Baseline Evaluation of WSD and Semantic Dependency in OntoSem

Sergei Nirenburg
Stephen Beale
Marjorie McShane
University of Maryland Baltimore County (USA)
email: `sergei@umbc.edu`

Abstract

This paper presents the evaluation of a subset of the capabilities of the OntoSem semantic analyzer conducted in the framework of the Shared Task for the STEP 2008 workshop. We very briefly describe OntoSem's components and knowledge resources, describe the work preparatory to the evaluation (the creation of gold standard basic text meaning representations) and present OntoSem's performance on word sense disambiguation and determination of semantic dependencies. The paper also contains elements of a methodological discussion.

1 Overview of OntoSem

OntoSem, which is the implementation of the theory of Ontological Semantics (Nirenburg and Raskin, 2004), is a text-processing environment that takes as input unrestricted raw text and carries out preprocessing followed by morphological, syntactic, semantic, and discourse analysis, with the results of analysis represented as a formal text-meaning representation (TMR) that can then be used as the basis for various applications. Text analysis relies on several knowledge resources, briefly described in the subsections below.

1.1 The OntoSem Ontology

The OntoSem ontology is a formal, language-independent, unambiguous model of the world that provides a metalanguage for describing meaning. It is a multiple-inheritance hierarchical collection of frames that contains richly interconnected descriptions of types of OBJECTs, EVENTs and PROPERTies. It is a general purposes ontology, containing about 9,000 concepts, that has a number of especially well developed domains that reflect past and ongoing application-specific knowledge acquisition. Each OBJECT and EVENT is described by several dozen properties, some property values being locally specified and others, inherited from ancestors.

Selectional restrictions in the ontology are multivalued, with fillers being introduced by a facet. The *value* facet is rigid and is used less in the ontology than in its sister knowledge base of real-world assertions, the fact repository (see Section 1.3). The facets *default* (for strongly preferred constraints) and *sem* (for basic semantic constraints) are abductively overridable. The *relaxable-to* facet indicates possible but atypical restrictions, and *not* blocks the given type of filler.

Event-oriented scripts encode typical sequences of EVENTs and the OBJECTs that fill their case-roles. Scripts are used to reason about both language and the world and, in addition to supporting text processing, can support simulation, as in our ongoing Maryland Virtual Patient project (see, e.g. McShane et al., 2007).

The number of concepts in the ontology is far fewer than the number of words or phrases in any language due to the existence of synonyms in language; the possibility of describing lexical items using a combination of ontological and extra-ontological (e.g., temporal) descriptors; the use of a single concept for each scalar attribute that describes all words on that scale (e.g., gorgeous, pretty, ugly); and the decision not to include language-specific concepts in the ontology.

As an example of the description of an ontological concept, consider an excerpt from the description of the concept ESOPHAGUS:

```
ESOPHAGUS
   IS-A           value ANIMAL-ORGAN
   LOCATION       sem   TRUNK-OF-BODY
   DISTAL-TO      sem   PHARYNX
   PROXIMAL-TO    sem   STOMACH
   LENGTH         sem   24
                  default-measure CENTIMETER
   INSTRUMENT-OF  sem   SWALLOW
   THEME-OF       sem   ESOPHAGEAL-CANCER
                        ACHALASIA ...
```

It is the richness of the property-based descriptions that permit the OntoSem ontology to be used for high-end applications like medical simulation and tutoring.

1.2 The OntoSem Lexicon

Even though we refer to the OntoSem lexicon as a semantic lexicon, it contains more than just semantic information: it also supports morphological and syntactic analysis and generation. Semantically, it specifies what concept, concepts, property or properties of concepts defined in the ontology must be instantiated in the TMR to account for the meaning of a given lexical unit of input.

Lexical entries are written in an extended Lexical-Functional Grammar formalism using LISP-compatible format. The lexical entry — in OntoSem, it is actually called a *superentry* — can contain descriptions of several lexical senses; we call the latter *entries*. As an example, consider the 2^{nd} sense of *take*:

```
(take-v2
   (cat v)
   (def "to begin to grasp physically")
   (ex "He took her hand as she got out of the car.")
   (syn-struc
      ((subject ((root $var1) (cat n)))
       (root $var0) (cat v)
       (directobject ((root $var2) (cat n)))))
   (sem-struc
      (HOLD
         (phase begin)
         (AGENT (value ^ $var1))
         (THEME (value ^ $var2)))))
```

The sem-struc says that the meaning of this word sense is the inceptive aspect ("phase begin") of the ontological event HOLD. The AGENT of HOLD is assigned the meaning of $var1 (the caret indicates "the meaning of") in the input text, and the THEME of HOLD is assigned the meaning of $var2. The OntoSem lexicon currently contains approximately 35,000 senses. For further information about the lexicon, see, e.g., McShane et al. (2005b).

A sister resource to the lexicon is the onomasticon, a lexicon of proper names linked to their respective ontological concepts: e.g., *IBM* is semantically described as CORPORATION.

1.3 The Fact Repository

The fact repository contains numbered remembered instances of concepts, with the numbers being used for disambiguation: e.g., HUMAN-FR88 is the 88^{th} human stored in the fact repository — e.g., *President Clinton*. Some aspects of "general world knowledge" are part of the seed fact repository used for all applications: e.g., France is recorded as NATION-FR47, and this information is available to all intelligent agents in our environment. This seed fact repository is then dynamically augmented as a given corpus is being processed. The fact repository also supports text processing, as for reference resolution: e.g., *President Clinton* in any text will be coreferential with HUMAN-FR88.

2 Text Meaning Representations (TMRs)

Section 3 includes an example of a TMR taken from the competition texts as well as a description of it. Here we will give a brief overview of the status of TMRs in OntoSem.

TMRs represent propositions connected by discourse relations (see Nirenburg and Raskin (2004), Chapter 6 for details). Propositions are headed by instances of ontological concepts, parameterized for modality, aspect, proposition time and overall TMR time.Each proposition is related to other instantiated concepts using ontologically defined relations. Coreference links form an additional layer of linking between instantiated concepts in the TMR as well as stored concept instances in the fact repository.

3 The STEP 2008 Shared Task

This section describes the evaluation of OntoSem results for the shared task at the STEP 2008 Workshop. Individual groups were allowed to make their own decisions with respect to a number of important parameters of the task, including, among others:

1. the nature of the metalanguage of semantic description (e.g., whether it relies on uninterpreted clusters of word senses, defined either within a language or cross-linguistically; whether it is based on a language-independent “interlingual” vocabulary; whether the latter is interpreted by assigning properties to vocabulary elements and constraints on the values of these properties, etc.);

2. the breadth of the coverage of phenomena (e.g., whether to include word sense disambiguation, semantic dependency determination, reference resolution, coverage of modality, aspect, time, quantification, etc.);

3. the depth of coverage of phenomena (e.g., the grain size of the description of word senses, the size of the inventory of semantic roles and other descriptive properties);

4. whether the analyzer is tuned to produce a complete result for any input; to produce partial results for all or some inputs; to produce output only for inputs it knows it can process;

5. whether (and how) the analyzer takes into account benign ambiguities, vagueness and underspecification;

6. whether the analyzer creates a semantic and pragmatic context for the input texts, thus modeling human ability to activate relevant knowledge not expressly mentioned in the text;

7. the practical end application(s) that a particular semantic analyzer aims to support.

In working on the shared task, our group has elected to test our system’s performance on word sense disambiguation (WSD) and semantic dependency determination. (For an early small-scale evaluation, see Nirenburg et al. (2004).) A prerequisite for our chosen evaluation experiment was filling the lacunae in lexical coverage. Two points are important to make here: a) we acquired what we consider a complete set of senses for each input word absent from our lexicon, not just the sense that was needed for the text — in other words, this was general-purpose acquisition; b) this was a part of routine ongoing work on resource acquisition and improvement in OntoSem. The only difference that these input texts made was with respect to the schedule of what to acquire first. Here are some basic statistics about our lexicon work. The input texts contained 270 lemmata, of which 36 (13%) were not originally in the OntoSem lexicon. 44 senses were added to the lexicon for the 36 words (these words were predominantly very specific, single-sense ones). In 12 cases, a sense was added to an existing lexicon entry (bringing the average number of senses for these 12 lemmata to 10.5). Finally, 5 word senses were added not because of any lacunae in the lexicon but just to make the life of the analyzer more difficult. In the end, the lexicon contained 1,168 senses for the 270 lemmata (an average of 4.33 senses per word).

Note that OntoSem processes more phenomena than WSD and dependency — aspect, time, speaker attitudes (called modalities in OntoSem), reference resolution and semantic ellipsis, discourse relations, unexpected input, metonymy, etc. In broad terms, the overall objective of the OntoSem analyzer, when viewed outside of the needs of this evaluation experiment, is to generate a significant amount of machine-tractable knowledge from the text, knowledge that includes not just a minimum of information gleaned from the text but also preference information obtained from the ontological and fact-repository substrate to be used in disambiguation heuristics and applications relying on the human ability to reconstruct meanings not overtly mentioned in the text for the purposes of reasoning and applications like question answering. For an overview of how TMRs produced by OntoSem can be used in lieu of traditional annotation schemes, see McShane et al. (2005a).

A standard example of using world knowledge activated in the process of text analysis is being able to infer (abductively) that once "virus" has been resolved to be the organism rather than the computer program, then if the word "operation" appears further on in the text, it is more probable that it means surgery rather than a computer operation or a military operation. The meaning extraction process also has a strong filtering ability to reject most of the senses of the words in the input as inappropriate for the particular text. This ability is not error-proof but the filtering capacity is quite strong even in the current state of the OntoSem analyzer; also note that the ratio of selected senses to those filtered away is a good measure of how much the static resources were tuned to a particular text or domain — the greater the number of senses per word, the less tuning occurred.

OntoSem distinguishes two stages of meaning representation producing, respectively, what is called basic and extended TMRs. The former covers the parts of the semantic representation that can be derived from the syntactic and lexical-semantic information and contains triggers (called "meaning procedures") for a variety of microtheories that require additional heuristics (and usually, as in the case of reference resolution, use general world knowledge from the ontology and fact repository as well as a window of text that is wider than a single sentence; for more on meaning procedures see McShane et al. (2004)). In this evaluation we constrained ourselves to the level of basic TMRs.

We will use the following relatively simple example sentence to demonstrate the scope of work of OntoSem.

> *Researchers have been looking for other cancers that may be caused by viruses.*

The basic TMR for this sentence is as follows:

```
SEARCH-103
   AGENT       RESEARCHER-102
   THEME       CANCER-104
   textpointer look
   word-num    3
   from-sense  look-v2
RESEARCHER-102
   AGENT-OF    SEARCH-103
   multiple    +
   textpointer researcher
   word-num    0
   from-sense  researcher-n1
CANCER-104
   THEME-OF    SEARCH-103
   CAUSED-BY   VIRUS-DISEASE-108
   multiple    +
   textpointer cancer
   word-num    6
   from-sense  cancer-n1
MODALITY-105
   TYPE        EPISTEMIC
   SCOPE       CAUSED-BY-109
   VALUE       0.5
   textpointer may
   word-num    8
   from-sense  may-aux1
CAUSED-BY-109
   DOMAIN      CANCER-104
   RANGE       VIRUS-DISEASE-108
   SCOPE-OF    MODALITY-105
   textpointer caused by
   word-num    10
   from-sense  cause-v1
VIRUS-DISEASE-108
   EFFECT      CANCER-104
   multiple    +
   textpointer virus
   word-num    12
   from-sense  virus-n1
```

We will describe select aspects of this TMR that apply to all frames. The head of the TMR is a SEARCH event with the instance number 103. Its AGENT is RESEARCHER-102 and its THEME is CANCER-104. Both of these case-role fillers also have their own frames that show inverse relations as well as other properties: e.g., RESEARCHER-102 has the property "multiple +", which indicates a set with cardinality > 1. A textpointer is the word in the text that gives rise to the given concept; its word number is shown, as is the appropriate lexical sense. The textpointer is included for the benefit of the human users, as an aid in debugging. It does not have any bearing on the meaning representation, as used by an application. Ontological concepts are unambiguous, as can be seen by the mapping of *virus* to the concept VIRUS-DISEASE rather than COMPUTER-VIRUS. Modalities are defined for type, scope and value, with values being on the abstract scale $\{0,1\}$. They are also defined for their attribution, which defaults to the speaker if no person is indicated explicitly in the text.

In this TMR, it so happens, there are no overt triggers for further processing — even though reference resolution is routinely triggered on all objects and events with the exception of those that are known not to require it (e.g., NPs with an indefinite article or universally known single entities, such as the sun).

The English gloss of the above TMR is as follows. There is one main event in the sentence — the searching event (represented by the word *look*). The agent of this event is a set of researchers and the theme of this event is a set of cancers, understood as diseases. This latter set is further qualified to include only those cancers that are caused by viruses, understood as organisms. The researchers are not sure at the moment of speech whether particular cancers are indeed caused — fully or partially — by viruses. It is known to the researchers and the author of the text that some cancer or cancers may, in fact, be caused by viruses (this is a contribution of the word *other* to the meaning of the sentence; another contribution of that word is posting a meaning procedure for blocking coreference of the cancer or cancers mentioned in this sentence with those mentioned in previous text). The search started before the time of speech and is still ongoing at the time of speech.

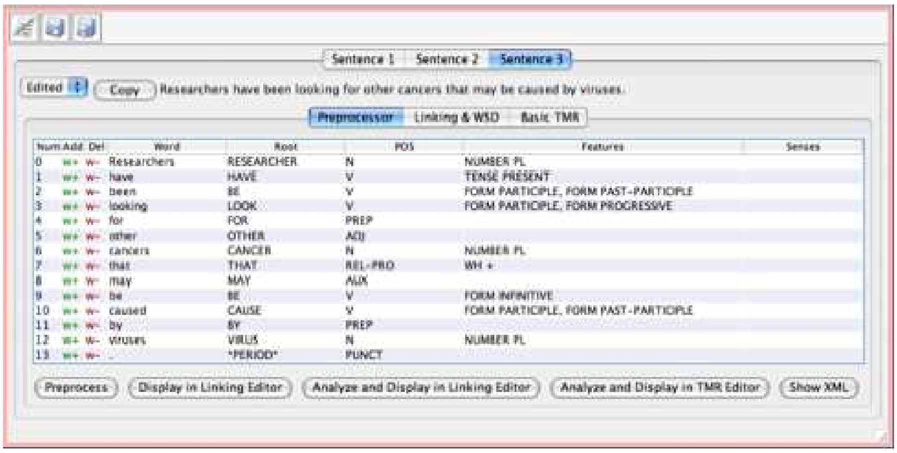
4 Creating Gold Standard TMRs

Gold standard TMRs form the basis for evaluating the results of automatic semantic analysis. To serve as useful measures, these TMRs must be created on the basis of the available static knowledge resources (in the case of OntoSem, mainly the lexicon and the ontology) and reflect the maximum of what a system could in principle do with the given resources.

Creating gold standard TMRs is similar to manually annotating texts using the metalanguage of OntoSem. Text annotation is a difficult and time-consuming (and, therefore, expensive) task. The deeper the level of annotation, the less reliable the results are. Even syntactic annotation poses problems and does not yield acceptable kappa scores measuring agreement among annotators (and, of course, agreement among annotators is not a fool-proof measure of the quality of an annotation). The annotation necessary for evaluating OntoSem TMRs is quite deep. Our experience showed that building gold standard TMRs entirely by hand is a very costly task — it requires highly skilled personnel and involves many searches in the knowledge resources for selecting appropriate TMR components.

In view of the above, we decided on semi-automatic production of gold standard TMRs as the most economical way of producing high-quality annotations. The process is, briefly, as follows. OntoSem operation is modularized into stages. Given an input, OntoSem runs the first stage and presents its results to the human validator/editor. The latter corrects any mistakes and submits the resulting structure to the next stage of OntoSem. The process repeats until OntoSem's final stage results are corrected and approved by the human validator, thus yielding a gold standard TMR.

Although “raw” TMRs can be cumbersome to read, the presentation format shown below — which is automatically generated from raw TMRs — reads rather easily as non-natural languages go. This demonstrates how our representation is actually quite NL-like, its role being not unlike the English “possibilistic” sentences of Schubert (2002). As concerns writing TMRs, people practically never do it — the most they do is check and, sometimes, correct the output of the automatic analysis system.


Sentence 1
Sentence 2
Sentence 3
Edited
Copy
Preprocessor
Linking & WSD
Basic TMR
Preprocess
Display in Linking Editor
Analyze and Display in Linking Editor
Analyze and Display in TMR Editor
Show XML

Researchers have been looking for other cancers that may be caused by viruses.

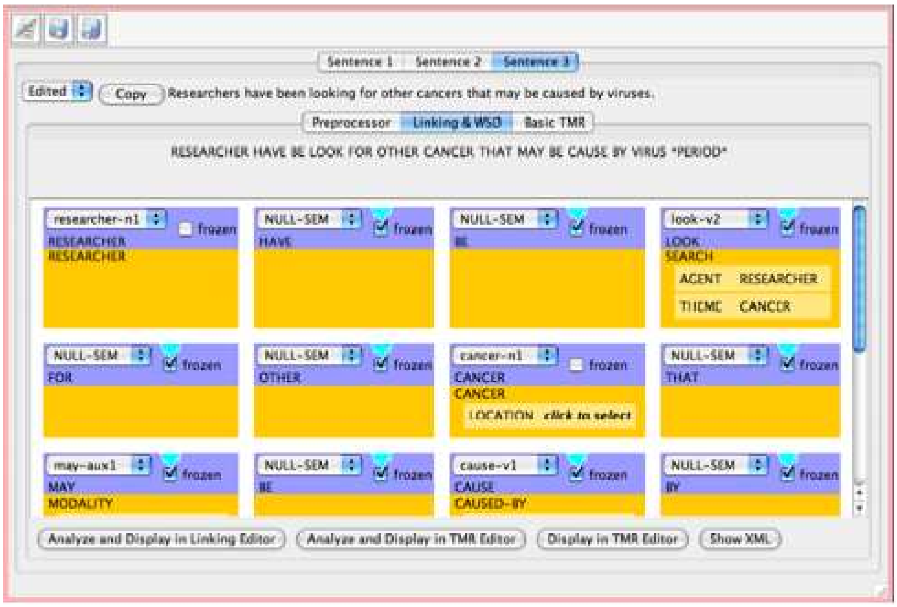
Num	Add	Del	Word	Root	POS	Features	Senses
0	w+	w-	Researchers	RESEARCHER	N	NUMBER PL	
1	w+	w-	have	HAVE	V	TENSE PRESENT	
2	w+	w-	been	BE	V	FORM PARTICIPLE, FORM PAST-PARTICIPLE	
3	w+	w-	looking	LOOK	V	FORM PARTICIPLE, FORM PROGRESSIVE	
4	w+	w-	for	FOR	PREP		
5	w+	w-	other	OTHER	ADJ		
6	w+	w-	cancers	CANCER	N	NUMBER PL	
7	w+	w-	that	THAT	REL-PRO	WH +	
8	w+	w-	may	MAY	AUX		
9	w+	w-	be	BE	V	FORM INFINITIVE	
10	w+	w-	caused	CAUSE	V	FORM PARTICIPLE, FORM PAST-PARTICIPLE	
11	w+	w-	by	BY	PREP		
12	w+	w-	viruses	VIRUS	N	NUMBER PL	
13	w+	w-	.	*PERIOD*	PUNCT		

Figure 1: The preprocessor editor of DEKADE

The user interfaces supporting the production of gold standard TMRs are incorporated in DEKADE, OntoSem’s Development, Evaluation, Knowledge Acquisition and Demonstration Environment. The editor for preprocessor results is illustrated in Figure 1.

The process of gold standard TMR creation has undergone modifications since the first version of DEKADE was deployed. In particular, experience showed that people find it difficult to edit the results of syntactic analysis, since it produces a densely populated chart of various options. So, instead of the syntax editor, we introduced a linking editor (see Figure 2) that helps to establish the correct linking between syntactic arguments and adjuncts on the one hand and semantic case roles and other ontological properties on the other. We will return to editing syntactic dependency structures once we devise or import an ergonomically appropriate method for this task. Figure 3 illustrates the editor of basic TMRs.

The semi-automatic methodology of creating gold standard TMRs has proved adequate. It takes a well-trained person on average less than a minute to correct preprocessing results for an average sentence of 25 words. Establishing correct linking between syntactic arguments and semantic case roles can take much longer. Together with the task of validating word sense selection (because of the peculiarities of the DEKADE editors), this task takes on average about 30 minutes per 25-word sentence. The time for final editing of the basic gold standard TMRs varies depending on how much material is present that does not relate to the “who did what to whom” component of meaning. However, the overall net time needed to create a gold standard TMR



Researchers have been looking for other cancers that may be caused by viruses.

RESEARCHER HAVE BE LOOK FOR OTHER CANCER THAT MAY BE CAUSE BY VIRUS *PERIOD*

researcher-n1 □ frozen
RESEARCHER
RESEARCHER

NULL-SEM ☑ frozen
HAVE

NULL-SEM ☑ frozen
BE

look-v2 ☑ frozen
LOOK
SEARCH
AGENT RESEARCHER
THEME CANCER

NULL-SEM ☑ frozen
FOR

NULL-SEM ☑ frozen
OTHER

cancer-n1 □ frozen
CANCER
CANCER
LOCATION *click to select*

NULL-SEM ☑ frozen
THAT

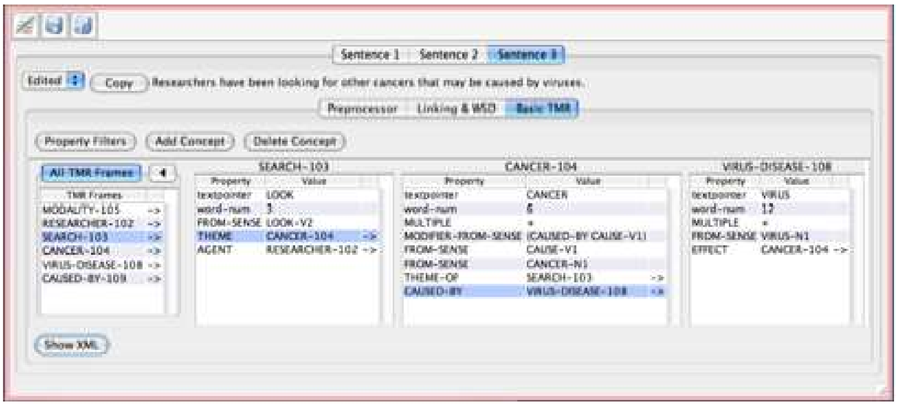
may-aux1 ☑ frozen
MAY
MODALITY

NULL-SEM ☑ frozen
BE

cause-v1 ☑ frozen
CAUSE
CAUSED-BY

NULL-SEM ☑ frozen
BY

Figure 2: The linking editor of DEKADE



Researchers have been looking for other cancers that may be caused by viruses.

TMR Frames	
MODALITY-105	->
RESEARCHER-102	->
SEARCH-103	->
CANCER-104	->
VIRUS-DISEASE-108	->
CAUSED-BY-109	->

SEARCH-103

Property	Value	
textpointer	LOOK	
word-num	3	
FROM-SENSE	LOOK-V2	
THEME	CANCER-104	->
AGENT	RESEARCHER-102	->

CANCER-104

Property	Value	
textpointer	CANCER	
word-num	6	
MULTIPLE	+	
MODIFIER-FROM-SENSE	(CAUSED-BY CAUSE-V1)	
FROM-SENSE	CAUSE-V1	
FROM-SENSE	CANCER-N1	
THEME-OF	SEARCH-103	->
CAUSED-BY	VIRUS-DISEASE-108	->

VIRUS-DISEASE-108

Property	Value	
textpointer	VIRUS	
word-num	12	
MULTIPLE	+	
FROM-SENSE	VIRUS-N1	
EFFECT	CANCER-104	->

Figure 3: The TMR editor of DEKADE

for a 25-word sentence is on the order of about 40 minutes.

Our methodology of semi-automatic creation of gold standard TMRs differs from the established rules of the game in the annotation-based evaluation business. We selected this approach because it a) significantly cuts the time needed for TMR production (we believe that the task would be simply impractical if attempted fully manually — the amount of annotation required being too extensive); b) simplifies the evaluation because it produces the TMR for the one paraphrase that OntoSem will (attempt to) produce automatically. The main efficiency gain in this semi-automatic production of gold standard TMRs is in using the OntoSem analyzer as a means of quickly retrieving and easily inspecting and selecting the lexical senses for the words in the input.

5 Results

Depending on a particular setting, OntoSem can seek one result for WSD or dependency determination, n-best results or both for each case of ambiguity. We chose to create both types of output. At the level of basic TMR there can in principle be one or more correct results, the latter case may signify a situation of underspecification or vagueness, to be resolved in OntoSem by special microtheories leading to the production of an extended TMR. In producing gold standard TMRs we always selected the single correct word sense, irrespective of whether we had to consult other parts of the input text or general world knowledge.

The results of OntoSem's performance on word sense disambiguation were evaluated against the gold standard TMRs and involved four different scores. In the case of each of the scores, performance on individual instances of word sense disambiguation was averaged over each sentence and text.

Score1 is 1 if the disambiguation was correct and 0 if it was not.

Score2 is 1 if the correct result is in the set of best results returned by OntoSem for a particular disambiguation instance such that the quality score generated for them by OntoSem is within 0.03 of the single best score (on the scale from 0 to 1). This score is 0 if the correct result is not within the above set. This measure was used because the significance of a preference at this fine-grain level is minimal.

Score3 takes into account the complexity of the disambiguation task by involving the number of senses from which the choice is made. Indeed, selecting out of 13 candidates is more difficult than out of, say, just 2. Thus, returning an incorrect result when there are 2 candidates earns a 0 but if there are more than 2 candidates, instead of 0 (as in Score1), we list the score of $1-(2/n)$, where n is the number of senses. A correct result is given a score of 1. As usual, cumulative scores were computed as simple averages over the input words.

Score4 was calculated using partially corrected results of the preprocessing and syntactic stages of the analysis process. This score did **not** take into account the complexity of the disambiguation task, as did Score3. It was, in fact, Score2 computed over partially corrected preprocessing results. The purpose of using it is to attempt to assign blame across the various components of the rather complex OntoSem system.

Semantic dependency results are scored as 1 when they are correct and 0 when they are incorrect. If an element of TMR is not a part of the semantic dependency structure but should be then we count that as an error. The results of the evaluation are summarized in the table below:

Text	WSD Score1		WSD Score2		WSD Score3		WSD Score4		Dependencies	
1	42/55	.78	43/55	.87	49.707/55	.90	48/55	.87	20/34	.59
2	30/40	.75	31/40	.78	37.030/40	.93	36/40	.90	15/20	.75
3	22/29	.76	23/29	.79	26.348/29	.91	24/29	.83	18/22	.82
4	75/114	.66	77/114	.68	96.966/114	.85	87/114	.76	44/84	.52
5	67/105	.64	67/105	.64	88.440/105	.84	85/105	.81	36/62	.58
6	82/129	.64	84/129	.65	104.807/129	.79	92/129	.71	45/99	.45
7	95/133	.71	95/133	.71	114.260/133	.86	101/133	.76	47/88	.53
Total	413/605	.68	420/605	.69	517.550/605	.86	474/605	.78	225/410	.55

6 Discussion

Our goal was not to get the best results for this particular task but rather to test some of the capabilities of the "raw" OntoSem analyzer. We have always advocated hybridization of methods, a direct consequence of our group's belief in task- rather than method-oriented approaches to system building. We fully expect to take that route when we are putting together the next end application. However, from the scientific point of view, it is important to assess the quality and promise of a particular method, even if it is known beforehand that it will be used in practical applications together with other methods.

Some practical limitations influenced our results. This is why we included the word "baseline" in the title of this paper. We intend to eliminate these limitations over time. The syntactic support for the system has been recently fully revamped to incorporate the Stanford parser. The work on deriving full syntactic dependency structures compatible with the requirements and coverage of the OntoSem syntax-to-semantics linking module from the results provided by the Stanford parser was not completed by the time of the evaluation. This means, among other things, that not all the diathesis transformations needed have been included.

In the current version of DEKADE, the automatic validator of lexicon acquisition does not yet indicate to acquirers when a new lexical sense has the same or very similar syntactic and semantic constraints. As a result, some of the word senses cannot currently be disambiguated using the standard selectional restriction-based method.

In addition to the above general limitations, there were some challenges specific to the particular input corpus. For example, the microtheory of measure has not yet been fully implemented in OntoSem.

We have not yet done enough to determine the contribution of preprocessing, syntax and the various semantic microtheories to the final result. We intend to pay more attention to this blame assignment task.

We restricted ourselves to evaluating just WSD and dependency determination because of time and resource limitations. It is clear that the quality of OntoSem's output for the other microtheories mentioned in Section 3 above, among others, must also be evaluated.

In addition to the above, we also plan to run an evaluation of OntoSem's performance on treating unexpected input, using the version of the lexicon existing before the shared task started.

We will also work on modifying the relative importance of heuristics from different sources. In particular, we will work toward reducing the influence of syntactic clues and thereby moving the center of gravity of the analysis process toward semantics proper.

The process of evaluating TMRs has benefits beyond assessing our progress. It facilitates debugging and enhancing the knowledge resources and processing modules of the system. Finally, we believe that the gold standard TMRs required for evaluation can also be used as an annotated training corpus for machine learning experiments in semantic analysis. We believe that the annotation task is quite feasible. If we estimate the time to create a gold standard basic TMR for a 25-word sentence takes one person-hour, counting the estimated time for acquiring the missing lexicon and ontology information, then it should be possible to create a 100,000-word corpus of gold standard basic TMRs in about two person-years.

References

McShane, M., S. Beale, and S. Nirenburg (2004). Some meaning procedures of ontological semantics. In *Proceedings of LREC-2004*.

McShane, M., S. Nirenburg, and S. Beale (2005a). An NLP lexicon as a largely language independent resource. *Machine Translation 19*(2), 139–173.

McShane, M., S. Nirenburg, and S. Beale (2005b). Text-meaning representations as repositories of structured knowledge. In *Proceedings of the Fourth Workshop on Treebanks and Linguistic Theories (TLT 2005)*.

McShane, M., S. Nirenburg, S. Beale, B. Jarrell, and G. Fantry (2007). Knowledge-based modeling and simulation of diseases with highly differentiated clinical manifestations. In *Proceedings of the 11th Conference on Artificial Intelligence in Medicine (AIME 07)*.

Nirenburg, S., S. Beale, and M. McShane (2004). Evaluating the performance of the OntoSem semantic analyzer. In *Proceedings of the ACL Workshop on Text Meaning Representation*.

Nirenburg, S. and V. Raskin (2004). *Ontological Semantics*. MIT Press.

Schubert, L. (2002). Can we derive general world knowledge from texts? In *Proceedings of the HLT Conference*.