Representing Computational Dictionaries in AI-Oriented Knowledge Representation Formalisms

Günther Görz, Erlangen-Nürnberg

30. März 2007

1 AI-Oriented Knowledge Representation Formalisms

The field of "Artificial Intelligence" experienced in the 1970s a transition form the so called power-based strategy for achieving machine intelligence to a knowledge-based approach. Whereas the power-based strategy focussed on general powerful algorithms, the shift to the knowledge-based paradigm was based on the conviction that machines would only be able to exhibit "intelligent behaviour", if they could access huge amounts of formally represented knowledge about particular application areas as well as of the common sense level in general — which turned out to be the far more difficult task. Researchers envisoned a broad range of applications for knowledge-based systems which, of course, should be able to deal with natural language, spoken or written, to communicate with their users or to perform content analysis for information retrieval and mechanical knowledge acquisition. So, from the viewpoint of knowledge-based systems, the human language faculty can be covered sufficiently by algorithmic methods operating on explicitly represented structures of linguistic knowledge. It is important to note that with this approach a considerable reduction of human language understanding to knowledge processing, i.e. automatic reasoning, is taken into account.

The type of knowledge to be represented for knowledge processing is explicit knowledge in propositional form, i.e. knowledge about facts and general laws or rules, resp., not other varieties of knowledge such as knowing people, places and things or how to perform tasks. Factual knowledge has been the central focus of epistemology, which determines our understanding of knowledge representation (KR) as a formal reconstruction of knowledge and its implementation. The latter step enables formally represented knowledge to become dynamic; not only the conditions that are necessary and jointly sufficient for factual knowledge have to be considered, but also the roles it plays in reasoning processes. For this purpose, various programming languages have been developed, most of them in a declarative, logic-oriented style.

For the following, we assume the framework of logic in its most general sense, just as Allen Newell expressed in 1982 that "there is no non-logical knowledge representation". According to a long-standing tradition, logic has been considered as the basis of any theory of argumentation. Traditionally, logic deals with the doctrines of concept formation, judgment (expressing assertions), and reasoning. As a reason for its widespread use, usually it is stated that standard logic has the advantage of a clear and well-defined semantics. In computer science, there are different uses of logic: (1) as a tool for the analysis, design and specification of modelling tasks and formalisms, (2) as a representation formalism or at least as the basis for such formalisms, and (3) as a programming language (Prolog). For further details on these general aspects, cf. [Thomason, 2003].

There have been various reasons for the development of dedicated languages for KR. One of them is rooted the problems with decidability and the computational complexity of standard first-order logic. Furthermore, for KR special expressive means are convenient, usually justified — and often also restricted — by epistemological considerations, but also influenced by the development of object-oriented programming languages. On a very general level, most KR languages offer constructs to model application domains in a object-centered fashion. The domain vocabulary is represented by a hierarchy of concepts and properties, where properties are inherited along the hierarchy, and associations are represented at the level of individuals. The decisive idea is the conception of a relational representation at the core of any KR system: The domain being modelled is abstractly represented as a set of objects and relationships between them, so concepts are represented by predicates and properties or features represented by relations, in some cases also by functions. Often, special language constructs are provided for predication and naming, and usually some mechanism for rules to state laws and constraints is available. In contrast to ordinary object-oriented programming languages, KR languages are equipped with an explicit inference mechanism which allows to draw conclusions in a logical sense. In contrast to "normal" programming languages which offer a symbolic representation level, Newell and others stipulated an explicit epistemic representation level with the "Knowledge Level Hypothesis" [Newell, 1982]: The symbol level provides symbolic structures like taxonomies, classes, or clauses, and operations as addition, modification, searching, whereas for the knowledge level a logical language with propositional expressions, a truth-theoretical interpretation and an entailment relation exhibiting truths implicit in others is required. According to Newell, knowledge is to be characterized entirely functionally, in terms what it does, not structurally, in terms of physical objects with particular properties and relations. This still leaves open the requirement for a physical structure for knowledge that can fill the functional role.

So, to enable a computational device to make use of the represented knowledge it must be equipped with some kind of an inference mechanism. In a logical framework, this means that the computational aspects of logic have to be addressed. Given a formal logical language with its syntax and semantics and the subsequent expressive power, first of all the reasoning problem has to be specified. Investigations of the reasoning problem are aimed at its decidability and computational complexity. Finally, a problem solving procedure, i.e. a specific implementation solving the reasoning problem, has to be provided, and the problems to be addressed at this level are its soundness and completeness and the practical aspects of complexity. Of course, an ideal computational logic should be expressive, and have decidable reasoning problems for which sound complete and efficient reasoning procedures are available. Unfortunately, as one might expect, the field of logic is not an ideal world at all. As we know, even the question for decidability cannot be answered positively for full first-order logic. Therefore, for practical purposes, specialized logic-based representation formalisms are needed. Among them, a family of sublanguages of first-order logic named "Description Logics" (DLs) has been developed, which explore the "most interesting" expressive decidable logics with classical semantics, and which can be equipped with "well-behaving" reasoning procedures (cf. section 2).

Considering the history of "symbolic" AI, description logics have been around for about twenty-five years, and have substituted most of the earlier KR languages, many of which had not been specified with such a dedicated logical rigor. Therefore, they usually lacked of a clear semantics. Among them are associative networks, sometimes also called "semantic networks" [Findler, 1979, Sowa, 1991] and "frames" [Minsky, 1977]. Associative networks can be best conceived as contiguous graphs where nodes represent concepts or classes and the edges connecting them represent (binary) relations. The biggest problem of such networks, used as a representation scheme, is on the semantic level: In general, there is no explicit formal semantics, but the meaning of nodes and links and the labels attached to them is hidden in access, construction, and modification procedures. To remedy this problem, Brachman [Brachman and Schmolze, 1985] suggested a semantic network language, KL-ONE, which provides a set of epistemologically motivated node and link types. Together with [Brachman and Levesque, 1984], which investigated a much simpler language from the viewpoint of logic and complexity theory, it can be seen as the starting point of DLs: With the concern on the formal meaning of constructs KR had become formal. The title of the last quoted paper refers to the second important family of AI KR languages: frames. Frames had been introduced by Minsky [Minsky, 1977], motivated by problems of computer vision and hypotheses from cognitive psychology. He claimed that the 'chunks' of reasoning, language, memory, and perception ought to be larger and more structured units, and their factual and procedural contents must be more intimately connected in order to explain the apparent power and speed of mental activities. In a frame representation language, the units represent prototypes with properties ("'slots") as opposed to classes or concepts in most associative networks and have attached procedures for slot filling available. Considering the static part and leaving the emphasis on prototypes aside, a frame system is built up from record-like structures, arranged in an inheritance hierarchy, and can easily be mapped to a standard logic representation (cf. [Hayes, 1980]). So, there is a striking correspondence between frames and semantic networks.

Both, frames and semantic networks, have been used in computational linguistics. A well known example is Schank's "Conceptual Dependency Theory" (cf. [Schank, 1981]), where lexical entries consisted of little semantic networks which were combined by the parser, a production rule engine, to larger networks as representations of the semantics of sentences. Whereas Schank's system with its particular psychological motivation has been discussed controversially, lacks a semantic foundation and seems rather ad hoc, Sowa's conceptual graphs [Sowa, 1984] take a similar approach, but with a logical foundation in a Peircean tradition. Basically, the building blocks of conceptual graphs are lexical entries with thematic roles. A late descendant of the semantic network tradition is Helbig's MULTINET system [Helbig, 2006], which offers a rich repertoire of modelling constructs. Furthermore, in computational linguistics a new family of grammar formalisms increased in popularity: constraint-based or unification grammars. Their fundamental representation scheme are (typed) feature structures, i.e. sets of attributes (features) represented as name-value pairs, where values can be either symbols or feature structures themselves. New feature structures are built up by means of the unification operation, which combines the attribute-value pairs of two feature structures, if they are compatible, i.e., if for any feature the corresponding values are not contradictory. Otherwise, the unification operation fails: e.g., a feature structure for a noun phrase cannot be built, if the determiner is of masculine and the noun of feminine gender. Shieber [Shieber, 1992] developed a simple constraint-based grammar formalism, PATR-II, which can be seen as the core of most contemporary grammar formalisms as LFG (cf. e.g. [Kaplan, 1995]), or HPSG [Pollard and Sag, 1994]. A grammar consists of lexical and syntactic rules, where rules in general have a context free "backbone" which is augmented by constraint equations. In the example mentioned, the noun phrase rule would contain a constraint enforcing the agreement of determiner and noun gender. Lexical rules associate a word class with a word form, and associate it morphosyntactic features as constraints. Upon lexical lookup, these features are introduced in the unification process along with the constraints of the appropriate grammar rules for the phrase in which the particular word occurs. In principle, feature logics are suitable as a uniform declarative representation formalism for all linguistic abstraction levels, where each level contributes specific constraints. So, feature structures have also been used for the representation of semantic structures, e.g. with a functional composition-based variant of Discourse Representation Theory [Fischer et al., 1996, Kohlhase et al., 1996]. The semantic component in lexicon entries consists of small discourse representation structures (DRSs) which are combined into larger DRSs according to composition constraints attached to the grammar rules by functional application, which in turn is implemented by unification. For a detailed account on the use of feature structures for computational lexicons cf. [Heid, 1997]. Both, feature logics and description logics, rely mainly on attributes as the primary notational primitives for representing knowledge, they can be jointly characterized as attributive description formalisms. In fact, the logical foundations of both turn out to be very similar [Nebel and Smolka, 1991]. As features are functions, not relations — as properties are in general —, feature structures can be represented in DLs, but except for very simple DLs like CLASSIC [Brachman and et al., 1991] there is no unification operator ("same-as") for complexity reasons.

2 How to Represent: Description Logics and Semantic Web Languages

Description Logics are structured fragments of classical first-order logic which offer expressive means according to KR requirements as described above: concepts (also classes or types) arranged in subsumption hierarchies, relations (properties, roles) and objects (individuals, instances). They can be characterized as sound and complete logics which are suitable to formalize theories and systems for expressing structured information and for accessing and reasoning with it in a principled way. For structured descriptions, a restricted set of epistemologically adequate language constructs is provided to express complex relational structures of objects. Of particular importance is the distinction between conceptual (terminological, "T-Box") knowledge and knowledge about instances (assertional, "A-Box"). For reasoning, automatic classification to determine the subsumption - i.e., universal (material) implication - lattice plays a central role.

Constructions of conceptual models begin with a small set of primitive concepts and properties. Further concepts are defined by composite expressions which express sufficient and necessary conditions. Compositional operators are negation (complement), conjunction, disjunction, and existential and also value restrictions for properties. For the syntax of DLs, a simple variable-free notation has been established, derived from frame languages and standard logic. The semantics of this language, which is known as $\mathcal{ALC}$, is defined model-theoretically, i.e. by associating a set-theoretical expression, denoting its interpretation in a domain *Delta*, with each syntactic construct, as shown in the following table for most of them.

A	$A^{\mathcal{I}} \subseteq \Delta$	primitive concept
R	$R^{\mathcal{I}} \subseteq \Delta \times \Delta$	primitive property
$\top$	Δ	universal concept (top)
$\bot$	$\emptyset$	empty concept (bottom)
$\neg C$	$\Delta \setminus C^{\mathcal{I}}$	complement
$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$	conjunction
$C \sqcup D$	$C^{\mathcal{I}} \cup D^{\mathcal{I}}$	disjunction
$\forall R.C$	$\{x \mid \bigwedge_y .R^{\mathcal{I}}(x,y) \rightarrow C^{\mathcal{I}}(y)\}$	universal quantification (value restriction)
$\exists R.C$	$\{x \mid \bigvee_y .R^{\mathcal{I}}(x,y) \wedge C^{\mathcal{I}}(y)\}$	existential quantification (existential restr.)

C and D denote concept expressions, either concept names or composite expressions, and R properties. A DL knowledge base Σ consists of a T-Box — defining a hierarchy of concepts with properties —, and an A-Box, which describes a certain situation in terms of instances: $\Sigma = \langle$TBox, ABox$\rangle$.

Concept definitions (non-recursive) in the T-Box are either partial, expressed with the $\sqsubseteq$, giving necessary conditions, or complete, specifying necessary and sufficient conditions. The latter case is expressed with an equal sign, where $C \doteq D$ iff $C \sqsubseteq D$ and $D \sqsubseteq C$. As an example for a complete definition let us define a student as a person who has a name (a string), an address and who has enrolled in a course. `Person` and `Course` are either primitive or defined concepts:

$\mathtt{Student} \doteq \mathtt{Person} \sqcap \exists \mathtt{HAS} - \mathtt{NAME.String} \sqcap \exists \mathtt{HAS} - \mathtt{ADDRESS.String} \sqcap \exists \mathtt{HAS} - \mathtt{ENROLLED.Course}$

Concepts need not be named. A partial definition of individuals who teach courses – which we could name `Teacher`, of course — could be, assuming `Undergrad` and `Professor` as given:

$\exists \mathtt{TEACHES.Course} \sqsubseteq \neg \mathtt{Undergrad} \sqcup \mathtt{Professor}$

A-Box expressions denote membership statements: $C(a), R(a,b)$, where a, b are individual constants. Examples are: $\mathtt{Student(john)}$, $\mathtt{HAS} - \mathtt{ENROLLED(john, cs415)}$, $(\mathtt{Student} \sqcup \mathtt{Professor})(\mathtt{mary})$.

As for reasoning, the fundamental inference relation is concept satisfiability, i.e. the proof that a concept expression has a model. All other kinds of inferences can be shown to be reducible to concept satisfiability: subsumption, general satisfiability, instance checking, retrieval, and realization. For DLs, tableau provers have become a de facto standard for proof procedures. For a comprehensive overview of DL, cf. [Baader et al., 2003].

On top of $\mathcal{ALC}$, a variety of more expressive DLs have been developed and investigated. A particularly important DL is called $\mathcal{SHIQ}$ to , which adds qualified number restrictions, General Inclusion Axioms (GCI), which allow for complex expressions on the l.h.s. of definitions, transitive and inverse properties, property hierarchies, and data types, e.g. real numbers. Its importance is due to the fact that it provides the basis for OWL-DL, a version of the (Semantic) "Web Ontology Language" OWL (cf. [Antoniou and van Harmelen, 2004, McGuinness and vanHarmelen, 2004]). In fact, for OWL-DL (qualified) number restrictions and nominals, i.e. classes with a singleton extension, have to be added [Horrocks and Sattler, 2005].

For the “Semantic Web”, a hierarchy of standardized representation languages based on XML has been proposed. XML has its roots in the markup language SGML with an emphasis on the separation of the representation of content (document structure) from layout. XML is meta language from which particular languages can be derived by supplying a grammar (DTD: Document Type Definition). In this fashion, XHTML – a cleaned-up variant of the well-known HTML where structure and layout are not separated — has been defined as a member of the XML language family; layout is either delegated to Cascading Style Sheets or achieved by means of the general XML transformation language XSLT. Recently, DTDs have been replaced by XMLSchema, which is more powerful and allows to define data types. With the addition of namespaces using Uniform Resource Identifiers (URIs) the basic layer of the Semantic Web language hierarchy is given.

The second layer consists of RDF (Resource Description Framework), a language to express associative triples (subject-predicate-object) which can be combined to associative networks as directed labeled graphs. RDF, conceived to express metadata, offers as modelling primitives instance-of, subclass and properties with range, domain, and cardinality restrictions. This layer is enriched by RDFS (RDFSchema), which provides a limited modelling vocabulary and allows to organize it in a typed hierarchy with facilities for the definition of classes and subclasses, and of properties and property hierarchies, but there is no commitment to an inference mechanism. The logic layer, built on top of that, is exactly the place where OWL-DL is located as an extension to RDFS(FA), a sub-language of RDFS with a first-order style semantics. In fact, two further versions of OWL have been defined: OWL-Lite, which is a simplified sublanguage of OWL-DL, and OWL-Full, which includes full RDF, but at the cost of undecidability. OWL-DL as a standardized language guarantees a stable future for a very expressive description logic. Hence, as an AI KR language for computational dictionaries, OWL-DL should be the first choice, because it offers the best compromise between expressiveness, decidability, complexity and efficiency of processing.

Future layers of the language tower will deal with extensions like rules and defaults which go definitely beyond DL, sacrificing at least completeness or even decidability. Actually, the introduction of rules is being considered. Rules have been used in KR for various purposes ever since; in general a rule specifies conditions which, evaluated in a given context, produce a result. Of its many uses, for (logic) programming, as triggers or integrity constraints, for transformations, and as logical conditions, only some are compatible with OWL. In particular, there is a useful intersection between DLs and logic programming. In this realm, the Semantic Web Rule Language (SWRL) [Horrocks and et al., 2004] has been proposed, which combines OWL with Horn (Prolog-like) rules to enable the expression of logical conditions and integrity constraints. Furthermore, with a combination of DL with slightly restricted function-free Horn rules a powerful query facility is provided; a feature which is of particular interest for computational dictionaries.

3 What shall be represented?

3.1 Lexicons and Formal Ontologies

The term ontology, which in philosophy denotes the doctrine of“what there is”, is used in KR in a much more restricted way. Formal ontologies define, which and in which way objects, substances, aggregates, changes, events, actions, time and place specifications, etc., can be represented and processed. The advantage of ontologies in information processing is that no external agreement is required on the meaning of the content words in use; they provide formal specifications of the meanings with a vocabulary of terms and using background knowledge or constraints of the domain of discourse. Among formal ontologies, at least two kinds can be distinguished, reference and application ontologies. Reference (or foundational) ontologies account for a generic, universal conceptual inventory, accounting for fundamental distinctions, and for foundational relations like parts and wholes (mereonymy), similarity, dependence, connection, inherence, and temporal order. Today, instead of just a single“upper level”, often a small set of foundational ontologies is provided, and current work aims even at a foundational ontologies library. Application ontologies provide conceptual models of particular application domains; usually they are connected to refe-

rence ontologies by linking their more specialized concepts with generic concepts of the latter. For the formal representation of ontologies, description logics provide the basic modelling constructs and have proven to be very well suited as ontology languages (*O*WL!).
Formal ontologies resemble a lot to hierarchical dictionaries like WordNet [Fellbaum, 1998], thesauri, and terminological systems, but usually they convey more information. Computational lexicons adhere to the actual use of language and therefore primarily capture the use of words, whereas ontologies are constructed according to epistemological principles and scientific theory construction. So one might argue that there is a difference in perspective on the construction of dictionaries vs. ontologies, but there is no clear separation in content, although there are pleas for integrating "everything" in a computational lexicon [Evens et al., 1992]. Nevertheless, in general empirical and encyclopedic knowledge is not contained in dictionaries, but is an essential component of formal ontologies — and is indispensable to understand the descriptive texts contained in dictionaries. . . In fact, the relation between dictionaries and ontologies is rather intricate: Looking at the "knowledge elements" and their properties in both, there are some important differences — for a detailed account cf. [Hirst, 2003]. Although the debate on the definition of word senses cannot be taken up here, it is worthwile to note that for their representation nearly everything brought up in KR can be found: concepts, prototypes, intensions, extensions, reduction to primitives, etc. There is also a formal coincidence wrt. the most important lexical relationships: synonymy as abstraction — an equivalence relation —, inclusion relations as hyponymy and hyperonymy, which are transitive and form (semantic) hierarchies as well as part-whole relationships meronymy and holonymy. Furthermore, opposites need to be represented, antonyms and complements, as well as associative relations (typicality, instrumentality, cause, and location). But there are also specific differences originating from linguistic categorizations that are not ontological, and which are not desirable in formal ontologies: overlapping word senses and near-synonymy, or as gaps in the lexicon relative to other languages which may originate from a different granularity of distinctions.
The gain for language processing from the kinds of general knowledge represented in ontologies is beyond doubt, if we think of, e.g., semantic interpretation, and anaphora or metonymy resolution. For the integration of linguistic and "ontological" knowledge, different approaches have been suggested: (1) In what one might call "lexicon first", the lexical hierarchy is interpreted ontologically, and therefore augmented with at least a minimal ontology of top concepts. This approach had been chosen by the past EurowordNet project to facilitate multilingual interoperability. An actual example is a revision of WordNet based on the DOLCE reference ontology and the methodology which had been developed along with it [Gangemi et al., 2002b, Gangemi et al., 2002a, Gangemi et al., 2003, Gangemi and P., 2003]. (2) The "ontology first" approach takes an ontology as the primary structure and attaches lexical entries to it. An actual example is LingInfo [Buitelaar and et al., 2006], where semantic and linguistic information are strictly distinguished. The latter comprises only lemmata with morphosyntactic features, whereas semantics, including the notions of "broader" and "narrower", is delegated to the ontology, which provides a semantic definition of the conceptual subsumption hierarchy; cf. also [Pazienza and Stellato, 2006]. Exactly this feature lacks in proposals like LMF, the Lexical Markup Framework [Francopoulo and et al., 2006], and SKOS (W3C Simple Knowledge Organization System), a format for the formal representation of thesauri, which mix lexical and semantic information. (3) A third approach leaves both separate, and links lexicon entries, e.g. WordNet synsets (synoym sets) with concepts of the ontology. Examples are the MikroKosmos system [Nirenburg et al., 1996, Onyshkevych and Nirenburg, 1992] or CONALD [Görz and Ludwig, 2005]. Advantages of such an architecture are configurability, portability, and scalability. For a comprehensive overwiew of ontologies and lexical resources, cf. [Huang et al., ress].

3.2 Lexicon Implementation with Description Logics

Taking for granted what linguistic information has to be represented in a computational lexicon — for a classical account cf. [Evens, 1988] — and assuming that access to related ontological (world and application) knowledge is indispensable for any modern language processing system, some implementation issues in the DL framework and beyond have to be considered. In any case, it comprises the representation of a hierarchy of lexical concepts, e.g. synsets. Mostly, a version of WordNet is included, either directly implemented in RDF/OWL [vanAssem et al., 2006a, vanAssem et al., 2006b], or ontologically augmented [Gangemi et al., 2002b, Gangemi et al., 2002a, Gangemi et al., 2003, Gangemi and P., 2003], or otherwise related to a formal ontology. Recent work aims at an improvement of the represented lexical semantics with an integration of WordNet and Fillmore's FrameNet [Oltramari, 2006]. An exemplary implmentation of a multilingual collocation dictionary in OWL-DL is presented in [Spohr and Heid, 2006].

Using OWL-DL as representation language has the advantage that powerful inference engines such as Racer [Haarslev and Möller, 2001] are available which perform automatic classification and consistency checking and support efficient retrieval. For input and editing as well as calling the reasoner directly, customizable tools such as Protégé [Noy et al., 2001, Knublauch et al., 2004] are disposable; in the case of particular requirements the implementation of a specialized editor is also an option ([Thabet et al., 2003]). Because OWL is an XML language, for data mangement and storage a variety of excellent software tools can be used without further effort; a widely used (open source) native XML database system is eXist (http://exist.sourceforge.net/).

3.3 Beyond Description Logics

Due to its inherent "open world" assumption, DLs allow for reasoning with partial information. This means that the underlying assumption in logic programming (Prolog), that information which is not stated positively as facts and rules in the database, is regarded as false, does not hold in DLs. But DLs do not allow to reason with defaults; because they are sublanguages of first-order logic, only monotonic inferences are allowed. So, dealing with a value to a property which shall only hold as long as there is no indication of the contrary, requires some extra machinery — at the cost of decidability. Without doubt, such a facility would be convenient in computational lexicography. The general importance non-monotonic reasoning is also reflected by the fact that it is addressed by a future layer in the Semantic Web language hierarchy.

Non-monotonic reasoning brings up an entirely new situation compared to standard logic and its semantics, where the monotonicity property holds: If a formula p follows from a set of premises Q, then p also follows from any superset of Q. As our set of beliefs grows, so does the set of conclusions that can be drawn. In contrast to common sense reasoning, conclusions are never withdrawn. As long as we insist on the consistency of knowledge bases, in the case of contradictions we have to identify the source of conflict and withdraw any belief depending on conflicting assumptions or inconsistent data. But often there are cases where the available knowledge only suffices to formulate general rules that usually apply, but also allow for exceptions, one might be forced to revise a general assumption if an individual case is encountered which instantiates an exception. Reasoning with defaults, revising beliefs, or asserting and retracting formulae in a knowledge base are typical cases of non-monotonic inferences. There are different approaches to non-monotonic reasoning for which we refer to Brewka's et al. comprehensive overview [Brewka et al., 1997]. One of them is Reiter's "default logic" — cf. also [Reiter, 2001] —, which delegates default knowledge to a separate representation layer of default rules. Basically, such rules encode that a certain fact holds as long there is no counterevidence. They allow to assume a (default) fact A, unless a condition c is met upon which another fact B is inferred. The hierarchy of Semantic Web languages envisages a layer of rules which could be used in this way, but this is still an open research issue. And whether that will meet the requirements in computational lexicography, is to be investigated.

Another issue beyond scope of this article, which focussed on a logic-based framework, are the questions of vagueness or fuzziness of lexical concepts. Current research approaches these problems with fuzzy control techniques and hybrid symbolic-stochastic representations.

Literatur

[Antoniou and van Harmelen, 2004] Antoniou, G. and van Harmelen, F. (2004). *A Semantic Web Primer.* The MIT Press, Cambridge, MA.

[Baader et al., 2003] Baader, F., Calvanese, D., McGuiness, D., Nardi, D., and Patel-Schneider, P., editors (2003). *The Description Logic Handbook — Theory, Implementations and Applications.* Cambridge University Press, Cambridge.

[Brachman and et al., 1991] Brachman, R. and et al. (1991). Living with classic : When and how to use a kl-one-like language. In *Principles of Semantic Networks*, chapter 14, pages 401–456. Morgan Kaufmann, San Mateo.

[Brachman and Schmolze, 1985] Brachman, R. and Schmolze, J. (1985). An overview of the kl-one knowledge representation system. *Cognitive Science*, 9:171–216.

[Brachman and Levesque, 1984] Brachman, R. J. and Levesque, H. J. (1984). The tractability of subsumption in frame-based description languages. In *Proceedings of the Annual Conference of the American Association for Artificial Intelligence (AAAI-84)*, pages 34–37, Palo Alto. AAAI, Morgan Kaufman.

[Brewka et al., 1997] Brewka, G., Dix, J., and Konolige, K. (1997). *Nonmonotonic Reasoning. An Overwiew*, volume 73 of *CSLI Lecture Notes.* CSLI Publications, Stanford, CA.

[Buitelaar and et al., 2006] Buitelaar, P. and et al. (2006). Linginfo : Design and applications of a model for the integration of linguistic information in ontologies. In *Proceedings of OntoLex 2006 (LREC 2006 Satellite Workshop)*, pages 28–32, Genova. ELRA.

[Evens, 1988] Evens, M. W., editor (1988). *Relational models of the lexicon. Representing knowledge in semantic networks.* Studies in Natural Language Processing. Cambridge University Press, Cambridge and etc.

[Evens et al., 1992] Evens, M. W., Dardaine, J., Huang, Y.-F., Li, S. M., Markowitz, J. A., Rinaldo, F. J., Rinaldo, M., and Strutz, R. (1992). For the lexicon that has everything. In *Lexical Semantics and Knowledge Representation : 1. SIGLEX Workshop, Berkeley, CA, USA, June 17, 1991 ; Proceedings*, volume 627 of *Lecture Notes in Computer Science : Lecture Notes in Artificial Intelligence*, pages 219–233. Springer.

[Fellbaum, 1998] Fellbaum, C., editor (1998). *WordNet : An Electronic Lexical Database.* Language, Speech, and Communication. MIT Press, Cambridge, Mass. and etc.

[Findler, 1979] Findler, N. V., editor (1979). *Associative networks : Representation and Use of Knowledge by Computers.* Academic Press, Orlando u.a.

[Fischer et al., 1996] Fischer, I., Geistert, B., and Görz, G. (1996). Incremental semantics construction and anaphora resolution using lambda-drt. In Botley, S. and Glass, J., editors, *Proceedings of DAARC-96 — Discourse Anaphora and Anaphor Resolution Colloquium*, pages 235–244, Lancaster. Poster in: Proceedings of the Fourth International Workshop on Parsing Technologies, Universitas Carolina, Prague and Karlsbad, 1995, 87–88.

[Francopoulo and et al., 2006] Francopoulo, G. and et al. (2006). Lexical markup framework (lmf). In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*, pages 233–236, Genoa, Italy. Association for Computational Linguistics.

[Gangemi et al., 2002a] Gangemi, A., Guarino, N., Masolo, C., and Oltramari, A. (2002a). Restructuring wordnet's top-level : The ontoclean approach. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC'02)*, page 10pp., Las Palmas, Spain. Association for Computational Linguistics.

[Gangemi et al., 2003] Gangemi, A., Guarino, N., Masolo, C., and Oltramari, A. (2003). Sweetening wordnet with dolce. *AI Magazine*, 24(3):13–24.

[Gangemi et al., 2002b] Gangemi, A., Guarino, N., Masolo, C., Oltramari, A., and Schneider, L. (2002b). Sweetening ontologies with dolce. In *Knowledge Engineering and Knowledge Management. Ontologies and the Semantic Web, 13th International Conference, EKAW 2002, Siguenza, Spain, October 1–4, 2002*, pages 166–181, Berlin. Springer Verlag.

[Gangemi and P., 2003] Gangemi, A. and P., N. R. V. (2003). The ontowordnet project : Extension and axiomatisation of conceptual relations in wordnet. In *Proceedings of the International Conference ODBASE 2003*, page 18pp.

[Görz and Ludwig, 2005] Görz, G. and Ludwig, B. (2005). Speech dialogue systems — a pragmatics-guided approach to rational interaction. *Künstliche Intelligenz*, (3):5–10.

[Haarslev and Möller, 2001] Haarslev, V. and Möller, R. (2001). Racer system description. In *International Joint Conference on Automated Reasoning, IJCAR'2001, June 18-23, Siena, Italy*, pages 701–705, Heidelberg and etc. Springer-Verlag.

[Hayes, 1980] Hayes, P. J. (1980). The logic of frames. In *Frame Conceptions and Text Understanding*, pages 46–61. deGruyter, Berlin.

[Heid, 1997] Heid, U. (1997). *Zur Strukturierung von einsprachigen und kontrastiven elektronischen Wörterbüchern*, volume 77 of *Lexicographica. Series maior.* Niemeyer, Tübingen.

[Helbig, 2006] Helbig, H. (2006). *Knowledge Representation and the Semantics of Natural Language.* Cognitive Technologies. Springer, Berlin and Heidelberg and New York.

[Hirst, 2003] Hirst, G. (2003). Ontology and the lexicon. In *Handbook on Ontologies in Information Systems*, pages 209–230. Springer, Berlin.

[Horrocks and et al., 2004] Horrocks, I. and et al. (2004). Swrl : A semantic web rule language combining owl and ruleml. Technical report, W3C. W3C Member Submission 21 May 2004.

[Horrocks and Sattler, 2005] Horrocks, I. and Sattler, U. (2005). A tableaux decision procedure for shoiq. In *Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence IJCAI-05, Edinburgh, UK*, pages 448–453, San Francisco. Morgan Kaufmann.

[Huang et al., ress] Huang, C.-R., Calzolari, N., Gangemi, A., Lenci, A., Oltramari, A., and Prevot, L., editors (2007, in press). *Ontologies and Lexical Resources for Natural Language Processing.* Cambridge Studies in Natural Language Processing. Cambridge University Press, Cambridge.

[Kaplan, 1995] Kaplan, R. M. (1995). The formal architecture of lexical-functional grammar. In *Formal Issues in Lexical-Functional Grammar*, volume 47 of *CSLI Lecture Notes*, pages 1–23. Center for the Study of Language and Information, Stanford, Calif.

[Knublauch et al., 2004] Knublauch, H., Musen, M. A., and Rector, A. L. (2004). Editing description logic ontologies with the protégé owl plugin. In *Proceedings of the 2004 International Workshop on Description Logics (DL 2004), Whistler, BC, Canada, June 6–8, 2004*, volume 104 of *CEUR Workshop Proceedings*, pages 479–498, Aachen. CEUR.

[Kohlhase et al., 1996] Kohlhase, M., Kuschert, S., and Pinkal, M. (1996). A type-theoretic semantics for lambda-drt. In *Proceedings of the 10th Amsterdam Colloquium*, pages 479–498, Amsterdam. ILLC.

[McGuinness and vanHarmelen, 2004] McGuinness, D. L. and vanHarmelen, F. (2004). Owl web ontology language overview. Technical report, W3C. W3C Recommendation, 10 February 2004.

[Minsky, 1977] Minsky, M. (1977). A framework for representing knowledge. In *The Psychology of Computer Vision*, pages 211–277. McGraw-Hill, New York.

[Nebel and Smolka, 1991] Nebel, B. and Smolka, G. (1991). Attributive description formalisms. . . and the rest of the world. In *Textunderstanding in LILOG: Integrating Computational Linguistics and Artificial Intelligence*, pages 439–454. Springer, Berlin etc.

[Newell, 1982] Newell, A. (1982). The knowledge level. *Artificial Intelligence Journal*, 18(1):87–127.

[Nirenburg et al., 1996] Nirenburg, S., Beale, S., Mahesh, K., Onyshkevych, B., Raskin, V., Viegas, E., Wilks, Y., and Zajac, R. (1996). Lexicons in the mikrokosmos project. In *Proceedings of the AISB'66 Workshop on Multilinguality in the Lexicon*, Brighton, UK. Society for Artificial Intelligence and Simulated Behavior.

[Noy et al., 2001] Noy, N., Sintek, M., Decker, S., Crubezy, M., Fergerson, R. W., and Musen, M. A. (2001). Creating semantic web contents with protege-2000. *IEEE Intelligent Systems*, 16(2):60–71.

[Oltramari, 2006] Oltramari, A. (2006). Lexipass methodology : a conceptual path from frames to senses and back. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*, pages 2311–2314, Genoa, Italy. Association for Computational Linguistics.

[Onyshkevych and Nirenburg, 1992] Onyshkevych, B. A. and Nirenburg, S. (1992). Lexicon, ontology, and text meaning. In *Lexical Semantics and Knowledge Representation : 1. SIGLEX Workshop, Berkeley, CA, USA, June 17, 1991 ; Proceedings*, volume 627 of *Lecture Notes in Computer Science : Lecture Notes in Artificial Intelligence*, pages 238–249. Springer.

[Pazienza and Stellato, 2006] Pazienza, M. T. and Stellato, A. (2006). Exploiting linguistic resources for building linguistically motivated ontologies in the semantic web. In *Proceedings of OntoLex 2006 (LREC 2006 Satellite Workshop)*, pages 20–27, Genova. ELRA.

[Pollard and Sag, 1994] Pollard, C. J. and Sag, I. A. (1994). *Head Driven Phrase Structure Grammar*. Studies in Contemporary Linguistics. Center for the Study of Language and Information etc., Stanford, Calif.

[Reiter, 2001] Reiter, R. (2001). *Knowledge in Action : Logical Foundations for Specifying and Implementing Dynamical Systems*. The MIT Press, Cambridge, MA.

[Schank, 1981] Schank, R. C., editor (1981). *Inside Computer Understanding : 5 Programs plus Miniatures*. The Artificial Intelligence Series. Erlbaum, Hillsdale, N.J.

[Shieber, 1992] Shieber, S. M. (1992). *Constraint-Based Grammar Formalisms : Parsing and Type Inference for Natural and Computer Languages*. A Bradford Book. MIT Press, Cambridge, Mass. u.a.

[Sowa, 1991] Sowa, J., editor (1991). *Principles of Semantic Networks*. Morgan Kaufmann, San Mateo.

[Sowa, 1984] Sowa, J. F. (1984). *Conceptual Structures : Information Processing in Mind and Machine*. The Systems Programming Series. Addison-Wesley, Reading, Mass. etc.

[Spohr and Heid, 2006] Spohr, D. and Heid, U. (2006). Modeling monolingual and bilingual collocation dictionaries in description logics. In *Proceedings of the EACL Workshop on Multi-Word-Expressions in a Multilingual Context — EACL 2006*, pages 65–72, Trento. Association for Computational Linguistics.

[Thabet et al., 2003] Thabet, I., Ludwig, B., Schweinberger, F.-P., Bücher, K., and Görz, G. (2003). Using eurowordnet within the speech operated system embassi. In Kunze, C., editor, *GermaNet Workshop: Anwendungen des deutschen Wortnetzes in Theorie und Praxis, Proceedings*, Tübingen. Gesellschaft für Linguistische Datenverarbeitung e.V. (GLDV).

[Thomason, 2003] Thomason, R. (2003). Logic and artificial intelligence. In *Stanford Encyclopedia of Philosophy*. CSLI, Stanford University, Stanford.

[vanAssem et al., 2006a] vanAssem, M., Gangemi, A., and Schreiber, G. (2006a). Conversion of wordnet to a standard rdf/owl representation. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*, pages 67–86, Genoa, Italy. Association for Computational Linguistics.

[vanAssem et al., 2006b] vanAssem, M., Gangemi, A., and Schreiber, G. (2006b). RDF/owl representation of wordnet. W3C working draft, 19 june 2006, W3C.