КАК МОДЕЛИРОВАТЬ ПОНИМАНИЕ ЕСТЕСТВЕННОГО ЯЗЫКА: ФОРМАЛЬНОЕ ПРЕДСТАВЛЕНИЕ СМЫСЛА

Богуславский И. М. (Igor.M.Boguslavsky@gmail.com),
ИППИ РАН, Москва, Россия;
Политехнический университет Мадрида, Мадрид, Испания

Диконов В. Г. (dikonov@iitp.ru),
Иомдин Л. Л. (iomdin@iitp.ru),
Тимошенко С. П. (timoshenko@iitp.ru),
ИППИ РАН, Москва, Россия

Ключевые слова: семантика, формальное представление смысла, онтология знаний, лексические функции, модель «Смысл — Текст», конверсивы

SEMANTIC REPRESENTATION FOR NL UNDERSTANDING

Boguslavsky I. M. (Igor.M.Boguslavsky@gmail.com)
Institute for Information Transmission Problems,
Russian Academy of Sciences, Moscow, Russia;
Universidad Politécnica de Madrid, Madrid, Spain

Dikonov V. G. (dikonov@iitp.ru),
Iomdin L. L. (iomdin@iitp.ru),
Timoshenko S. P. (timoshenko@iitp.ru),
Institute for Information Transmission Problems,
Russian Academy of Sciences, Moscow, Russia

While mainstream semantic parsing mostly consists in word sense disambiguation, semantic role labeling and assigning WordNet/FrameNet categories, deeper NL understanding requires much more. It includes understanding of the meaning of words, extralinguistic knowledge and is based on a more intricately elaborated representation of this meaning than that provided by standard resources. For example, the semantic model should not only know that *ask for, implore* and *demand* belong to the same REQUEST frame. It should also formally represent the very idea of an incentive speech act (e.g. 'X tells Y that he wants him to do Z') and even the difference between such request varieties as represented by the words listed. Our aim is to build a semantic analyzer supplied with this kind of semantic knowledge and capable of constructing semantic representations that convey this knowledge and can be used for inferences. However, before constructing a parser, one should define the target representation. The focus of this paper is to propose a semantic representation richer than usually considered. Since the depth of representation is an important decision in language modeling, the topic deserves a detailed discussion. Our paper demonstrates selected NL phenomena untreatable by state-of-the-art parsers and semantic representations proposed for them.

Key words: semantic representation, ontology, Meaning-Text Theory, formal sense representation, lexical functions, semantic parsing

1. Introduction

Mainstream shallow semantic parsing mostly includes named entity recognition, word sense disambiguation, semantic role labeling and assigning WordNet/FrameNet categories (e.g. Shi, Michalcea 2004). For example, the sentence

(1) *Messi scored a goal*

is typically assigned a semantic representation of the type

(2) *'A person named Messi is the agent of a GoalEvent'.*

Deeper NL understanding requires much more. In should include understanding of words' meaning, take into account available world knowledge and, ideally, involve as much inference as possible. In a sense, the level of text understanding is determined by the amount of inferences the cognitive agent can make. For sentence (1) such an understanding would include at least the following explicit data:

(3) (a) 'Messi is the captain of the Argentina national football team and a player of FC Barce-lona' [encyclopedic knowledge],

(b) 'Messi hit the ball, which resulted in the ball being located in the goal of the opposite team; as a result, the score of the team for which Messi was playing increased by 1' [linguistic knowledge: the explicit interpretation of the expression *score a goal*].

To obtain this level of understanding, one should have access to both linguistic and world knowledge and have an inference engine.

A project aiming at this type of semantic analysis has been initiated at IITP RAS. The ultimate goal of this project is to build a broad coverage semantic parser. In this paper, however, we will focus on one aspect of this work — the appropriate semantic representation and its depth. As opposed to usual routine of discussion we will proceed bottom-up rather than top-down: instead of describing a semantic language and illustrating it by linguistic examples, we will depart from some non-trivial linguistic phenomena which are rarely (if ever) tackled by existing semantic parsers and show how they are represented in our language (Section 2). This material is worth discussing because the depth of representation is a crucial decision in language modeling, which should be taken irrespective of the way in which the parser processes the text. Logically, the target representation precedes the construction of the parser. Once we decide on the range of linguistic phenomena to be covered and devise a formal representation for them, we can choose a strategy of parser building. Our strategy is primarily rule-based. Obtaining semantic representation of the kind proposed below entirely by machine learning methods would require large annotated corpora, which are difficult and expensive to produce. A rule-based system may be viewed as a convenient step towards semi-automatic creation of such a corpus. On the other hand, we believe that knowledge-intensive methods have important advantages over data-driven ones, as far as the transparency and explanatory power is concerned.

Although the main contribution of the paper is theoretical, the feasibility of the representations proposed is confirmed by their being based on the available resources (a lexicon, an ontology, a rule-based engine) that we briefly describe in Section 3. In Section 4 we will present related work, and conclude in Section 5.

2. Selected issues of semantic analysis

2.1. Normalization and paraphrasing

Semantic analysis rules operate on Normalized Syntactic Structure and produce Basic Semantic Structures (SemS, see Section 3 below). One of the first tasks that should be done is the canonization. It includes restoring subjects of non-finite verbs *(I want to run → I want: I run),* processing of ellipsis, comparative constructions and the like. We will illustrate one canonization pattern: elimination of semantically void collocates. This operation is performed by means of a paraphrase generator based on Lexical Functions (LF) (Mel'čuk 1996; Apresjan, Cinman 2002). The paraphrase generator is a system of rules relying on a rich dictionary of lexical functions. In the semantic analyzer, the generator reduces sentences containing collocate LFs to the canonical form without these LFs. Some examples are:

(4) → (5), (6) → (7), (8) → (9).

(4) *John <u>has</u> respect for his teachers / John's teachers <u>enjoy</u> his respect / John <u>treats</u> his teachers with respect*

(5) *John respects his teachers*

(6) *The police <u>gave</u> the protesters an order to disperse / The protesters <u>were ordered</u> by the police to disperse / The protesters <u>received</u> an order from the police to disperse.*

(7) *The police ordered the protesters to disperse.*

(8) *The experts should <u>submit</u> / <u>prepare</u> / <u>make</u> / <u>produce</u> a report on chemical weapons.*

(9) *The experts should report on chemical weapons.*

Strictly speaking, the sentences in these pairs are not fully synonymous. However, the semantic representation we are striving at does not aim to account for all subtleties of meaning. The level of granularity of semantic representations should be determined by the task for which they are constructed. The immediate objective of our semantic representations is to support inference. For this aim, semantic differences that can be observed in these pairs are not relevant. Should an application require finer-grained representations, paraphrasing rules should be made more precise. An example of a subtler representation is given in the next section.

2.2. Semantic definitions of NL words and ontology concepts

Deep NL understanding requires much more elaborated meaning representation than that provided by standard resources. For example, semantic parsers based on FrameNet annotate verbs like *ask for, implore* and *demand* by relating them to the same REQUEST frame. It is true that all the three verbs have the same set of roles. The generalized frame REQUEST allows us to capture this similarity, but we also want to preserve the knowledge about their difference. First, one should make explicit the very idea of the incentive speech act (e.g. 'X tells Y that he wants him to do Z'). This should be made in a formal language so that it could be used for inferences. Second, since our ultimate aim is to model natural language as fully as possible, it is desirable to account for the semantic difference between the varieties of this speech act. There exist many NL speech act types in which the agent informs the addressee that he wants the latter to do something.

Below, we give definitions of three of them: *ask* (as in *He asked to open the window), implore* (as in *They implored her to help)* and *demand* (as in *She demanded an explanation).* Roughly, the difference between *ask* and *demand* is that the one who is asking does not think that the addressee is obliged to fulfill the request, while the one who is demanding assumes that the addressee must do it. Imploring adds to asking the idea that fulfilling the request is very important for the agent so in persuading the addressee to do it he tries to affect his feelings. In the definitions, variables are marked with the ? sign. For brevity, the ontological class to which the variable belongs is encoded by the name of the variable.

(10) *ask for (?Agent1,?Agent2,?Action)* [='?Agent1 tells ?Agent2 that he wants him to do ?Action; ?Agent1 does not think that ?Agent2 must do ?Action']

```
hasAgent(Tell,?Agent1)
hasRecipient(Tell,?Agent2)
```

```
hasObject(Tell,Want)
hasSubject(Want,?Agent1)
hasObject(Want,?Action)
hasAgent(?Action,?Agent2)
hasScope(Negation,Opinion)
hasSubject(Opinion,?Agent1)
hasObject(Opinion,?Action)
hasScope(MustModality,?Action)
```

(11) *implore (?Agent1,?Agent2,?Action)* [='?Agent1 asks ?Agent2 to do ?Action; it is very important for ?Agent1 that ?Agent2 realizes ?Action; ?Agent1 tries to affect the feelings of ?Agent2']

SemS of (11) consists of the SemS of (10) plus the following:

```
hasSubject(Important,?Action)
hasObject(Important,?Agent1)
hasSubject(Degree,Important)
hasValue(Degree,high)
hasAgent(Affect,?Agent1)
hasObject(Affect,Feeling)
hasSubject(Feeling,?Agent2)
```

(12) *demand (?Agent1,?Agent2,?Action)* [='?Agent1 tells ?Agent2 that he wants him to do ?Action; ?Agent1 thinks that ?Agent2 must do ?Action']

```
hasAgent(Tell,?Agent1)
hasRecipient(Tell,?Agent2)
hasObject(Tell,Want)
hasSubject(Want,?Agent1)
hasObject(Want,?Action)
hasAgent(?Action,?Agent2)
hasSubject(Opinion,?Agent1)
hasObject(Opinion,?Action)
hasScope(MustModality,?Action)
hasAgent(?Action,?Agent2)
```

2.3. Converse terms

Natural languages have hundreds of converse terms, i.e. pairs of words that denote the same situation but differ in the syntactic status of their arguments. Obvious examples are *husband — wife, buy — sell, to the right of — to the left of, more — less, better — worse,* etc. Although these words are not synonyms, if we swap positions of arguments we obtain equivalent assertions:

(13) *John is Mary's husband = Mary is John's wife.*

(14) *John bought a house from Mary = Mary sold a house to John.*

(15) *The table is to the right of the window = The window is to the left of the table.*

(16) *John likes physics more than geography = John likes geography less than physics.*

Since converse terms refer to the same situation, it is sufficient for a semantic language and ontologies to contain only one term of the pair. In our semantic language, we have only one correlate for the 'more'/'less' pair — concept MORE. In representing this meaning, we differ from some other approaches (such as e.g. (Nirenburg, Raskin 2004), which treat 'more' as a binary relation: A>B. Our MORE concept has three arguments: A — "what is more?", B — "more than what?", C — "by how much is A more than B?". In sentence (17) the arguments of MORE are: A=John's height, B=Bill's height, C=3 cm.

(17) *John is 3 cm taller than Bill.*

On the other hand, one can opt for having both members of the converse pair. For instance, we represent *husband* and *wife* by different concepts, because these social roles are bound by different conventions and stereotypes which have to be described in the ontology.

An interesting case of converse relations, which as far as we are aware was first introduced in (Boguslavsky 2009), is the relationship between *all* and *only.*

(18) *Here are all my documents*
("for any document x of mine, it is true that x is here").

(19) *Here are only my documents*
("for any *x* that is here, it is true that *x* is my document").

This allows us to have only one semantic unit — a two-place predicate `All`, which covers both *all* and *only.* Here are semantic structures for sentences (22) and (23):

(20) *All the children who guessed the riddle got a prize.*

(21) *Only the children who guessed the riddle got a prize.*

(20a)
```
hasElements(Set,Child)
hasAgent(Guess,Child)
hasObject(Guess,Riddle)
hasAgent(Get,Set)
hasObject(Get,Prize)
hasSubject(All,Set)
hasObject(All,Get)
```

(21a)
```
hasElements(Set,Child)
hasAgent(Guess,Child)
hasObject(Guess,Riddle)
hasAgent(Get,Set)
hasObject(Get,Prize)
hasSubject(All,Get)
hasObject(All,Set)
```

2.4. Evaluation of objects

Evaluation of objects and events plays an enormous role in our life, everyday behavior and common sense reasoning. Therefore the world knowledge modeled by the ontology should contain manifold information on what is good and, bad and for whom. For many situations, we are aware that they are either beneficial or detrimental to the interests of some of their participants. For example, if somebody dies, is sick, late for an appointment, gets ruined, receives a rebuke, or fails an exam, by default this is bad for him. If, on the other hand, he recovers from an illness, gets an award, is promoted or attains his aim, then, again by default, it is beneficial for him. Some situations are estimated differently from the point of view of their different participants. For example, a victory (in a conflict, debate, sports competition, etc.) is beneficial for the winner and adverse for the loser. We will demonstrate that this kind of information can play a role in text understanding. Then we will show how it is incorporated in our Ontology and used for semantic analysis.

Consider sentence (22) and its two possible continuations — (23) and (24).

(22) *In the first tour FC Spartak overwhelmed FC Dynamo.*

(23) *In the second tour FC Zenith suffered the same fate.*

(24) *In the second tour FC Zenith managed to achieve the same thing.*

Both (23) and (24) contain the anaphoric expression *the same* that refers to sentence (22). In both cases, a situation is described that is similar to (22), the only difference being that one of the clubs is replaced with Zenith. In (23) an analogy is drawn between Zenith and Dynamo, and in (24) between Zenith and Spartak. In other words, (23) is unambiguously understood as 'Spartak overwhelmed Zenith', while (24) means that '*Zenith overwhelmed Dynamo*'. It is noteworthy that even though neither (23) nor (24) explicitly specifies the opponent of Zenith, it is "calculated" from the evaluation semantics.

To be able to draw these conclusions, the system should dispose of the following knowledge:

(a) "P is fate suffered by X" implies that P is not in the interests of X;

(a) "X managed to achieve P" implies that P was among X's aims and P is beneficial for X;

(a) "victory of X over Y" is beneficial for X but not for Y.
This knowledge is incorporated into the system as follows:

- The Ontology contains an `Evaluation` concept, which has 4 slots: the agent of the evaluation (`hasAgent`), the object or event under evaluation (`hasObject`), the value of the evaluation (`hasValue`) — good or bad and the beneficiary, i.e. someone for whom the object or event is beneficial or adverse (`hasBeneficiary`).
- This concept is introduced into the description of the concepts which include a default evaluation (cf. examples above). The `WinEvent` concept, which has slots for the winner (`hasWinner`) and for the loser (`hasLoser`) and which covers both a victory and a defeat, is assigned the following properties, among others:

```
hasWinner(WinEvent,?SportAgent1)
hasLoser(WinEvent,?SportAgent2)
hasObject(Evaluation-01,WinEvent)
hasValue(Evaluation-01,good)
hasExperiencer(Evaluation-01,?SportAgent1)
hasObject(Evaluation-02,WinEvent)
hasValue(Evaluation-02,bad)
hasExperiencer(Evaluation-02,?SportAgent2)
```

- A reference to evaluation is included into semantic rules that interpret natural language evaluating expressions. X *suffered the fate of P* contains the component "P is estimated to be bad for X". In our semantic language it is represented as follows:

```
hasObject(Evaluation,P)
hasValue(Evaluation,bad)
hasBeneficiary(Evaluation,X)
```

- Expressions like *X succeeded in / achieved P* include in their definition a reference to P being the aim of X, which in its turn implies that P is beneficial for X:

```
hasObject(Evaluation,P)
hasValue(Evaluation,good)
hasBeneficiary(Evaluation,X)
```

Now, let us see how this knowledge helps interpret sentences (23) and (24). As mentioned above, proposition (22) serves as the antecedent of '*the same*', so theoretically, it can be introduced into the SemS of both (23) and (24) in two different ways:

(24a)
```
hasWinner(WinEvent,Zenith)
hasLoser(WinEvent,Dynamo)
```

(meaning that Zenith beat Dynamo like Spartak beat Dynamo) or

(23a)
```
hasWinner(WinEvent,Spartak)
hasLoser(WinEvent,Zenith)
```

(meaning that Zenith lost to Spartak like Dynamo lost to Spartak).

However, taking into account that the meaning of *Zenith suffers a fate* assigns to Zenith the role of the beneficiary of a negative evaluation, while in the *WinEvent* it is the winner who benefits, version (24a) should be rejected for sentence (23). In a similar way, (23a) is rejected for (24).

3. Semantic analysis in ETAP

The cases analyzed above make part of a small corpus manually annotated with semantic structures. The corpus comprises several hundred sentences which are partly extracted from the articles on football published at various sports portals and partly composed by ourselves. This corpus is used for developing a rule-based semantic analyzer capable of building gold standard structures. As of now, more than a hundred sentences have been processed by the analyzer and assigned correct semantic structures. Once a rule-based analyzer is constructed, it will open the possibility to considerably augment a corpus, which could then be used for refining and evaluating the analyzer, as well as for developing other semantic parsers.

Our analyzer will be described in detail at a later stage when more experiments have been conducted and more data accumulated. Now we will only give a brief sketch of its architecture and resources used.

The analyzer is a new module of the ETAP-3 linguistic processor (see e.g. Apresjan et al. 2003). Before being sent to semantic analysis, the text is subjected to morphological analysis, dependency parsing, and normalization. The semantic analysis consists in two major steps. First, Normalized Syntactic Structures of all sentences are individually transformed into Backbone Semantic Structures (BSemS). At this canonization stage, missing arguments are restored and semantically void collocates are eliminated (see Section 2.1 above). Then all meaningful words are replaced by their definitions. Second, BSemSs are enriched with the world knowledge and the contextual knowledge from the previous text and thus converted to Enhanced Semantic Structures.

Linguistic information is contained in two kinds of resources: the combinatorial dictionary and several sets of rules. World knowledge is contained in the Ontology, while contextual knowledge is stored in the Fact Repository. The ontology we constructed for the analyzer has two sources. We compiled a small domain ontology of football, reusing the existing football ontologies (e.g. http://www.lgi2p.ema.fr/~ranwezs/ontologies/soccerV2.0.daml). Then we merged it with a general ontology developed on the basis of SUMO (http://www.ontologyportal.org/), which we partially restructured and complemented with a large set of properties. In our analyzer the ontology plays a two-fold role. On the one hand, it is a source of structured information about the world. It is composed of a hierarchy of concepts and instances supplied with properties. Many concepts belong to various classes at a time, so that they inherit properties from multiple sources. On the other hand, the ontology serves as a metalanguage for semantic representation. It is an inventory of semantic units that make up semantic structures.

4. Related work

A popular resource for developing shallow semantic parsers is FrameNet, mentioned above in Section 2.2. Semantic definitions of frames are intended for humans and are not written in a formal language. Therefore, they structures produced by FrameNet-based semantic parsers cannot be used for inference.

There are several directions in which semantic processing relying on ontologies is currently carried out. Our approach to semantic analysis is closely related to the OntoSem approach, with which we share several important ideas, although our linguistic framework is substantially different (Nirenburg, Raskin 2004), (Akshay Java et al. 2006), (Akshay Java et al. 2007), (Raskin, Taylor 2010), (Raskin et al. 2010).

Still another linguistic model underlies a series of papers on FuncGram — an advanced semantic Knowledge Base rooted in the Lexical-Constructional Model (Mairal Usón, Periñán-Pascual 2009, Periñán-Pascual, Arcas-Túnez 2010 a, b, Mairal Usón 2010).

Semantic processing based on OWL-implemented ontologies and Descriptive Logic does not allow accounting for exceptions. Interesting work is being done in order to incorporate common sense reasoning, which is inseparable from defeasible statements (Fahlman 2011), (Carlson et al. 2012). In (Bouayad-Agha et al. 2012a), (Bouayad-Agha et al. 2012b) a two-layer ontology is used for NL generation.

Modern QA systems use ontologies as the core knowledge component. They are often used to annotate original data obtained from the web sites and other sources of unstructured or loosely structured texts. The annotated data is stored in the databases and retrieved to answer the user's questions (Shiyan Ou et al. 2008). (Fernandez et al. 2011), (Cardoso et al. 2010) describe semantically-aware QA working on structured data modeled by an ontology.

There is much research on semantic parsing within the machine learning paradigm. Interesting results have been obtained in supervised and unsupervised semantic parsing in (Ge and Mooney, 2005), (Poon and Domingos, 2009), (Titov and Klementiev, 2011), (Clarke et al. 2010), (Liang et al. 2011). A combination of machine learning and rule-based approaches is used for semantic processing in (Moldovan et al., 2010). However, for the kind of structure we are interested in, no annotated corpora are available.

5. Conclusion

Deep understanding of NL requires more expressive semantic representation than is currently used in most state-of-the-art semantic parsers. It should be equally well-suited for expressing lexical meanings and world knowledge. Such a representation can be built on the basis of RDF-style subject-predicate-object triples. We showed a variety of NL phenomena that are conveniently expressed in such a language. To the best of our knowledge, some of this material is introduced in the computational semantics area for the first time. This is true for semantic definitions of many concepts in 2.2. Our approach to the evaluation topic is also new: it differs from the approaches used in the sentiment analysis domain which are prevalent today. We showed how lexical meanings can be decomposed, semantically void collocates can be identified and eliminated not affecting the argument structure, converse terms can be properly processed, general semantics modifiers can be contextually interpreted, lexical semantics (including evaluation) can be used in hard cases of anaphora resolution. The semantic language illustrated in this paper is used in the semantic analyzer currently under development within the multifunctional ETAP linguistic processor.

References

1. *Akshay Java* et al. 2006 — Akshay Java, Tim Finin and Sergei Nirenburg. Text understanding agents and the Semantic Web. Proceedings of the 39th Hawaii International Conference on System Sciences, Vol.3, pp. 62b, Kauai HI, 2006.
2. *Akshay Java* et al. 2007 — Akshay Java, Sergei Nirenburg, Marjorie McShane, Timothy Finin, Jesse English, Anupam Joshi. (2007) Using a Natural Language Understanding System to Generate Semantic Web Content. International Journal on Semantic Web and Information Systems, 3(4), pp. 50–74..
3. *Apresjan, Cinman* 2002 — Apresjan, Ju. D., Cinman, L. L. (2002) The Formal Model for Sentence Paraphrasing in NLP systems [Formal'naja model' perifrazirovanija predlozenij dlja sistem pererabotki tekstov na estestvennyx jazykax]. Russian Language in The Scientific Context [Russkij jazyk v naucnom osvescenii]. No. 4, pp. 102–146.
4. *Apresjan* et al. 2003 — Apresjan, Jury, I. Boguslavsky, L. Iomdin, A. Lazursky, V. Sannikov, V. Sizov, L. Tsinman. ETAP-3 Linguistic Processor: a Full-Fledged NLP Implementation of the MTT. MTT 2003, First International Conference on Meaning — Text Theory (June 16–18, 2003). Paris: Ecole Normale Superieure, pp. 279–288.
5. *Apresjan* et al. 2009 — Apresjan Ju., I.Boguslavsky, L.Iomdin, L.Cinman, S.Timoshenko. Semantic Paraphrasing for Information Retrieval and Extraction. Flexible Query Answering Systems. 8th International Conference, FQAS 2009, Roskilde, Denmark, October 26–28, 2009. Lecture Notes in Artificial Intelligence, Vol. 5822, pp. 512–523.
6. *Boguslavsky* 2009 — Boguslavsky, Igor. Enlarging the Diversity of Valency Instantiation Patterns and Its Implications. Lecture Notes in Artificial Intelligence. Logic, Language, and Computation: 7th International Tbilisi Symposium on Logic, Language, and Computation, TbiLLC 2007, Tbilisi, Georgia, October 1–5, 2007. Bosch, P; Gabelaia, D; Lang, J (Eds.) Springer-Verlag Berlin, Heidelberg, 2009, p. 206–220.
7. *Bouayad-Agha* et al. 2012a — Bouayad-Agha N., Casamayor G., Mille S., Wanner L. Perspective-Oriented Generation of Football Match Summaries: Old Tasks, New Challenges. ACM Transactions on Speech and Language Processing, 2012. 9(2).
8. *Bouayad-Agha* et al. 2012b — Bouayad-Agha N., Casamayor G., Mille S., Rospocher M., Serafini L., Wanner L. From Ontology to NL Generation of Multilingual User-Oriented Environmental Reports. Proceedings of NLDB 2012: 17th

International Conference on Applications of Natural Language Processing to Information Systems. Groningen, 2012.

9. *Cardoso* et al. 2010 — Cardoso, N., Dornescu I., Hartrumpf S. and Leveling J. (2010) Revamping question answering with a semantic approach over world knowledge. CLEF Labs 2010, Multiple Language Question Answering, 2010 (MLQA10), Padua, Italy.
10. *Carlson* et al. 2012 — Carlson T., Van Lifferinge S., Holt E., Smith R., Covington M., Potter W. Application of Defeasible Domain-Specific Knowledge to the Description of Gothic Cathedrals in the ARC Project. Proceedings of the 2012 International Conference on Artificial Intelligence. Hamid R. Arabnia et al (eds.) vol. 1, p. 172–178.
11. *Clarke* et al. 2010 — Clarke, J., D. Goldwasser, M. Chang and D. Roth. (2010). Driving Semantic Parsing from the World's Response. Proceedings of the Fourteenth Conference on Computational Natural Language Learning (CoNLL-2010).
12. *Dornescu* 2009 — Iustin Dornescu. (2009). EQUAL: Encylopaedic Question Answering for Lists. Working notes for the CLEF 2009 Workshop. Corfu, Greece.
13. *Fahlman* 2011 — Scott E. Fahlman. (2011) Using Scone's multiple-context mechanism to emulate human-like reasoning. Proceedings of the AAAI Fall Symposium on Advances in Cognitive Systems, 2011. pp. 98–105.
14. *Ferrandez* et al. 2011 — Oscar Ferrandez, Christian Spurk, Milen Kouylekov, Iustin Dornescu, Sergio Ferrandez, Matteo Negri, Ruben Izquierdo, David Tomas, Constantin Ora-san, Guenter Neumann, Bernardo Magnini, Jose Luis Vicedo. (2011). The QALL-ME Framework: A specifiable-domain multilingual Question Answering architecture. Web Semantics: Science, Services and Agents on the World Wide Web. Vol. 9, Issue 2, July 2011, p. 137–145.
15. *Ge and Mooney* 2005 — Ruifang Ge, Raymond J. Mooney. (2005). A Statistical Semantic Parser that Integrates Syntax and Semantics. Proceedings of the Ninth Conference on Computational Natural Language Learning. Ann Arbor, MI, pp. 9–16, June 2005.
16. *Liang* et al. 2011 — Percy Liang, Michael Jordan, Dan Klein. (2011). Learning Dependency-Based Compositional Semantics. Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies — v. 1, p. 590–599.
17. *Mairal Uson, Perinan-Pascual* 2009 — Mairal Uson, R. y J. C. Perinan-Pascual. "The anatomy of the lexicon component within the framework of a conceptual knowledge base". Revista Española de Lingüística Aplicada 22 (2009), pp. 217–244.
18. *Mel'cuk* 1996 — Mel'cuk, I. Lexical Functions: A Tool for the Description of Lexical Relations in a Lexicon. Wanner, L. (ed.) Lexical Functions in Lexicography and Natural Language Processing,Amsterdam, Philadelphia, pp. 37–102.
19. *Moldovan* et al., 2010 — Moldovan, D., Tatu, M., Clark, Ch. (2010). Role of Semantics in Question Answering. In: Phillip C.-Y. Sheu, Heather Yu, C. V. Ramamoorthy, Arvind K. Joshi, Lotfi A. Zadeh (Eds.) Semantic Computing, pp. 373–420.
20. *Nirenburg, Raskin* 2004 — Nirenburg, S., and Raskin, V. (2004). Ontological Semantics. The MIT Press. Cambridge, Mass., London, England.

21. *Periñán-Pascual, Acas-Tunez* 2010a — Perinan-Pascual, J. C. and F. Arcas-Tunez. (2010) Ontological Commitments in FungramKB.. Procesamiento del Lenguaje Natural 44.
22. *Periñán-Pascual, Arcas-Tunez* 2010b — Perinan-Pascual, J. C. and F. Arcas-Tunez. (2010) The architecture of unGramKB. Proceedings of ELRA Conference. Malta.
23. *Periñán-Pascual, Mairal Usón* 2010 — Perinan-Pascual, J. C. and R. Mairal Uson. (2010) "La Gramatica de COREL: un lenguaje de representation conceptual". Onomazein 21. Universidad de Chile.
24. *Poon, Domingos* 2009 — Poon, H., & Domingos, P. Unsupervised semantic parsing. Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing Volume 1 EMNLP 09 (p.1).
25. *Raskin, Taylor* 2010 — Victor Raskin, Julia Taylor. Fuzzy Ontology for Natural Language. 29th International Conference of the North American Fuzzy Information Processing Society, Toronto, Canada, July 2010.
26. *Raskin* et al. 2010 — V. Raskin, C.F.Hempelmann, J. Taylor. Application-guided Ontological Engineering. International Conference on Artificial Intelligence, Las Vegas, NE, July 2010.
27. *Shi and* 2004 — Lei Shi and Rada Mihalcea. Open Text Semantic Parsing Using FrameNet and WordNet. Proceedings HLT-NAACL — Demonstrations '04 Demonstration Papers at HLT-NAACL 2004. p. 19–22
28. *Shiyan Ou* et al 2008 — Shiyan Ou, Victor Pekar, Constantin Orasan, Christian Spurk, Matteo Negri. Development and Alignment of a Domain-Specific Ontology for Question Answering. Proceedings of LREC 2008. 26 May — 1 June 2008, Morocco, Marrakech, pp. 2221–2228.
29. *Titov, Klementiev* 2011 — I. Titov, A. Klementiev. A Bayesian Model for Unsupervised Semantic Parsing. Learning Dependency-Based Compositional Semantics. Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies — v. 1,USA, Oregon,.Portland. pp. 1445–1455.