptual Representations via *traditional* Natural Language Processing

The previous section has shown diverse IR statistical approaches for mono and multi lingual document retrieval. The usual adopted document representation is the simple *bag-of-words*. This causes several problems in term of language translation since it does not allow the system to rely on precise information.

Given such problem, IR studies have been directed to the designing of more effective document representations. Documents are still described as pairs *<feature, weight>*, but, a more complex and effective feature design is applied. Such studies aim to achieve a representation more *conceptual* than the one provided by simple words. The consequences for multi-language applications are straightforward: a representation based on concept rather than on words, eliminate all problems relates to the ambiguity. This allows the cross-language system to achieve the same retrieval performance of the monolingual systems.

Unfortunately, none of the advanced linguistic representations proposed in the literature have been shown to improve optimal pure statistical approaches based on the simple *bag-of-words.*
The major reasons for such failure are the following: (a) complex representations capture just a small piece of information more than the *bag-of-words* and (b) such representations are derived automatically, thus the (few) errors introduced in the retrieval process compensate the poor gain in accuracy provided by the richer feature space.

3.1 Linguistically Inspired Conceptual Representation

Documents embody the highest expressions of the human linguistic skills. This intuitive observation has led researchers in document retrieval to consider linguistic aspects in the modeling of more complex document representations. Some of the well-known feature models experimented in the last decades are:

- ***Lemmas***, i.e., the base form of rich morphological categories, like nouns or verbs. In this representation, lemmas replace the words in the target texts, e.g., *acquisition* and *acquired* both transform in *acquire*. This should increase the probability to match the target concept, e.g., *the act of acquiring* against texts that express it in different forms, e.g., *acquisition* and *acquired*. Lemmatization improves the traditional stemming techniques used in IR. In fact, the stems are evaluated by making a rough approximation of the real root of a word. As a consequence, many words with different meanings have common stems, e.g., *fabricate* and *fabric*, and many stems are not words, e.g., *harness* becomes *har*.
- ***Simple n-grams***, i.e., sequences of words selected by applying statistical techniques. Given a document corpus all consecutive *n*-sequences of (non-function) words are generated, i.e. the *n*-grams[1]. Then statistical selectors based on *n*-gram frequencies are applied to select those most *relevant* for the target domain. Typical used selectors are *mutual information,* χ^2 or *document frequency*.
- ***Nouns Phrases***, e.g., Proper Nouns and Complex Nominals. Simple regular expressions, e.g. N^+ (i.e., every sequence of one or more nouns), based on word categories (e.g., nouns, verbs and adjectives) can be used to select complex terms like *Minister of Finance* and discard the non-feasible term *Minister formally*. The words *Ministers* and *Finance*, in the first phrase, are often referred to as *head* and *modifier,* respectively. More modifiers can appear in a complex nominal, e.g., the phrase *Satellite Cable Television System* is composed of the tree nouns *Satellite*, *Cable* and *Television* that modify the head *System*.

[1] Traditionally, in IR *n*-grams refer to sequence of characters but they are also used to indicate sequence of words.

- ***<head, modifier$_1$,…, modifier$_n$>*** *tuples*. Parsers, e.g., [Charniak, 2000]; [Collins, 1997]; [Basili *et al.*, 1998c] are used to detect complex syntactic relations like *subject-verb-object.* These can be used to select complex phrases, e.g., *Minister announces plans*, from texts. An interesting property is that these tuples can contain non adjacent words, i.e. tuple components can be words linked by a long distance dependency, e.g. in [Strzalkowski and Jones, 1996] the *subject-verb* and *verb-object* pairs (i.e. the <*head, modifier*> pairs) were used. Hardly, such tuples can be detected via pure statistical models. The aim of phrases is to improve the precision on concept matching. For example, documents in an *Economic* category could contain the phrase *company acquisition* whereas an *Education* category could include term like *language acquisition*. If the word *acquisition* alone is used as an individual feature, it will not be useful to distinguish between the two above categories. The whole phrases, instead, give a precise indication of the document content.
- ***Semantic concepts***, each word is substituted with a representation of its meaning. Assigning the meaning of a content word depends on the definition of word senses in semantic dictionaries. There are two ways of defining the meaning of a word. First, the meaning may be explained, like in a dictionary entry. Second, the meaning may be given through other words that share the same sense, like in a thesaurus. For example, WordNet [Miller, 1991] encodes both forms of meaning definitions. Words that share the same sense are said to be *synonyms* and in WordNet, a set of synonym words is called *synset*. The advantage of using word senses rather than words is a more precise concept matching. For example, the verb *to raise* could refer to: (a) *agricultural texts*, when the sense is *to cultivate by growing* or (b) *economic activities* when the sense is *to raise costs*.

3.2 Limitations of traditional NLP-based conceptual representations

The above techniques appear as a feasible way to improve IR systems, nevertheless, the use of NLP in IR has produced controversial results and debates.

In TREC-5 and TREC-6 [Strzalkowski and Jones, 1996]; [Strzalkowski and Carballo, 1997], document retrieval models based on stems were slightly improved by phrases, noun phrases, *head-modifier* pairs and proper names. However, their evaluation was done on *ad-hoc* retrieval mode only, since the less efficient NLP techniques could not be applied to the same *testing-set* of the pure statistical models. This prevented the comparison with the *state-of-the-art* retrieval systems.

In [Strzalkowski *et al.*, 1998; Strzalkowski and Carballo, 1997] a high improvement of retrieval systems was obtained using topic expansion technique. The initial query was expanded with some related passages not necessarily contained inside the relevant documents. The NLP techniques used in TREC-6 were used to further increase the retrieval accuracy. The success of the above preliminary experiments was not repeated in TREC-8 [Strzalkowski *et al.*, 1999] as the huge amount of data made impossible the correct application of all required steps. The conclusion was that the higher computational cost of NLP prevents its application in operative IR scenario. Another important conclusion was:

NLP representations can increase basic retrieval models (e.g., SMART) that adopt simple stems for their indexing but if advanced statistical retrieval models are used NLP does not produce any improvement [Strzalkowski et al., 1998].

In [Smeaton, 1999] a more critical analysis is made. In the past, the relation between NLP and Machine Translation (MT) has always been close. Thus, much of the NLP research has been tailored to the MT applications. This may have prevented the NLP techniques to be compatible with task such as retrieval, categorization or filtering. Thus, when pure retrieval aspects of IR are considered, such as the statistical measures of word overlapping between queries and documents, the NLP that has been developed recently, has little influence on IR. This because such NLP does not help to overcome the major problem of IR, i.e. the retrieval of documents which do not contain many, or, any of the query terms. Indeed, current IR is not able to handle the cases in which different words are used to represent the same meaning or concepts in documents and queries. Moreover, polysemous words, which can have more than one meaning, are treated as any other word. Thus, Smeaton suggested to drop the idea of using NLP techniques for IR, and proposed to exploit NLP resources like WordNet.

In this perspective Smeaton used WordNet to define a semantic similarity function between noun pairs. The purpose was to retrieve documents that contain terms similar to those included inside the query. As many words are polysemous, a Word Sense Disambiguation algorithm was developed to detect the right word senses. Unfortunately, such algorithm showed an accuracy ranging between 60-70%, which was to low to obtain positive results. Indeed, improvement were obtained only after the senses were manually validated.

Other studies using semantic information to improve IR were carried out in [Sussna, 1993] and [Voorhees, 1993; 1994]. They report the use of word semantic information for text indexing and query expansion respectively. The poor results obtained in [Voorhees, 1994] show that the semantic information taken directly from WordNet without performing any kind of WSD does not help IR at all. In contrast, in [Voorhees, 1998], promising results on the same task were obtained after that the senses of selected words were manually disambiguated.

In summary the analysis of the literature reveals that the most likely reasons for the failure of NLP in IR are the following:

- High computational cost of NLP models, due prevalently to the use of the parser to detect syntactic relations, e.g., the <*head, modifier*> pairs. This prevented a systematic comparison with *the-state-of-the-art* statistical models.
- Small improvements when complex linguistic representation is used. This may be caused either by NLP errors in deriving complex structures or by the poor information that complex features brought more than the *bag-of-words*.
- The lack of accurate WSD tools, in case of semantic representation:
    a) The ambiguity of the words causes the retrieval of a huge number of irrelevant documents if all senses for each query words are introduced, or;
    b) if a WSD with 60% is employed to disambiguate document and query word senses, the retrieval precision decrease proportionally to the error, i.e. 40%.

3.2.1 Results in Text Categorization

Literature work has shown the failure of conceptual representations on ad-hoc document retrieval tasks. However, there are other document retrieval problems such as document filtering and text categorization for which traditional NLP techniques may improve the *bag-of-words* models. Unfortunately, even for these retrieval tasks, such techniques seem not to be successful. In the following, we report the conclusive discussion of the large experimentation carried out in [Moschitti, 2003, Moschitti and Basili, 2004].

Syntactic information in TC

NLP derived phrases seem intuitively superior to the *bag-of-words*, nevertheless, literature results, e.g. [Moschitti, 2003, Moschitti and Basili, 2004] have shown that phrases produce small improvement for weak Text Categorization algorithms, i.e., Rocchio [Ittner, 1997], and no improvement for theoretically motivated machine learning algorithms, e.g., *Support Vector Machines [Joachims, 1997]*. Possible explanations are:

- Word information cannot be easily subsumed by the phrase information. As an example, suppose that in the target document representation *proper nouns* are used in place of their compounding words. Then, suppose that, our task is to design a classifier which assigns documents to a *Politic* category, i.e. describing political events. The training documents could contain the feature `George Bush` derived by the proper noun *George Bush*. If a political test document contains the `George Bush` feature, it will have chances to be classified in the correct category. On the contrary, if the document contains only the last name of the president, i.e., *Bush*, the match of the feature `Bush` against the category feature `George Bush` will not be triggered. This is confirmed by the findings in [Caropreso *et al.*, 2001], which show that replacing *n*-grams with individual tokens produces a decrease of the Rocchio classifier accuracy .
- The information added by the sequence of words is very poor. Note that, a sequence of words classifies better than its compounding words only if two conditions occur:
    a) The individual words of the sequence appear in the wrong documents. For example the words *George* and *Bush* are included in a document not related to the Political category.
    b) Some documents that contain the whole sequence *George Bush* are correctly categorized in the Political category.

Thus, on the one hand, the sequence *George Bush* is a strong indication of political category; on the other hand, the individual words, *Bush* and *George,* are not related to the political category. This is very improbable in natural language documents since many co-references between two referentials in which one of them is a word sequence are triggered by a common subsequence (e.g. *Bush* co-refers with *George Bush*). The same situation frequently occurs for the complex nominals, in which the head is used as a short referential. This suggests that terms are rarely not related to their compounding words.

Although the previous considerations leave few room for an effective use of traditional natural language processing representations, several researches have claimed to have applied them succesfully:

- In [Furnkranz *et al.*, 1998] advanced NLP has been applied to categorize the HTML documents. The main purpose was to recognize the student home pages. For this task, the simple word *student* cannot be sufficient to obtain a high accuracy since the same word can appear, frequently, in other University pages. To overcome this problem, the AutoSlog-TS, Information Extraction system [Riloff, 1996] was applied to automatically extract syntactic patterns. For example, from the sentence *I am a student of computer science at Carnegie Mellon University*, the patterns: *I am* <->, <-> *is student*, *student of* <->, and *student at* <-> are generated. AutoSlog-TS was applied to documents collected from various computer science department and the resulting patterns were used in combination with the simple words. Two different TC models were trained with the above set of features: Rainbow, i.e. a bayesian classifier [Mitchell, 1997] and RIPPER. The positive result reported by the authors is a higher Precision when the NLP-representation is used in place of the *bag-of-words*. This improvement was obtained for recall lower than 20% only. The explanation was that the above NLP-patterns have low coverage, thus they can compete with the simple words only in low recall *zone*. This result, even if important, cannot be accounted as an evidence of the superiority of the *NLP-based TC*.
- [Mladenic and Grobelnik, 1998] report the experiments using *n*-grams. These have been selected by using an incremental algorithm. The web pages in the Yahoo categories, *Education* and *References* were used as reference corpus. Both categories contain a sub-hierarchy of many other classes. An individual classifier was designed for each sub-category. The classifiers were trained with the *n*-grams contained in the few available training documents. The results showed that *n*-grams produce an improvement about 1 percent point (in terms of *Precision* and *Recall*) for *Reference* category and about 4% on the *Educational* category. This latter outcome may represent a good improvement over the *bag-of-words*, but we have to consider that:
  a) Although a cross validation was carried out, the experiment were done on 300 documents only.
  b) The adopted classifier is *weak*, i.e. a *Bayesian* model which is not very accurate. Its improvement using *n*-grams does not prove that the best figure classifier improves too.
  c) The task is not standard: many sub-categories (e.g., 349 for *Educational*) and few features for each classifier. There are no other researches that have measured the performance on this specific task, thus, it is not possible to compare the results.

  As the best hypothesis, we can claim that an efficient classifier (averagely accurate) has improved its accuracy, using *n*-grams. The task involved few data and many categories.
- In [Furnkranz, 1998] is reported the experimentation of *n*-grams for *Reuters-21578* and 20 NewsGroups corpora. *n*-grams were, as usual, merged with the words to improve the *bag-of-words* representation. The selection of features was done using the simple document frequency [Yang and Pedersen, 1997]. Ripper was trained with both *n*-grams and simple words. The improvement over the *bag-of-words* representation, for the Reuters corpus was less than 1%. For 20 NewsGroups no enhancement is reported.
- Other experiments of *n*-grams using Reuters corpus are reported in [Tan *et al.*, 2002]. Only bigrams were considered. Their selection is slightly different from the previous work, as Information Gain was used in combination with document frequency. The experimented TC models were Naïve Bayes and Maximum Entropy classifiers [Nigam *et al.*, 1999] both trained with bigrams and words. On *Reuters-21578* the authors present an improvement of 2% for both classifiers. The achieved accuracies were 67.07% and 68.90% for Naive Bayes and Maximum Entropy, respectively. Thus, using phrases, we are able to slightly improve TC models which perform about 20% less than the best figure model (SVMs achieve almost 88%). The consequence is that even [Tan *et al.*, 2002] do not provide a clear evidence that simple NLP-derived features as the *n*-grams, are useful for TC. A higher improvement was reported for the other experimented corpus, i.e. some *Yahoo* sub-categories. Again, we cannot validate

these finding as such corpus is not standard. A standard corpus allows researchers to replicate the results. Note also that, it is not possible to compare the performance with [Mladenic and Grobelnik, 1998] as the set of documents and *Yahoo* categories are quite different.

- On the contrary, in [Raskutti *et al.*, 2001] were experimented bigrams using *SVM* on the *Reuters-21578*. This enables the comparison with (a) the literature results and (b) the best figure TC model. The feature selection algorithm that was adopted is interesting. They used the *n*-grams over characters to weight the words and the bigrams inside categories. For example, the sequence of characters "to build*"* produces the following 5-grams: "to bu", "o bui", "buil" and "build". The occurrences of the *n*-grams *inside* and *outside* categories were used to evaluate the *n*-gram scores in the target category. In turn *n*-gram scores are used to weight the characters of a target word. For instance, the character "o" of the word "score" in the "to score by" context receives a contribution from the 5-grams, "o scor", " score", "score", "core", and "ore b". The 5-gram scores are apportioned giving more ratio to the most centered *n*-gram, i.e. the scores are multiplied respectively by 0.05, 0.15, 0.60, 0.15 and 0.5. These weights are used to select the most relevant words and bigrams. The above set as well as the whole set of words and bigrams were compared on *Reuters-21578* fixed *test-set*. According to the expriments, SVM improved about 0.6% when bigrams were added either to all words or to the selected words.
- This may be important because to our knowledge is the first improvement on SVM using phrases. However, we have to consider that:
  a) No cross validation was applied. The fact that bigrams improve SVM on the Reuters fixed *test-set* does not prove that they improve the general SVM accuracy. Indeed, in [Dumais *et al.*, 1998, Moschitti and Basili, 2004], SVM reaches an f-measure over 87%, that is higher than 86.2% obtained by Raskutty et al. with bigrams (even if they experimented with a larger number of categories which determines a more difficult task).
  b) The improvement on simple words reported in [Raskutti *et al.*, 2001] is 0.6% = 86.2% - 85.6%. If we consider that the Std. Dev. in the experiments [Moschitti, 2003], [Bekkerman *et al.*, 2001] is 0.4/0.6%, the improvement is not statistically sufficient to assess the superiority of bigrams.
  c) Only, the words were used, special character strings and numbers were removed. These strongly affect the results as suggested in [Moschitti and Basili, 2004]. The *only words* model may be improved by bigrams but it provides a baseline lower than the bag-of-words on general strings. Consequently, we cannot state that phrases increase the best figure classification approach. On the contrary, another corpus experimented in [Raskutti *et al.*, 2001], i.e., *ComputerSelect* shows higher accuracy when bigrams are used, i.e. 6 percent points. But again the *ComputerSelect* collection is not standard. This makes difficult to replicate the results.
- The above literature, favorable to the use of phrases in TC, shows that these latter do not affect the accuracy (or at least the best classifier accuracy) on the Reuters corpus. This could be related to the structure and content of its documents, as it has been pointed out in [Raskutti *et al.*, 2001]: Reuters news are written by journalists to disseminate information and hence contain precise words that are useful for classification, e.g., *grain* and *acquisition* whereas other corpora such as *Yahoo* or *ComputerSelect* categories contain words like *software* and *system*, which are useful only in context, e.g., *network software* and *array system*.

Semantic information in TC

The experiments on word senses carried out in [Moschitti, 2003, Moschitti and Basili, 2004] show that there is not much difference between senses and words. This because word senses in category documents tend to be always the same. Moreover, different categories are characterized by different words rather than different senses. The consequence is that words are sufficient surrogates of exact senses.

Another study on the clustering of words which encode similar conceptual information was carried out in [Bekkerman *et al.*, 2001]. They applied the Information Bottleneck (IB) feature selection technique to cluster similar features. The important idea was that a classical feature-filtering model cannot achieve good performance for the text classification problem as it is usually not related to the adopted machine learning algorithm. The IB clusters words according to their relationship with categories. More precisely, it attempts to derive a good trade-off between the minimal number of word clusters and the maximum mutual information between the clusters and document categories.

The information bottleneck method relates to the distributional clustering approach that has been shown not particularly useful to improve ”weak” TC model performance (e.g., Naive Bayes TC). However, a more powerful TC model like SVM was shown to take advantage of word clustering techniques. SVM trained with IB derived clusters was experimented on three different corpora: Reuters, WebKB and 20 NewsGroups.
Only the 20 NewsGroups corpus showed an improvement over the bag-of-words. This was explained by studying the ”complexity” of the involved corpora. The above analysis revealed that Reuters and WebKB corpora require a small number of features to obtain optimal performance. The conclusion is that if the target corpus is enough *complex* the IB can be applied to reduce its complexity (i.e. to reduce the number of relevant features by clustering together those thar are *similar*) and consequently to increase the SVM accuracy. The improvement on 20 NewsGroups, using the cluster representation, was 3 percent points only.

3.3 Conclusions

This section has discussed how NLP technologies have been used to increase precision and expressivity in IR processes. Here the major technologies and achievements have been discussed. The controversial fact that, among the several linguistically motivated representations modeled so far, none has significantly improved traditional quantitative validation criteria (such as precision and recall) with respect to state-of-art purely statistical approaches has also been discussed.
The major reasons for such limitations are the following: (a) the complex representations provided by the adopted language processing technologies capture just a small piece of information more than simpler *bag-of-words* representations and (b) the errors that occur in their automatic extraction introduce noise in the retrieval process; this decreases the overall performance.
However, such study provides some clues to design more effective conceptual representations: (a) the accuracy of the conceptual representation extraction should be very high and (b) in order to be effective for indexing purposes, the novel extracted concepts should be meaningful for the application domain and the user needs.

4 Conceptual Retrieval based on Ontologies and Semantic Metadata

The previous section has shown that traditional NLP techniques are unable to derive accurate and effective conceptual representations and hence cannot improve IR systems. However, the fact that we cannot derive automatically an effective semantic information does not imply that words are the only interesting indexing objects.

In this section, we show that approaches to document retrieval containing both traditional *bag-of-words* and semantic information encoded by a markup language (as for example in the Semantic Web) are superior to the traditional IR models. The reasons for the success of such Semantic Web approach relates exactly on the solution of the drawbacks of traditional NLP-based concept extraction, i.e.:

a) At the moment, the markup annotation is carried out manually, thus it is reliable and very accurate. This means that the document retrieval systems are not affected by the errors of automatic extraction.

b) The information is annotated according to the target application domain thus it is suitable to characterize documents according to the user needs. As users can formulate queries with relevant concepts the retrieval of significant documents increases.

One the most interesting results is that the accuracy of the markup information makes possible to apply inferential mechanisms. These, in turn, enable the users to retrieve information by expressing the queries with high level of conceptual abstraction.

From a cross-language point of view, the availability of a precise conceptual representation allows us to design very accurate and effective CLIR systems.

4.1 Basic Semantic Web Ideas

Recent literature work on conceptual retrieval is critically based on the knowledge representation problem. Indeed, in the literature, several approaches that effectively and efficiently model such problem have been developed. To understand current retrieval systems based on conceptual information a brief introduction on the most used knowledge representation components and methods is needed. In particular, we have detected three common ingredients in such systems: Ontology, Metadata, RDF and DAMN+OIL:

Ontologies
An ontology is typically a hierarchical data structure containing all the relevant entities and their relationships and rules within that domain (e.g., a domain ontology). The computer science usage of the term ontology is derived from the much older usage of the term ontology in philosophy. The entities and the rules in the ontology are generally expressed into a vocabulary, containing both domain independent and domain specific terms. An “upper ontology” is an ontology not tied to a particular domain that describes general entities and their relations.

Metadata
Metadata represents information about the data in the individual databases and can be seen as an extension of the concept of schema in structured databases. It may describe or be a summary of the information content of the individual databases in an intentional manner. It typically represents constraints between the individual media objects which are implicit and not represented in the databases themselves. Some types of metadata may also capture content independent information like location and time of creation.

For example, a broadcast news metadata can consist of additional information related to the speaker’s name, the recording data and time, the type of recorded sound. Deeper metadata may be information related to some people involved in the news, such as political entities, corporations or locations.

Knowledge Representation

The ontology-based approaches to IR require a systematic description of the underlying domain model as a semantic model agreed among the users and practitioners of the domain. Such specification requires languages, models and infrastructures able to act as standards for knowledge representation and exchange, and for automatic reasoning.

The Resource Description Framework (RDF) is a language for representing information about resources in the World Wide Web. It is particularly intended for representing metadata about Web resources, like title, author, and modification date of a Web page, copyright and licensing information about a Web document, or the availability schedule for some shared resource. By generalizing the concept of a "Web resource", RDF can also be used to represent information about “things“ that can be identified on the Web, even when they cannot be directly retrieved on the Web.

OWL is the result of a joint research effort among two different knowledge representation languages aiming to support ontology definition and management on the Web: the DARPA Agent Markup Language Ontology (DAML-OIL) and the Ontology Interface Layer (OIL) developed by a network of excellence supported by the European Commission.

The DAML+OIL language was created in response to the DARPA Agent Markup Language (DAML) sprang from a U.S. government-sponsored effort in August 2000, which released DAML-ONT, a simple language for expressing more sophisticated RDF class definitions than those permitted by RDFS.

As an example, we can use DAML+OIL to make ontology definitions. This is done by giving a name for a class, which is the subset of the universe which contains all objects of that type. If we are working in a domain of animals, we will want to define a kind of thing called animal. To do this, we use a Class tag. `Class`, e.g.`<daml:Class rdf:ID="Animal">`

The Ontology Inference Layer OIL is a proposal for a web-based representation and inference layer for ontologies, which combines the widely used modeling primitives from frame-based languages with the formal semantics and reasoning services provided by description logics. **OIL** adds a simple Description Logic to RDF Schema: It thus allows to define axioms that logically describe classes, properties, and their hierarchies.
The language DAML+OIL built on both OIL and DAML-ONT, was submitted to the W3C as a proposed basis for OWL, and was subsequently selected as the starting point for OWL
OWL extends the Class, SubClass, Property and Subproperty primitives of RDF with Restrictions on Range, Domain, existential and cardinality, with Combinators (like union, intersection and complement for sets and symmetric, transitive for relations) and Equivalence and Inverse properties for functions.

OWL is made available in three species with increasing expressive power and decreasing levels of efficiency with respect to inference (e.g. classification and subsumption). *OWL lite* that gives support to the definition of classification hierarchies and simple constraint features; *OWL DL* that keeps decidability and completeness of description logics but restricts the definitions of properties (they cannot be classes) and classes (they cannot be individuals); OWL Full meant for maximum expressiveness and the syntactic freedom of RDF with no computational guarantee.

The increasing availability of tools for editing/creation of ontological resources, for reasoning and aligning over ontologies makes OWL a current standard for most Semantic Web resources and applications. The W3C is promoting OWL as the reference KR language in the Semantic Web[2]. Where earlier languages have been used to develop tools and ontologies for specific user communities (particularly in the sciences and in company-specific e-commerce applications), they were not defined to be compatible with the architecture of the World Wide Web in general, and the Semantic Web in particular. OWL instead uses both URIs for naming and the description framework for the Web provided by RDF to add to ontologies the possibility of being distributed across the network, scalability and compatibility features desirable for Web standards.

[2] http://www.w3.org/2004/OWL/

4.2 Semantic Web IR

In this paragraph, we will describe several systems, focusing on a semantic web approach for information retrieval. Each system is based on (a) some ontologies of different complexities and representation purposes and (b) a classical IR applied to raw texts enriched with metadata provided by such ontologies.

A typical example of a Semantic Web IR model is described in [Shah et al., 1997]. This system aims to retrieve documents that contain both free text and semantically enriched markup. To provide a uniform environment for the indexing and the retrieval process, both documents and queries are marked up with statements in the DAML+OIL semantic web language. The resulting system following from this approaches is called OWLIR (Ontology Web Language and Information Retrieval) which focuses on addressing three scenarios (involving semantically marked up web pages and text documents):

1. Information retrieval (IR) - e.g., identify and rank relevant pages or documents for a query looking for detail descriptions concerning USA and Afghanistan leaders.
2. Simple question answering (Q&A) - e.g., *who is the president of the USA?*
3. Complex question answering - e.g., `what is the current situation in Afghanistan`

To carry out the previous three steps, OWLIR uses two primary components:

1. a set of ontologies designed using DAML+OIL which allow users to specify their interests in different events. For example, users can annotate the announcement events which consist of several information such as the speaker, the media type and the broadcast channel.
2. a hybrid information retrieval mechanism based on WONDIR framework. This is a system that carries out text extraction, annotation, inference on knowledge encoded in ontologies. To infer document similarity, a language model approach, is used in lieu of traditional Boolean or vector-space models.

OWLIR bases its reasoning functionality on DAMLJessKB system [20] which facilitates the reading of DAML+OIL pages, and allows the user to reason over read information. DAMLJessKB is a rule engine and scripting environment written in the Java language that can be used to write applications that have the capacity to "reason" using knowledge supplied in the form of declarative rules. DAMLJessKB uses the Rete [Rete, 1982] algorithm, a very efficient mechanism for solving the many-to-many matching problem, to process rules.

A peculiar aspect of OWLIR is that the ontology becomes a means of communication between the user and the system and helps to overcome the bottlenecks in information access, typical of keyword searches. It supports information retrieval based on the actual content of a page and helps navigate the information space based on semantic concepts. OWLIR uses the metadata information added during the text extraction process to infer additional semantic relations. These relations are used to decide the scope of the search and to provide more accurate responses.

As a case study of the OWLIR system, the authors create a “natural ontology”, which follows the concept of “Natural Kinds OF" from the field of philosophy [Quine et al, 1977]. The resulting Event ontology relates to the University domain, which in turn is used to formulate semantic queries and to deliver exactly the information we are interested in. Event categories follow the natural kind of events that are prominent in a university e.g. Movie Showing, Seminars, Sport events etc. Every event has common properties like Date and Time of event, Organizer and Place of the event etc. Events may be academic or non academic, free or paid, open or by invitation.

The experimental analysis reported is aimed to measure the extent to which Precision and Recall is improved with the use of semantic markup. The measurement was extended to three different types of document: text only, text with semantic markup and text with semantic markup that has been augmented by inference. The resulting value of average precision is 25.86% for unstructured data, 66.15% for structured data plus free text and 85.48% for the case of structured data plus inferred data and free text. The corresponding Recall values are, respectively, 20%, 85% and 90%. This experimental values show the increasing of quality in the retrieval process introduced by the proposed model.

Another example of system that uses ontology and document similarity based on events is the Swoogle system [Li Ding et al, 2001]. This is a crawler-based indexing and retrieval system for the Semantic Web, i.e., for Web documents in RDF or OWL.

The Swoogle system extracts metadata for each discovered document, and computes relations between documents. Discovered documents are also indexed by an information retrieval system which can use either character N-Gram or URIrefs as keywords to retrieve relevant documents and to compute the similarity among a set of documents.

Swoogle identifes three categories of metadata:

- basic metadata which includes syntactic and semantic features of a Semantic Web Document (SWD), i.e. an on line document, accessible by Web users and software agents;
- relations, which include the explicit semantics between individual SWDs,
- analytical results such as Semantic Web Ontology/Semantic web DataBase (SWO/SWDB) classification, and SWD ranking.

The retrieval task is accomplished using a transformation function to put the document with semantic information into a document-only format. This function encodes the metadata into a text-based representation and pipes this information to the document itself. Then, the retrieval process with metadata is reduced to the classical retrieval task, using a standard information retrieval engine.

Actually, the Swoogle system is on-line and has discovered and analyzed over 11,000 semantic web documents. However, the authors do not report either any metric for evaluating the quality of the retrieval process or any results.

In [Faaborg, 2001], it is described another system which uses each document's metadata items to create a navigational interface that allows users to filter documents by means of attributes and locate information with fewer navigational steps.

The main idea is that, when searching for information, people automatically group documents in clusters. [Pirolli et. al, 1995] from Xerox PARC advocate that as people forage for information they are likely to cluster what they find into unique groups based on the relevance of their particular query. [Robertson et. al, 1998] from Microsoft Research note that, to locate information on the Web, users often cluster information by using the *favourite links folder* or documents that contain a list of related hyperlinks.

A clear way of improving information retrieval is to have a system robust enough to consider such clusters. To achieve this we need to discover the attributes that users consider when personally groups documents together. By creating a navigation system that automatically filter documents based on these attributes, nodes in the interlinked hierarchies of the retrieved documents should begin to resemble their personal information clusters. This is because the system allows them to filter on the exact same attributes they use to personally group documents together.

The study on the above approach aims to measure the correlation between the quantity and the quality of metadata (attributes) available to build the directories and the response time for the retrieval process. The authors examined three different document directory classifications, by Audience, by Subject and by Geographic location. The results prove that, the larger the number of metadata used to describe the documents is, faster the users react on average.

A totally different approach to information retrieval, based on logic formalism was studied in [Zhang et al, 2003]. In this work, the authors propose an enhanced model that tightly integrates IR with formal query and reasoning to fully use both textual and semantic information for searching in Semantic Portals. The model extends the search capabilities of existing methods and can answer more complex search requests. It uses and combines fuzzy description logic (DL) IR model and a formal DL query method.

In DL-based IR models, documents and queries are defined as DL individuals and concepts respectively. The content, structure, layout and thesauri information of all the documents are also described in DL and form a document knowledge base Σ. A document *d* is relevant to a query *Q* if *d* is logical consequence of the knowledge base given the query *Q*, i.e. $\Sigma \Rightarrow d : Q$. The IR problem is then reduced to the DL instance retrieval problem which can be answered by a DL reasoning engine. This leads the authors to the intuition of extending the rule $\Sigma \Rightarrow d : Q$ to one that retrieves both documents and metadata objects and can also integrate them together to obtain an enhanced model (based on textual and semantic information).

The evaluation process shows that a system using both textual and semantic information as the one described here, can drastically increase the performance in Precision/Recall of one system using only the textual information. The authors compared such two approaches on simple and complex queries.

In both cases the DL system increases the average Precision from 73.4% to 95.2% (in the simple case) and from 40.8% to 96.9% (in the complex case).

It is also interesting to consider the work reported in [Malet et al, 1999] for its effort of defining a Metadata Standard (i.e. its requisites and methods). The authors based their work on a proposed metadata standard, the Dublin Core Metadata Element Set, which has recently been submitted to the Internet Engineering Task Force. This model also incorporates the National Library of Medicine's Medical Subject Headings (MeSH) vocabulary and MEDLINE-type content descriptions. The model defines a medical core metadata set that can be used to describe the metadata for a wide variety of Internet documents. The MCM-MESH subject metadata tags allow authors to describe the subject of a document to a high level of accuracy and present subject data in a format that both human users and Web crawlers can understand. Tagging a document with MeSH numbers metadata syntax will permit a Web crawler to retrieve documents with hierarchic subject content descriptions.

4.3 Latent semantic approach

LSI is a statistical technique which correlates words on the basis of their occurrence patterns in the documents. It also does the clustering of words based on the statistical analysis, which form the dimensions of the multi-dimensional vector space. In section 7.4.1 we focus on the potentiality of latent semantic indexing. In this section, we will present some systems, using LSI techniques for indexing purposes.

The first system that we describe attempts to extract metadata with a latent semantic approach [Shklar, 2002]. Such system, namely InfoHarness, can be sumarized into the following components:

1. The InfoHarness Server (IH Server), which uses document's metadata to traverse the document set, search between the document, and retrieve the original information.

2. the HTTP Gateway, a networking component, which is used to pass requests from HTTP clients to the IH Server and then responses back to the clients.

3. The Repository Generator, which is used for the off-line generation of metadata that represents the desirable view on the structure and organization of the original information.

InfoHarness uses an InfoHarness Object (IHO) to encapsulates metadata, and organize metadata into hierachies; each IHO may have one or multiple children, as well as multiple parents. An IHO can be referred to one document or set of documents. The IHO are subdivided into two main categories: Content-based IHO and Contend-descriptive IHO. Examples of content-based metadata include full-text indices of InfoHarness collections, and are extracted using semantic analysis on the document. Content-descriptive metadata may include concept as *location, ownership, creation-date, etc.* This metadata is inferred without the need of semantic analysis on the document. All the IHO are grouped together to form the InfoHarness Abstract Type Hierarchy (ATH)

The important advantage of InfoHarness is the flexible access to arbitrary heterogeneous information without any relocation or reformatting of the data. The InfoHarness Abstract Type Hierarchy is stable in the sense that it does not have to be modified to support new user-specific terminal classes. This hierarchy has been constructed to achieve the dual flexibility in the representation, as well as the presentation of data. However, the authors do not report the performance evaluation of the proposed system, so we are not able to understand concretely how the proposed solutions can be useful to the retrieval task we are studying.

In some cases, the statistical analysis on the document set it's not computationally convenient and is preferable to have some other methodology. The WAIS index [Kahle et al. 1991] is a complete inverted index on the document contents and does not involve any statistical analysis. We can summarize some interesting technologies used in WAIS:

1. Size of a Collection: The WAIS indexing technique is preferred if there are just a few documents in the collection. Statistical techniques like LSI require a large sample space to prevent meaningless answers at the time of query processing.

2. Frequency of Modification: WAIS is also preferred if the collection is modified frequently (addition, deletion and update of documents). This is because extensive statistical computations are required by LSI to re-compute the word clusters.

3. Types of Queries: The *WAIS* index is suitable for keyword-based queries but not for finding documents that do not contain the exact keywords (e.g. searching for car it may not be possible to retrieve documents containing the word automobile).
4. Domain Structure of Information: It would be advantageous to use LSI if all documents are from the same domain or display a typical usage correlation pattern or structure. If the documents are chosen at random from different domains, WAIS is likely to work better.

4.4 Semantic WordNet Kernel

The previous section has shown that the LSI approach is currently used in modern systems that aim to provide an effective conceptual retrieval. LSI indeed provides a term to term matrix that specifies the conceptual information common to term pairs.

The major drawback of the LSI approach is that it is strictly domain specific, i.e. if we change domain or document corpus we will most likely need to re-evaluate the latent semantic matrix. This causes efficiency problem and prevents a fast system reuse.

An alternative to the LSI has been recently proposed in [Basili et al, 2005], where the semantic lexical knowledge contained in the WordNet hierarchy is used for supervised text classification task in a novel way. Intuitively, the main idea is that the documents *d* are represented through the set of all pairs in the vocabulary $<t,t'> \in V \times V$ originating by the terms $t \in d$ and all the words $t' \in V$, e.g. the WordNet nouns.

In this way, instead of a set of terms, a set of pairs are used for document representation. Consequently, when the similarity between two documents is evaluated, the matching between the above pairs is used in place of the usual term matching. The weight given to each matched pair is proportional to the similarity that the two terms compounding the pair have in WordNet (Miller,91).

The authors proved that such document similarity (which is a valid kernel) is equivalent to summing the similarities of each term t_1 of the first document with each term t_2 of the second document, where the term to term similarity is provided by WordNet. Such approach has two advantages: (a) a well defined space which supports the similarity between terms of different surface forms based on external knowledge and (b) no term or sense clusters is explicitly evaluated.

The spaces which embed the above pair information may contain $O(|V|^2)$ dimensions. If we consider only the WN nouns (about 10^5), such space contains about 10^{10}dimensions which is not manageable by most of the learning algorithms. Thus the authors applied kernel methods to solve this problem since such techniques allow the learning machines to use an implicit representation space. The kernel approach along with Support Vector Machines (SVMs) [Vapnik, 95] allowed the authors to achieve an accuracy higher than the one based on the *bag-of-words*.

The main drawback of such approach is that the general prior knowledge contained in WordNet is not very useful when there is a sufficient amount of training documents. Thus the approach should be used only in poor training data conditions.

4.5 Strong Ontology-driven approaches

In some systems, ontology hierarchies are used in more depth than in others. We can introduce the term “strong ontology approach”, to indicate approaches that critically use ontologies. In general, in the “strong” approach, ontologies are used both to represent documents and metadata and are also involved in the retrieval process using hierarchal structures.

In [Kashyap et al., 1996] the authors present a three level architecture for representing multimedia data:

1. Ontologies, this level refers to the terminology/vocabulary which characterizes the content of information in a database irrespective of the media type. The vocabulary in general shall contain both domain-independent and domain-specific terms.
2. Metadata, these represent the information about the data in the individual databases; it can be a *summary* of the information content of the individual databases. Some metadata may also capture content-independent information like “location” and “time of creation”.
3. Data, this is the actual raw data which might be representend in any of the media type.

In particular the Metadata level, which is crucial for the proposed system, can be subdivided into three categories:

- Content dependent metadata, it depends only on the content of the original data and can be automatically extracted.
- Content descriptive metadata, it can not be extracted automatically from the contents and relates into characteristics which could be determined by a cognitive process. Content descriptive metadata can be also classified into:
    - Domain-dependent, which uses domain-specific concepts. An example of domain dependent metadata would characterize the set of images in a GIS database containing forest land cover.
    - Domain-independent metadata, which relies on no such domain-specific concept. A typical example of a domain-independent metadata would be the one which describes the structure of a multimedia document.
- Content independent metadata, it does not depend on the content, as, for example, the document's date of creation, the file size, its location and so on.

For the retrieval task, the user can be guided to the construction of the query with the help of the ontology, to increase the query expressiveness. The authors propose a graphical interface to show the ontology content in the form of hierarchical attributes that the users can select to search in the document set.

When the user defines a query in terms of metadata described in an ontology, we can use a correlation engine to establish correspondence between the query and the document set. The authors introduce the terms "meta correlation" to describe a correlation between different documents in different media by using the information contained into their metadata.

In the previous example of a GIS database, we can produce some document's metadata information such as:

```
[(region[block(bounds[33N<=latitude<=34N,84W<=longitude<=85W])]
(relief[moderate,steep])]
```

which encodes the following information:

- All "block" fall within the latitudes 33N and 34N and longitudes 84W and 85W
- All "block" have either moderate or steep relief

A typical user's query on this system could be "`Get me all regions having moderate relief and population greater than 200`", which can be translated into:

```
[(region X) (population [>200]) (relief [moderate])]
```

and can be solved using an algorithm like:

- compare the query metadata with only the database metadata having information about population and retrieve the latitude and longitude values of all the areas having population greater than 200
- Query the image database and retrieve the images having a moderate relief, together with their latitude and longitude
- Intersect the two results set

This process permits in general to extract documents from multiple databases containing documents of different media.

4.5.1 Ontology-driven Information Retrieval in KIM

Within the context of information retrieval (IR), KIM[3] represents an approach based on the implementation of semantically enabled IR techniques. The functionality of KIM is ontology dependent. Analysis showed that it would be most appropriate to implement KIM on top of a basic upper-level

[3] KIM Platform, see http://www.ontotext.com/kim/

ontology and allow for future modular extensions for specific tasks and domains. This section provides a brief description of how KIM exploits the idea of upper-level ontology to the effect that it provides an effective and novel solution to the IR task.

The KIM Approach to Information Retrieval

The KIM platform provides a novel Knowledge and Information Management infrastructure and services for automatic semantic annotation, indexing, and retrieval of unstructured and semi-structured content/documents. It differs from other systems and approaches in that it performs semantic annotation and provides IR services based on the results. To do this in a consistent fashion, it performs information extraction based on an ontology and a massive knowledge base.

Semantic Annotation

KIM platform implements semantic annotation as an innovative model for automatic semantic content enrichment and it enables new information access methods and extends the existing ones. Moreover, KIM enables new applications such as highlighting, indexing and retrieval, categorization, generation of advanced metadata, smooth traversal between unstructured text and the available relevant knowledge. The most significant advantage of semantic annotation is its wide applicability - to any sort of text – web pages, regular documents, text fields in databases, transcripts of news, documentaries, movies, etc.

The rationale behind the semantic information extraction approach employed in KIM has roots in the conception that certain entities, a content refers to, are of significant importance for the meaning (semantics) of the content they appear in. In the field of Natural Language Processing (NLP), and particularly, within the Information Extraction tradition, as NEs are considered people, organizations, locations, and entities of the like kind, for whose reference a (proper) name is used. To clarify why named entities constitute an important part of the semantics of the documents, consider the sentence “the first president of the United States”. To understand the meaning of the sentence it is not enough to understand the meanings of the constituent words. In addition a background knowledge is required, namely about: the president being a person; the president designating “the person who holds the office of head of state of the United States government” (WordNet) and not “an executive officer of a firm or corporation” for example; that “United States” is an abbreviation of the name of a state, positioned in North America, etc. Similar considerations apply when querying a DB – a keyword search would yield a lot of irrelevant results; step-wise restrictions or boolean operator sentences will only limit the amount of irrelevant result. Yet, the presence of highly trained librarian who mediates between the archive/collection and the user is required to gain adequate search results by means of clarifying the exact domain of the search, the user interests and needs and so on. Named entities, when compared to words, have different nature as a result – different semantics, and thus require different approach. Unlike words, named entities denote an (often concrete) individual and not a class or any member of the class. When describing the meaning of words, lexical semantics and/or common sense would suffice; to understand the meaning of named entities more specific knowledge about the world is required.

The semantic annotation process in KIM is based on a simple model of real-world entity classes (ontology) and a massive knowledge base. Semantic annotation is a generation of specific metadata. It is a process of assigning to the named entities in the text links to their semantic descriptions. This sort of metadata, also called “semantic annotation”, provides both class and instance information about the entities. The semantic annotation (metadata) has certain prerequisites for its representation: (1) an ontology (or at least, a taxonomy), which defines the entity classes; (2) entity identifiers, which allow entities to be distinguished and linked to their semantic descriptions; (3) a knowledge base with entity descriptions.

The PROTON upper-level ontology

To perform IR as well as for the representation of the specific metadata, KIM makes use of basic upper-level ontology (PROTON[4]). The usefulness of an ontology, in view of the IR process, is best illustrated when considering the tasks of classification, archive organization, access and retrieval of

[4] PROTON: see http://proton.semanticweb.org/

information and content. An ontology models the domain sub-world by identifying classes of instances, their attributes and inter-relations. The richness, hierarchical structure, and the design principle of an ontology, as well as the underlying instance data encoded with respect to the ontology, would result in its capability to provide access and to retrieve relevant and/or exhaustive information.

PROTON, the ontology used in the KIM platform, encompasses a carefully meant set of descriptive metadata that suffices for the description of all the main attributes of an information resource. Though PROTON was not designed especially for the purpose of multimedia domain modeling, it has nevertheless the capability to represent it. Even without further extension, PROTON is sufficient to represent, for example the type of the document, its being a news emission, a movie; having a certain producer, director, creation date, etc. At the same time it includes means with enough expressivity to describe the content, i.e. the things the material refers to – a person, an event the person has role in or she is involved in and so on.

There are several, often contradicting or at least inconsistent or divergent definitions of what an ontology is, and yet it is undisputable that it is of a great use for the purpose of semantic IR. Instead of providing a formal definition of an ontology, in what follows we will assume that an ontology could be meaningfully described when considering its usefulness for a particular purpose or for a wide range of tasks. A basic upper-level ontology is a kind of a background or pre-existing knowledge that can be used to provide formal semantics (i.e. machine-interpretable meaning) to any sort of information: databases, catalogues, documents, web pages, transcripts of news, documentaries, etc. To allow machine-interpretable semantics is an advantage when aiming at the efficiency of automatic information processing. What is more important, based on an ontology, the process of IR is enhanced with precise, structured, user-need defined, and manageable techniques for querying and retrieval.

Besides the many possible classifications based on different characteristics an ontology might possess, one approach, illuminating for the understanding of the usefulness of an ontology, defines two types: upper-level (roughly domain-independent) and domain-specific. Having in mind that an ontology defines (consist of) hierarchy of classes and hierarchy of relationships between them and/or between their instances, it is easy to understand why an upper-level ontology is to be preferred. Namely, because it possess the means of encoding and representing the most basic and common aspects of any considerable description no matter of the specificity of the domain (weather forecast, popular science documentary, etc.), and despite the specificity of the task in view (for example - classification of movies, access to news emissions, description of the themes of the documentaries), and so on. Using an upper-level ontology has the advantage that it provides easy extendibility and domain-specific ontologies may be plugged onto it. PROTON was designed with the requirement to make it suitable for open-domain general purpose semantic annotations as well as to allow easy extensions according to specific needs.

PROTON ontology contains about 300 classes and 100 properties, providing coverage of the general concepts necessary for a wide range of tasks, including semantic annotation, indexing, and retrieval of information.

PROTON consists of four modules (see Fig.1), organized in three levels. The levels organize the modules according to their importance for the operation of an application. The Basic Level, occupied by the system module, comprises all the necessary meta-primitives and technical notions. The system classes and properties are usually not presented to the end-user, but their inclusion is absolutely necessary so that any application that uses PROTON can operate properly.

The System module defines basic classes such as entity, lexical resource and lexical source, necessary for the encoding of the most basic and common aspects of any considerable description no matter of the specificity of the domain or of the task. The Entity class subsumes all the existing and conceivable things we can see, witness, talk about, or think of. By the introduction of a common class, the ontology allows the definition of common properties, applicable to all of the class members, while at the same time it introduces them just once, thus avoiding the redundancy, characteristic of ordinary classification schemes. In a classification scheme for each subtopic a separate entry is provided. For example, one can find “capital of” as a subtopic of any “country-name topic”, to the effect that a search based on the keyword “capital” would have to browse all the topics containing the specialization “capital” and to the effect that the catalog increases more rapidly than the number of the catalogued items. In the system ontology it is defined that entities (i.e. the instances of protons:Entity) could have multiple names (instances of protons:Alias), that information about them could be extracted from particular protons:EntitySource-s, etc.

The LexicalResource class serves for the encoding of lexical resources (like the suffixes “AG”, “Ltd.” etc. in a company name, first names of persons, and so on), which are used to encode clues for the IE process. Alias, an important sub class of the LexicalResource branch, represents the alternative

names an Entity has. The hasAlias relation is used to link an Entity to its alternative names. The relation is very useful and widely exploited by KIM in its IR and IE modules to represent the set of names associated with an entity. It is useful both in the entity recognition and disambiguation tasks. Furthermore, Alias might be used for cross-linguistic IR, in case that the names of a certain entity in different languages are included in the KB or in case of translation of the ontology elements. In fact, KIM KB, has already several aliases for locations in English, French, Spanish, Italian and sometimes the local transcription of the location name.

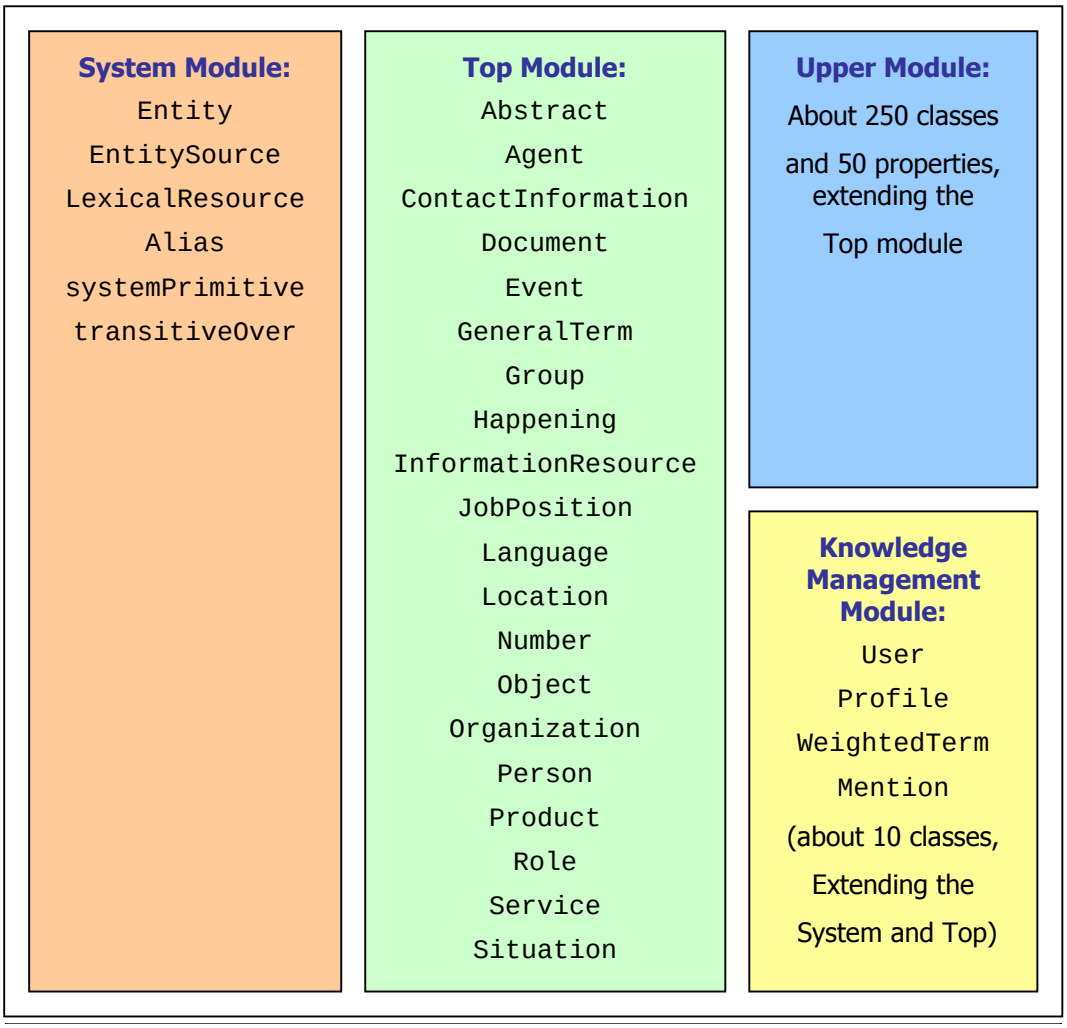

Fig. 1. PROTON (PROTo ONtology) Modules

The Top module forms the second layer of PROTON. To be more specific, it extends the basic level in that it contains further specializations. The Top module consists of specializations of the entity class – object, abstract, happening, and of their sub-classes whose inclusion is useful for the representing entities with a cross-domain importance. First a distinction is made between entity types, such as Object – real-world things, i.e. existing entities such as agents, locations, vehicles; Happening – events and situations as for example they are reported in the news; and Abstract – abstractions that are neither objects nor happenings, i.e. concepts that lack a real-world counterpart, such as “economy”, “politics”, etc. Both objects and abstracts are useful when describing resources (documents, audio and/or video material, transcripts of news broadcasts, etc.) as well as their content concepts like the “industry sector” in industry related news, or organizations being referenced, people being shown in a multimedia material, etc. The entity’s subclasses are specialized down to real entity types of general importance: meetings, military conflicts, job positions; commercial, government, and other types of

organizations; people; and various locations. Things like numbers, time, money, and other specific values are also covered.

The third layer of PROTON comprises of two independent ontologies - PROTON Upper module or PROTON KM (Knowledge Management) module. Both define more specific classes. The top module contains general classes, useful, among others, for the following tasks: (1) the knowledge discovery; (2) metadata generation; and (3) intelligent knowledge access tools. The top-level classes represent the most general, basic, and comprehensive concepts, part of the knowledge about the world. KM (Knowledge Management) module – contains 38 classes of slightly more specialized entities specific for Knowledge Management tasks and applications.

The KIM Knowledge Base

Besides the ontology, for the purposes of semantic annotation, indexing, and retrieval of documents, KIM uses a seed knowledge base (KB). The knowledge base (KB), in this context, is a body of formal knowledge about entities, a means for the representation of non-ontological formal knowledge. It consists of instance data – descriptions of entities and their interrelations, i.e. for each entity, the KB contains information about the entity's type, aliases (incl. a main alias - official or well-known name), attributes, and relations. A KB contains, for the most part, instance knowledge/data while the ontology defines the classes, relations, attributes, as well as additional constraints and dependencies, necessary for the modeling of the domain. The ontology is a kind of a schema for the KB. Both, the ontology and the KB, are to be stored in a semantic store – the system for formal knowledge reasoning and management, which provides the basic operations of storage and retrieval.

KIM KB provides coverage of popular real-world entities of common interest, which are considered well-known and thus not explicitly introduced in the documents. Most important and used entities in the KIM KB are geographic names and organizations. The entities that represent geographical features are imported from GNS (GEOnet Names Server) and other sources. They are organized so as to represent instances of Location (and its subclasses) having the property subRegionOf as it is applied between Continents, GlobalRegions, Countries, and other subclasses of Location. Some of the subtypes of Location, contained in KIM KB are - Country; Province ; County; CountryCapital; City; Ocean; Sea, etc. The locations are given together with several of their aliases, including in English, French, sometimes the transcription of the local name; as well as with their geographic coordinates (Long/Lat), the designator (DSG) and Unique Feature Index (UFI), according to GNS. All this provides a useful basis for cross-linguistic querying and retrieval. The entities in the KB are derived or collected from various sources like geographical and business intelligence gazetteers.

The KB hosts two kinds of knowledge about entities. The first kind is the so called pre-populated knowledge as it was imported or acquired from trusted sources. The entities acquired this way are of general importance and provide enough clues so that the IE process performs well on inter-domain content. The second kind is automatically extracted knowledge as it was discovered in the process of semantic annotation (using IE techniques) or by means of other knowledge discovery and acquisition methods. The second type of knowledge provides a means for permanent enrichment of the KB. To enable the recognition of entities and relations not part of the KB, and so to facilitate the IE process, the KB contains a collection of lexical resources with coverage of the suffixes in an organization's name; personal names, time lexica, currency prefixes and others.

The representation of the named entities in the KB is given with entity's Semantic Description containing Aliases (for example: Florida and FL); Relations to other entities (hasPosition relation between Person and job Position); Attributes (for geographic entities - latitude and longitude).

More details about the KIM and PROTON technology for Ontology-driven IR can be found in [Kiryakov et al., 2004] [Popov et al., 2004] [Terziev et al., 2005] [Kiryakov et al, 2005] [Dowman et al., 2005].

4.6 Conclusions

This section has shown several attempts to design new systems which exploit conceptual information available in the form of ontologies and metadata. Although, the ideas and results are promising the case studies along with the evaluation procedures seem to be in a early stage. The consequence is that such technology represents a intriguing and interesting research topics rather than a stable and ready to use methodology for information retrieval and management.

5 Source of Complexity and Achievements

This part of the deliverable will investigate the role of independent technologies in the functionalities required by the MAD system. Emphasis to the applicability of each technology given its current state-of-art performances will be given.

Preliminary investigation will also involve some application dependent issues, like the availability of already processed, i.e. indexed and documented, audiovisual material. This information in fact may have an impact on the final set of functionalities as, for example, available documented material can be profitably used for improving the metadata extraction procedures, by means of machine learning algorithms.

5.1 IR, IE and CLIR in Prestospace

5.1.1 Semantic Metadata extraction as Information Extraction and Web Mining

Web services actually tend to offer functional and non-functional requirements and capabilities in an agreed, machine-readable format. The target is to provide automated services for discovering, selection and binding of information as a native capability of middleware and applications. However, major limitations are due to the lack of clear and processable semantics. Multimedia data posed even more critical problems as their semantics often depends on multiple and independent aspects: functional information, e.g. data format and processing constraints, application criteria, e.g. the different commercial constraints that may be applied, as well as content information, e.g. the topics to which a TV program refers or the genre of a song or video clip. In particular audio-visual data suffers from the fact that they are particularly rich in content and the level of semantic description is not easily detected from the different co-operating information pieces that give rise to a variety of abstraction levels (the video content vs. the environment sound as well as the speaker's comments).

Thus, methods of Information Extraction from multimedia data have to face specific problems to support realistic Semantic Web scenarios:

- They must capture *levels of abstraction* able to express content at the visual level as well as at the sound (or speech) level.
- Given the richness of the audio-visual information and the usually large size of target archives they must be *efficient* and *scalable*.
- They should be as much *adaptable* as possible even in the early development phases to afford problems of realistic size. In particular, methods of machine learning for the construction of the required large knowledge bases and rule sets are needed.
- They must be *robust* with respect to the noise and complexity (often incompleteness) of the source data.

The above issues are central in the MAD area activities of PrestoSpace, as it involves either documentation and publication processes. In this scenario, the need for making digitised data accessible through intelligent information retrieval interfaces must be approached as a combination of pre-processing (content feature extraction), automation semantic metadata extraction from raw texts (when available), extraction from possible external sources (e.g. the Web), ontology-driven extraction and indexing as well as advanced retrieval. Ontological resources can be thus used as a reference model for the extraction and retrieval steps. Moreover, multilingual access has to be guaranteed.

The PrestoSpace MAD components should face all the above problems. However, in the remaining sections, we will survey requirements especially targeted to semantic extraction and retrieval from textual information, as they mainly characterize the tasks related to cross-linguistic Information Retrieval in MAD (MAD-CLIR).

The MAD-CLIR subsystem should make use of human language technologies for Information Extraction (IE) from automatic transcribed speech and for robust shallow grammatical analysis of incoming AV data. Moreover, given the size of archived material, topical categorisation is an important step to govern the complexity issues in the management and publication processes towards large audiences. It is to be stressed that redundancy at the data level is to be explored. The noisy nature of the extracted data (e.g. errors in the ASR or mistakes in the grammatical recognition) should be controlled by making available to the overall extraction system of source material as much as possible. In this perspective larger data sets should be taken into account than just the source AV data. For example initial IE results over input raw data (e.g. the transcriptions local to the input broadcasted news) should be then exploited to search texts or pages equivalent on the Web (weakly equivalent or strongly related). This should improve the overall precision by extending the evidence about the input and by adjusting the possibly noisy metadata. External material available on the "parallel" Web texts is in fact linguistically more reliable and even more comprehensive, thus complementary with respect to the implicit information local to the AV data.

The final picture is that textual material in input should be processed by adding the following evidences as principled metadata:

- Terminological and lexical evidence local to the AV input data (via ASR when possible)
- Named-Entity recognition from local data as well as from external sources
- Automatically suggested hyperlinks between the target AV data to be archived (e.g. individual news in broadcasted TV journals) and distributed (and trusted) sources (e.g. Web-based newspaper portals and pages)
- Ontological information in terms of IR representation comprising ontology indexes about topical classes (e.g. Education vs. Sport, Foreign Politics vs. Economics), upper-level concepts (e.g. geographical locations, organizations or individuals) and specific instances (e.g. George Bush, or USA vs. United States)

This rich variety of information expected from semantic extraction in MAD poses then several requirements to CLIR. First, the modelling of the user interface according to different (and integrated) querying capabilities:

- Full text search as usually applied by mostly popular search engines
- Natural Languages Questions
- Semantic access through the navigation of ontological information (i.e. concepts, relations and instances) and reference to the ontological indexes in the source AV documents

Second all the above modalities should be offered in a language independent fashion. Although the discussion of technological solutions to support the above processes are not the direct target of the present document, they have a relevant impact on the requirements posed to the CLIR technology. In the next sections we will thus outline some research lines as general technical guidelines, that the presented survey suggest.

5.1.2 Conceptual Retrieval through ontology-driven IR and CLIR

The form of conceptual retrieval foreseen in the MAD documentation and publication platform should provide the targeted user (i.e. the validator archivist in one case and the final user in the other) with two major capabilities:

- *Look for data at the proper level of abstraction*. During validation as well as during the final browsing of the archives the user should not to take a specific care on the implicit system modalities of indexing. For example, we should not assume any specific user acknowledge about the different ways a piece of information is named or referred into the AV data. Individuals are often referred to with different terminologies and most of them are perfectly equivalent from the retrieval point of view: *George Bush* or *G.W. Bush* or *the current USA president* are ways expressing slightly different semantic information about an individual, but these differences are in general irrelevant for most querying scenarios.

- *Accessing the system in any of the supported language* without differences*,* i.e. independently on the possibly different languages characterizing the different AV material archived. As different languages are targeted in PrestoSpace, users should use any of the supported natural languages to querying the system against any processed AV data. This possibility is offered by the relative language-neutral nature of the metadata. The foreseen ontology-based indexing is thus a solution as concepts and relations are by their nature notions agreed across language and nationality barriers. However, especially for natural language querying, there is a relevant need of mapping question information (in the source language) to the textual and metadata information characterizing the archived material. This aspect emphasizes the requirements posed in MAD by the traditional views in CLIR.

The next two sections will deal separated by sketching possible solutions offered by the PrestoSpace project to the above two aspects. They thus represent a kind of look-ahead of this document that partly reflects and summarizes some of the project work in progress.

5.1.2.1 Conceptual Retrieval through ontology-driven IR and CLIR

To understand the role that KIM plays for the enhancement of the functionalities of the MAD platform, it is useful first to consider the infrastructure and services of the KIM Platform. In the course of the introduction, a note will be made on the applicability of certain modules within KIM for the IR process, and in particular for the purpose of retrieving digitized audiovisual material.

To perform IR, KIM makes use of basic upper-level ontology (PROTON) and massive KB. PROTON contains about 300 classes and 100 properties, thus covering wide range of the general concepts required for the semantic annotation, indexing, and retrieval of documents. PROTON has been designed with the intent to serve as a basis for the creation of various domain-specific ontologies. The KB, contains almost 80 000 entities and their aliases (including aliases in several European languages). The entities have been collected from various sources like geographical and business intelligence gazetteers. The KB covers real-world entities that could be referred in content across a wide variety of domains (e.g. Locations are important News, Tourism, Documentaries, etc). The KIM World KB contains entity descriptions that represent a named entities in terms of their aliases, relations to other entities, attributes and the entity's proper class.

Semantic IR needs a specific form of information extraction based pre-processing called semantic annotation. KIM analyzes texts and recognizes references to entities (such as persons, organizations, locations, dates). Then it tries to match the reference with a known entity that has a unique URI and description. Alternatively, a new URI and description are generated automatically. Finally, the reference in the document is annotated with the URI of the entity. The metadata, created in this way can be used for indexing, retrieval, visualization and automatic hyper-linking of documents. Therefore, semantic annotation, as performed by KIM, is the generation of specific metadata. It is the process of assigning to the entities in the text links to their semantic descriptions in the KB. It is important to mention that the semantic annotation process is applicable to any sort of content that could be used with traditional information extraction techniques: web pages, documents, text fields in databases, transcripts of news emissions, etc.

One of the specifics of semantic annotation is that it may provide more precise information as to the NEs type than the systems based on flat NE type sets lacking taxonomy or other sort of definitions would. While the result of the traditional NE recognition approach gives only few basic types the entities might belong to, for example:

<Person>Lama Ole Nydahl</Person>,

After a semantic annotation has been performed, more specific NEs type is also given:

<ReligiousPersonID="http://..kim/Person111111">Lama Ole Nydahl</ReligiousPerson>

Having semantic annotation over the content KIM can index the content with respect to the mentioned entities. The metadata is used as an index pointer for the respective entity during the retrieval process. Because of the specifics of indexing that KIM provides, new (semantically-enhanced) access methods (or user-need definitions) are enabled. The user could specify queries consisting of constraints about the types of the entities, relations obtaining among entities, and/or entity's attributes. This means that the user could specify the NEs to be referred to in the content of interest, using name restrictions (e.g. a Person whose name ends with '*Alabama*'). An example of a query consisting of pattern restrictions over entities is:

> Give me all materials referring a Person that hasPosition "*CEO*" within a Company, locatedIn a Location with name "*UK*".

To answer the query, KIM applies the semantic restrictions over the entities in the instance base. The resulting set of entities is matched against the index, produced by the semantic indexing of the processed materials. Then the referring content is being retrieved with relevance ranking according to these NEs. Queries of this kind could also be combined with conventional keyword search (full text search) and thus, could benefit from the combination of both approaches (e.g. via intersection or union).

To illustrate the usefulness of semantic IR as compared to classical IR, consider the following example: the user attempts to retrieve the documents containing information about a telecom company positioned in Europe. The IR of usual sort cannot retrieve the materials about *Vodafone*, because it cannot link *Vodafone* to the description "*a telecom in Europe*", namely because it cannot infer that *Vodafone*, being a mobile operator, is a kind of a telecom; or that *Vodafone* being positioned in UK, means it is positioned in Europe, etc. Having the background information in the ontology and the instance base, and semantically-enabled information extraction modules, the semantic retrieval techniques would produce results that are superior to the ones of traditional IR in two ways. Having precise proper class information (e.g. that an entity is `CommercialOrganization`) searches like "*give me materials about Commercial Organizations that have ImpEx in their name*" would result only in materials that contain such organizations, while a FTS with the keyword *ImpEx* would result in all materials that contain the phrase regardless of the entity type associated (if any). Another scenario is when you search for an entity but you specify only one of its names, like *Beijing* and not *Pekin*. A FTS for *Beijing* would return just documents containing exactly this alias of the city, while the semantic IR would consult the instance base and implicitly expand the query to all the aliases of the city. This is not done explicitly since the name itself is not used as a pointer to the indexed content, instead KIM uses the entity ID which is the same no matter which alias has been used *Pekin* or *Beijing*.

5.1.2.2 Query Translation as a Wordnet-based expansion process

In the scenario foreseen in the documentation and publication processes in PrestoSpace also textual (i.e. keyword based) or natural language queries must be supported. The query is given in one language (e.g. English), and the AV items to be searched may be expressed in a different language (e.g. Italian). A typical example can be the query "*Iraqi war"* that, in order to properly trigger the retrieval process, should be translated into "*Guerra in Iraq*" as reportages or news about the conflict (*conflitto* in Italian) are meant to be the target.

As discussed in previous sections, this CLIR task is usually tackled by statistical or knowledge based approaches. The first are based on large corpora of aligned texts, i.e. texts in both languages, obtained via direct sentence translation. They show good precision and recall performances but they are strictly tight to the availability of parallel texts. Moreover, they are highly sensitive to domain variations, making thus hard their portability among domains and collections (i.e. different and heterogeneous AV data streams). Dictionary or Knowledge based approaches have a lower accuracy, but do not require training data. They are currently based on bilingual dictionaries. However, they are even more sensitive to domain variations. The result is that they are also difficult to adapt to the dynamically changing application domain. For example, Google does not use this technology.

Notice how MAD in PrestoSpace is targeted to heterogeneous streams in different languages (at the moment English and Italian) so that *adaptivity* is a critical issue. A proper approach to be pursued in PrestoSpace should thus combine the advantages of the different technologies. Hybrid techniques that integrate Knowledge Based methods and unsupervised learning not tight to parallel corpora seem the best solution. The KB methods should thus be exploited to minimize the requirements posed on training material during adaptation. General audience of the system (especially for the publication processes) is not meant to be specialized in any domain. General knowledge in the (source) language is the expected. General purpose resources (e.g. Wordnet, [Miller,1991]) can here play an important role. Statistical methods can also be used in forms of unsupervised learning from the target material without posing much emphasis on the manual annotation due for training purposes in supervised models. The availability of large data sets (due to the size of foreseen archives approached in PrestoSpace) suggests that unsupervised statistical models can be successfully applied in the medium and long run.

An important modeling issue, complementary to the above arguments, is the fact that the MAD solution to CLIR should be oriented to query rather than to document translation, as manual validation of

translations is not applicable, given the size of the archives. Documentation in PrestoSpace is devoted to verify, adjust and refine the set of metadata associated automatically to AV data. Any process foreseeing a translation of individual documents is thus just inapplicable in PrestoSpace. It would increase in fact, rather than limit, the documentation costs as it would require the documenter to validate also the translation metadata with an unaffordable additional effort. Translation of the queries on-the-fly seems here thus the best approach.

As a general indication and outcome of this survey, recent results in the area of unsupervised semantic disambiguation and document classification should be sought as important contributions to the query translation task. The first set of methods is useful as translations can be seen as senses and sense disambiguation of words in (source language) queries is a way to capture the proper lexical translations and build the corresponding target language queries. The second line of research is useful to explore the methodologies (i.e. vector spaces and metrics) adopted by weakly and successful supervised classification paradigms that can be useful to model the translation problem.

Semantic Disambiguation.

Recent work in word sense disambiguation suggest that any statistical approach to disambiguation should account for the significant differences that characterize sense distributions (i.e. probability distributions of senses given the words) across different domains. Predominant (i.e. most likely) senses in one domain (or collection) do not generalize, and change from one test scenario to another.

{McCarthy-et-al-2004} proposes a method to estimate predominant senses via a large untagged corpus, reaching high accuracy on standard WSD benchmarks. The underlying assumption is that predominant senses are not stable across domains and corpora, and that the notion of predominance can be modeled through an unsupervised corpus based analysis. In that work also syntagmatic evidence is taken to build thesauri of semantically similar words, that in turn results in an effective modeling of the proper sense probability distributions. The main limitation of that approach is that the sense distribution is modeled once for all contexts on the basis of the corpus adopted for unsupervised learning, However, the notion of predominance is even more dynamic than the one suggested by McCarthy and colleagues. It is certainly given by the entire corpus as it embodies most of the agreement among writers and readers regarding the semantic contribution of individual lexical items. The corpus should thus enter in process by providing source evidence to estimate “global” sense probabilities. However, predominance also depends on individual occurrences of target words, e.g. document sentences or paragraphs, or the shorter text chunks expressing the queries.

A significant step further in disambiguation would be to capture both the global evidence embodied by the corpus as well as the implicit domain local to each context. Each sentence in fact suggests a topical context where the semantics of target word is highly constrained. When such topical context is made available the disambiguation process is easier as it can exploit both evidences.

While estimating mot likely senses given the global or local domain is a problem that involve metrics of sense similarity and distances, the ways of capturing local domain evidence is a different task aiming to explore combination of several sources as a source of semantic evidences. The former aspect will be briefly discussed in the immediately following section. The ways of capturing local domains is the focus of the successive section.

Unsupervised semantic similarity estimation.

A large area of research related to semantic disambiguation is represented by the deep work made on word sense disambiguation (e.g. [lesk:86], [schultze:pedersen:95], [yarowsky:92], [yarowsky:95], [ng:96], [Agirre&Rigau, 1996], [Mihalcea:99], [dagan:2000], [buitelaar:01], [Pianta et el., 2002], [McCarthy et al., 2004], [Gliozzo:2005]). Most work has been done on ways of estimating semantic similarity between word (pairs) according to distance metrics based on resources (e.g. dictionary definitions as in [lesk:86], or lexical hierarchies as in [Agirre&Rigau, 1996]) or on corpora (e.g. [yarowsky:92], [Gliozzo:2005]).

A well known methods is based on the notion of conceptual density ([Agirre&Rigau, 1996], [Basili et al., 2004]). Conceptual density (CD) measures the similarity among target words as a function of the informational utility of a lexical hierarchy able to represent (i.e. subsume) most senses of the targets. It depends on the topological structure of the underlying lexical semantic network and given the target words results in the selection of one or more sub-hierarchies generalizing most of the target word senses. It results at the same time in:

- (explanatory outcome) a number of generalizations (higher level senses) of lexical word senses useful to subsume (i.e. explain) the target words

- (quantitative outcome) A quantification of the quality of the different reachable generalizations according to the conceptual density metrics: the most dense if the hierarchy wrt to the source words the higher is the score. This score can be easily interpreted as a similarity score and give also rise to a distance metrics. A probabilistic interpretation of the scores allows to trigger sense disambiguation as a statistical task ([Basili et al, 2004]).

In [Basili et al, 2004] efficient algorithms for the derivation of both explanatory and quantitative information about the best generalizations as they can be found in Wordnet [Miler, 91] are presented. Notice how one of the advantages of this metrics is that it applies to pairs as well as *n*-ary sets of words. Moreover, CD only depends on the network structure and does not require any training over labeled examples.

The source information of this method is a set of associated words *w'* that should suggest the proper lexical sub-hierarchies for disambiguation, i.e. isolating the best senses and neglecting the odd senses irrelevant for the target set. Given a target noun *w* (e.g. a noun in a query) ways of building the target set for trigger the conceptual density disambiguation method may depend on

- *Syntagmatic evidence*, i.e. grammatical properties local to the context of *w*, e.g. the syntax of the query.
- *Paradigmatic evidence*, e.g. the topological properties of the ontological context for *w*. If *w* is the name of a concept *Cw* in an ontology, aliases *w'* of *Cw* or the names *w'* of super-ordinates or sub-ordinates concepts of *Cw* can be added to the target set. The resulting outcome of the CD method can be seen as an explanation (disambiguation) of *w* as the name of *Cw*.
- *Associative evidence*, as words *w'* can be collected from those contexts surrounding each occurrence of *w* in an underlying corpus.

The above properties of the CD method makes it appealing for query translation. For each word w of the query in fact, a specific target set can be derived with the above mentioned different methods. For example, given the noun *conflict*, and a query like "*reportages about Iraqi conflict",* syntagmatic or associative evidence can be collected outside the query[5]to expand this word into a target set like: {*war, battle, enemy, army, …,* } expressing other synonyms, syntagmatically equivalent or topically associated words. When CD is applied to the derived target set it will converge more easily to the proper sense 3 of the conflict word, i.e. *"a hostile meeting of opposing military forces in the course of a war",* rather than the irrelevant sense "*opposition between two simultaneous but incompatible feelings*". Only in this case the proper translations (i.e. *conflitto* e *Guerra*) can be activated and added to the target language (i.e. Italian) query.

An important consequence of sense preferences obtained over Wordnet in the above manner is that individual senses (i.e. synsets in the American English, i.e. Princeton, Wordnet) can be mapped in other languages as well, given the availability of multilingual extensions of Wordnet, called Multiwordnet [Pianta et al., 2002]. This is thus not only true for the English-Italian language pair (of interest in PrestSpace) but also for a variety of other languages. Individual senses, i.e. synsets, are mapped across languages so that Italian names (i.e. synsets expressed into Italian nouns) are available: the selection of a synset in one language can easily trigger the extraction of corresponding nouns in the target language, as a form of translation via query expansion.

LSA-based domain modeling.

For each target noun *w* in a query, the unsupervised similarity metrics over Wordnet, as suggested in the previous section, requires a target set, i.e. a set of nouns *w'* semantically associated with *w*. Associative evidence occurs when lexical entries (like conflict) are mapped into sets of topically related words, like *war, battle, enemy* or *army.* Notice that we are not looking necessarily for paradigmatic relations, like for example synonymy. The notion of semantic field here can be useful as the example demonstrate: the overall set *war, battle, enemy* or *army* constitutes a semantic field of *war*. Target sets should provide us with information for disambiguation and semantic fields can play here a role. In the example they enable to catch the proper translation more easily than in a more complex reasoning or statistical manner. If a method to map individual occurrences of words (e.g. queries) to semantic fields is available, than a lexical description of a semantic field would be accessible and CD disambiguation would make its work.

[5] In the next section we will show a model for acquiring such an expansion from unlabelled thematic corpora.

The notion of semantic field is rather close to the notion of local semantic domain discussed before. As Latent Semantic Indexing methods suggest, a local domain can be obtained as regions in the LSA space derived by SVD (Singular Value Decomposition). The semantic effect of the LSA method is to generate a space where terms and documents are defined in the same space. The virtual document given by a query as well as the target nouns (to be translated) when mapped via LSA transformations, is a point in the LSA space and is thus close to other semantically related terms. The surroundings of such region can be assumed to be populated with semantically correlated words. It is thus a useful model for the notion of the target set needed for the CD estimation. LSA-like analysis can then be integrated with CD as a preparatory step to derive associative lexicons. First SVD decomposition of the source term-document matrix is run by deriving the linear transformations required for mappings. Then for each query, a virtual document in the transformed LSA space is derived. The sets of terms close enough to it are computed and added to the target set. CD disambiguation can then be run over the resulting lexical set and sense preferences are derived.

The LSA method appears promising in the PrestoSpace application, as the SVD decomposition, it is based on, can be applied without much effort:

- It does not require any annotated corpus as it runs over the simple term-document matrix
- It can be fed with information derived from any collection, and in particular with the AV document already available (i.e. ASR from video or audio programs).
- It is robust with respect to errors in transcriptions
- It is language neutral so that it can be applied to material in different languages

Dynamic Domain-driven Query Translation.

The result of the integration of the above methods is a fully unsupervised process of query translation whose applicability to PrestoSpace-like scenarios is very high. First, it consistently exploits the wide available resources of semantic evidence (i.e. corpora dealing with the user/application domains of interest), so realizing a powerful model of query translation *on-the-fly*. The scale of the AV archives targeted by MAD methodologies in PrestoSpace guarantee the availability of such large collections of (ASR transcribed) AV material. They are thus able to consistently trigger LSA modeling as well as to estimate sense disambiguation probabilities with precision.

Second, the methodology is highly portable throughout domains and applications as it combines learning methods that are fully unsupervised. This is thus ideal for a technology like PrestoSpace where heterogeneous archives are targeted. For this reason it represents an effective (and efficient) approach that should be explored in PrestoSpace.

5.2 Benchmarking

A final remark is needed for discussing aspects related to the delivery of the CLIR methodologies in PrestoSpace MAD. As the size and heterogeneity of the AV archives foreseen in PrestoSpace are challenging for the IR technologies, a specific methodology must be followed to assess the resulting CLIR models adopted in the MAD subsystem.

Objective measures in IR are only allowed when quantitative evaluation is applied, in terms of *precisions* and *coverage* (also called *recall*) over controlled data sets. These data sets work as oracles for the target IR system and are specifically tailored to the target applications. Different results are usually obtained when similar measures are applied, for the same IR systems/models, to different data sets. The superiority of a methodology with respect to others has to be established by investigating how well the system performs on data derived from collections that are close to (i.e. significantly overlapping with or totally immerse in) the data over which the final system will work. The activity of producing controlled material, i.e. oracles for the different sub-tasks, is very costly but it is the only way to derive objective estimations of the realistic system performances in real scenarios.

The kind of capabilities expected for the CLIR system in MAD are very different. They range from AV data topical categorization to Web alignment; from retrieval of individual news items to automatic generation of hyperlinks, to query translation. Each of the above mentioned tasks will require a careful specific analysis of performances, coupled with error analysis for fine-tuning in the last phases of the project. For example, mistakes in the ontology-based retrieval may depend on lacks in coverage of the available Knowledge Bases (i.e. missing instances of some concept or missing properties or relations in the ontology itself), in problems of the decision-making algorithms required to deal with ambiguous

cases or in misleading annotations produced over AV data sets by the IE algorithms. Similar problems may arise in cross-linguistic information retrieval where query translation may fail due to lacks in the Wordnet lexical coverage or in errors in the sense disambiguation algorithms (see section 5.2.1). As the error analysis is essential for the tuning of the MAD system before the end of the project lifecycle, the size and quality of the controlled data sets for the different activities is a critical issue for the final success of the project.

It is thus suitable that specific effort is spent by the technical groups involved in the CLIR activities as well as by the user groups involved in MAD to build extensive and reliable controlled (i.e. annotated) data for performance evaluation and error analysis. Specific collections should thus be built as repositories from which specific oracles can be extracted. These will involve: oracles for the benchmarking of the full text retrieval (i.e. sets of queries with the corresponding correct answers as individual AV items satisfying the requests); oracles for the ontology based retrieval as complex queries based on the available concepts and relations coupled with exact answers; oracles for cross-linguistic retrieval including sets of queries in the source language with AV answers in (possibly more than one) target language. Notice that specific parallel corpora should be created made by multimedia material in different languages insisting on the same domain and events: for example a collection of TV broadcasts from RAI and BBC over the same time period could be used as a starting point both for training and testing CLIR models. A specific attention should be also devoted to methods for evaluating automatic hyperlinking capabilities among individual AV items as well as Web documents, as expected in the MAD semantic analysis. Protocols for evaluating the quality and usefulness of individual links should be defined and, accordingly, test material should be created for quantitative evaluation.

Although the costs of the specific development of test material is high, the resulting resource would be invaluable in light of the technical assessment internal to the project as well as a general guideline and reference for the future development of the multimedia data indexing and delivery technology in Europe.

Bibliography

[ACE, 2001] NIST ACE - Automatic Content Extraction Benchmark, at http://www.nist.gov/speech/tests/ace/index.htm

[Agirre&Rigau, 1996], E. Agirre and G. Rigau, Word Sense Disambiguation using Conceptual Density, Proceedings of the 16$th International Conference on Computational Linguistics (COLING-96), Copenhagen, Denmark, 1996.

[Anick, 1994] Anick, P.J. Adapting a full-text information retrieval system to the computer troubleshooting domain, Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 349-358, 1994.

[Appelt et al., 1993] D. Appelt and J. Hobbs and J. Bear and D. Israel and M. Kameyama and A. Kehler and D. Martin and K. Meyers and M. Tyson, "SRI International FASTUS system: MUC-6 test results and analysis", In Proceedings of 16th MUC, Columbia, MD. 1993.

[Ballesteros, L., & Croft, W. B]. (1997). Phrasal translation and query expansion techniques for cross-language information retrieval. Retrieved October 30, 2004, from http://ciir.cs.umass.edu/publications/

[Bartell, et al. SIGIR '92] Bartell, B.T., Cottrell, G.W., Belew, R.K. Latent Semantic Indexing is an optimal special case of Multidimensional caling, Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 161-167, 1992.

[Basili et al, 2005] Roberto Basili, Marco Cammisa and Alessandro Moschitti, Effective use of wordnet semantics via kernel-based learning. In Proceedings of the 9th Conference on Computational Natural Language Learning (CoNLL 2005), Ann Arbor(MI), USA, 2005

[Basili et al., 1998] Roberto Basili, Maria Teresa Pazienza, and Fabio Massimo Zanzotto. Efficient parsing for information extraction. In Proc. of the ECAI98, Brighton, UK, 1998.

[Basili et al., 2004], R. Basili, M. Cammisa, F.M. Zanzotto, A semantic similarity measure for unsupervised semantic disambiguation, Proceedings of the Language, Resources and Evaluation LREC 2004 Conference, Lisbon, Portugal, 2004.

[Bekkerman et al., 2001] Ron Bekkerman, Ran El-Yaniv, Naftali Tishby, and Yoad Winter. On feature distributional clustering for text categorization. In Proceedings of the 24th annual international ACM

SIGIR conference on Research and development in information retrieval, pages 146-153. ACM Press, 2001.

[Berry, 1995] Berry, M.W., Dumais, S.T., O'brien, G.W. Using linear algebra for intelligent information retrieval, SIAM Review, Vol. 37, No. 4, pp. 573-595, December 1995.

[Berry, M., & Young, P]. (1995). Using latent semantic indexing for multilingual information retrieval. Computers and the Humanities, 29(6), 413-429.

[Bikel et al., 1999] An Algorithm that Learns What's in a Name," Daniel Bikel, Richard Schwartz, & Ralph M. Weischedel, Journal of Machine Learning, vol. 34, n. 1-3, 211-231, (1999).

[Braschler, M., Peters, C., & Schauble, P.] (2000). Cross-language information retrieval (CLIR) track overview. Retrieved October 30, 2004, from http://trec.nist.gov/pubs/trec8/papers/trec8ov.pdf

[Buitelaar and Sacaleanu, 2001], P. Buitelaar and B. Sacaleanu, Ranking and Selecting Synsets by Domain Relevance, Proc.\ of NAACL Workshop WordNet and Other Lexical Resources: Applications, Extensions and Customizations, Pittsburgh, 2001.

[Caropreso et al., 2001] Maria Fernanda Caropreso, Stan Matwin, and Fabrizio Sebastiani. A learner-independent evaluation of the usefulness of statistical phrases for automated text categorization. In Idea Group Publishing, Hershey, US, 2001.

[Charniak, 2000] Eugene Charniak. A maximum-entropy-inspired parser. In Proceedings of the 1st Meeting of the North American Chapter of the ACL, pages 132-139, 2000.

[Cohen, 1995] Cohen, J. Highlights: Language- and Domain-Independent Automatic Indexing Terms for Abstracting, Journal of the American Society for Information Science, 46(3), pp. 162-174, 1995.

[Collins, 1997] Michael Collins. Three generative, lexicalized models for statistical parsing. In Proceedings of the ACL and EACLinguistics, pages 16-23, Somerset, New Jersey, 1997.

[Cooper et al., 1992] Cooper, W.S., Gey, F.C., Dabney, D.P. Probabilistic retrieval based on staged logistic regression, Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp.198-210, 1992.

[Cooper, 1991] Cooper, W.S. Inconsistencies and misnomers in probabilistic IR, Proceedings of the 14th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 57-61, 1991.

[Cooper, 1994] Cooper, W.S. The formalism of probability theory in IR: A foundation or an encumbrance?, Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 242-247, 1994.

[Cooper, 1995] Cooper, W.S. Some inconsistencies and misidentified modeling assumptions in probabilistic information retrieval, ACM Transactions on Information Systems, Vol. 13, No. 1, pp. 100-111, January 1995.

[Croft, W. B., Broglio, J., & Fujii, H.] (1996). Applications of multilingual text retrieval. Retrieved October 29, 2004, from the IEEE database.

[Dagan,2000], I. Dagan, Contextual Word Similarity, in Handbook of Natural Language Processing, Rob Dale and Hermann Moisl and Harold Somers Eds., Marcel Dekker Inc, 2000.

[Damashek et al., 1995] Damashek, M. Gauging similarity with n-grams: Language-independent categorization of text, Science, Volume 267, pp. 843-848, February 1995.

[Davis, M.W., & Dunning, T. E.] (1995). A TREC evaluation of query translation methods for multi-lingual text retrieval. Retrieved October 30, 2004, from http://trec.nist.gov/pubs/trec4/t4_proceedings.html

[Deerwester et al., JASIS, 1990] Deerwester, S., Dumais, S., Furnas, G., Landauer, T., Harshman, R. Indexing by latent semantic analysis, Journal of the American Society for Information Science, 41(6), pp. 391-407, 1990.

[Deerwster, 1990] S. Deerwester, "Indexing by Latent Semantic Indexing", Journal of the American Society for Information Science, 41(6), 1990

[Dowman et al., 2005] Web-Assisted Annotation, Semantic Indexing and Search of Television and Radio News. In Proc. of the 14th International World Wide Web Conference. Chiba, Japan, 2005.

[Dumais et al., 1998] Susan T. Dumais, John Platt, David Heckerman, and Mehran Sahami. Inductive learning algorithms and representations for text categorization. In Georges Gardarin, James C. French,

Niki Pissinou, Kia Makki, and Luc Bouganim, editors, Proceedings of CIKM-98, 7th ACM International Conference on Information and Knowledge Management, pages 148-155, Bethesda, US, 1998. ACM Press, New York, US.

[Faaborg, 2001] Alexander J. Faaborg, Cornell University Human Computer Interaction Group, "Leveraging Metadata to Improve Information Retrieval in Directory Interfaces", 2001

[Fluhr, C.] (1996). Multilingual information retrieval. In A. Zaenen (Ed.), Survey of the State of the Art in Human Language Technology. Retrieved October 29, 2004, from http://cslu.cse.ogi.edu/HLTsurvey/ch8node7.html#SECTION85

[Furnkranz et al., 1998] J. Furnkranz, T. Mitchell, and E. Rilof. A case study in using linguistic phrases for text categorization on the www. In Working Notes of the AAAI/ICML, Workshop on Learning for Text Categorization, 1998.

[Furnkranz, 1998] Johannes Furnkranz. A study using n-gram features for text categorization. Technical report oefai-tr-9830, Austrian Institute for Artificial Intelligence., 1998.

[Gliozzo, Strapprava, 2005], A. Gliozzo and C. Strapparava, Domain Kernels for Text Categorization, Proceedings of the Ninth Conference on Computational Natural Language Learning (CoNLL-2005), 2005.

[Haddouti, H.] (1999). Survey: multilingual text retrieval and access. Retrieved October 29, 2004,

[Han, 2001] "Automatic Document Metadata Extraction using Support Vector machines" Hui Han, Department of Computer Science and Engineering, The Pennsylvania State University University, 2001

[Harman, 1992] Harman, D. User-friendly systems instead of user-friendly front-ends, Journal of the American Society for Information Science, 43(2), pp 164-174, 1992.

[Hayashi, Y., Kikui, G., & Susaki, S.] (1997). TITAN: a cross-linguistic search engine for the WWW. Retrieved November 1, 2004, from http://www.slt.atr.jp/~gkikui/pub_web_access.html

[Hull, D. A., & Grefenstette, G.] (1996). Querying across languages: a dictionary-based approach to multilingual information retrieval. In Sparck Jones, K., & Willet, P. (Eds.), Readings in Information Retrieval. San Francisco, CA: Morgan Kaufmann Publishers.

[Hull, SIGIR '94] Hull, D. Improving text retrieval for the routing problem using Latent Semantic Indexing, Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp.282-291, 1994.

[Joachims, 1997] Thorsten Joachims. A probabilistic analysis of the rocchio algorithm with tfidf for text categorization. In Proceedings of ICML97 Conference. Morgan Kaufmann, 1997.

[Kahle et al., 1991] B. Kahle and A. Medlar, Information Servers", Connexions - The Interoperability Report, 5(11), Nov. 1991.

[Kashyap et al., 1996] Vipul Kashyap, Kshitij Shah, Amit Sheth, "Metadata for building the multimedia patch quilt", 1996.

[Kiryakov et al, 2005] OWLIM - a Pragmatic Semantic Repository for OWL. in Proc. of Int. Workshop on Scalable Semantic Web Knowledge Base Systems (SSWS 2005), WISE 2005, 20 Nov, New York City, USA. Springer-Verlag LNCS series, LNCS 3807, pp.182-192.

[Kiryakov et al., 2004] Semantic Annotation, Indexing, and Retrieval. Elsevier's Journal of Web Sematics, Vol. 2, Issue (1), 2005. http://www.websemanticsjournal.org/ps/pub/2005-10

[Kopena, 2000] J. Kopena, http://plan.mcs.drexel.edu/projdesign/software/damljesskb/

[Korfhage, 1997] Korfhage, R.R. Information Storage and Retrieval, John Wiley and Sons, New York, 1997.

[Lee, 1994] 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, pp. 182-190, 1994.

[Lesk,1986], M. Lesk, Automated Sense Disambiguation Using Machine-Readable Dictionaries: How to Tell a Pine Cone from an One Cream Cone, Proceedings of the 1986 SIGDOC Conference, Toronto, Canada, 1986.

[Li Ding et al., 2001] Department of Computer Science and Electronic Engineering University of Maryland Baltimore County, Baltimore MD 21250, USA, 2001.

[Lin, C. H., & Chen, H.] (1996). An automatic indexing and neural network approach to concept retrieval and classification of multilingual (Chinese-English) documents. Retrieved November 2, 2004, from http://ai.bpa.arizona.edu/papers/chinese93/chinese93.html

[Malet et al., 1999] "A Model for Enhancing Internet Medical Document Retrieval with "Medical Core Metadata" Gary Malet, DO, Felix Munoz, Richard Appleyard, PhD, Williamo Hersh, MD, JAMIA 1999

[McCarthy et al., 2004], D. McCarthy and R. Koeling and J. Weeds and J. Carroll, Finding predominant senses in untagged text, Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics, Barcelon, Spain, 280-287, 2004.

[Mihalcea, 1999], R. Mihalcea and D. Moldovan, A Method for Word Sense Disambiguation of Unrestricted Text, Proceedings of ACL 1999, College Park Maryland, 1999.

[Miller,90], G. Miller, An On-Line Lexical Database, International Journal of Lexicography, 13,4,235-312, 1990

[Mitchell, 1997] Tom Mitchell, editor. Machine Learning. McCraw Hill, 1997.

[Mladeni´c and Grobelnik, 1998] Dunja Mladeni´c and Marko Grobelnik. Word sequences as features in text-learning. In Proceedings of ERK-98, the Seventh Electrotechnical and Computer Science Conference, pages 145-148, Ljubljana, SL, 1998.

[Moschitti and Basili, 2004] Alessandro Moschitti and Roberto Basili, Complex Linguistic Features for Text Classification: a comprehensive study. In proceedings of the 26th European Conference on Information Retrieval Research (ECIR 2004), Sunderland, U.K., 2004.

[Moschitti, 2003] Alessandro Moschitti. Is text categorization useful for word sense disambiguation or question answering? In Proceedings of the 2nd Annual Research Symposium of the Human Language Technology Research Institute, Dallas, Texas, 2003.

[MUC-3] Proceedings of the 3rd conference on Message understanding 1991, San Diego, California May 21 - 23, 1991.

[MUC-4] Proceedings of the 4th conference on Message understanding 1992, McLean, Virginia June 16 - 18, 1992.

[MUC-5] Proceedings of the 5th conference on Message understanding 1993, Baltimore, Maryland August 25 - 27, 1993.

[MUC-6] Proceedings of the 6th conference on Message understanding 1995, Columbia, Maryland November 06 - 08, 1995.

[Ng, 1996], H. T. Ng, Getting Serious about Word Sense Disambiguation, Proceedings of the ACL SIGLEX Workshop on Tagging Text with Lexical Semantics: Why, What, and How?, Washington, DC, USA, 1996.

[Nigam et al., 1999] K. Nigam, J. Lafferty, and A. McCallum. Using maximum entropy for text classification. In IJCAI-99 Workshop on Machine Learning for Information Filtering, pages 61-67, 1999.

[Oard, D. W., & Dorr, B. J.] (1996). A survey of multilingual text retrieval. Retrieved October 27, 2004, from http://citeseer.ist.psu.edu/oard96survey.html

[Oard, D. W.] (1997). Cross-language text retrieval research in the USA. Retrieved November 1, 2004, from

[Oard, D. W.] (1997). Serving users in many languages: cross-language information retrieval for digital libraries. D-Lib Magazine. Retrieved November 1, 2004, from http://www.dlib.org/dlib/december97/oard/12oard.html

[Pazienza,1997], Maria Teresa Pazienza, Information Extraction: A Multidisciplinary Approach to an Emerging Information Technology, International Summer School, SCIE-97, Frascati, Italy, 14-18, 1997, Springer, Lecture Notes in Computer Science, 1299,ISBN:3-540-63438-X, 1997.

[Pearce et at., 1996] Pearce, C., Nicholas, C. TELLTALE: Experiments in a dynamic hypertext environment for degraded and multilingual data, Journal of the American Society for Information Science, 47(4), pp. 263-275, 1996.

[Pianta et al., 2002], Emanuele Pianta, Luisa Bentivogli, Christian Girardi, MultiWordNet: developing an aligned multilingual database, Proceedings of the First International Conference on Global WordNet, Mysore, India, 2002.

[Pirolli et. al, 1995] Peter Pirolli , Stuart Card, Information foraging in information access environments, Conference proceedings on human factors in computing systems, p.51-58, May 07-11, 1995, Denver, Colorado, United States.

[Popov et al., 2004] KIM - a semantic platform for information extraction and retrieval, Journal of Natural Language Engineering, Vol. 10, Issue 3-4, Sep 2004, pp. 375-392, Cambridge University Press.

[Porter, 1980] Porter, M. An Algorithm for Suffix Stripping, Program, 14(3), pp. 130-137, 1980.

[Porter, 1997] Porter, M. An Algorithm for Suffix Stripping, in Readings in Information Retrieval, Sparck Jones and Willett, eds., pp. 313-316, 1997.

[R. Cost et al. 2001] , A Case Study in the Semantic Web and DAML. In International Semantic Web Working Symposium (SWWS), July 2001].

[Raskutti et al., 2001] Bhavani Raskutti, Herman Ferr´a, and Adam Kowalczyk. Second order features for maximising text classification performance. In Proceedings of ECML-01, 12th European Conference on Machine Learning. Springer Verlag, Heidelberg, DE, 2001.

[Rete et al, 1982] C. Forgy. Rete: A fast algorithm for the many object pattern match problem. Articial Intelligence, 1982.

[Riloff, 1996] Ellen Riloff. Automatically generating extraction patterns from untagged text. In AAAI/IAAI, Vol. 2, pages 1044-1049, 1996.

[Robertson et. al, 1998] George Robertson , Mary Czerwinski , Kevin Larson , Daniel C. Robbins , David Thiel , Maarten van Dantzich, Data mountain, Proceedings of the 11th annual ACM Symposium on User Interface Software and Technology, p.153-162, November 01-04, 1998, San Francisco, California, United States

[Salton and Buckley, 1988] Salton, G., Buckley, C. Term-weighting approaches in automatic text retrieval, Information Processing & Management, 24(5), pp. 513-523, 1988.

[Salton et al., 1983] Salton, G., Fox, E.A., Wu H. Extended boolean information retrieval, Communications of the ACM, 26(11), pp. 1022-1036, 1983.

[Salton, 1989] Salton, G. Automatic text processing: The transformation, analysis, and retrieval of information by computer, Addison-Wesley, Reading, MA, 1989.

[Salton, G.] (1970). Automatic processing of foreign language documents. Retrieved November 1, 2004,

[Schutze and Pedersen, 1995], Information retrieval based on word senses, Symposium on Document Analysis and Information Retrieval, 1995.

[Schutze et al, 1997] Schutze, H., Silverstein, C. A Comparison of Projections for Efficient Document Clustering, Proceedings of the 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 74-81, 1997.

[Shah et al, 1996] Urvi Shah and Tim Finin and Yun Peng, "Information Retrieval on the semantic web", University of Maryland, Baltimore, MD 21227, 1996.

[Sheridan, P., & Ballerini, J. P.] (1996). Experiments in multilingual information retrieval using the SPIDER system. October 30, 2004

[Shklar, 2002] "InfoHarness: Use of Automatically Generated Metadata for Search and Retrieval of Heterogeneous Information" Leon Shklar et al.Bell Communications Research, 1991

[Simmons, 1965], R. F. Simmons. Answering English questions by computer: A survey. Communications of the ACM, 8(1):53--70, 1965.

[Smeaton, 1999] Alan F. Smeaton. Using NLP or NLP resources for information retrieval tasks. In Tomek Strzalkowski, editor, Natural language information retrieval, pages 99-111. Kluwer Academic Publishers, Dordrecht, NL, 1999.

[Strzalkowski and Carballo, 1997] Tomek Strzalkowski and Jose Perez Carballo. Natural language information retrieval: TREC-6 report. In Text REtrieval Conference, 1997.

[Strzalkowski and Jones, 1996] Tomek Strzalkowski and Sparck Jones. NLP track at trec-5. In Text REtrieval Conference, 1996.

[Strzalkowski et al., 1998] Tomek Strzalkowski, Gees C. Stein, G. Bowden Wise, Jose Perez Carballo, Pasi Tapanainen, Timo Jarvinen, Atro Voutilainen, and Jussi Karlgren. Natural language information retrieval: TREC-7 report. In Text REtrieval Conference, pages 164-173, 1998.

[Strzalkowski et al., 1999] Tomek Strzalkowski, Jose Perez Carballo, Jussi Karlgren, Anette Hulth Pasi Tapanainen, and Timo Jarvinen. Natural language information retrieval: TREC-8 report. In Text REtrieval Conference, 1999.

[Suen, 1979] Suen, C.Y. N-gram statistics for natural language understanding and text processing, IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-1 (2), pp. 164-172, 1979.

[Sussna, 1993] M. Sussna. Word sense disambiguation for free-text indexing using a massive semantic network. In ACM Press New York, editor, The Second International Conference on Information and Knowledge Management (CKIM 93), pages 67-74, 1993.

[Tan et al., 2002] C.-M. Tan, Y.-F. Wang, and C.-D. Lee. The use of bigrams to enhance text categorization. accepted for publication in Information Processing and Management, 2002.

[Terziev et al., 2005] PROTON ontology, SEKT IP deliverable, http://proton.semanticweb.org/D1_8_1.pdf

[TREC 8, 1999] NIST Special Publication 500-246: The Eighth Text Retrieval Conference (TREC 8), 1999.

[van Rijsbergen, 1979] van Rijsbergen, C.J. Information Retrieval (2nd ed.), Butterworths, London, 1979.

[Voorhees, 1993] Ellen M. Voorhees. Using wordnet to disambiguate word senses for text retrieval. In Robert Korfhage, Edie M. Rasmussen, and Peter Willett, editors, Proceedings of the 16th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval. Pittsburgh, PA, USA, June 27 - July 1, 1993, pages 171-180. ACM, 1993.

[Voorhees, 1994] Ellen M. Voorhees. Query expansion using lexical-semantic relations. In W. Bruce Croft and C. J. van Rijsbergen, editors, Proceedings of the 17th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval. Dublin, Ireland, 3-6 July 1994 (Special Issue of the SIGIR Forum), pages 61-69. ACM/Springer, 1994.

[Voorhees, 1998] Ellen M. Voorhees. Using wordnet for text retrieval. In C. Fellbaum, editor, WordNet: An Electronic Lexical Database, pages 285-303. The MIT Press, 1998.

[W. V. Quine. Naming, Necessity and Natural Kinds, chapter Natural Kinds. University Press, 1977]

[Wilks & Catizone, 1999] Y.Wilks, R. Catizone, Can We Make Information Extraction More Adaptive in Pazienza (Ed) Information Extractions Berlin Springer-Verlag, 1999.

[Yang and Pedersen, 1997] Yiming Yang and Jan O. Pedersen. A comparative study on feature selection in text categorization. In Proceedings of ICML-97, pages 412-420, Nashville, US, 1997.

[Yarowsky,1992], D. Yarowsky, Word-Sense Disambiguation Using Statistical Models of {Roget}'s Categories Trained on Large Corpora, Proceedings of the 14th International Conference on Computational Linguistics (COLING-92), Nantes, France, 1992.

[Yarowsky,1995], D. Yarowsky, Unsupervised Word Sense Disambiguation Rivaling Supervised Methods, Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics (ACL-95), Cambridge, MA, 1995.

[Zamora et al., 1981] Zamora, E.M., Pollock, J.J., Zamora, A. The use of trigram analysis for spelling error detection, Information Processing and Management, 17, pp. 305-316, 1981.

[Zhang et al., 2003] Lei Zhang, Yong Yu, Jian Zhou, ChenXi Lin, APEX Data and Knowledge Management Lab, Dept of Computer Science and Engineering, Shanghai University, Shanghai, 200030 CHINA, 2003