Web Semantic Search with TUCUXI*

(Extended Abstract)

Roberta Benassi[1], Sonia Bergamaschi[1,2], Maurizio Vincini[1]

[1] Dip. di Ingegneria dell'Informazione - Universita' di Modena e Reggio Emilia
[2] IEIIT-CNR Bologna
lastname.firstname@unimore.it

Abstract. Traditional search engines rely on keywords to locate Web documents that best fit a user's query. Since words extracted from their context do not always capture the intended meaning, the relevance of the retrieved documents is affected by the natural language ambiguity. TUCUXI is a semantic search tool that replaces keywords with an ontology-based expression of the user's requests. TUCUXI judges the relevance of a document by performing a semantic matching between the user-provided ontology and the *Map of Meanings*, a simplified - but semantically rich - representation of the source text.

Key words: *Basi di dati e Semantic Web, Ontologies, Lexical Chaining*

1 Introduction

The Web is said to be a huge, distributed and dynamic collection of documents, with no intrinsic and coherent organization besides the linkage structure. Current *Search Engines* are, essentially, Information Retrieval systems for the WWW which identify relevant documents w.r.t users' queries by merging keyword-based matching techniques[1] and other aspects such as the page authority degree [2]. Despite this, the information overload problem is not effectively faced. In fact, users can express their needs just by keywords, while they are usually looking for concepts. Thus, the relevance and quality of the retrieved documents are strongly affected by the well-known problems of synonymy (two or more words with the same meaning), polysemy (a word with several meanings) and by the fact that existing search tools do not take into account lexical cohesion, which is essential for natural language comprehension[3].

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 queries, where meanings and concepts to be searched are not ambiguous; (**b**) an effective semantics-based retrieval of documents; (**c**) software agents to carry out sophisticated tasks such as intelligent strategies for Web explorations. In this paper we

* This work is supported in part by the EC's 5^{th} Framework IST program through the Sewasie project (IST-2001-34825 www.sewasie.org) in the Semantic Web Action Line.

present TUCUXI - In**T**elligent H**U**nter Agent for **C**oncept **U**nderstanding and Le**X**ical Cha**I**ning - a semantic search tool that exploits WordNet[4] to provide a conceptual representation of Web pages (*Map of Meanings*). TUCUXI[3] adopts a *Domain Common Thesaurus* (*DCT*)[5, 6] as the user-provided ontology. The relevance of retrieved documents is judged by comparing the *Map of Meanings* with the DCT, thus TUCUXI will be able to select relevant Web pages on the basis of concepts rather than keywords. We first introduce the DCT synthesis by means of the MOMIS system (Section 2); then Section 3 explains the *Map of Meanings* extraction. Encouraging results coming from the comparison between TUCUXI and Google, are presented in Section 4.

2 The MOMIS framework

MOMIS, (**M**ediator Envir**O**nment for **M**ultiple **I**nformation **S**ources) is a framework for structured and semistructured data sources information extraction and integration[5, 6]. The information integration process creates a conceptualization of the underlying domain (i.e. a *domain ontology*) via the generation of a reconciled, integrated *Global Virtual View* (GVV) of the involved sources.

1. Heterogeneous data sources are presented to the system in a standard way, that is, wrappers extract local source schemata and translate them into a common data model based on ODL_{I3} language;
2. in order to exploits semantics of terms describing schemas' items (e.g., class names and attributes), the integration designer is asked to choose their meanings from WordNet. That is, each schemas' item is *annotated* with one or more WordNet synsets[6];
3. MOMIS generates a *Domain Common Thesaurus (DCT)* of the involved local sources (Fig. 1(b)) which contains intra and inter-schema knowledge in the form of synonymy (SYN); hypernymy/hyponymy (BT/NT); meronymy/holonymy (RT); equivalence (SYN_{ext}); generalization (BT_{ext}) and aggregation (RT_{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* (the integration designer can directly supply new relationships to capture specific domain knowledge) and *inferred relationships* (via equivalence and subsumption computation);
4. starting from the DCT and the local schemata descriptions, MOMIS generates a global reconciled schema (GVV) plus sets of mappings to the local sources. Then, the GVV is semiautomatically annotated by associating each item of the global schema to meanings extracted from the local sources [6].

To present the scenario in which TUCUXI works, let us consider the domain in (Fig. 1(a)) where, under the supervision of the integration designer, three different sources have been integrated: a relational source named *University* storing

[3] Pron. "tookooshee", the common name of a South American river dolphin.

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)
```

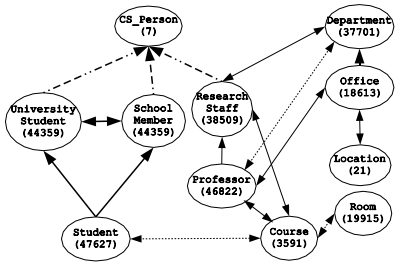
Computer_Science *Source*

```
CS_Person(first_name,last_name)
Professor: CS_Person(title,belongs:Office,rank)
Student: CS_Person(year,takes:set(Course),rank)
Office(description,address:Location)
Location(city,street,number, county)
Course(course_name,taught_by:Professor)
```

Tax_Position *Source*

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

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



(b) ***MOMIS-generated DCT. Thick arrows are BT, NT and SYN relationships, while thin ones are RT relationships. Dashed ones are inferred relationships while solid ones explicitly given relationships.***

Fig. 1. DCT generation. Each class name in 1(b) is annotated with a WordNet synset.

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*). Let us suppose that the integration designer needs to extend the obtained domain ontology by integrating new Web sources[6] about courses at CS departments, e.g., courses' information and professors that teach the lessons, courses' location, professors and research staff. Thus, (s)he can locate interesting Web pages by querying a search engine with appropriate keywords or providing TUCUXI with (part of) the MOMIS-generated DCT (Fig. 1(b)). The DCT expresses in an unambiguous manner the meanings to be searched and semantic relations between them. To compare the results provided by the two different approaches, we ask Google to perform a *site-restricted* search (w.r.t queries in Tab. 1). More precisely, we queried Google about documents within the computer science departments' sites of four, among the most prestigious, US universities: Berkeley, New York, Princeton and Stanford. To explain how TUCUXI preserves source texts' semantics we will refer to some sentences extracted from *http://cs.stanford.edu/Courses/index.html*, one of the most relevant page according to Google (w.r.t Stanford CS site and Query 1).

Query 1	*"computer science"* and *courses* and *professor* and *information*
Query 2	*course* and *location* and *"computer science"* and *department*
Query 3	*"computer science" department* and *professor* and *"research staff"*

Table 1. Queries submitted to Google.

Algorithm 1 TUCUXI's Word Sense Disambiguation

Input: *WNx: the WordNet lexical Database and its extensions if any*
S={s_i: s_i is one of the n possible synsets contained in the text}, an ordered set CW={w_j: w_j is one of the k candidate nouns in the text}, WS_j={ws_l: ws_l is one of the t possible meanings of w_j} with $j = 1..k$, the set of scoring criteria C.

for each synset $s_i \in$ S
- build the list RS_i by retrieving its *related synsets* from WNx, i.e, hypernyms, hyponyms, siblings, cousins, meronyms and holonyms;

for each synset $s_i \in$ S
- select the words in CW whose ws_l=s_i;
- update cohesion vote for the nouns whose ws is contained in RS_i (according to relationship strength and relative positions of words in text, i.e scoring criteria C);

for each noun w_j
- select the ${ws_l}^{best}$ synset in WS_j (with the highest preference score or the most frequent one in case of a tie) and store it in the list BU;
- nullify the cohesion votes expressed by $ws_l \neq {ws_l}^{best}$

Update S by deleting the s_i that are not preserved (and the related list RS_i;)
Output: *BU which stores the most reasonable meaning for each noun in CW, the preserved synsets and their related ones.*

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

Example 1. Extracted from *http://cs.stanford.edu/Courses/index.html.*

3 Lexical Chaining for Map of Meanings extraction

The comprehension of natural language may be the key to preserve documents' semantics. As observed by Hasan and Halliday[3], human readers understand the meaning of written texts because each language has a particular set of possibilities for making sentences hang together *(cohesion)* and following a logical sense *(coherence)*. In this work we particularly address to **lexical cohesion** as the way to identify semantic relationships between words [7–9]. Lexical cohesion can be achieved through **reiteration** (reinforcement of a concept through repetition of terms, use of synonyms and substitution of a term with its broader/narrower terms) and **collocation** (regular combination of words which tend to co-occur in similar lexical environments). Thus, if we are able to assign meanings to words and identify semantic relations between terms (such as hypernymy, hyponymy, meronymy and holonymy relations), we will obtain **lexical chains**. A lexical chain is, formally, a cluster of related words, representing concepts[4] that are naturally connected each others[3, 7]. The first step is to *understand nouns' meanings* by exploiting WordNet[4][5]. Starting from sentences in Example 1, we identified the candidate nouns (CW) and their possible synsets as

[4] We assume that concepts are best expressed by nouns[7].
[5] Thesaurus deficiencies in specific domains are expected to be amended by MOMIS-WordNet extensions[10].

Table 2. Candidate Nouns from Example 1 and Their WordNet Meanings. Because of a MOMIS' WordNet extension, cs is registered as a contraction of computer science.

CW	Synsets and WordNet Gloss (Meanings)
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	33347 - formal accusation of a crime
(2)(7)	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 that studies (with the aid...
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 ...

shown in Tab. 2. Via the *word sense disambiguation* heuristic in Alg. 1, an incremental process guided by the cohesion property, we selected the most appropriate candidate nouns' meanings as the ones that best stick together. Then, we formed lexical chains as described in Alg. 2. Each lexical chain has a *cohesion degree* (ChD) representing the strength of semantic relationships (such as hypernymy/hyponymy, meronymy/holonomy...) between (disambiguated) nouns (Tab. 3(b)). Only strongly connected clusters, the so-called *Strong Chains*[11], form the *Map of Meanings (MM)* of the given text (Fig. 2(b)).

Both MM and DCT are graphs, i.e. nodes are meanings and edges are semantic connections between nodes. To compare them, we propose the *Synset Matching SM* similarity measure in (1), where N_{sS} is the number of common (shared) concepts in the two graphs, N_{sMM} is the number of concepts in MM and N_{sDCT} is the number of concepts in DCT.

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

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 S and their related ones (hypernyms, hyponyms...), the set of scoring criteria C*

Create an empty array L;
for all $bu_j \in BU$ **do**
- add bu_j to the chain in L whose elements establish the strongest connection with it (through the bu_j synset or the related ones and according to the scoring criteria C);
if no chains are suitable **then** create a new chain in L with bu_j;
else update the score of the selected chain;

end for
Calculate the $avg(score)$ of the lexical chains and the standard deviation $stDev$;
Delete the chains following the Barzilay and Elhadad' criterion[11]: $score \leq avg + 2 * stDev$ (other pruning criteria if necessary).

Output: *The survived lexical chains = Map of Meanings*

The SM measure grows rapidly as the number of common synsets increases. Nevertheless, if we consider the perfect synset match only, we will underestimate the page similarity degree. For example, the concept *Course* in the 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. Since WordNet-provided relationships, such as hypernymy/hyponymy (e.g *course-seminar*) and meronymy/holonymy (e.g *faculty-professor*), indicate semantic relatedness between concepts[12], 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 ($i = 1, \ldots, m$), if they are not the same synset (such a case is considered in SM). $Score(MM)$ and $Score(DCT_{tr})$ are the lexical cohesion degree of the MM and in the DCT_{tr} respectively and are calculated as the sum of the relations weights (Fig. 2(a) and 2(b)). DCT_{tr} stands for *transformed* DCT: since the cohesion degree takes into account lexical relationships only, it could be necessary to cluster the DCT into lexical chains (Fig. 2(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 WordNet, *Student* has a more general meaning than *University_Student*.

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

Definition 1. *A document is said to be relevant when RS, the whole Relevance Similarity measure (3), exceeds the user-defined threshold.*

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

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

4 Empirical results

In this section we evaluate the ability of TUCUXI to filter the *Google*' results. With respect to queries in Tab. 1, the integration designer was asked to distinguish between relevant and not relevant documents from Berkeley, New York, Princeton and Stanford cs departments' sites. After that, we retrieved, for each query and for each site, the first 100 results proposed by Google. According to the designer's decisions, the precision (P), the recall (R) and the F-measure (F) of Tucuxi (RS value >= 80%, in [13] we showed how the user-defined threshold influences the TUCUXI's overall performance) and Google are depicted in

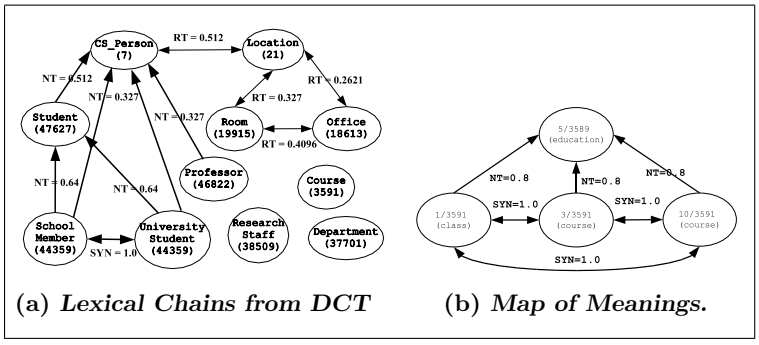
Table 3. Selected Meanings and Lexical chains from Example 1.

(a) *Word Sense Disambiguation*

CW	Meaning	Score
class(1) course(3)(10)	3591	3.8
information(2)(7)	27555	2.512
computer science(4) cs(9)	28610	2.0
education(5)	3589	3.4
center(6)	27928	2.024
undergraduate(8)	47915	1.0

(b) *Lexical Chains*

Chain#	ChD	Nouns/Meanings
1	5.4	class(1)/3591 - course(3)(10)/3591 education(5)/3589
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



(a) *Lexical Chains from DCT* **(b) *Map of Meanings.***

Fig. 2. The DCT and the Map of Meanings shared the synset 3591 only.

Tab. 5. Encouraging results can be explained by the fact that TUCUXI manages *meanings*, not mere keywords: the concept of *course = education imparted in a series of lessons* is detected even if the word *course* does not appear in the text, i.e. only the term *class* (synonym for the synset 3591) is used. With respect to Query 1 (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 university*[6]. Google, at present, is not able to do so.

5 Related Work and Conclusions

TUCUXI provides a semantic retrieval of documents by exploiting the lexical cohesion property of written texts[3][8]. One of the most interesting semantic search tools is described in [14], where Web pages are previously annotated with machine readable metadata (SHOE markup tags). Our approach, which does not require any annotation phase, is suitable both for the future Semantic Web and the Web as it is at present. Future work will design intelligent semantic-driven strategies for Web exploration.

[6] 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.*

Table 4. First 10 Google Results from Stanford (Query 1).

Address	TUCUXI's Score	Professor's 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 5. TUCUXI vs Google: Precision, Recall and F-Measure in %.

Dataset	Query 1						Query 2						Query 3					
	Google			*Tucuxi*			*Google*			*Tucuxi*			*Google*			*Tucuxi*		
	R	P	F	R	P	F	R	P	F	R	P	F	R	P	F	R	P	F
www.cs.berkeley.edu	88	40	55	88	94	90	86	38	53	70	97	81	87	28	42	96	100	98
www.cs.nyu.edu	93	58	71	98	96	97	96	45	61	78	51	62	94	80	86	85	93	89
www.cs.princeton.edu	52	39	45	98	94	96	83	46	59	55	83	66	80	26	39	76	76	76
www.cs.stanford.edu	94	26	41	94	85	89	78	45	57	98	93	95	23	38	29	39	62	48
Arithmetic Mean	**82**	**41**	**53**	**94**	**92**	**93**	**86**	**43**	**58**	**75**	**81**	**76**	**71**	**43**	**49**	**74**	**82**	**78**

References

1. Baeza-Yates, R., Ribeiro-Neto, B.: Modern Information Retrieval. (Add. Wesley)
2. Page, L., et al.: The pagerank citation ranking: Bringing order to the web. In: Stanford Dig. Lib. Technologies Project. (1998)
3. Halliday, M.A.K., Hasan, R.: Cohesion in English. Longman (1976)
4. Miller, A.: Wordnet: A lexical database for english. Comm. of ACM **38(11)** (1995)
5. Bergamaschi, S., et al.: Semantic integration of semistructured and structured data sources. In: SIGMOD Record. (1999)
6. Beneventano, D., et al.: Synthesizing an integrated ontology. In: IEEE Internet Computing, The Zen of The Web 7(5). (2003)
7. Morris, J., Hirst, G.: Lexical cohesion by thesaural relations as an indicator of the structure of text. In: Computational Linguistics 17(1). (1991)
8. Hoey, M.: Patterns of Lexis in Text. (Oxford University Press)
9. Galley, M., McKeown, K.: Improving word sense disambiguation in lexical chaining. In: IJCAI. (2003)
10. Benassi, R., Bergamaschi, S., Fergnani, A., Miselli, D.: Extending a lexicon ontology for intelligent information integration. In: ECAI'04. (2004)
11. Barzilay, R., Elhadad, M.: Using lexical chains for text summarization. In: ISTS'97 Workshop, ACL. (1997)
12. Budanitsky, A., Hirst, G.: Semantic distance in wordnet: An experimental application-oriented evaluation of five measures. In: Workshop NAACL 2001. (2001)
13. Benassi, R., Bergamaschi, S.: Tucuxi: the intelligent hunter agent for concept understanding and lexical chaining. In: DBGROUP Tech. Report. (Jan. 2004)
14. Heflin, J., Hendler, J.: Searching the web with shoe. In: AAAI Workshop. (2000)