Natural Language Engineering **8** (4): 359–373. © 2002 Cambridge University Press
DOI: 10.1017/S1351324902003029 Printed in the United Kingdom

The role of domain information in Word Sense Disambiguation

BERNARDO MAGNINI, CARLO STRAPPARAVA, GIOVANNI PEZZULO and ALFIO GLIOZZO
ITC-irst, Istituto per la Ricerca Scientifica e Tecnologica, I-38050 Trento, Italy
e-mail: {magnini,strappa,pezzulo,gliozzo}@itc.it

(*Received 1 November 2001; revised 25 July 2002*)

Abstract

This paper explores the role of domain information in word sense disambiguation. The underlying hypothesis is that domain labels, such as MEDICINE, ARCHITECTURE and SPORT, provide a useful way to establish semantic relations among word senses, which can be profitably used during the disambiguation process. Results obtained at the SENSEVAL-2 initiative confirm that for a significant subset of words domain information can be used to disambiguate with a very high level of precision.

1 Introduction

The purpose of this paper is to investigate the role of *domain information* in Word Sense Disambiguation (WSD). The hypothesis is that domain labels (such as MEDICINE, ARCHITECTURE and SPORT) provide a powerful way to establish semantic relations among word senses, which can be profitably used during the disambiguation process. In particular, we assume that domains constitute a fundamental semantic property on which textual coherence is based, such that word senses occurring in a coherent portion of text tend to maximize their belonging to the same domain.

Figure 1 illustrates this behavior providing an example selected from the *English_lexical_sample* task at SENSEVAL-2. The target word, i.e. the word to be disambiguated, is the second occurrence of 'chairs' in a double box. Let us suppose that for some words in the passage and for each sense of those words in WORDNET a domain label is available, and that such words are already disambiguated in the text. We can note that several words (e.g. 'sofa', 'Bang and Olufsen stereo', 'living room', 'dinner table') carry the FURNITURE domain; few words (e.g. 'games', 'backgammon', 'chess') carry the PLAY domain; and just one word carries the LITERATURE domain. To disambiguate 'chairs' it seems natural to rely on the fact that FURNITURE is the prevalent domain in the text. This leads to correctly choose the sense of 'chairs' more closely related to FURNITURE, discarding the other non-related senses, such as for instance, `chair#2` (i.e. professorship) and`chair#3` (i.e. chairperson).

From the plush Connolly hide leather sofa$_F$ and chairs$_F$ in the livingroom$_F$ to the Bang and Olufsen stereo$_F$, and remote control television$_F$ complete with video,you're surrounded by the HIGHEST QUALITY. The inlaid$_F$ chequerboard top of the coffeetable$_F$ houses all kind of games$_P$, including backgammon$_P$, chess$_P$ and Scrabble$_P$. You'll also find a selection of books,from Queen Victoria's Highland journals, to the very latestbestselling thriller$_L$. The dinnertable$_F$ and chairs$_{??}$ are elegant yetcomfortable, and you can be assured of the finest tableware$_F$ and crystal for meals at home.

Fig. 1. Domain information in a sample text (from SENSEVAL-2). Subscripts: Furniture = F, Play = P, Literature = L.

However, to implement the above intuitions a number of issues need to be addressed. First, it seems necessary to rely on a lexical resource where words are associated with domain information, in addition to senses. For this purpose an extended version of WORDNET, WORDNET DOMAINS, has been developed, which provides a domain annotation for every synset. Section 2 describes the main characteristics of this resource and discusses some relevant issues, such as the selection of an appropriate set of domain labels and the annotation procedure.

As a next step, the availability of WORDNET DOMAINS, allows us to undertake a domain-oriented analysis of a text. For instance, it is possible to determine the prevalent domain for a text (or for a portion of it) and to investigate how it comes out from the words in that text. In addition, a measure of text coherence can be calculated on the basis of a 'one domain per discourse' hypothesis (as opposed to a 'one sense per discourse' hypothesis). A relevant outcome of this analysis is that a rather limited number of words, i.e. *domain words*, actually contributes to determine the prevalent domain of a portion of text. These words will be central in our approach to WSD based on domain information. Section 3 addresses the main ideas of a textual analysis based on domains.

Results obtained from a domain-oriented analysis of texts are necessary to implement effective WSD algorithms based on domain information. Our basic strategy is first to calculate the prevalent domain for a portion of text, then to compare this domain with domains associated with single word senses, and finally to select the sense that maximizes the similarity. Section 4 presents the algorithms implemented.

Section 5 reports on our participation at the SENSEVAL-2 initiative on WSD for the *English_all_words* and the *English_lexical_sample* tasks, evaluating the domain-based approach with respect to other systems. Results confirm that for a significant subset of words domain information can be used to disambiguate with a very high level of precision. Finally, section 6 provides a view of our work in the context of the relevant literature that addresses the role of domain information.

2 Domains and WordNet

In our usage, a *domain* is a set of words between which there are strong semantic relations. An approximation to domains are Subject Field Codes, used in

Table 1. WordNet *senses and domains for the word 'bank'*

Sense	Synset and Gloss	Domains	Semcor
#1	depository financial institution, bank, banking concern, banking company (a financial institution . . .)	Economy	20
#2	bank (sloping land . . .)	Geography, Geology	14
#3	bank (a supply or stock held in reserve . . .)	Economy	–
#4	bank, bank building (a building . . .)	Architecture, Economy	–
#5	bank (an arrangement of similar objects . . .)	Factotum	1
#6	savings bank, coin bank, money box, bank (a container . . .)	Economy	–
#7	bank (a long ridge or pile . . .)	Geography, Geology	2
#8	bank (the funds held by a gambling house . . .)	Economy, Play	
#9	bank, cant, camber (a slope in the turn of a road . . .)	Architecture	–
#10	bank (a flight maneuver . . .)	Transport	–

Lexicography (e.g. in Proctor (1978)) to mark technical usages of words. Although this information is useful for sense discrimination, in dictionaries it is typically used only for a small portion of the lexicon. WordNet Domains (Magnini and Cavaglià 2000) is an attempt to extend the coverage of domain labels within an already existing lexical database, WordNet (version 1.6) (Fellbaum 1998). As a result, WordNet Domains can be considered an extension of WordNet in which synsets have been annotated with one or more domain labels, selected from a hierarchically organized set of about two hundred labels. Domain labeling is complementary to what is already in WordNet. First, a domain may include synsets of different syntactic categories: for instance, Medicine groups together senses from nouns, such as `doctor#1` and `hospital#1`, and from verbs, such as `operate#7`. Secondly, a domain may include senses from different WordNet sub-hierarchies (i.e. deriving from different 'unique beginners' or from different 'lexicographer files'). For example, Sport contains senses such as `athlete#1`, deriving from `life_form#1`, `game_equipment#1` from `physical_object#1`, `sport#1` from `act#2`, and `playing_field#1` from `location#1`. Finally, domains may group senses of the same word into thematic clusters, which has the important side-effect of reducing the level of ambiguity when we are disambiguating to a domain. Table 1 shows an example. The word 'bank' has ten different senses in WordNet 1.6: three of them (i.e. `bank#1`, `bank#3` and `bank#6`) can be grouped under the Economy domain, while `bank#2` and `bank#7` both belong to Geography and Geology. Grouping related senses is an emerging topic in WSD (see, for instance, Palmer, Fellbaum, Cotton, Delfs and Dang (2001)).

The annotation methodology was mainly manual and based on lexico-semantic criteria which take advantage of the already existing conceptual relations in WordNet.

First, about 200 domain labels were selected from a number of dictionaries and then structured in a taxonomy according to the Dewey Decimal Classification (DDC (Comaroni, Beall, Matthews and New 1989)). The DDC classification was chosen as it guarantees good coverage and because it is easily available. The resulting domain taxonomy is a subgraph of DDC.

The annotation task consists of interpreting a WordNet synset with respect to the DDC classification. First, a small number of high level synsets was manually annotated with their pertinent domain. Then, an automatic procedure exploited some of the WordNet relations (i.e. hyponymy, troponymy, meronymy, antonymy and pertain-to) to extend the manual assignments to all the reachable synsets. As an example, this inheritance-based procedure marks the synset {`beak, bill, neb, nib`} with the code Zoology, starting from the synset {`bird`} and following a 'part-of' relation. However, there are cases in which the inheritance procedure has to be blocked, inserting 'exceptions', to prevent incorrect propagation. There are WordNet synsets that do not belong to a specific domain, but rather appear in texts associated with any domain. For this reason, a Factotum label has been created which basically includes two types of synsets: (i) *generic* synsets, which are hard to classify in a particular domain, such as `man#1` (i.e. an adult male person); and (ii) *stop sense* synsets, which appear frequently in different contexts, such as numbers, week days, colors, etc.

For the purpose of WSD, we have considered 43 disjoint labels (i.e. we have used Sport in place of Volley or Basketball, which are subsumed by Sport). This set allows a good level of abstraction without losing relevant information and, in addition, overcomes the problem of applying learning techniques to domains not well enough represented in available texts. The production of WordNet Domains took a total of 2 person-years. The complete list of domains and the number of annotations in WordNet are reported in Table 2.

3 Domains and texts

The availability of WordNet Domains makes it possible to analyze the content of a text in terms of domain information. Two related aspects will be addressed: section 3.1 proposes a test to estimate the number of words in a text that brings relevant domain information. Section 3.2 reports on an experiment whose aim is to verify the 'one domain per discourse' hypothesis. These experiments make use of the Semcor corpus, the portion of the Brown corpus tagged with WordNet synsets.

3.1 Domains and words

Words in a text do not behave homogeneously as far as domain information is concerned. In particular, we have identified three different roles that a word can assume in a given text (the same word can play a different role in a different text):

- *Text Related Domain (TRD) words*: words that have at least one sense that contributes to determine the domain of the whole text; for instance, the word 'bank' in a text concerning Economy is likely to be a text related word.

Table 2. *Domains distribution over* WordNet *synsets*

Domain	#Syn	Domain	#Syn	Domain	#Syn
Factotum	36820	Biology	21281	Earth	4637
Psychology	3405	Architecture	3394	Medicine	3271
Economy	3039	Alimentation	2998	Administration	2975
Chemistry	2472	Transport	2443	Art	2365
Physics	2225	Sport	2105	Religion	2055
Linguistics	1771	Military	1491	Law	1340
History	1264	Industry	1103	Politics	1033
Play	1009	Anthropology	963	Fashion	937
Mathematics	861	Literature	822	Engineering	746
Sociology	679	Commerce	637	Pedagogy	612
Publishing	532	Tourism	511	Computer_Science	509
Telecommunication	493	Astronomy	477	Philosophy	381
Agriculture	334	Sexuality	272	Body_Care	185
Artisanship	149	Archaeology	141	Veterinary	92
Astrology	90				

- *Text Unrelated Domain (TUD) words*: words that have senses belonging to specific domains (i.e. they are non generic words) but do not contribute to the domain of the text; for instance, the occurrence of 'church' in a text about Economy does not probably affect the whole topic of the text.
- *Text Unrelated Generic (TUG) words*: words that do not bring relevant domain information at all (i.e. the majority of their senses are annotated with Factotum); for instance, a verb like 'to be' is likely to fall in this class, whatever the domain of the whole text.

To provide a quantitative estimation of the distribution of the three word classes, an experiment has been carried out on the Semcor corpus using WordNet Domains as a repository for domain annotations. In the experiment we considered 42 domains (Factotum was not included). For each text in Semcor, all the domains were scored according to their frequency among the senses of the words in the text. The three top scoring domains are considered as the prevalent domains in the text. These domains have been identified for the whole text, without taking into account possible domain variations that can occur within portions of the text. Then each word token of a text has been assigned to one of the three classes according to the fact that (i) at least one domain of the word is present in the three prevalent domains of the text (i.e. a TRD word); (ii) the majority of the senses of the word have a domain, but none of them belongs to the top three of the text (i.e. a TUD word); (iii) the majority of the senses of the word are Factotum and none of the other senses belongs to the top three domains of the text (i.e. a TUG word). Then each group of words has been further analyzed by part of speech and the average polysemy with respect to WordNet has been calculated.

The results (Table 3) show that just a few words (21%) actually carry domain information which is compatible with the prevalent domains of the whole text, with a

Table 3. *Word distribution in Semcor according to the prevalent domains of the texts*

Word class	Nouns	Verbs	Adjectives	Adverbs	All
TRD words	18732 (34.5%)	2416 (8.7%)	1982 (9.6%)	436 (3.7%)	21%
Polysemy	3.90	9.55	4.17	1.62	4.46
TUD words	13768 (25.3%)	2224 (8.1%)	815 (3.9%)	300 (2.5%)	15%
Polysemy	4.02	7.88	4.32	1.62	4.49
TUG words	21902 (40.2%)	22933 (83.2%)	17987 (86.5%)	11131 (93.8%)	64%
Polysemy	5.03	10.89	4.55	2.78	6.39

significant (79.4%) contribution of nouns. TUG words (i.e. words whose senses are tagged with FACTOTUM) are, as expected, both the most frequent (i.e. 64%) and the most polysemous words in the text. This is especially true for verbs (83.2%), which often have generic meanings that do not contribute to determining the domain of the text.

3.2 One domain per discourse

The One Sense per Discourse (OSD) hypothesis is that there is a strong tendency for multiple uses of a word to share the same sense in a well-written discourse. Depending on the methodology used to calculate OSD, Gale, Church and Yarowsky (1992) claim that OSD is substantially verified (98%), while Krovetz (1998), using WORDNET as a sense repository, found that 33% of the words in Semcor have more than one sense within the same text, basically due to the high sense granularity of WORDNET.

Following the same line, a One Domain per Discourse (ODD) hypothesis would claim that multiple uses of a word in a coherent portion of text tend to share the same domain. If demonstrated, ODD would reinforce the main hypothesis of this work, i.e. that the prevalent domain of a text is an important feature for selecting the correct sense of the words in that text.

To support ODD an experiment has been carried out using WORDNET DOMAINS as a repository for domain information. We applied to domain labels the same methodology proposed by Krovetz (1998) to calculate sense variation: to invalidate the OSD hypothesis it is sufficient just one occurrence of a word in the same text with different meanings. A set of 23,877 ambiguous words with multiple occurrences in the same document in Semcor was extracted and the number of words with multiple sense assignments was counted. Semcor senses for each word were mapped to their corresponding domains in WORDNET DOMAINS and for each occurrence of the word the intersection among domains was considered. To understand the difference between OSD and ODD, let us suppose that the word 'bank' (see Table 1) occurs three times in the text with three different senses (e.g. `bank#1`, `bank#3`, `bank#8`). This case would be inconsistent with OSD but would be consistent with ODD because the three senses all belong to the same domain (i.e. ECONOMY).

Table 4. *One Sense per Discourse vs. One Domain per Discourse*

Pos	Cases[a]	Exceptions to OSD[b]	Exceptions to ODD[c]
All	23877	7469 (31%)	2466 (10%)
Nouns	10291	2403 (23%)	1142 (11%)
Verbs	6658	3154 (47%)	916 (13%)
Adjectives	4495	1100 (24%)	391 (9%)
Adverbs	2336	790 (34%)	12 (1%)[d]

[a] Given a text in Semcor we have a *case* for each lemma that has more than one occurrence in that text and that is ambiguous in WordNet.
[b] Number of cases in which a lemma has multiple senses in the same text.
[c] Number of cases in which a lemma has multiple domains in the same text.
[d] Adverbs have few exceptions to ODD because they often have general senses that are labeled with the Factotum domain. In these cases ODD is verified in a trivial way, because all senses of an adverb have the same label.

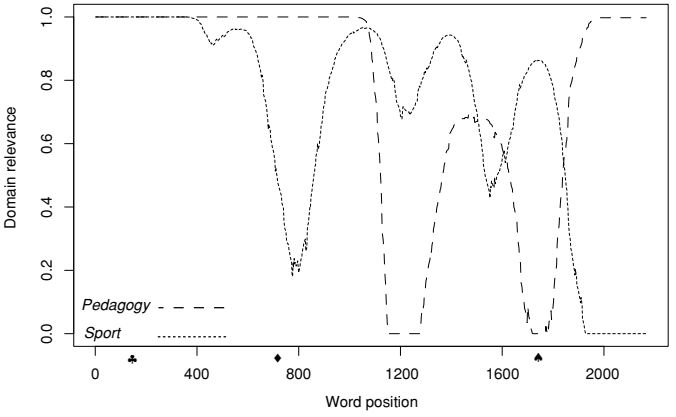

Fig. 2. Domain variation in the text `br-e24` from the Semcor corpus. ✣ ... The Russians are all trained as dancers before they start to study gymnastics .... ♢ ... If we wait until children are in junior-high or high-school, we will never manage it. ... ♠ ... The backbend is of extreme importance to any form of free gymnastics, and, as with all acrobatics, the sooner begun the better the results. ...

Results of the experiment, reported in Table 4, show that ODD is verified, corroborating the hypothesis that domain coherence is an essential feature of texts (i.e. there are only a few relevant domains in a text). Exceptions to ODD (10% of word occurrences) might be due to *domain variations* within Semcor texts, which are quite long (about 2000 words). In these cases the same word can belong to different domains in different portions of the same text. Figure 2, generated after having disambiguated all the words in the text with respect to their possible domains, shows how the relevance of two domains (*domain relevance* is defined in section 4.2), Pedagogy and Sport, varies through a single text.

As a consequence, the idea of 'relevant domain' actually makes sense within a portion of text (i.e. a context), rather than with respect to the whole text. This

also affects WSD. Suppose, for instance, the word 'acrobatics' (third sentence in Figure 2) has to be disambiguated. It would seem reasonable to choose an appropriate sense considering the domain relevant in a portion of text around the word, rather than relevant for the whole text. In the example, the local relevant domain is SPORT, which would correctly cause the selection of the first sense of 'acrobatics'.

4 Domains and WSD

A number of suggestions emerges from the domain-oriented text analysis carried out in section 3. We have evidence that domains can play a role in WSD, as they are able to define portions of text which are coherent with respect to word senses. A formal definition of a relevant domain with respect to the word to be disambiguated is required. It also emerged that words can be grouped into two classes with respect to their relation to the relevant domain: 'text related domain words', i.e. those that contribute to the definition of the relevant domain of the text and that can be disambiguated by means of a comparison with the relevant domain, and 'generic words', which lie outside the scope of domain-based techniques.

4.1 General methodology

The WSD algorithm described in this paper is an evolution of the approach proposed in Magnini and Strapparava (2000). The basic idea underlying that work was that the disambiguation of a word in its context is mainly a process of comparison between the domain of the context and the domains of the word's senses. One drawback of this approach is that it does not consider domain variations which would be especially appropriate for rather long texts. In addition, the methodology was completely supervised, being based on information from WORDNET DOMAINS. In order to overcome these problems the present approach considers portions of text within which domain relevance is calculated, and suggests a common framework to integrate domain information acquired from annotated texts.

The data structure that collects domain information is called a *domain vector*; this is a vector whose length is the number of considered domains. For the experiments reported in this section we used 43 of the domain labels defined in WORDNET DOMAINS. Domain vectors are proposed as a common notation to merge and easily manage information concerning either portions of texts or senses of words. We distinguish two kinds of domain vectors: (i) *text vectors*, which represent the relevance of a portion of text with respect to each domain in the considered set; and (ii) *sense vectors*, which represent the relevance of a sense of a given word with respect to each considered domain. Text vectors are computed according to the senses of the words of the text, while sense vectors are induced by the system by exploiting either training examples or information extracted from annotated resources, such as WORDNET DOMAINS.

To disambiguate a word occurrence w in a portion of text **T**, the text vector $\vec{T}_w$ for the portion **T** of text around w and the vectors $\vec{s}_1, \vec{s}_2, \ldots, \vec{s}_k$ for all the senses $s_1, s_2, \ldots, s_k$ of w must be computed. Then the system chooses the sense whose vector

maximizes the similarity with T_w. For the purposes of the following explanation, a text **T** is represented as a list of pairs $\langle lemma, POS \rangle$ provided by the output of a Part Of Speech tagger. Lemmas are indexed by their position in the text. The notation T_p is used to refer to the word in position p of text **T**.

4.2 Domain relevance

We represent the *relevance* of a domain with respect to a text as a positive real number in the range [0, 1]. Given a domain, it approaches 1 for texts highly related to such domain, 0 for unrelated ones. For example, a text whose topic is 'September 11th attack on the Twin Towers' could have domain relevance 1 with respect to POLITICS or MILITARY, and 0 for unrelated domains, such as SPORT.

To compute the relevant domain for a portion of text around a word in position T_p, the algorithm first identifies the subsequence of words from $T_{(p-c)}$ to $T_{(p+c)}$, where $2c$ is the size of context given to the algorithm as a parameter. Our tests on Semcor showed that the disambiguation performance decreases when c is smaller than 50. As a second step the algorithm collects all domain annotations in WORDNET DOMAINS for all the synsets of the selected words and computes the frequency of each domain in this set. However, we found that the frequency of a domain in a text does not imply its relevance in that text. For instance, it may happen that POLITICS is the most frequent domain in a news, even if the most relevant domain is actually a different domain, such as VETERINARY. This is because words about VETERINARY are less frequent than words about POLITICS, so that fewer word have a greater impact on domain relevance. Our hypothesis is that a domain is relevant for a text if its frequency in that text is significantly higher than in texts unrelated with that domain.

To estimate the relation between frequency and relevance we assume that in a balanced generic corpus the number of texts relevant to a certain domain D is equally distributed; thus mean and standard deviation for the frequency of D in such a corpus will tend to mean and standard deviation for random texts. In our experiments, mean and standard deviation for each domain in WORDNET DOMAINS have been determined over the LOB Corpus (Johansson 1986), considered as a balanced generic corpus for English. Domain relevance is evaluated using theorems about *normal distribution*: if the frequency for a domain D computed on a text **T** is significantly higher than the mean frequency of D on the corpus (e.g. it exceeds more than twice the standard deviation), then D is relevant with respect to **T**.

For example, suppose we want to evaluate the relevance of ECONOMY in the sentence 'Today I draw money from my bank'. The algorithm will collect all the domains for each sense of each word. The noun 'bank' has five occurrences of ECONOMY in its synsets out of a total of 10 senses (see Table 1), the noun 'money' has three occurrences out a total of three, and the verb 'draw' has one occurrence out a total of 33. Then the total frequency of ECONOMY is 1.53. Suppose that the mean frequency of ECONOMY for texts of this length in the LOB corpus is 0.2 and that the standard deviation is 0.1. These values represent the frequency distribution of ECONOMY in unrelated texts. As a consequence, ECONOMY will not be relevant

in texts in which its frequency is in the range [0, 0.4], while it will be considered relevant in texts with significantly higher frequency, as it is the case of our example.

4.3 Text vectors and sense vectors

A text vector is a domain vector extracted from a portion of text. Given a set of domains $\mathbf{D} = \{D_1, D_2, \ldots, D_n\}$, a text $\mathbf{T}$ and a word at position p, the text vector $\vec{T}_p$ will be the n-dimensional vector such that the component i is the relevance of D_i for $\mathbf{T}$ at the position p. Given a context, intuitively $\vec{T}_p$ represents the relevant domains for a point p of the text. Text vectors computed on different positions of the same text could differ, and many domains could be relevant for the same text.

A sense vector $\vec{s}$ is a domain vector extracted from a word sense. It provides two relevant pieces of information: its length represents the frequency of occurrence of that sense, and its direction represents the 'mean' vector of the texts where the sense usually occurs. The most natural way to build sense vectors is to apply supervised techniques to training data. However, in case training data are not available, simple sense vectors can be built by exploiting the information in WordNet Domains. This possibility, which we have adopted for the *all_words* task in Senseval-2, makes the proposed approach very flexible.

In the first case (i.e. training data is available), our current method for building sense vectors is quite simple, despite which it proved to be effective. A sense vector is built as the sum of the text vectors of its examples, thus capturing the direction of the mean vector of texts in which the sense typically occurs. This method is particularly effective for generic senses (i.e. Factotum), which typically occur in various types of texts and produce vectors without a dominant dimension. However, given that texts often have few prevalent domains (see the discussion in section 3.1), a high number of examples is necessary to produce vectors for generic senses. In this work (i.e. training over the Senseval-2 *lexical_sample* task) sense vectors were made only if at least 10 examples were available for a given sense. Figure 5 shows an example of a text vector and two sense vectors built from training data.

In the second case (i.e. training data not available), sense vectors have been built using WordNet Domains and Semcor. In this case a sense vector has 1's in the respective domain positions in WordNet Domains and 0's elsewhere, while it has the length proportional to the sense frequency in Semcor. For example, the vector for `bank#1` (see Table 1) would be ((Economy . 20) (Architecture . 0) ... (Sport . 0)). If the sense is annotated with Factotum, its sense vector has the direction of a vector of 1's for each component and is normalized assigning its length to 1. For instance, the vector for `bank#5` is ((Economy . $1/\sqrt{n}$) (Architecture . $1/\sqrt{n}$) ... (Sport . $1/\sqrt{n}$)) where n is the number of domains considered.

4.4 Disambiguation procedure

The disambiguation process of a word occurrence T_p consists of a simple comparison between the text vector $\vec{T}_p$ and the vectors of all the senses of such word. To take into account both the direction (i.e. the domain) and the length (i.e. the frequency)

Table 5. *Sense vectors ($\vec{s}_1$ and $\vec{s}_2$) and text vector ($\vec{T}_8$) for the text* **T** *'Today I have drawn money from my bank', for a subset of domains*

	SPORT	MEDICINE	ECONOMY	GEOGRAPHY
$\vec{s}_1$ (*Bank#1*)	0.02	0.08	1.73	0.04
$\vec{s}_2$ (*Bank#2*)	0.005	0.03	0.04	0.69
$\vec{T}_8$	0.2	0.005	1	0.03

of sense vectors, the dot product between $\vec{T}_p$ and each sense vector is computed. The result is a ranked list of senses for T_p and the final selection is based on a fixed threshold.

Three possible output situations are considered. If the match of one sense significantly exceeds the threshold, that sense is selected. If more than one good match is achieved (e.g. two senses of the same domain with similar frequency), the current strategy is simply not to assign a sense to the word, as no other disambiguating information is available. Finally, if no good match is achieved (as happens, for instance, with highly polysemous generic words), no sense is selected.

As an example, suppose one wants to disambiguate the word 'bank' in the sentence 'Today I have drawn money from my bank' with respect to the senses s_1: `bank#1` and s_2: `bank#2`. This situation is reported in Table 5, where both the sense vectors and the text vector for a subset of domains are represented. The dot product between $\vec{T}_8$ and $\vec{s}_1$ is 1.7356, whereas the dot product between $\vec{T}_8$ and $\vec{s}_2$ is 0.06185, allowing the correct selection of `bank#1`.

5 Results and discussion

The major goal in ITC-IRST's participation at SENSEVAL-2 was to test the role of domain information in word sense disambiguation. Results should be interpreted bearing in mind that no syntactic or semantic information except domain labels has been used. Figures 3 and 4 show the results of our system in two tasks we participated in (i.e. *English_all_words* and *English_lexical_sample*). For each task we report results for five intervals of word polysemy (i.e. 'all', [0–2], [3–4], [5–7], >= 8). The graph on the left compares precision and recall of our system with respect to a baseline (i.e. Most Frequent for *all_words* and Lesk for *lexical_sample*). Precision and recall have been calculated according to the scoring policy of SENSEVAL-2 (i.e. $P = correct/(wrong + correct)$, $R = correct/(wrong + correct + unattemped)$). The graph on the right shows precision considering parts of speech separately.

In general, the system compared very well with respect to other SENSEVAL-2 participants, achieving one of the highest scores (0.75 and 0.66, respectively for *all_words* and *lexical_sample*) as far as precision is concerned, at the cost of lower recall. The best SENSEVAL-2 system (i.e. SMUaw), which attempted all instances, achieved 0.69 and 0.64, respectively on the two tasks. The *all_words* task seems to benefit from the domain approach. A reason for this is that texts are long enough to

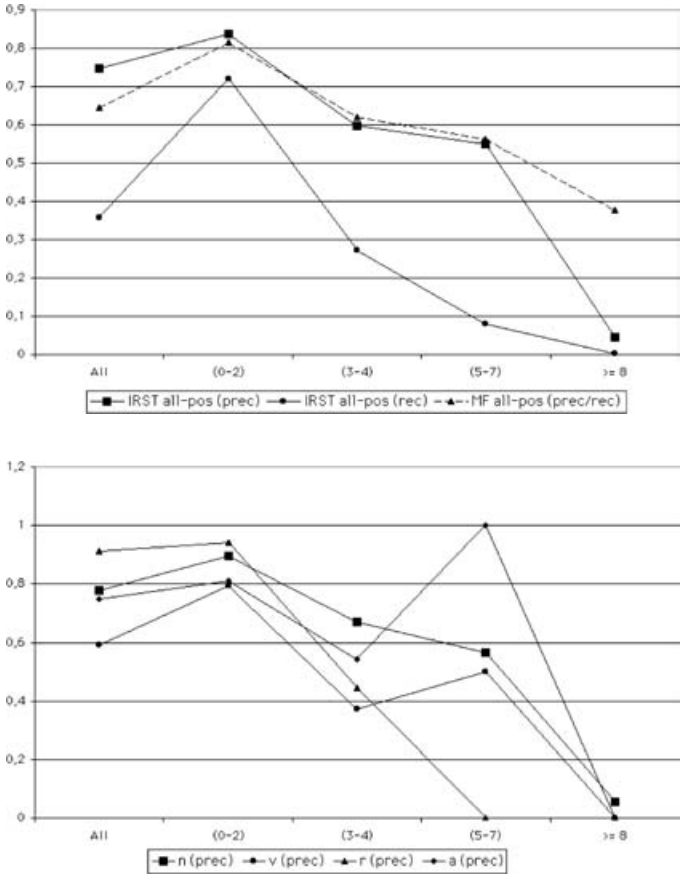

Fig. 3. *All_words Task*. On the left a comparison of our system with the 'most frequent' (MF) baseline. On the right the precision of our system for each POS.

provide an accurate context within which domains are coherent. We used a window of 100 content words around the target word. The *lexical_sample* task was inherently more difficult, for two reasons. First the context provided for disambiguation was generally shorter than the 100 words we used to build text vectors. Second, the high number of FACTOTUM words to be disambiguated resulted in a recall even lower (i.e. about 0.24) than for the *all_words* task. However, even if on average we have a better precision for the *all_words* task, it is significant that for the *lexical_sample* task there is less decrease in precision as the polysemy increases. This is due to the availability of training data, which let us produce more reliable sense vectors than those derived from WORDNET DOMAINS.

Perhaps the most evident drawback of the system is the loss of recall as word polysemy increases. In the *all_words* task we have ten points of difference with respect to the Most Frequent baseline at the [0, 2] polysemy interval which become around 40 points at the [5, 7] interval. This confirms the evidence reported in section 3.1 that just a few words in a text carry relevant domain information. Most of the words actually behave as FACTOTUM, as if they can equally occur in almost every

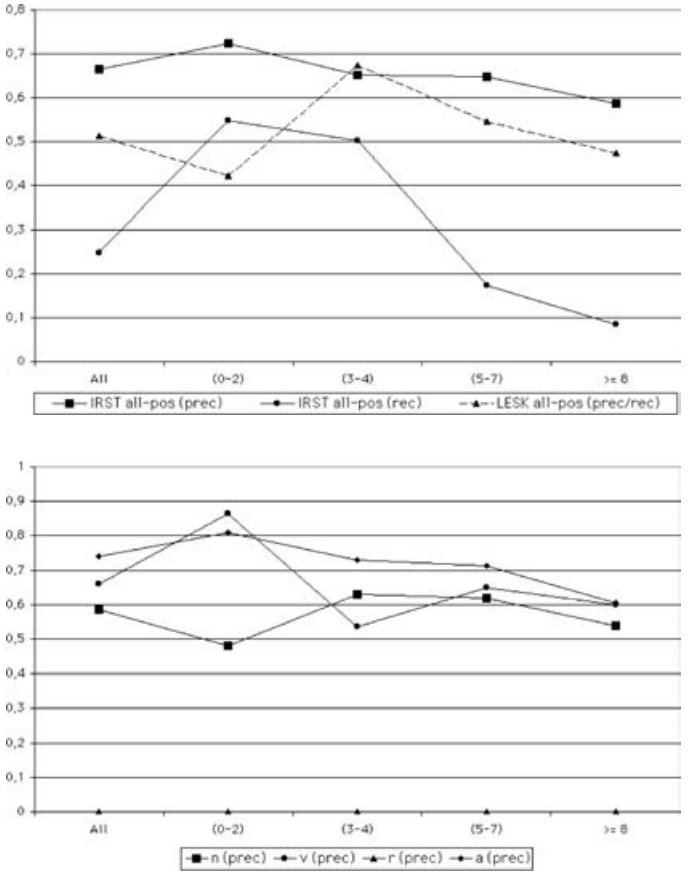

Fig. 4. *Lexical_sample Task*. On the left a comparison of our system with the 'Lesk' baseline. On the right the precision of our system for each POS.

domain. As an example, all the senses of the verb 'begin' in the *lexical_sample* task belong to FACTOTUM and our system did not provide any answers for its occurrences. Those words lie outside the domain approach and their senses must be captured using local information.

In a further investigation we analyzed the set of words in the *all_words* task for which we gave a good answer. The total set of good answers amounts to 882 word occurrences, from which we exclude 444 monosemous words. Out of the 438 remaining polysemous words 168 are TUG words (i.e. the majority of the senses are labeled with FACTOTUM, see section 3.1) and the remaining 270 can be considered as 'domain words' (TRD + TUD words). We then calculated the set difference among the answers belonging to the 'domain words' and the answers given by other systems participants at SENSEVAL-2 for the same set of words. It come out that systems with better recall than our system have a significant amount of 'domain words' for which a wrong answer was given. For instance, the SMUaw system (0.68 precision and 0.68 recall) did not give the correct answer to 36 (13%) occurrences of domain words.

This seems to be a strong indication that, at least for a subset of words, domain information plays a role in disambiguation which is not addressed by other systems.

6 Related work

The importance of domain information in relation to WordNet and to WSD has been remarked by several works in the last years. Perhaps the first attempt to introduce domains in WSD dates to Cowie, Guthrie and Guthrie (1992), where subject fields provided in LDOCE definitions were used to disambiguate word senses by means of simulated annealing techniques. Specific issues concerning WSD and sense tuning in the context of specialist domains have been addressed by Basili, Della Rocca and Pazienza (1997) and Cucchiarelli and Velardi (1998). More recently, Gonzalo, Verdeijo, Peters and Calzolari (1998) emphasized the role of domain information in relation to WordNet synsets. Following this line, Magnini and Strapparava (2000) introduced 'Word Domain Disambiguation' (WDD) as a variant of WSD, where for each word in a text a *domain* label (among those allowed by the word) has to be chosen instead of a *sense* label. We also argued that WDD can be applied to disambiguation tasks that do not require fine-grained sense distinctions, such as information retrieval and content-based user modeling.

A closely related work is that of Buitelaar and Sacaleanu (2001), which describes a method for determining the relevance of GermaNet synsets with respect to a specific domain. Term Relevance of a synset with respect to a domain is calculated by summing up term relevances for words in the synset and in its hyponyms (with a penalty for missing hyponyms). Finally, a methodology for the integration of domain-specific information into generic synsets is suggested in Vossen (2001).

7 Conclusions

We have described an approach to WSD based on domain information. The underlying assumption is that domains establish semantic relations among word senses that can be profitably used during the disambiguation process. The disambiguation algorithm takes advantage of domain annotations manually added to WordNet synsets. Results obtained at Senseval-2 show that for a significant subset of words a domain-based disambiguation achieves a high degree of precision compared to other systems.

Currently, the system uses just domain information. For the future, to improve the system recall, we plan to integrate the domain-based approach with supervised approaches that make use of local information, such as word collocation and grammatical context.

References

Basili, R., Della Rocca, M. and Pazienza, M. T. (1997) Contextual word sense tuning and disambiguation. *Appl. Artif. Intell.* **11**: 235–262.

Buitelaar, P. and Sacaleanu, B. (2001)Ranking and selecting synsets by domain relevance.*Proc. NAACL Workshop Wordet and Other Lexical Resources: Applications, Extensions and Customizations*, Pittsburgh.

Comaroni, J. P., Beall, J., Matthews, W. E. and New, G. R. (editors) (1998) *Dewey Decimal Classification and Relative Index*. Forest Press, Albany, NY.

Cowie, J., Guthrie, J. and Guthrie, L. (1992) Lexical disambiguation using simulated annealing. *Proc. of COLING-92*, 359–365. Nantes, France.

Cucchiarelli, A. and Velardi, P. (1998) Finding a domain-appropriate sense inventory for semantically tagging a corpus. *Natural Lang. Eng.* **4**(4): 325–344.

Fellbaum, C. (ed.) 1998 *WordNet. An Electronic Lexical Database*. MIT Press.

Gale, W., Church, K. and Yarowsky, D. (1992) One sense per discourse. *Proc. 4th ARPA Workshop on Speech and Natural Language Processing*, pp. 233–237. Harriman, NY.

Gonzalo, J., Verdejio, F., Peters, C. and Calzolari, N. (1998) Applying EuroWordNet to cross-language text retrieval. *Comput. and the Humanities*, **32**(2–3): 185–207.

Johansson, S. (1986) *The Tagged LOB Corpus*. Norwegian Computing Centre for the Humanities.

Krovetz, R. (1998) More than one sense per discourse. Technical Report, Princeton, NJ. NEC Research Institute.

Magnini, B. and Strapparava, C. (2000) Experiments in word domain disambiguation for parallel texts. *Proc. SIGLEX Workshop on Word Senses and Multi-linguality*, Hong-Kong.

Palmer, M., Fellbaum, C., Cotton, S., Delfs, L. and Dang, H. T. (2001) English tasks: All-words and verb lexical sample. *Proceedings SENSEVAL-2*, Toulouse, France.

Vossen, P. (2001) Extending, trimming and fusing WordNet for technical documents. *Proc. NAACL Workshop WordNet and Other Lexical Resources: Applications, Extensions and Customizations*, Pittsburgh.