TUCUXI: the InTelligent Hunter Agent for Concept Understanding and LeXical ChaIning*

Roberta Benassi Sonia Bergamaschi Maurizio Vincini
Universita' degli Studi di Modena e Reggio Emilia
Dipartimento di Ingegneria dell'Informazione
905, Via Vignolese - 41100 Modena, Italy
lastname.firstname@unimore.it

Abstract

In this paper we present TUCUXI, an intelligent hunter agent that replaces traditional keywords-based queries on the Web with a user-provided domain ontology, where meanings to be searched are not ambiguous. TUCUXI judges the relevance of the retrieved pages by matching the domain ontology against a simplified, but semantically rich, document representation (Map of Meanings). The Map of Meanings extraction involves the Lexical Chaining *technique, from the Natural Language Processing (NLP) research field.*

1. Introduction

Query results from traditional search engines are often less of use because of the information overload (pages and pages of links) and the semantic ambiguity of natural language. Current retrieval techniques are inadequate: firstly, keyword-based queries do not capture the user's needs, since many terms can express the same concept (synonymy) or a term can have several meanings (polysemy). Secondly, to rapidly answer queries, many search engines employ a *bag-of-word* representation of documents where intended meanings and semantic relations between meanings are completely lost. Thus, we think that the new generation of search tools should focus its efforts over three main aspects: (**a**) an ontology-based expression of the user's needs, where the meanings and the concepts to be searched are not ambiguous; (**b**) the ability to exploit semantics for documents retrieval; (**c**) the introduction of software personal agents to carry out sophisticated tasks such as intelligent strategies for Web exploration.

The SEWASIE project (**SE**mantic **W**eb and **A**gents in **I**ntegrated **E**conomies) aims at outperforming current search engines enabling users to easily find strategic information via an intelligent and integrated access to a collection of heterogeneous data sources. In order to provide such an access, current *Semantic Web*[6] approaches rely on the a-priori existence of (generic) ontologies to enrich information sources with meaningful machine-processable metadata. Instead, the SEWASIE's component called MOMIS OntologyBuilder creates a more accurate conceptualization of the domain of interest (domain ontology) starting from sources to be integrated. In this scenario, discovering new (interesting) Web sources could enrich the domain ontology.

TUCUXI[1] is the InTelligent HUunter Agent for Concept Understanding and LeXical ChaIning. It adopts a domain ontology as the expression of the user's needs. The relevance of the retrieved pages is judged by matching the user-provided domain ontology against the *Map of Meanings*, a simplified, but semantically rich, document representation. In Section 2 we discuss how the output of an integration process within the MOMIS OntologyBuilder can create a domain ontology, while Section 3 illustrates our contribution for information discovery: we firstly explain how to extract the Map of Meanings through the *Lexical Chaining process*; secondly, how to semantically match the domain ontology against the Map of Meanings; thirdly, how to automatically explore the Web looking for concepts rather than keywords. Section 4 presents some encouraging results from the comparison between TUCUXI and Google, one of the most prestigious keyword-based search engines.

* This work is supported in part by the EC's 5^{th} Framework IST program through the Sewasie project (IST-2001-34825) in the Semantic Web Action Line. The SEWASIE consortium includes the University of Modena and Reggio Emilia (Sonia Bergamaschi, coordinator) the Universities of Aachen RWTH (M. Jarke), Roma La Sapienza (M. Lenzerini, T. Catarci), Bolzano (E. Franconi) and the companies IBM Italia (G. Vetere), Thinking Networks AG (C. Engels) and CNA (A. Tavernari) as user organization.

[1] Pronunciation 'tookooshee". TUCUXI is a South American river dolphin which employs echolocation to find fishes in murky waters.

University *Source*

```
Research_Staff(name,dept_code,room_code)
 FK: dept_code references Department
 FK: room_code references Room
School_Member(name,faculty,year)
Department(dept_name,dept_code,budget)
Room(room_code,seats_number,notes)
```

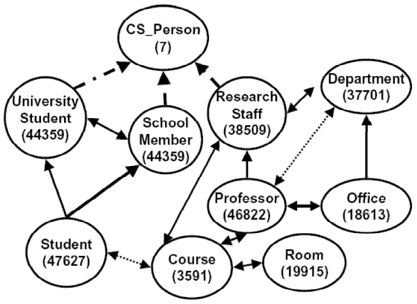
Computer_Science *Source*

```
CS_Person(first_name,last_name)
Professor: CS_Person(title,belongs:Office,rank)
Student: CS_Person(year,takes:set(Course),rank)
Office(description)
Course(course_name,taught_by:Professor)
```

Tax_Position *Source*

```
University_Student(name,stud_code,fac_name,fee)
```

(a) ***Sources to be integrated.***



(b) ***DCT: thick arrows = BT, NT, SYN relationships; thin ones= RT relationships; thin ones= RT relationships; dashed ones= inferred relationships, solid ones= explicitly given relationships.***

Figure 1. UNI Scenario DCT, where a WordNet synset is assigned to each class name.

2. MOMIS OntologyBuilder

Unlike current *Semantic Web* approaches, the MOMIS OntologyBuilder starts from local sources to generate an accurate conceptualization of the domain of interest. More precisely, it provides a reconciled *Global Virtual View* (GVV) of heterogeneous data sources[5]. To generate the GVV, heterogeneous data sources are firstly described in a standard way, i.e. wrappers extract local source schemata and translate them into ODL_{I3}[4], a modified version of the *Object Definition Language*. Secondly, the integration designer is asked to annotate each item of the sources description (names of classes and attributes) with one or more meanings according to the WordNet lexicon ontology. WordNet[14] is a lexical database that organizes nouns, verbs, adjectives and adverbs into synonyms sets (*synsets*) each representing one underlying lexical concept. The price of imposing the syntactic categorization is a certain amount of redundancy that conventional dictionaries avoid; on the other hand, the advantage is that fundamental differences in the semantic organization of syntactic categories can be clearly seen and exploited: WordNet is founded on the semantic relations between synsets. When no satisfactory synsets can be associated to a item, the integration designer can extend the WordNet ontology on (her)his own, thanks to WNEditor[3]. Then, starting from the annotated sources, MOMIS generates a *Domain Common Thesaurus* (DCT), which contains intra and inter-schema knowledge in the form of synonyms (SYN); broader/narrower terms (BT/NT); meronymy/holonymy (RT); equivalence (SYN$_{ext}$) and generalization (BT$_{ext}$) relationships. The DCT is incrementally built by adding *schema-derived relationships* (automatic extraction of intra schema relationships from each schema separately), *lexicon-derived relationships* (inter-schema lexical relationships derived by the annotated sources and WordNet interaction), *designer-supplied relationships* (specific domain knowledge capture) and *inferred relationships* (via equivalence and subsumption computation). Then, the DCT is exploited together with local sources to generate and semi-automatically annotate the global reconciled schema (GVV)[4]. In order to explain how we discover Web information sources for the GVV enrichment (Section 3), we briefly describe two case studies among several scenarios in which we tested TUCUXI. The DCT in Fig. 1(b) represents the *UNI Scenario* and derives from the integration of a relational source *University*, storing data about students and staff, an object-oriented database *Computer_Science* about people at the CS department and a file system about students' fees (*Tax_Position*) (Fig. 1(a)). The DCT in Fig. 2(b) represents the *TEX Scenario*, where a relational source *Suppliers* stores companies selling clothing and the object oriented source *Catalog* is about available wears in a store (Fig. 2(a)).

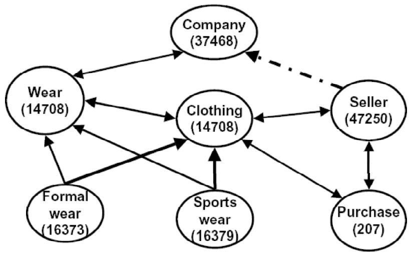
Suppliers *Source*

```
Seller(name,phone,address)
Clothing(code, description)
Purchase(number, name, code, date, price)
 FK: name references Seller
 FK: code references Clothing
```

Catalog *Source*

```
Company(name, address)
Wear(code,description,number,produced_by:Company)
SportsWear:Wear(fiber)
FormalWear:Wear(sales_promotion)
```

(a) ***Sources to be integrated***



(b) ***DCT***

Figure 2. TEX Scenario DCT, where a WordNet synset is assigned to each class name.

3. Information discovery with TUCUXI

3.1. Map of Meanings generation

Our approach relies on the comprehension of natural language to preserve the text semantics and provide a conceptual-based retrieval of documents. To do so, only surface properties of texts can be exploited, since robust NLP techniques enabling machines to fully understand documents are not still available. In particular, to take into account that words representing concepts are naturally connected each others, we have to identify the terms' meanings and retain semantic relations between them. The linguistic formalization of text properties due to Halliday and Hasan[12] supports our intuition: human readers are able to understand a written text because each language has a set of possibilities for making sentences hang together (*cohesion property*) and achieving a logical sense (*coherence property*). While coherence depends on the readers' point of view, cohesion has a more objective nature and, as Hoey observed[13], it is mainly provided by lexis: *lexical cohesion* can be achieved through **reiteration** (reinforcement of a concept by repeating a term or using its synonyms and hypernyms/hyponyms) and **collocation** (regular combination of words which often co-occur, such as meronyms/holonyms)[15]. Lexical cohesion is a clue to identify relations between words and therefore represents the context in which terms appear. As shown by Morris and Hirst[15], a thesaurus can be exploited to extract semantic relationships between terms and construct *lexical chains*. Lexical chains are, formally, *groups of semantically related words*. Henceforth we will refer them also as *clusters* that can be built in three steps: Step I identifies words suitable to be included in lexical chains (*candidate words*); Step II associates them an appropriate WordNet synset (*word sense disambiguation*); Step III generates clusters of related synsets. Only strongly connected clusters (i.e. clusters in which disambiguated words reveal a strong cohesion), named *strong lexical chains*, form the semantic representation of the text (*Map of Meanings*, MM). Henceforth we will refer to the example about the UNI Scenario to illustrate the MM generation (Fig. 3(a)).

Step I - Selection of candidate words. Assuming that concepts are best expressed by nouns, *candidate words* are selected by a part-of-speech tagger. In addition, a shallow parser identifies compounds, i.e. words created by 2 or more simple terms. Maintaining a compound as a whole unit (e.g. *computer_science* in Fig. 3(a)) captures more semantics than considering the terms separately.

Step II - Word Sense Disambiguation. WordNet meanings of the candidate words are collected. We expect that user-provided WordNet extensions[3] will amend thesaurus deficiencies in specific topic lexicon. For each retrieved meaning, we acquire a set of related synsets (i.e. its hypernyms, hyponyms, siblings, cousins, meronyms and holonyms[2]. The idea is to (partially) solve the ambiguity of natural language through an incremental process guided by the cohesion property. We call *basic unit* the association between a candidate word and one of its synsets[3]. Then, according both to the strength of lexical relationships and the relative position in text, each basic unit expresses cohesion votes for itself and for the other basic units with a semantically related synset. As described in Alg. 1, for each candidate word, just the basic unit with the highest preference score is retained, while the others are eliminated.

Algorithm 1 TUCUXI's Word Sense Disambiguation

```
Input: WordNet lexical Database (WNx) and its extensions if any
S={$s_i$: $s_i$ is one of the 1..n possible synsets contained in the text}, an ordered set
CW={$w_j$: $w_j$ is one of the 1..k candidate words in the text}, $WS_j$={$ws_l$: $ws_l$
is one of the 1..t possible meanings of $w_j$}, $j = 1..k$, scoring criteria $C$.
for $i = 1$ to $n$ do
    ask WNx for $s_i$ hyponyms, hypernyms, siblings,... meronyms and holonyms
    build the list of related synsets $RS_i$;
end for
for $i = 1$ to $n$ do
    select the words in CW whose $ws_l$=$s_i$;
    update cohesion vote for the words whose $ws$ is contained in $RS_i$ (according
    to relationship strength and position of words in text, i.e scoring criteria C);
end for
for $j = 1$ to $k$ do
    select the ${ws_l}^{best}$ meaning in $WS_j$ (with the highest score or the most frequent one in case of a tie)
    store the ${ws_l}^{best}$ meaning in the basic units list $BU$;
    for all $ws_l$ in $WS_j$ do
        if $ws_l$!= ${ws_l}^{best}$ then
            nullify the votes expressed by the $ws_l$ synset of the word $w_j$ in the previous phase;
        end if
    end for
end for
Update S by deleting the $s_i$ that are not preserved (and the related list $RS_i$;)
Output: a list $BU$ of basic units, which stores the most reasonable meaning for
each word in CW, a list of preserved synsets and their related ones.
```

Step III - Lexical Chaining for Map of Meanings generation The lexical chaining process is a clustering algorithm (Alg. 2) that takes in input the survived basic units (step II) and produces as output a set of clusters of disambiguated words (Tab. 1). Since we assume that MM consists of strongly connected clusters only, we implemented several strategies to select them. On the basis of our qualitative experiments (we do not show them, because of the lack of space), we observed that the Barzilay and Elhadad's criterion[2] (see Alg. 2 for more details) is a good strategy to identify strong chains, but when documents contain a lot of textual information further pruning strategies, such as a user-defined

[2] It is worth noting that *hypernymy* (*hyponymy*) is the semantic relation of being superordinate (subordinate) or to belonging to a higher (lower) class or rank. Since the WordNet organization of nouns is based on hypernymy/hyponymy relations, it is easy to deduce the meaning of siblings or cousins of a given synset. *Meronymy* (partwhole/HAS-A relation) is held between a part and the whole. *Holonymy*, on the contrary, between a whole and its parts.

[3] For istance, the noun class in Fig. 3(a) has 4 possible meanings, so it forms 4 basic units.

Sentences extracted from
http://www.cs.stanford.edu/Courses/index.html

Class information & Courses.
The Computer Science Education Center
has information on undergraduate CS courses.

Candidate words	Selected Meaning
class(1)	3591
information(2)	27555
course(3)	3591
computer science(4)	28610
education(5)	3589
center(6)	27928
information(7)	27555
undergraduate(8)	47915
cs(9)	28610
course(10)	3591

(a) *Word Sense disambiguation*

Candidate Words	Possible Meanings (Synsets and WordNet Glosses)
class(1)	37377 - a collection of things sharing a common attribute; 38085 - a body of students who are taught together; 37296 - people having the same social or economic status… 3591 - education imparted in a series of lessons or class meetings…
information (2)(7)	33347 - formal accusation of a crime 38929 - a collection of facts from which conclusion may be drawn 27555 - knowledge acquired through study or experience…
course(3)(10)	3591 - education imparted in a series of lessons… … 15044 - a circumscribed area of land or water…
computer science(4)	28610 - the branch of engineering science …
education(5)	3589 - activities that impart knowledge; 28190 - knowledge acquired by learning and instruction… …
center(6)	39134 - an area that is approximately central …
undergraduate(8)	47915 - a university student who has not yet received a first degree
cs(9)	62950 - a soft silver-white ductile metallic element 28610 - the branch of engineering science …

(b) *Candidate words and Their Possible Meanings.*

Figure 3. Word sense disambiguation of some sentences extracted from http://www.cs.stanford.edu/Courses/index.html (see Tab. 2 and 4).

number of chains to be retained, have to be adopted. The MM of Fig. 3(a) is shown in Fig. 4(b).

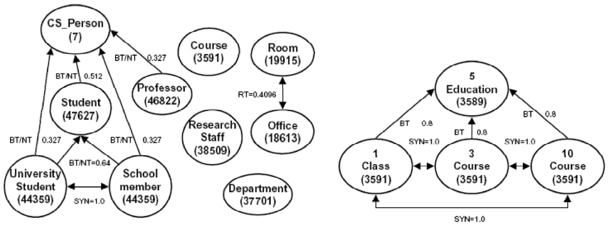
Algorithm 2 TUCUXI's Lexical Chaining Process

```
Input: BU={bu_j: bu_j represents the w_j word in CW and the most reasonable meanings ws_l^best in WS_j}, a list of preserved synsets PS and their related ones, scoring criteria C
Create an empty array $L$;
for all $bu_j \in BU$ do
    add $bu_j$ to the chain in $L$ whose basic units establish the strongest connection with it (through the $bu_j$ synset or the related ones);
    if no chains are suitable then
        create a new chain in $L$ with $bu_j$
    else
        update the score of the selected chain, according to C;
    end if
end for
Calculate the $avg(score)$ of chains and the standard deviation $stDev$;
Delete (not strong) chains (Barzilay and Elhadad' criterion: $score_{chain} \leq avg(score) + 2 * stDev$, other pruning criteria if necessary).

Output: The survived lexical chains.
```

Chain #	Score	Basic Units
1	5.4	class(1)/3591 course(3)/3591 education(5)/3589 course(10)/3591
2	2.024	center(6)/27928 information(2)/27555 information(7)/27555
3	1.0	computer_science(4)/28610 cs(9)/28610
4	0	undergraduate(8)/47915

Table 1. Lexical Chains of Fig. 3(a).



(a) *Transformed DCT. Lexical cohesion degree = 5,2837.* (b) *MM of Fig. 3(a). Lexical cohesion degree = 5,4.*

Figure 4. Graphs to be matched. They share 3591 only, RS of MM = 40%.

3.2. Semantic Matching between the Map of Meanings and the Domain Common Thesaurus

Both *MM* and *DCT* are graphs, i.e. nodes are synsets and edges are semantic connections between synsets. A trivial similarity measure is the *Synset Match* in (1), where N_{sS} is the number of common concepts in the two graphs, N_{sMM} is the number of concepts of MM and N_{sDCT} is the number of concepts of DCT.

$$SM = \begin{cases} 1 - \exp\left(-\frac{N_{sS}^2}{N_{sMM}}\right), & ifN_{sMM} < N_{sDCT}; \\ 1 - \exp\left(-\frac{N_{sS}^2}{N_{sDCT}}\right), & otherwise. \end{cases} \quad (1)$$

The SM measure grows rapidly with the increasing number of synsets in common, but if we consider the perfect synset match only, we will underestimate the page similarity degree. For instance, the concept *Course* in the UNI Scenario *DCT* is a broader term of *Seminar=a course offered for a small group of advanced students*, so a page with the latter meaning should be judged more relevant than documents with no *course*-related concepts at all. Since WordNet-provided relationships, such as hypernymy/hyponymy (e.g *course-seminar*) and meronymy/holonymy (e.g *faculty-professor*) indicate semantic relatedness between concepts[7], we exploit them in the cohesion parameter CM (2), where w_{ij} represents the weight associated to the relationship (or path of relationships) between the j^{th} synset of DCT ($j = 1, \ldots, t$) and the i^{th} synset of MM ($i = 1, \ldots, m$)[4] $Score(MM)$ and $Score(DCT_{tr})$ represent the lexical cohesion degree among synsets in the MM and in the DCT respectively and are calculated as the sum of the relations weights (Fig. 4(a) and 4(b)). Since the cohesion degree takes into account lexical relationships only, it could be necessary to *transform* the DCT by clustering its synsets into lexical chains (Fig. 4(a)). For instance, in the DCT of Fig. 1(b), *Student*, since it represents a *computer_science* student, is a subset of *University_Student*, while in the WordNet knowledge *Student* has a more general meaning than *University_Student*.

$$CM = \begin{cases} \frac{\sum_{j=1,\ldots,t}^{i=1,\ldots,m} w_{ij}}{Score(DCT_{tr})}, & if Score(DCT_{tr}) > Score(MM); \\ \frac{\sum_{j=1,\ldots,t}^{i=1,\ldots,m} w_{ij}}{Score(MM)} & otherwise. \end{cases} \tag{2}$$

Definition 1 *A document is said to be relevant when RS, the whole Relevance Similarity measure (1), exceeds the user-defined threshold.*

$$RS = \begin{cases} 1 - \exp\big(-(\frac{N_{sS}^2}{N_{sMM}}) + (a \cdot CM)\big), & N_{sMM} < N_{DCT}; \\ 1 - \exp\big(-\frac{N_{sS}^2}{N_{sDCT}} + (a \cdot CM)\big), & otherwise. \end{cases} \tag{3}$$

The parameter a in RS is calculated as $a = 1/(N_{sDCT} + N_{sMM})$.

3.3. The semantic-based crawling strategy

Crawlers are well-known programs used by Google-like search engines to collect documents from the Web. To face scalability and performance issues, the so-called *focused crawlers* implement Web navigation strategies to retrieve the maximum number of interesting pages while visiting the minimal set of irrelevant ones[8]. In essence, they assume that the Web has specific linkage structure in which pages on a specific topic are likely to link to other documents on the same topic (*short range topical locality*)[1]. Instead an *intelligent* crawler gradually learns the linkage structure during the exploration. Since TUCUXI exploits semantics to judge the relevance of a document, we propose a semantic-driven Web exploration strategy that follows the intelligent crawling framework in [1]. Each time a web page is retrieved, it is parsed to split the text from the addresses of referred pages (URLs). The crawler keeps track of the already visited pages as well as the unexplored ones (*candidate URLs*). For each candidate URL, some features (*facts*), such as content of the pages which link to it, are available during exploration, thus can be used to define the order in which documents have to be retrieved (Def. 1). More precisely, let us denote C as the event that makes a retrieved page to be relevant. The probability $P(C)$ that the document associated to a candidate URL will satisfy Def. 1 is the ratio between the number of relevant retrieved pages (N_{rel}) and the number of already crawled ones (N_{cwl}).

$$P(C) = N_{rel}/N_{cwl} \tag{4}$$

Let E be a known fact about a candidate URL. If we consider E, then the probability for a candidate URL to satisfy Def. 1 could be increased ($P(C|E) > P(C)$). The idea is to evaluate $P(C|E)$ with $P(E)$ (the probability that a candidate URL has the feature E) and $P(C \cap E)$ (the probability that a candidate URL has the feature E and satisfies Def. 1, at the same time):

$$P(C|E) = P(C \cap E)/P(E) \tag{5}$$

Then, for the event C and the feature E, we define the *Interest Ratio* (6).

$$I(C,E) = P(C|E)/P(C) = P(C \cap E)/(P(C) \cdot P(E)) \tag{6}$$

Let $\mathcal{E}$ be the composite event of the occurrences of a set of k events $E_1 \ldots E_k$, i.e. $\mathcal{E} = E_1 \cap E_2 \cap \ldots E_k$. The Composite Interest Ratio is:

$$I(C, \mathcal{E}) = \prod_{i=1}^{k} I(C, E_i). \tag{7}$$

The higher is the composite ratio, the higher is the probability of the candidate URL to satisfy Def. 1, thus it should be retrieved as soon as possible. We exploit four interest ratios to determine the retrieval order of candidate URLs. Three of them are the same described in [1]: *Link Based Learning* I_{lin}, *Sibling Based Learning* I_{sib} and *Content Based Learning* I_{con}, and we add a fourth one named *Synset Based Learning* I_{syns}. Briefly, I_{lin} expresses the short range topic locality; I_{sib} assumes that a candidate URL has more probability to satisfy Def. 1 if many of its siblings[5] also satisfy it. I_{con} considers the set of words in the DCT and determines if there is some kind of relation between the probability to satisfy Def. 1 and words appearing in inlinking pages of a candidate URL. Due to lack of space, please see [1] for the formalization of I_{lin}, I_{con} and I_{sib}, while the *Synset Based Learning* (Def.2) determines if

[4] If they are not the same synset: such a case is considered in SM.

[5] A web page is said to be a sibling of a candidate URL, when it is referred by the same page as the candidate.

there is some kind of relation between the probability to satisfy Def. 1 and meanings in inlinking pages of a candidate URL.

Definition 2 *Let S_1, S_2,...,S_t the t synsets in the Domain Common Thesaurus. The event U_j is true when the synset S_j is present in one of the pages pointing to the candidate URL. The whole Synset Based Learning interest ratio is:*

$$I_{syns}(C) = \prod_{j=1}^{t} \left(P(C \cap U_j)/(P(C) \cdot P(U_j))\right) \qquad (8)$$

The interest ratios are combined together to prioritize candidate URLs. The priority function PV is (9), where w_{lin}, w_{sib}, w_{con} and w_{syns} are user-defined weights. The higher is a weight, the higher is the importance of the associated interest ratio.

$$PV = w_{lin} \cdot I_{lin}(C) + w_{sib} \cdot I_{sib}(C) + w_{con} \cdot I_{con} + w_{syns} \cdot I_{syns} \qquad (9)$$

The algorithm for a semantic-driven Web exploration is summarized as Alg. 3.

Algorithm 3 Semantic-driven exploration strategy

```
Input: an empty hash table HT to trace features of crawled pages; a Candidate URL list CL, URLs, a DCT, a priority function $PV$, a relevance threshold RTH.
Insert given URLs in CL ($PV$=0);
while CL not empty do
    Select the candidate URL in CL with the highest $PV$;
    Retrieve the associated web page $WP$;
    Extract the $WP$ semantic document representation;
    Calculate the $RS$ value of $WP$;
    Extract the URLs contained in $WP$ and add them to CL (if not crawled yet); add $WP$ and its features to HT;
    Update the $PV$ for the pages in CL, according to features traced in HT;
end while
```

Query 1	UNI	*"computer science" & courses & professor* and *information*
Query 2	UNI	*course & location & "computer science" & department*
Query 3	UNI	*"computer science" & professor & "research staff"*
Query 4	TEX	*sportswear & sellers*
Query 5	TEX	*"athletic wear" & suppliers*
Query 6	TEX	*"smart clothing" & wholesaler*

Table 2. Queries submitted to Google.

4. Empirical results

The effectiveness of the proposed approach is first evaluated as the ability of TUCUXI to filter the *Google*'s results. For the *UNI Scenario* in Fig. 1(b), TUCUXI retrieved (all the accessible) pages from Berkeley, New York State, Princeton and Stanford computer science departments' sites (four of the most prestigious US universities) and performed its semantic analysis and matching. With respect to the queries 1, 2, 3 in Tab 2, a user was asked to distinguish between relevant and not relevant documents. After that, for each query and for each site, we retrieved the first 100 results (if any) proposed by Google[6]. According to the user decisions, the precision (P) and recall (R) of Tucuxi (RS value >= 80%) vs Google are depicted in Tab. 3. These encouraging results can be explained by the ability of TUCUXI to manage *meanings*, not mere keywords. For example, concerning the first query, TUCUXI detects the concept of *3591, course = education imparted in a series of lessons* even if the word *course* does not appear in the text, i.e. only the synonym *class* is used. In addition, as shown in Tab. 4, TUCUXI recognizes and rankes in a different way, when *Professor* is associated to more specific meanings than *46822 someone who is member of the faculty at a college or university*[7]. Google, at present, is not able to do so. Similar experiments (queries 4, 5, 6 in Tab 2) about *TEX Scenario* are carried out on *www.usawear.org*, *www.texweb.com*, *www.texbuyer.com* and *www.textilefiberspace.com*, (CNA's case studies within the SEWASIE project). Due to lack of space, we show precision and recall results only (RS value >= 80%, Tab. 3). As well as in the UNI Scenario, we can say that TUCUXI performs better than Google, since it manages meanings. For instance, thanks to the WordNet lexicon knowledge, TUCUXI recognizes that pages about companies selling *T-shirts* and *sport suits* are interesting even if they do not contain the word *sportswear* (e.g. http://www.texbuyer.com/com/79487.htm, RS=81% , never retrieved by Google). Finally, the harvest ratio measures the rate at which relevant pages are acquired (4). In both the scenarios, TUCUXI (RS value >= 80%) outperforms well-known navigation policies such as *Random* and *Breadth-first* strategies (Fig. 5(a) and 5(b)). Thus, considering also encouraging precision/recall results, our semantic-driven strategy can be exploited for automatically discovering interesting Web sources.

5. Conclusive remarks

As the best of our knowledge, the selective exploration of the Web with a semantic analysis of documents has been never pursued in research. Recently, a new crawler framework was proposed in [10], where, relying on future Semantic Web metadata, an "ontology" is used to focus the Web search. Instead, our lexicon-driven crawling is suitable both for Semantic Web and the Web as it is at present.

To ensure an effective enrichment of a domain ontology, we aim at discovering data-intensive Web sites, whose pages display information from a back-

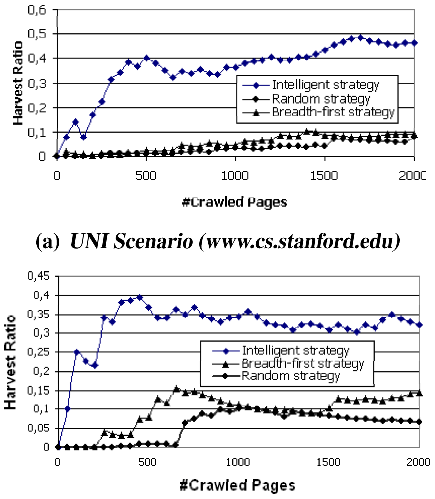
[6] By means of a site-restricted search.

[7] such as *43770, Associate professor = a teacher lower in rank than a full professor* or *43768, Assistant professor = a teacher lower in rank than an associate professor.*

UNI Scenario Dataset	Query 1				Query 2				Query 3			
	Google		*Tucuxi*		*Google*		*Tucuxi*		*Google*		*Tucuxi*	
	R	P	R	P	R	P	R	P	R	P	R	P
www.cs.berkeley.edu	88	40	88	94	86	38	70	97	87	28	96	100
www.cs.nyu.edu	93	58	98	96	96	45	78	51	94	80	85	93
www.cs.princeton.edu	52	39	98	94	83	46	55	83	80	26	76	76
www.cs.stanford.edu	94	26	94	85	78	45	98	93	23	38	39	62
Arithmetic Mean	**82**	**41**	**94**	**92**	**86**	**43**	**75**	**81**	**71**	**43**	**74**	**82**

TEX Scenario Dataset	Query 4				Query 5				Query 6			
	Google		*Tucuxi*		*Google*		*Tucuxi*		*Google*		*Tucuxi*	
	R	P	R	P	R	P	R	P	R	P	R	P
www.usawear.org	78	43	85	86	83	58	93	97	82	41	88	95
www.texweb.com	67	34	79	84	78	66	82	91	74	32	75	92
www.texbuyer.com	57	23	69	82	78	62	86	97	74	52	78	98
www.textilefiberspace.com	67	34	75	80	64	56	73	96	78	44	87	99
Arithmetic Mean	**67**	**33**	**77**	**83**	**76**	**60**	**83**	**95**	**77**	**42**	**82**	**96**

Table 3. TUCUXI vs Google: Precision and Recall in %.



(a) ***UNI Scenario (www.cs.stanford.edu)***

(b) ***TEX Scenario***

Figure 5. Crawling strategies comparison.

end database. The semantic analysis we proposed derives from the research field of the automatic generation of summaries[2][11], where documents are supposed to be linguistically well-formed. Nevertheless, Web pages from data-intensive sources are rarely rich of textual information. In such a case, we think that TUCUXI's lexical chaining technique could be useful to improve the semantic annotation of data extracted by RoadRunner[9].

Addresses	**TUCUXI RS%**	**46882 related Synsets**
www.cs.stanford.edu/Degrees/phd-req.html	73%	43768
www.cs.stanford.edu/Courses/ Schedules/2003-2004autumn.html	17%	43768 (not 46882)
www.cs.stanford.edu/Courses/index.html	10%	
www.cs.stanford.edu/Admissions/faq.html	39%	43768 (not 46882)
www.cs.stanford.edu/Admissions/index.html	32%	43768 (not 46882)
www.cs.stanford.edu/News/index.html	93%	43770 43768
www.cs.stanford.edu/News/news2002.html	65%	
www.cs.stanford.edu/News/news2001.html	40%	43768 (not 46882)
www.cs.stanford.edu/News/News1997.html	83%	43770 43768
www.cs.stanford.edu/News/News1998.html	46%	

Table 4. First 10 Google's results from www.cs.stanford.edu (Query 1).

References

[1] C. C. Aggarwal, F. Al-Garawi, and P. S. Yu. Intelligent Crawling on the World Wide Web with Arbitrary Predicates. In *WWW'01*, 2001.

[2] R. Barzilay and M. Elhadad. Using Lexical Chains for Text Summarization. In *ISTS'97 ACL*, 1997.

[3] R. Benassi, S. Bergamaschi, A. Fergnani, and D. Miselli. Extending a Lexicon Ontology for Intelligent Information Integration. In *ECAI'04*, 2004.

[4] D. Beneventano, S. Bergamaschi, F. Guerra, and M. Vincini. Synthesizing an Integrated Ontology. In *IEEE Internet Computing, Special Issue on The Zen of the Web*, 2003.

[5] S. Bergamaschi, S. Castano, and M. Vincini. Semantic Integration of Semistructured and Structured Data Sources. *SIGMOD Record*, 28(1), 1999.

[6] T. Berners-Lee, J. Hendler, and O. Lassila. The Semantic Web. *Scientific American*, May 2001.

[7] A. Budanitsky and G. Hirst. Semantic Distance in Wordnet: An Experimental Application-Oriented Evaluation of Five Measures. In *Workshop on WordNet and Other Lexical Resources. NAACL 2001*, 2001.

[8] S. Chakrabarti, M. van den Berg, and B. Dom. Focused Crawling: a new Approach to Topic-Specific Web Resource Discovery. *Computer Networks*, 1999.

[9] V. Crescenzi, G. Mecca, and P. Merialdo. Roadrunner: Towards Automatic Data Extraction from Large Web Sites. In *Proceedings of the 27th VLDB Conference*, 2001.

[10] M. Ehrig and A. Maedche. Ontology-Focused Crawling of Web Documents. In *SAC'03*, 2003.

[11] M. Galley and K. McKeown. Improving Word Sense Disambiguation in Lexical Chaining. In *IJCAI'03*, 2003.

[12] M. A. K. Halliday and R. Hasan. *Cohesion in English*. Longman, London, 1976.

[13] M. Hoey. *Pattern of Lexis in Text*. Oxford, 1991.

[14] G. A. Miller. Wordnet: a Lexical Database for English. *Commun. ACM*, 38(11):39–41, 1995.

[15] J. Morris and G. Hirst. Lexical Cohesion Computed by Thesaural Relations as an Indicator of the Structure of Text. *Computational Linguistics*, 17(1):21–48, 1991.