ELSEVIER

Artificial Intelligence 135 (2002) 145–198

Artificial Intelligence

www.elsevier.com/locate/artint

Understanding metonymies in discourse

Katja Markert [a,*], Udo Hahn [b]

[a] *Language Technology Group, Division of Informatics, University of Edinburgh, 2 Buccleuch Place, Edinburgh EH8 9LW, UK*

[b] *Computational Linguistics Lab, Freiburg University, Werthmannplatz 1, D-79085 Freiburg, Germany*

Received 30 January 2001

Abstract

We propose a new computational model for the resolution of metonymies, a particular type of figurative language. Typically, metonymies are considered as a violation of semantic constraints (e.g., those expressed by selectional restrictions) that require some repair mechanism (e.g., type coercion) for proper interpretation. We reject this view, arguing that it misses out on the interpretation of a considerable number of utterances. Instead, we treat literal and figurative language on a par, by computing both kinds of interpretation independently from each other as long as their semantic representation structures are consistent with the underlying knowledge representation structures of the domain of discourse. The following general heuristic principles apply for making reasonable selections from the emerging readings. We argue that the embedding of utterances in a coherent discourse context is as important for recognizing and interpreting metonymic utterances as intrasentential semantic constraints. Therefore, in our approach, (metonymic or literal) interpretations that establish referential cohesion are preferred over ones that do not. In addition, metonymic interpretations that conform to a metonymy schema are preferred over metonymic ones that do not, and metonymic interpretations that are in conformance with knowledge-based aptness conditions are preferred over metonymic ones that are not. We lend further credit to our model by discussing empirical data from an evaluation study which highlights the importance of the discourse embedding of metonymy interpretation for both anaphora and metonymy resolution. © 2001 Elsevier Science B.V. All rights reserved.

Keywords: Metonymy; Natural language processing; Figurative language; Discourse; Anaphora

* Corresponding author.
E-mail addresses: markert@cogsci.ed.ac.uk (K. Markert), hahn@coling.uni-freiburg.de (U. Hahn).

PII: S0004-3702(01)00150-3

1. Introduction

Metonymy is a phenomenon in natural language, which is usually defined as a figure of speech in which the speaker is "*using one entity to refer to another that is related to it*." [35, p. 35]. [1] This can be illustrated by the following examples:

(1) "**The ham sandwich** is waiting to pay."

(2) "He read **Shakespeare**."

In Example (1), it is not the ham sandwich that is waiting, but the person who ordered it. In Example (2), it is not the person named 'Shakespeare' who is being read, but the works, plays or poems written by him (see Lakoff and Johnson [35] or Fass [18] for many more similar examples). Metonymy poses a serious problem for semantic analysis both in terms of *recognition* and *interpretation*, i.e., uncovering the implicit lexico-semantic or conceptual relationships in the metonymic utterance. In Example (1) "***the person who ordered*** (*the ham sandwich*)" and in Example (2) "***the works, plays or poems written by*** (*Shakespeare*)" have to be recovered to interpret the sentences correctly.

When it comes to the computation of metonymies, researchers usually assume some a priori lexical or conceptual specification as a reflection of canonical usage patterns of lexical items—characterizing their literal meaning—relative to which a metonymic usage has to be determined. It is then a commonly held view in the field of metonymy resolution to consider semantic restrictions that predicates (especially verbs) impose on their arguments as a valid indicator to recognize metonymies [18,30,44]. In the examples above these predicate-argument restrictions require, e.g., the subject of "*wait*" to be a PERSON and the direct object of "*read*" to be a kind of DOCUMENT. [2] The prevailing computational scheme for dealing with metonymies can then be summarized in the following way: Only after *selectional restriction violations* (*SRVs*) have been encountered, metonymy resolution is triggered; in all other cases, the literal reading is preferred, by default.

We will present arguments that this SRV-based approach is neither sufficient for the recognition of metonymies (i.e., there exist metonymies which do not violate selectional restrictions), nor are SRVs always indicative of metonymic usage. Our first major claim can then be summarized as follows. We reject the assumption that metonymic language stands for a deviation from language norms and instead propose a mechanism which computes literal and metonymic interpretations *independently* from SRVs. [3]

The second major claim we make relates to the *discourse embedding* of metonymies. Almost all approaches to metonymy processing focus on *intra*sentential analysis. In our

[1] We will not discuss problems associated with the lack of precision of this definition; cf. [26].

[2] Some researchers refer to these predicate-argument restrictions as *selectional restrictions*, e.g., Fass [18], others call them *sortal constraints* (Harabagiu [29]). We shall use the term *'selectional restriction'* to characterize any semantic constraint on argument positions of a predicate related to its lexical or conceptual specification (e.g., in terms of the proper agent or patient of a verb).

[3] Of course we are not the first researchers to have noticed that figurative language is not always indicated by SRVs. But whereas some theoretical linguists, e.g., Ortony [43], have also made this point, the *computational* approaches to metonymy still focus on SRVs. A more detailed discussion of these claims is presented in Section 2.1.

data, however, we found a significant number of metonymies which can only be resolved through *inter*sentential analysis, i.e., by incorporating criteria of textual cohesion. This has immediate implications for the resolution of nominal anaphora, since it indicates a strong *interdependence* between both types of referential language phenomena, which has not been focused on so far.

It should be evident from this discussion that treating literal and metonymic interpretations on a par and incorporating readings originating from the discourse level opens up the box of Pandora, i.e., we allow a possibly large number of ambiguities to arise. We therefore introduce a few general heuristic principles by which alternative readings can be ranked on a preference scale. These principles incorporate criteria that relate to referential success, conventionality and aptness of a metonymy, as well as intrasentential semantic constraints. We will show that the integration of all these criteria is really mandatory for the adequate treatment of metonymies *and* literal readings.

One might, finally, concede that metonymies constitute a challenging semantic phenomenon *per se* but still question the necessity to deal with a possibly exotic phenomenon in 'real' natural language systems. The contrary is true. Stallard [53] reports on a 27% performance improvement when metonymy resolution is incorporated into a question-answering system about airline reservations. Kamei and Wakao [33] argue for the necessity of metonymy resolution for machine translation. We found metonymic expressions in 15% of the utterances in a German language corpus of information technology test reports [38]. Even for language engineering tasks, metonymy resolution is a sheer necessity as evidenced by a named entity recognition problem in the *British National Corpus* (*BNC*). [4] In a small-scale experiment we found, e.g., that approximately 50% of 100 randomly drawn occurrences of "*BMW*" referred metonymically to cars or motorcycles, while the other half referred literally to the company. The interdependence of anaphora and metonymy resolution is obvious in everyday language use, e.g., "*The treaty has to be signed by the American government.* ***The United States*** *announced* ...". Therefore any information extraction system wanting to extract, e.g., all the activities of the American government has to recognize the metonymy in the second sentence.

The remainder of the article is organized as follows: In Section 2 we elaborate on the claims made above by considering the relation between metonymies and SRVs (Section 2.1), their embedding into a larger discourse context (Section 2.2), and the need for a comprehensive integration of metonymy resolution criteria, including principles for the heuristic ranking of literal and metonymic readings (Section 2.3). In Section 3 we introduce the basic knowledge representation requirements of our approach. We start with the knowledge representation schema in Section 3.1, and then turn to the fundamental search procedure, the `Path Finder`, which computes literal and metonymic readings on a par (Section 3.2). In Section 3.3 several criteria are proposed to distinguish literal and metonymic readings by means of a so-called `Path Classifier`. Section 4 deals with the discourse embedding of metonymies. After a discussion of how (nominal) anaphora and metonymies interact (Section 4.1), we formalize these interactions (Section 4.2) and turn to the discourse constraints by which reasonable interpretations can be ranked, resulting in a

[4] The BNC is a 100 million word corpus of contemporary British English including a wide variety of domains and genres (http://www.info.ox.ac.uk:80/bnc).

summary of our metonymy resolution algorithm (Section 4.3). In Section 4.4 this algorithm is adapted to an incremental processing mode. Section 5 is devoted to the evaluation of our approach. We discuss occurrence data of the various linguistic phenomena (Section 5.1), as well as the resolution rates we achieve with our approach (Section 5.2). Section 6 contains a discussion of related work. Finally, in Section 7 we point out some shortcomings of our solutions and propose extensions.

2. Major claims for metonymy resolution

In the following, we will substantiate the claims sketched in the introduction and show how they differ from assumptions made in previous work on metonymies.

2.1. Metonymies and selectional restrictions

The majority of approaches to metonymy resolution use the violation of selectional restrictions as a defining property for metonymies: [5]

> "Predicates impose constraints on their arguments that are often violated. When they are violated, the arguments must be coerced into something related that satisfies the constraints. This is the process of metonymy resolution."
>
> (Hobbs et al. [30, p. 79])

> "*Type coercion*: A semantic operation that converts an argument to the type that is expected by a function, where it would otherwise result in a type error."
>
> (Pustejovsky [44, p. 425])

We fundamentally disagree with this view and argue instead that a computational approach to metonymy recognition relying merely on constraint violations has serious drawbacks regarding coverage and methodological cleanliness. [6]

Metonymies without SRVs. Metonymies can occur without necessarily causing SRVs as illustrated by Examples (3) and (4):

(3) A: "[. . .] First of all, anybody like **Johnson**, better than **Shakespeare?**"
B: "I do."
A: "You do. Why?"
B: "Just cos erm, I don't really like **Shakespeare**."
(Author-for-Document metonymy, BNC)

[5] Pustejovsky [44] also considers *syntactic* type violations but concentrates, nevertheless, on local semantic or syntactic predicate-argument restrictions.

[6] There are also arguments against the cognitive plausibility of an approach where figurative readings are considered only *after* literal readings have been rejected because they violate semantic restrictions (for overviews, cf. [21,22]). Such considerations that, indeed, played a role for the design of the metonymy resolution procedure we propose will not be discussed in this article (cf. [25]).

(4) “Je vis alors tomber *le papillon*. [. . .] Je me suis dit: Semblable est mon destin. [. . .] Comme **ce ver**, j’ai rampé dans la fange.”

(“I saw this butterfly fall. I said to myself: Similar is my destiny. [. . .] Like **this caterpillar** I have crawled around in the mud.”)

(Caterpillar-for-Butterfly metonymy, from a text by Zola, quoted by Bonhomme [3, p. 185])

Therefore, SRVs cannot be taken as a defining characteristic for metonymies. One layer of indeterminacy comes in as verbs such as “*like*”, “*put*”, “*convince*” do not impose particularly severe restrictions on the types of their arguments (cf. [49] for a measure of the strength of selectional restrictions). Such *soft* selectional restrictions often lack reliable clues on literal and metonymic readings of the arguments of the verb (see Example (3)). Therefore, all approaches relying on SRVs as a triggering condition for metonymy recognition miss out the interpretation of some metonymies at least, thereby lowering recall. An empirical evaluation of this claim will be provided in Section 5.1.

In those cases where previous work acknowledges that metonymies without SRVs exist (e.g., [18]), the violation criterion is nevertheless used as a simple and ‘neat’ trigger for recognizing at least the majority of metonymies. The following discussion will challenge this claim, too.

Direct versus indirect semantic constraints. It is often not clearly stated which sorts of semantic constraints are covered by a particular metonymy resolution algorithm. Some authors use mainly restrictions verbs impose on their arguments and state this clearly—e.g., Fass [17] who explores these as well as adjective-noun constructions only, or Amghar et al. [1]. Of course, other kinds of predicate-argument restrictions may also be relevant, e.g., restrictions of nouns on their modifying prepositional phrases. Approaches covering these restrictions (e.g., [30]) include a greater variety of metonymies. But there are still more complex semantic restrictions than local predicate-argument restrictions, which we will call *indirect semantic constraints*.

In Example (5), [7] the subject “*Quantum*”—the name of a company—does not immediately violate a verb restriction as companies can achieve goals. [8] But there is an *indirect* violation between the *two* arguments of the verb as companies cannot achieve certain access times.

(5) “In der Leistung konnte die LPS 105 ebenfalls weitestgehend überzeugen. Laut Core-Test2.8. erreicht **die Quantum** eine mittlere Zugriffszeit von 16.5 ms, [. . .].”

(“The performance of the LPS 105 [known to be a hard disk developed by Quantum, K.M. & U.H.] was mostly convincing. According to Core-Test2.8, **the Quantum** achieves an average access time of 16.5 ms, [. . .]”)

[7] German examples are taken from one of our two application domains, *viz.* test reports from the information technology domain. Their translation into English will maintain the original German word order only when it is necessary for the understanding of the metonymy.

[8] Note that the use of the definite determiner with company names in German does not indicate a metonymic meaning, but is used with the literal meaning as well.

(From H. Lengner, “Festplatten von 80 MB bis 120 MB: Quantum ProDrive LPS 105AT.”, *PC Praxis*, Vol. 1, 1992, p. 54)

Recognizing these and even more complex violations (e.g., ones mediated by subordinate clauses) requires a quite sophisticated metonymy resolution and semantic interpretation procedure. Also, it is usually hard to tell whether indirect semantic constraints are violated—an issue normally not addressed in previous work. Often not only subject and (in)direct objects, but *all* arguments and possibly some adjuncts of the verb have to be taken into account—in Example (6), the subject and the object of the verb seem to point to a violation (and thus to a metonymic reading of “*the Quantum*”), but the sentence-final prepositional phrase “*with its product*” readjusts a literal reading interpretation.

(6) “According to CoreTest2.8, **the Quantum** achieves an average access time of 16.5 ms with its product.”

These problems are crucial for incremental approaches to metonymy resolution as the relevant semantic information might not be available when metonymy recognition takes place.

In the following, the notion of *semantic constraints* will include both direct semantic constraints (called predicate-argument constraints or selectional restrictions, interchangeably), and indirect semantic constraints. With the term SRV, we refer to the violation of *direct* constraints only.

SRVs pointing to non-metonymic readings. Even if SRVs occur, they may also give rise to non-metonymic readings. One example are metaphoric readings of the verb instead of metonymic ones of the noun phrase. For instance, a verb like “*wait*” restricts its subject fairly clearly to instances of the concept class PERSON (see also the senses of “*wait*” in WORDNET [19]), giving rise to explanations of the metonymic reading of “*sandwich*” in Example (1). But Example (7) shows that there are still other reasonable interpretations for non-human subjects, too.

(7) “Upstairs Henry [...] pulled his pyjamas on. His purple horrors, were neatly laid out on a chair again [...], but the tight buttonless shirt had been replaced by a new one. [...] His mother was infuriating but she cared about him, deep down. All the time he’d been going on about the buttons popping off, **this shirt** had been up here, waiting for him.” (BNC)

In this example, the preceding context indicates that the shirt is indeed a shirt, thus precluding a metonymic reading of ‘a person wearing a shirt’. Instead “*wait*” is used metaphorically. We do not handle metaphors in our approach, although we will discuss it as a possible extension of our work in Section 7.

Moreover, SRVs interact heavily with the way how restrictions are encoded. Therefore, violation judgments are dependent on the specification depth of the underlying knowledge base or lexicon—the flatter it is, the less likely are violations to be detected because the relevant fine-grained distinctions cannot be expressed. So, the *granularity* of knowledge specifications becomes an issue for metonymy resolution, too (see also [4]). Although this is in some sense true for any specification of semantic or conceptual knowledge and cannot be fully avoided, metonymy resolution algorithms should provide safeguards to deal with systematic granularity problems, but rarely do so.

2.2. Metonymies and the discourse context

Previous approaches to metonymy resolution usually disregard the broader discourse context. This neglect is obviously rooted in the assumption that if metonymies are so easily recognized via selectional restrictions *within* the sentence, then the context spanned *between* several sentences might not be relevant for recognition. But what is more, discourse context even does not play a role during the interpretation phase—one notable exception being [37]. The interpretation phase normally capitalizes on local selectional restrictions, together with some knowledge about the metonymic expression (and perhaps knowledge about conventions for metonymic usage). We argue instead that the discourse embedding of metonymic expressions is often crucial for *both* recognition and interpretation.

Firstly, as Examples (4) and (5) reveal, the discourse context often allows to recognize metonymies even if there are no or only indirect intrasentential clues. In both examples the preceding sentence(s) have as a current local focus a butterfly (not a caterpillar) or a hard disk (not a company), with which the metonymic noun phrases under scrutiny corefer. Interpreting the noun phrases metonymically regardless of SRVs allows to establish referential cohesion in a straightforward way, which is not true for the literal reading. In Example (4), metonymy recognition is *fully* dependent on the previous context, since no clue for a metonymic reading is available within the sentence.

Secondly, Examples (4) and (5) suggest that anaphora resolution is *dependent* on prior metonymy resolution for making the proper referent accessible. In Example (5), "*Quantum*", the name of a company, cannot be resolved as being coreferent to "*LPS105*", a hard disk, without prior metonymy resolution. If it is not available, the anaphor resolution will fail and the discourse representation will appear incoherent, though the actual discourse is not.

Thirdly, when choosing between different metonymic readings those which establish referential cohesion should often be preferred over ones that do not. In Examples (8) and (9) the selectional restriction of "*buy*" is strong enough to signal a metonymic reading for "*the Shakespeare*", but not strong enough to indicate what it is metonymic for. A solution to this problem depends on the context—in Example (8) the (conventional) reading `Shakespeare-for-Play` would be preferred, whereas in Example (9) the reading `Shakespeare-for-Bust` is most likely. The latter example indicates how discourse-rooted referential preferences can override the preference for conventional metonymic readings.

(8) "John could not decide whether to buy the play by Shakespeare or the play by Goethe. In the end, he bought **the Shakespeare**."

(9) "John could not decide whether to buy the bust of Shakespeare or the bust of Goethe. In the end, he bought **the Shakespeare**."

Summarizing our arguments, anaphora and metonymy resolution are often *co-dependent*. Hence, we argue for an integrated model that accounts for the systematic interdependencies between them. Such a model is especially important for (although by no means restricted to) incremental approaches to natural language analysis, as semantic violations may not be recognized at the time when metonymy or anaphora resolution is or should be carried out.

As a caveat, we concede that anaphora resolution is not always sufficient to overcome the shortcomings of the SRV approach. This holds especially for non-definite metonymic noun phrases, whose resolution may require rich world knowledge—in Example (3), e.g., the knowledge that the speaker lives in the 20th century and is thus unlikely to talk about liking *the person* ‘Shakespeare’.

2.3. Feature integration for metonymy resolution

In the previous two subsections, we focused on two particularly relevant criteria for metonymy processing, *viz.* the violation of or conformance with selectional restrictions at the sentence level and the discourse embedding of metonymy interpretation via the interactions of nominal anaphora and metonymies. There are, however, additional criteria one has to consider.

Firstly, any relation between two objects can give rise to a metonymy (cf. Nunberg [41] and Example (1)). One basic condition, though, is that such a relation is possible at all—in other words, that the metonymic reading is consistent with the current encoding of the underlying *world knowledge*. A metonymy of the type “*I read **X***” (meaning “*I read the works of X*”) is precluded if *X* cannot be an author of a document (e.g., if *X* is a non-human animal).

A second well-known property of metonymies is that although, in principle, every relation can give rise to metonymies, some relations do so especially often and systematically. Such relations, *R*, give rise to a *metonymy schema* of the form `X-for-Y`, where *R* holds between *Y* and *X*. So the *written-by* relation (Example (2)) can be used for any author, giving rise to an `Author-for-Document` schema. Typical schemata include `Producer-for-Product` (“*I parked **the BMW** outside*”), `Part-for-Whole` (“*There are a lot of **good heads** in this university*”) or `Place-for-Institution` (“***The White House** announced* ...”); lists of schemata can be found, e.g., in Stern [54] and Lakoff and Johnson [35]. Computational approaches may use schemata to constrain and disambiguate interpretations by preferring schematic and therefore more conventional interpretations over unconventional ones [1,30].

A third observation relates to the *aptness* of metonymic interpretations, i.e., the *typicality* of the relation between the two objects in question. Regarding, e.g., `Part-for-Whole` metonymies, we claim that a more significant (or visible) part of an object is more likely to stand for the object than a non-significant (invisible) one. Thus, the screen of a TV is more likely to stand for the TV metonymically than one of its push-bottons (insignificant) or transistors (invisible from the outside). This issue has been mostly neglected. We have so far regarded the two properties of typicality and visibility as good indicators of aptness. One reviewer made the suggestion that the functionality of parts might also be an indicator of aptness (the screen of a TV being strongly related to the TV’s function). This would have the advantage that aptness could be explained in terms of the qualia structures of components. We will briefly touch upon the issue of aptness in Section 4.3, leading to a preference of apt metonymies over non-apt ones (for a detailed discussion, cf. Hahn and Markert [25]).

Some metonymic utterances are even marked at the level of *morphosyntax*, thus yielding a fourth set of constraints. `Producer-for-Product` metonymies, e.g., often require a definite article in the singular ("*I parked* ***the BMW*** *outside*"), whereas literal readings do not ("*BMW bought Rover*").[9] These observations can be exploited for metonymy recognition, but like SRVs they are often non-conclusive as they can apply to literal readings as well, as is exemplified by "*Even the mighty IBM had problems*" (BNC). In addition, these interactions are language-dependent. In German, e.g., the definite article used with literal PRODUCER readings is much more common than in English.

Finally, *language idiosyncracies* also have an effect on the preference and possibility of interpretations. Schemata, e.g., have language-specific gaps. The `Animal-for-Meat` schema for the English language, e.g., is normally not applicable to "*pig*" as the word "*pork*" exists [7].

None of the features discussed so far explains metonymic usage alone. Even a strength-based ranking of criteria has to be evaluated carefully, since they can override each other. So, e.g., extremely unconventional metonymies can be made plausible in a suitable context, or language-specific idiosyncracies might override all other, usually stronger constraints in some cases. We claim that an adequate model of metonymy resolution should take all of these features into account. We present here an integrated model, using five of them, namely, world knowledge, intrasentential semantic constraints, discourse embedding, schematization and aptness constraints, with emphasis on the first three of them. We disregard morphosyntactic evidence and language-specific lexical idiosyncracies. This reflects our intention to propose a largely *language-independent* model of metonymy processing. We are, however, aware of the fact that this limited model needs to be supplemented by language-dependent criteria. In summary, the following principles characterize our model:

- *Principle* 1. All literal and metonymic interpretations must be consistent with the available knowledge representation structures as a formal model of a given domain (so-called world knowledge). In the case of metonymic interpretations, this requires the reconstruction of a conceptual relation between the conceptual correlate of the surface expression and the intended referent. This principle is formalized in Section 3.2.
- *Principle* 2. All interpretations must be consistent with the intrasentential context, especially with applicable selectional restrictions. This does *not* mean, however, that a literal interpretation consistent with the intrasentential context is always preferred over a metonymic one consistent with the intrasentential context (see Section 3.2).
- *Principle* 3. Interpretations that establish referential cohesion are preferred over ones that do not (see Section 4.3).

[9] Many more of such phenomena have been discussed, e.g., by Copestake and Briscoe [12] and Pustejovsky [44].

- *Principle* 4. Metonymic interpretations that conform to a schema are preferred over metonymic ones that do not (see Section 3.3 for the definition of appropriate schemata).
- *Principle* 5. Metonymic interpretations that fulfil aptness conditions are preferred over metonymic ones that do not (see Section 4.3).

For now, we distinguish between principles that hold *absolutely* (Principles 1 and 2), i.e., principles that *must* be fulfilled by every interpretation, and principles that hold only *relatively*, i.e., that only express a preference (Principles 3 to 5). Furthermore, there are principles that hold for literal and metonymic readings (Principles 1, 2 and 3) and ones that are only applicable to metonymic readings, by definition (Principles 4 and 5). Obviously, these criteria are not mutually exclusive. Some of the problems arising from non-exclusiveness can be eliminated by a model of preference rankings (cf. Table 12 in Section 4.3). This proposal will then be evaluated by considering empirical evidence in Section 5.

3. Conceptual interpretation

In order to interpret Example (1) one has to know that sandwiches are eaten by humans who can be agents of waiting; the understanding of Example (2) requires to know that Shakespeare was a writer and that therefore a relation between him and books (a common object of reading) is likely; for Example (5) one needs to know that Quantum is a company producing hard disks. Since metonymies are so deeply rooted in the conceptual knowledge of some domain, metonymy resolution is here considered as part of the conceptual interpretation of an utterance. The underlying knowledge about these kinds of relations, instances and concepts is kept in a knowledge base (see Section 3.1). Conceptual interpretations, i.e., the identification of relations between concepts and instances, are computed by a search algorithm—the `Path Finder`—operating on this knowledge base (cf. Section 3.2). The distinction whether these interpretations can be considered as literal or as metonymic is carried out by a path classification algorithm, the `Path Classifier` (cf. Section 3.3).

3.1. Representation of conceptual knowledge

We use a KL-ONE-type terminological knowledge representation language (cf. Woods and Schmolze [60] for a survey) to represent conceptual knowledge of a domain. Since our application takes texts from the information technology domain as input, we will focus on concepts and relations from this domain in our examples.

The concept hierarchy of our knowledge base consists of a set of concept names $\mathcal{F} = \{$COMPUTER-SYSTEM, PRINTER, HARD-DISK-DRIVE, $\ldots\}$ and a subclass relation $\sqsubseteq_{\mathcal{F}} = \{$(LASER-PRINTER, PRINTER), (PRINTER, PRINTER), (NOTEBOOK, COMPUTER-SYSTEM), (COMPUTER-SYSTEM, OBJECT), $\ldots\} \subset \mathcal{F} \times \mathcal{F}$. This relation is both transitive and reflexive. If $C \sqsubseteq_{\mathcal{F}} D$ holds, we call C a *subconcept* of D and D a *superconcept* of C. The intersection of two concepts C and D is written as $C \sqcap D$. The set of instance names is denoted by $\mathcal{I}$, and every instance i is an element of at least one concept C (written as $i \in C$). For every instance $i \in \mathcal{I}$ the (direct conceptual) class of i is defined by $\bigcap_{C \in \mathcal{F}, i \in C} C$. We express this by $class(i) = C$ or by i *inst-of* C and call C also the *smallest* concept containing i.

The set of relation names $\mathcal{R} = \{$*Has-Physical-Part*, *Has-Laser*, *Clock-Frequency-of*, *Produced-by*, $\ldots\}$ contains the labels of all conceptual roles. These are also organized into a hierarchy by the relation $\sqsubseteq_{\mathcal{R}} = \{($*Has-Laser*, *Has-Physical-Part*$)$, $($*Clock-Frequency-of*, *Property-of*$) \ldots\} \subset \mathcal{R} \times \mathcal{R}$. The same properties and notions of *subrelation* and *superrelation* hold as for the concept hierarchy. If the relation R holds between two instances a and b, the tuple (a, b) is said to be an instance of R (written as $(a, b) \in R$ or aRb). If (a, b) is an instance of R, then $a \in domain(R)$ holds, e.g., *domain*(*Laser-of*) = LASER; b is called a *role-filler* of R or an element of *range*(R), e.g., *range*(*Laser-of*) = LASER-PRINTER. Domain and range restrictions of relations incorporate selectional restrictions; e.g., the range of the role *Agent* of the concept WAIT is restricted to PERSON. A relation R is called the *inverse* of a relation S (written as $R = S^{-1}$) iff for every instance (a, b) of S the tuple (b, a) is an instance of R and *vice versa*. Thus, *Laser-of* and *Has-Laser* are inverses of each other. We may also restrict a relation R to a concept C, expressed as ${}_{C}|R$. This restricted relation contains as instances all $(a, b) \in R$ with $a \in C$; similarly, the range restriction $R|_C$ contains all $(a, b) \in R$ with $b \in C$.

The constructs we use are very common except for two particularities. Firstly, we use fairly *specific* relations like, e.g., *Has-Laser*, which allow the specification of particularly fine-grained constraints at the concept level. Consider, e.g., cardinality restrictions associated with the concept LASER-PRINTER. The general relation *Has-Part* cannot be restricted to one role filler, whereas the specific relation *Has-Laser* can. The use of those rather specific relations is motivated by knowledge engineering requirements emerging from the host system SYNDIKATE [28] the metonymy resolution module is embedded in, not by the metonymy resolution task. In effect, the use of very specific relations is mostly irrelevant for our claims, since metonymies usually address rather general levels of conceptual constraints.

Secondly, we use *relation hierarchies* as they are a suitable vehicle to explain inconsistencies with regard to relation transitivity. So, e.g., it is well-known that the *Has-Part* relation is not always transitive [15]. Chaffin [8] considers different subrelations of *Has-Part* instead, which are transitive, whereas the general relation is not. We have, by and large, adopted Chaffin's relation hierarchy, especially the hierarchy of *Has-Part* relations (cf. Table 1), as it is supported by empirical data from psychological experiments and has been widely acknowledged (it is, e.g., incorporated in WORDNET [19]).

Fig. 1 depicts a fragment of the relation hierarchy from our knowledge base. Relations are depicted without boxes, domain and range restrictions appear below the corresponding relation and are marked by `D:` and `R:`, respectively. Concepts are indicated by rounded boxes. The figure also depicts that *Has-Part*, as well as the two relations *Contains* and *Made-of*, are subrelations of an *Includes* relation. This is motivated by empirical data from Chaffin et al. [9]. The *Contains* relation denotes spatial containment as in BOTTLE—MILK, the *Made-of* relation makes explicit the link between an object and its material as in SHEET OF PAPER—PAPER.

The relation hierarchy will be used to constrain the search for conceptual interpretations by the `Path Finder` (although the `Path Finder` operates without this restriction

Table 1
Subrelations of *Has-Part*

Relation	Examples
Has-Physical-Part	cup–handle, laser printer–laser, tree–trunk
Has-Member	forest–tree, firm–manager, soccer team–defender
Has-Portion	cake–piece of cake, toner powder–toner particle
Has-Phase	shopping–paying, programming–testing
Has-Feature	rodeo–cowboy, halloween–masks
Has-Subregion	New York–Harlem, desert–oasis

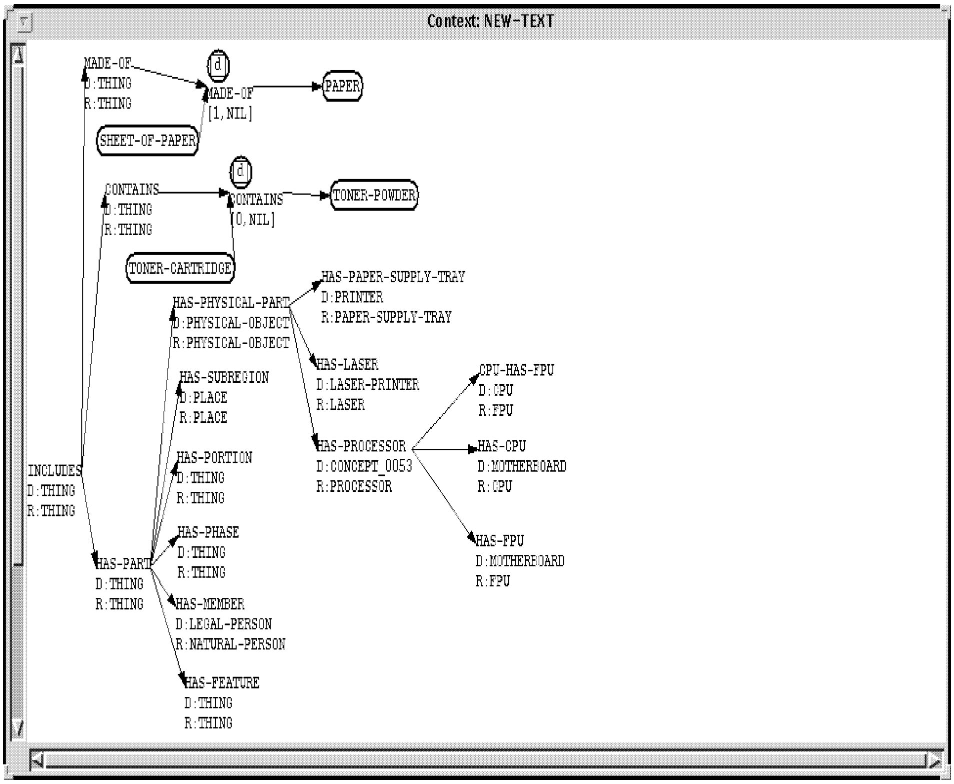

Fig. 1. *Has-Part*-relations in a concrete knowledge base (IT domain).

as well), and for handling some granularity problems. It also eases the specification of regularities underlying metonymic relations in the sense that abstract specifications (e.g., *Has-Part*) get instantiated by specific relations (e.g., *Has-Laser*) by the `Path Classifier`.

3.2. *Computing interpretations:* `Path Finder`

We associate a lexical item w with a conceptual correlate, $w.\text{CON} \in \mathcal{F}$ (e.g., "*computer*" is associated with COMPUTER-SYSTEM). If a syntactic link between two lexical items, w and w', is to be allowed, the concepts $w.\text{CON}$ and $w'.\text{CON}$ must be conceptually related. [10] So, for Example (10), the conceptual relatedness of the concepts CPU and COMPUTER-SYSTEM has to be checked.

(10) "The CPU of the computer."

In order to determine conceptual relatedness, we employ the `Path Finder`, an algorithm which performs an extensive search in the domain knowledge base looking for series of relations connecting two given concepts (e.g., CPU and COMPUTER-SYSTEM are connected by a *Part-Of* relation). The basic algorithm works with domain knowledge constraints only, but can be supplemented by additional ones, e.g., discourse constraints, if needed.

Definition 3.1 (*Conceptual relatedness*). Two concepts C and D, both from $\mathcal{F}$, are *conceptually related* iff a well-formed conceptual relation path (short: well-formed path) exists between C and D. A well-formed path has to be connective and non-cyclic. The set of well-formed paths between C and D is called $P(C, D)$.

We consider the existence of a well-formed path between the conceptual correlates of two lexical items in a syntactic relation to be equivalent to a *conceptual interpretation* of that syntactic relation.

Connectivity ensures that a relation path between two concepts does not have any "breaks":

Definition 3.2 (*Connectivity*). A series of relations $R_i \in \mathcal{R}$ $(i = 1, \ldots, n)$ and concepts $C_j \in \mathcal{F}$ $(j = 0, \ldots, n)$, $n \in \mathbb{N}$, is *connective* or establishes a *connective path* from C to D $(C, D \in \mathcal{F})$ iff:

- R_i is a (possibly inherited) conceptual role of C_{i-1}, and C_i is the smallest concept in $\mathcal{F}$ with $\mathit{range}(_{C_{i-1}}|R_i) \sqsubseteq_{\mathcal{F}} C_i$, i.e., R_i has the range C_i when restricted to C_{i-1} for all $i = 1, \ldots, n$.
- $C_0 = C \wedge (C_n \sqsubseteq_{\mathcal{F}} D \vee D \sqsubseteq_{\mathcal{F}} C_n)$, i.e., the path begins at C and its end point C_n is either a subconcept or a superconcept of D. [11]

A connective path will be denoted as $(C_0, R_1, C_1, R_2, \ldots, R_n, C_n)$ or simply as $(R_1, \ldots, R_n)$. The number n is called the *length of the path*.

[10] For details of our approach to conceptual interpretation from syntactic (dependency) structures, cf. Romacker et al. [50]. Syntactic links we consider for the purpose of metonymy interpretation include verbs and their arguments (subjects, objects and PPs), as well as the links between nouns and their pre- or postmodifiers.

[11] That the end point C_n is required to be a subconcept or a superconcept of D, and not to be a superconcept of D only, leads to the following consequence: A connective path from C to D is not necessarily a connective path from D to C. We circumvent this problem by calling two concepts conceptually related if a path from C to D or from D to C exists.

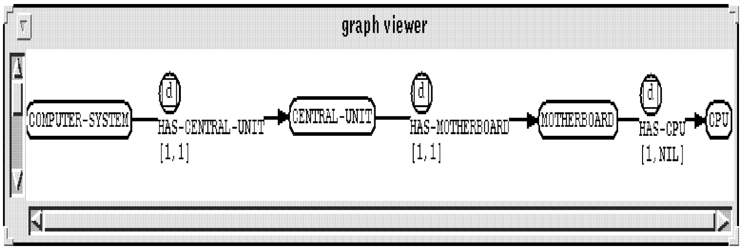

Fig. 2. Path between COMPUTER-SYSTEM and CPU.

Table 2
Non-cyclic paths

Non-Cyclic $((R_1 \ldots R_n)) : \Leftrightarrow \forall\, i, j \in \{1, \ldots, n\}, i \neq j :$
$\neg\exists S \in \mathcal{R}, S \neq S^{-1} : ((R_i \sqsubseteq_{\mathcal{R}} S) \wedge (R_j \sqsubseteq_{\mathcal{R}} S^{-1}))$

An example of a connective path between COMPUTER-SYSTEM and CPU (see Example (10)) is the path $p_1 =$ (COMPUTER-SYSTEM, *Has-Central-Unit*, CENTRAL-UNIT, *Has-Motherboard*, MOTHERBOARD, *Has-Cpu*, CPU), which is depicted in Fig. 2.

Furthermore, we require a well-formed path to be *non-cyclic*.

Definition 3.3 (*Cyclic and non-cyclic paths*). A connective path $(R_1, \ldots, R_n)$ is *cyclic* iff it contains two non-identical relations which are inverses to each other or have an inverse superrelation. Otherwise, the path is called *non-cyclic*. These requirements are formalized in Table 2.

The path $p_2 =$ (PRINTER, *Printer-of*, COMPUTER-SYSTEM, *Has-Central-Unit*, CENTRAL-UNIT) from PRINTER to CENTRAL-UNIT is connective, but cyclic and therefore not well-formed, as *Printer-of* is a subrelation of *Part-of* and *Has-Central-Unit* is a subrelation of the inverse *Has-Part*. This criterion favors a unidirectional search in the knowledge base as cyclic paths often express some *similarity* between the starting and end point of a search (e.g., for p_2 we may consider a printer and a central unit as parts of the same computer), as opposed to semantic *relatedness* (for this distinction, cf. Resnik [48]).

Note that this criterion does not depend on the existence of relation hierarchies. With no such hierarchies available, it is then reduced to a path not including relations and their direct inverses. In such a simple model using only a 'flat' *Has-Part* relation, p_2 would be considered as a path of the form (*Part-of, Has-Part*), while p_1 is composed of a series of *Has-Part* relations.

A warranted side effect of the cyclicity criterion is that we achieve more independence of path length criteria for evaluating a connective path (cf., e.g., Norvig [40] for an approach which relies partly on explicit path length restrictions). A path of length 3, like the one from COMPUTER-SYSTEM to CPU in Fig. 2, is non-cyclic as it contains only relations of the type *Has-Part*, whereas a path of length 2, like the one from PRINTER to CENTRAL-UNIT, is cyclic. The independence of path length criteria translates directly into a greater independence of granularity biases in the knowledge base. If, e.g., *Has-Part* relations are specified in great detail, we may easily end up with rather long chains of *Has-Part* relations between objects, which nevertheless should not be penalized in comparison to less developed areas of the knowledge base.

One implicit restriction in Definition 3.2 is not to allow concept specialization during the search. This becomes clear when we look at Example (5) again, repeated in a shorter form as Example (11):

(11) "The performance of the LPS105 was mostly convincing. **The Quantum** achieves an average access time of 16.5 ms."

The `Path Finder` looks for paths between the concepts ACHIEVE and QUANTUM when interpreting the subject relation between "*achieves*" and "*Quantum*". The path which mirrors the *correct* metonymic interpretation, viz. "*Quantum*" is a `Producer-for-Product` metonymy for a hard disk drive, should look something like $p :=$ (ACHIEVE, *Achieve-Agent*, HARD-DISK-DRIVE, *Produced-by*, PRODUCER). This path, however, is not connective as the role filler restriction associated with *Achieve-Agent* is not as specific as HARD-DISK-DRIVE, but restricted to OBJECT, in general. OBJECT, however, does not have a *Produced-by* role as only instances of PRODUCT have such a role. Specialization from OBJECT to the more specific concept HARD-DISK-DRIVE is in general not allowed in the search, simply to avoid combinatorial explosion of the search procedure (imagine a specialization from OBJECT to all its subconcepts and following up all emanating relations). At the same time, uncontrolled (i.e., not context-specific) specialization would be equivalent to making an unsupported hypothesis about the identity of objects (e.g., tentatively assume an OBJECT to be a HARD-DISK-DRIVE or any other subclass of OBJECT) or postulate a metonymic reading not supported by the discourse context. We do, however, allow specialization to a particular set of concepts made salient in the previous discourse context. [12] This leads to a refined definition of connectivity:

Definition 3.4 (*Extended connectivity*). A series of relations $R_i \in \mathcal{R}$ $(i = 1, \dots, n)$ and concepts $C_j \in \mathcal{F}$ $(j = 0, \dots, n)$, $n \in \mathbb{N}$, is *connective* or establishes a *connective path* from C to D $(C, D \in \mathcal{F})$ *with regard to a concept list* $(D_1, \dots, D_m)$ $(D_k \in \mathcal{F}\colon\ k = 1, \dots, m)$ iff:

- For all $i \in \{1, \dots, n\}$ holds: R_i is a (possibly inherited) conceptual role of C_{i-1}, and C_i is the smallest concept in $\mathcal{F}$ with $\mathit{range}({}_{C_{i-1}}|R_i) \sqsubseteq_{\mathcal{F}} C_i$, *or* R_i is a conceptual role of some $D_k \in (D_1, \dots, D_m)$ with $D_k \sqsubseteq_{\mathcal{F}} C_{i-1}$ and C_i is the smallest concept in $\mathcal{F}$ with $\mathit{range}({}_{D_k}|R_i) \sqsubseteq_{\mathcal{F}} C_i$.
- $C_0 = C \wedge (C_n \sqsubseteq_{\mathcal{F}} D \vee D \sqsubseteq_{\mathcal{F}} C_n)$ (see Definition 3.2).

The original Definition 3.2 of connectivity is the same as that of extended connectivity with regard to the empty concept list. We denote specialization in a path with $\sqsupseteq_{\mathcal{F}}$. The path from ACHIEVE to QUANTUM can then be written as (ACHIEVE, *Achieve-Agent*, OBJECT, $\sqsupseteq_{\mathcal{F}}$, HARD-DISK-DRIVE, *Produced-by*, PRODUCER). It is connective with

[12] These concepts are determined and ordered by a centering algorithm [56], an issue not relevant for this exposition.

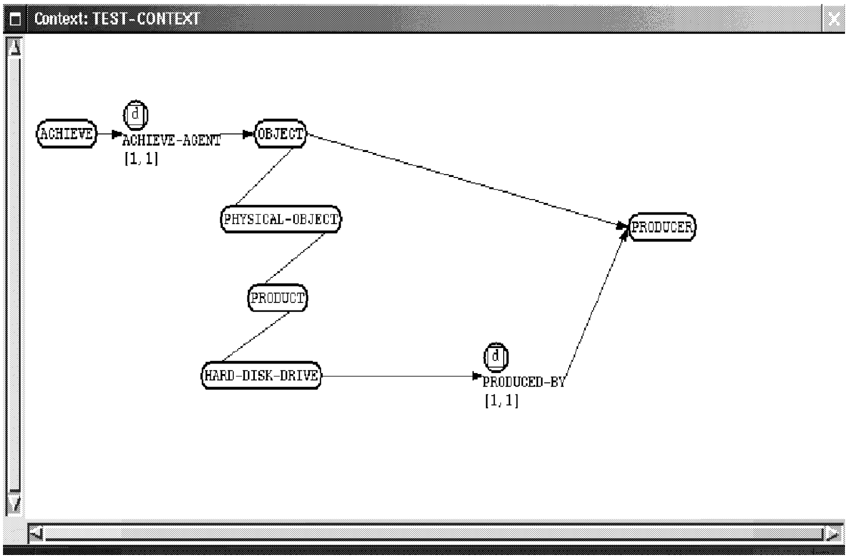

Fig. 3. Path between ACHIEVE and PRODUCER.

regard to any list $(D_1, \ldots, D_m)$ containing the concept HARD-DISK-DRIVE and it is not connective, otherwise. The corresponding path is shown in Fig. 3. [13] The short-hand notation $(R_1, \ldots, R_n)$ ignores concept specialization; so the length of the path is neutral with respect to concept specialization.

3.3. *Distinguishing literal from metonymic interpretations:* `Path Classifier`

The `Path Finder` computes all well-formed paths between two concepts w.CON and w'.CON, which correspond to conceptual interpretations of a syntactic relation between w and w'. It does not distinguish between literal and metonymic readings or choose between them *a priori*. The former task is assigned to the `Path Classifier`. Predefined path patterns are used to distinguish between a subset $\mathcal{LP}$ of all types of well-formed paths which are labeled "*literal*", another subset $\mathcal{MP}$ which is labeled "*metonymic*", and all remaining paths $\mathcal{UP}$ which are labeled "*unclassified*". A literal path between w.CON and w'.CON mirrors a literal interpretation of both w **and** w', whereas a metonymic path mirrors a metonymic interpretation of w **and/or** w'. Unclassified paths mirror literal or metonymic interpretations that cannot be classified with the current path patterns. These patterns make use of relation transitivity and metonymic schemata. They are defined as generally as possible to reduce the amount of manual specification. From those general path patterns and by virtue of the hierarchical organization of conceptual relations, specific conceptual role chains can be derived by a simple pattern matching algorithm.

[13] The picture also shows another path between ACHIEVE and PRODUCER as PRODUCER is also a subconcept of OBJECT. We will come back to this later in Example (13).

3.3.1. Literal paths

If both w and w' are meant literally, w fulfils the selectional restrictions of w' (the converse is not necessarily true, see Section 2.1). Then the conceptual correlates of w and w' can normally be linked directly via a relation R (e.g., a verb being directly linked to its subject by an *Agent* role). On the other hand, if such a direct link exists, local semantic constraints allow a literal interpretation of both w and w' (although other interpretations via longer paths may exist as well and may be preferrable by, e.g., discourse constraints). Thus, all paths of length 1 are included in $\mathcal{LP}$.

Due to different granularity levels in knowledge bases, a literal usage can also be intended if *no direct* link between two conceptual correlates can be found. This holds, in particular, when the conceptual specifications are fairly detailed. The relational structure, e.g., between the concepts CPU and COMPUTER-SYSTEM (cf. Example (10)) can be expressed in many different ways, depending on the intended specificity. Knowledge base KB_1 might specify a *Has-Part* relation between the two concepts directly (i.e., COMPUTER-SYSTEM, *Has-Cpu*/*Has-Part*, CPU); alternatively, knowledge base KB_2 might contain a series of *Has-Part* relations between them, e.g., (COMPUTER-SYSTEM, *Has-Central-Unit*, CENTRAL-UNIT, *Has-Motherboard*, MOTHERBOARD, *Has-Cpu*, CPU) (see Fig. 2). In the second case, the path falls short of the 'length 1' constraint from above but still expresses a literal interpretation, though more fine-grained.

In order to accomodate our notion of literalness to this phenomenon we incorporate empirical observations about the transitivity of relations made by Chaffin [8] and Huhns and Stephens [31]. As already described in Section 3.1, Chaffin [8] distinguishes between several subrelations of the general *Has-Part* relation. He claims that any of these *sub*relations are transitive, while the *Has-Part super*relation, in general, is not. Thus, a relation chain containing only relations of one of these subtypes is again *a relation of the same subtype*, whereas a relation chain containing several different types of *Has-Part* relations does not constitute an admitted *Has-Part* relation any more. Accordingly, we have included the path patterns (*Has-Physical-Part**), (*Has-Member**), (*Has-Portion**), (*Has-Phase**), (*Has-Feature**), (*Has-Subregion**) (cf. Table 1) and the corresponding inverses like (*Physical-Part-of**) in $\mathcal{LP}$, but exclude (*Has-Part**) from it. The abbreviation (R^*) for $R \in \mathcal{R}$ describes every well-formed path $(R_1, \ldots, R_n)$ with $R_i \sqsubseteq_{\mathcal{R}} R$ for every $i \in \{1, \ldots, n\}$. We refer to the first six of these patterns as *transitive part-whole patterns* and their inverses as *transitive whole-part patterns*. Compositionality of relation types other than *Has-Part* relations has not received that much attention in the literature, one of the rare exceptions being the study by Huhns and Stephens [31]. Drawing on their results, we have as yet included (*Contains**), (*Contained-by**) and (*Connected-to**) in $\mathcal{LP}$. The entire set of literal paths is summarized in Table 3.

As we regard all these paths as literal, all the above-mentioned paths between CPU and COMPUTER-SYSTEM are literal, too—either because they are of length 1 or instantiate a transitive part-whole pattern. So, their lexical correlates in Example (10) can receive a literal interpretation (under these paths), regardless of knowledge base granularity effects.

Table 3
Literal paths

Path patterns	Comments	Path patterns	Comments
$(R, R \in \mathcal{R})$	Paths of length 1		
(*Has-Physical-Part**)		(*Physical-Part-of**)	
(*Has-Member**)		(*Member-of**)	
(*Has-Phase**)	Transitive	(*Phase-of**)	Transitive
(*Has-Feature**)	Part-whole	(*Feature-of**)	Whole-part
(*Has-Portion**)	Patterns	(*Portion-of**)	Patterns
(*Has-Subregion**)		(*Subregion-of**)	
(*Connected-to**)	Self-inverse role		
(*Contains**)		(*Contained-by**)	

3.3.2. Metonymic paths

We now assume that w is metonymic for an instance of the concept class B. We also assume for reasons of simplicity that w' is literal and will later extend our definition to other cases.

In order to determine metonymic path patterns consider the conceptual link from the concept w'.CON to w.CON. A corresponding well-formed conceptual path $p = (C_0, R_1, C_1, \ldots, R_n, C_n)$ with $n \in \mathbb{N}$, $n > 1$, $R_i \in \mathcal{R}$ $(i = 1, \ldots, n)$, and $C_i \in \mathcal{F}$ $(i = 0, \ldots, n)$ must, first, link $w'.\text{CON} = C_0$ to B via $p_1 := (R_1, \ldots, R_j)$ for some $j \in \{1, \ldots, n-1\}$. The concept B is then linked to C_n (a sub- or superconcept of w.CON) via $p_2 := (R_{j+1}, \ldots, R_n)$.

The path is therefore divided into two halfs with the 'breakpoint' C_j $(= B)$. As w' is literal, $p_1 \in \mathcal{LP}$ must hold, whereas p_2, the other half of the path, must characterize the metonymic relation between B and w.CON. We concentrate here on metonymies referring to established schemata [18,34] such as `Whole-for-Part`, `Part-for-Whole`, `Producer-for-Product`, `Container-for-Contents`, and `Material-for-Object` metonymies. The metonymic relations corresponding to these schemata {*Part-of, Has-Part, Produced-by, Contained-by, Made-of*} constitute the set $\mathcal{MR}$. If w' instead of w is metonymic, the inverse relations $\mathcal{MR}^{-1} =$ {*Has-Part*, *Part-of*, *Produces*, *Contains*, *Material-of*} must be considered. This list of metonymic relations is, of course, incomplete and can be augmented on demand. Augmenting the list has to be done manually at the moment. As it only involves adding a further high-level relation name (e.g., *Attribute-of*) to the set of metonymic relations, this is no major drawback.

In the case of a `Producer-for-Product` metonymy (such as with Example (11)—"***The Quantum*** *achieves* ..."), $j = n - 1$ and $R_n =$ *Produced-by* must hold (the full path reads as $p :=$ (ACHIEVE, *Achieve-Agent*, $\sqsupseteq_{\mathcal{F}}$, HARD-DISK-DRIVE, *Produced-by*, PRODUCER)) (see Fig. 3). For a `Part-for-Whole` or `Whole-for-Part` metonymy, $j < n - 1$ may be possible, as all transitive part-whole and whole-part patterns (e.g., (*Has-Physical-Part**)) also express a single *Has-Part* or *Part-of* relation (see the explanations of literal paths in Table 3). For notational convenience, we now consider all transitive part-whole and whole-part patterns and all other literal paths as a single relation so that we may write (*Has-Physical-Part**) $\sqsubseteq_{\mathcal{R}}$ *Has-Part* $\in \mathcal{MR}$ or (*Has-Feature**) $\in \mathcal{MR}$. We summarize these considerations in the definition of metonymic paths below:

Definition 3.5 (*Metonymic paths*). A well-formed conceptual path $p := (C_0, R_1, \ldots, C_{n-1}, R_n, C_n)$ from $w'.\textsc{con}$ to $w.\textsc{con}$ is a (*simple and one-sided*) *metonymic path*, iff it is not literal and $j \in \{1, \ldots, n-1\}$ exists such that:

$$\big((R_1, R_2, \ldots, R_j) \in \mathcal{LP} \wedge (R_{j+1}, \ldots, R_n) \in \mathcal{MR}\big)$$
$$\vee \big((R_{j+1}, \ldots, R_n) \in \mathcal{LP} \wedge (R_1, R_2, \ldots, R_j) \in \mathcal{M}R^{-1}\big).$$

In the first case, w is metonymic for an instance in C_j under the interpretation p (abbreviation: $metonymic_p(w, C_j)$). In the second case, w' is metonymic for an instance in C_j under the interpretation p.

This definition captures only those cases where either w or w' is metonymic, but not both ("one-sided metonymic path"); it also captures only non-recursive (or "simple") metonymies. In recursive metonymies, the underlying referent of w is related via a series of different relation types to w (see Example (14)).

If w and w' are both metonymic, then p is composed of three subpaths, viz. $p_1 \in \mathcal{MR}^{-1}$, $p_2 \in \mathcal{LP}$ and $p_3 \in \mathcal{MR}$. If w is a recursive metonymy and w' literal, p consists of two subpaths, $p_1 \in \mathcal{LP}$ and a path $p_2 \notin \mathcal{MR}, \notin \mathcal{LP}$, but where p_2 can be divided into a series of paths which are all in $\mathcal{MR}$. Consider the following examples:

- In Example (12), "*computer*" is a `Whole-for-Part` metonymy for its processor, as clock frequency is a property attributed to a processor, not to a computer as such.[14]

  (12) "The clock-frequency of **the computer**."

  The syntactic relation between $w' =$ "*clock frequency*" and $w =$ "*computer*" is checked by searching for a well-formed path between the corresponding concepts CLOCK-FREQUENCY and COMPUTER-SYSTEM. The `Path Finder` locates the path (CLOCK-FREQUENCY, *Clock-Frequency-of*, CPU, *Cpu-of*, MOTHERBOARD, *Motherboard-of*, CENTRAL-UNIT, *Central-Unit-of* COMPUTER-SYSTEM), a metonymic one. With the above notation, $n = 4$, $j = 1$, $p_1 = (R_1) = (\textit{Clock-frequency-of}) \in \mathcal{LP}$, and $p_2 = (\textit{Cpu-of}, \textit{Motherboard-of}, \textit{Central-Unit-of}) \sqsubseteq_{\mathcal{R}} \textit{Part-of} \in \mathcal{MR}$ holds (see also Fig. 2 for path p_2). We deduce by this path pattern that "*computer*" denotes a `Whole-for-Part` metonymy for an instance of the concept CPU $= C_1$.
- In the previous example only a metonymic path between CLOCK-FREQUENCY and COMPUTER-SYSTEM is found. We do not need to resolve ambiguities between literal and metonymic readings. If we go back to Example (5) repeated below in a shorter form, we encounter such a (local) ambiguity between literal and metonymic readings.

  (13) "[...] the LPS 105 was mostly convincing [...]. **The Quantum** achieves an average access time of 16.5 ms."

[14] The difference between granularity phenomena and metonymies is not always clear-cut as a comparison with Example (10) reveals.

The `Path Finder` finds two well-formed paths between ACHIEVE and QUANTUM, namely (ACHIEVE, *Achieve-Agent*, OBJECT) and (ACHIEVE, *Achieve-Agent*, OBJECT, $\sqsupseteq_{\mathcal{F}}$, HARD-DISK-DRIVE, *Produced-by*, PRODUCER) (see also Fig. 3). The first has length 1 and is therefore literal. The second is a metonymic path where $n = 2$ and $j = 1$ holds. The metonymic relation in question is *Produced-by*. Hence, "*Quantum*" can be interpreted as a `Producer-for-Product` metonymy for an instance of the class HARD-DISK-DRIVE.

- Example (14) illustrates a recursive metonymy:

  (14) "The clock-frequency of **the IBM**."

  Here "*the IBM*" is used metonymically for the processor of a computer developed by IBM—the recursive metonymy is therefore a composition of a `Producer-for-Product` and a `Whole-for-Part` metonymy (cf. Example (12) for an explanation of the latter case and Example (13) for the former case). The correct path found by the `Path Finder` (among other ones) is $p =$ (CLOCK-FREQUENCY, *Clock-Frequency-of*, CPU, *Cpu-of*, MOTHERBOARD, *Motherboard-of*, CENTRAL-UNIT, *Central-Unit-of*, COMPUTER-SYSTEM, *Produced-by*, PRODUCER). The path has to be divided into three subpaths: $p_1 = (\textit{Clock-Frequency-of}) \in \mathcal{LP}$, $p_2 = (\textit{Cpu-of}, \textit{Motherboard-of}, \textit{Central-Unit-of}) \sqsubseteq_{\mathcal{R}} \textit{Part-of} \in \mathcal{MR}$, mirroring the `Whole-for-Part` metonymy, and $p_3 = (\textit{Produced-by}) \in \mathcal{MR}$, mirroring the `Producer-for-Product` metonymy.

3.3.3. Summarizing our claims

We equate conceptual interpretations with the identification of conceptual paths between two concepts. The `Path Finder` computes conceptual interpretations of syntactic relations. It operates as a search algorithm taking as input two concepts and yielding as output relation paths between them. These paths reflect series of relations in the knowledge base and must obey some well-formedness criteria. In particular, the algorithm makes sure that these interpretations are consistent with the knowledge about relations encoded in the knowledge base, therefore fulfilling *Principle* 1.

The basic finder algorithm takes only local information into account (two lexical items and their conceptual correlates). It is therefore compatible with incremental interpretation and does not need complete information about all the concepts and semantic constraints that are introduced in an utterance.

The computation of individual paths is independent of the computation of other paths. Thus, the existence of a literal interpretation does not exclude the computation of metonymic interpretations. So the algorithm is independent of SRVs (cf., e.g., the interpretation of Example (13)—"*The Quantum achieves* ..."). This is in conformance with *Principle* 2.

The `Path Classifier` makes use of well-known schemata of metonymies (e.g., `Whole-for-Part`), which are specified as general as possible. The explicit encoding of domain-specific metonymies (e.g., `Computer-for-CPU`) is therefore not necessary. Special instances of schemata are derived on the fly by instantiating specific relations linked in the relation hierarchy. The schemata can also easily be extended by including additional relations in $\mathcal{MR}$, without changing the general definition of metonymic paths in Definition 3.5.

The algorithm developed so far covers literal interpretations, including one aspect of granularity concerning transitive relations in a knowledge base. It covers and distinguishes schematized metonymies, including recursive metonymies. Non-schematized metonymies are included as unclassified paths—these interpretations are computed although detailed information about the underlying referent is not available and a distinction from granularity phenomena, not yet included in the set of literal paths, is not possible. Details about occurrences of schematized and non-schematized metonymies in our domain are given in Section 5.1.4. The `Path Finder` and `Path Classifier` cover metonymies with and without SRVs.

We cannot, however, handle metonymies that already violate syntactic constraints like so-called *logical metonymies*, in which a noun stands for the activity denoted by an absent verb [44,59].

(15) "He began **the book**."

In Example (15), "*the book*" stands for the action "*reading/writing the book*". If "*begin*" is specified in the grammar to only take a gerund or a *to*-infinitive as object, this metonymy leads to syntactic violations. Since the `Path Finder` is only triggered to compute interpretations of *admissible* syntactic relations in our approach, we are not able to handle these metonymies, since a syntactic linkage is precluded. Note that this deficiency is not implied by the `Path Finder` specification, but by its triggering condition.

We currently do not take language-specific idiosyncracies into account when computing an interpretation. Thus, the `Path Finder` will also compute interpretations of metonymies which are exceptions to a schema in a given language. So, it would compute the interpretation "I eat the meat of a pig" for the sentence "*I eat* ***pig***", although the sentence is normally not licensed in English.

4. Discourse constraints in disambiguation

Up to now, we focused on the computation of literal and metonymic readings, applying Principle 1 (world knowledge consistency) and Principle 2 (consistency with selectional restrictions). Despite the constraints both these principles introduce into the computation process the `Path Finder` usually identifies more than one possible interpretation. This raises the problem of disambiguation among alternative readings. We here rely mainly on discourse constraints, especially reference constraints. In Section 4.1 we illustrate various interaction patterns of metonymies and anaphora in depth, thus elaborating on our remarks in Section 2.2. In particular, we want to point out that, first of all, not all metonymies lead to a reference change (cf. the notion of *predicative metonymies* introduced by Stallard [53] and Nunberg [42]) and, secondly, referential cohesion is not limited to coreference as is evident from bridging phenomena [10]. In Section 4.2 we will provide a formal framework which captures these interactions. Section 4.3 leads us to a ranking for alternative readings based on the *Principle of Referential Success* and the *Principle of the Direct Trigger*. In

Section 4.4 we will extend the non-incremental algorithm for metonymy resolution from Section 4.3 to the incremental case.

4.1. Interaction patterns between metonymies and anaphora

We restrict the notion of (nominal) *anaphora* to definite noun phrases which are coreferent with their antecedent, a noun phrase (including pronouns) in a previous utterance. Apart from morphological and binding criteria, which are not an issue here [55], a conceptual generalization relation must hold between the conceptual correlates of the anaphor and its antecedent in the literal usage case.

Furthermore, we restrict the notion of *bridging phenomena* to textual ellipsis [27]. A *textual ellipsis* is a definite noun phrase whose reference is completely determined by its conceptual relation to an antecedent noun phrase in a previous utterance. In contrast to nominal anaphora, however, the textual ellipsis and its antecedent are not coreferent, but are connected by a conceptual relation (excluding generalization relations), which is not mentioned explicitly in the text, but has to be inferred from the background knowledge.

The following examples will clarify the interactions between anaphors, textual ellipses and metonymies. They are extracted from our IT corpus but their syntactic structure has been extremely simplified to make them more easily accessible and modified to reveal more clearly the various interaction patterns.

Given a definite noun phrase w, we may distinguish between the following usage patterns: [15]

I. w is anaphoric

For all three subcases let $\tilde{w}$ be the antecedent of w. For reasons of simplicity we will assume the antecedent $\tilde{w}$ to be literal. [16]

1. w is *literal*.
   This is the standard case of a nominal anaphor:

   (16) "10 minutes before *the notebook* switches off, it starts beeping. 5 minutes later the display of *the computer* starts flashing as well."

   The literal denotation of $w =$ "*computer*" allows for anaphora resolution to $\tilde{w} =$ "*notebook*" as the conceptual correlates, COMPUTER-SYSTEM and NOTEBOOK, respectively, stand in a conceptual generalization relation, i.e., the conceptual correlate of the antecedent is a subconcept of the conceptual correlate of the anaphoric expression. For Example (16), ($\tilde{w}.con =$) NOTEBOOK $\sqsubseteq_{\mathcal{F}}$ COMPUTER-SYSTEM ($= w.con$) holds. As w is literal, it fulfills the required selectional restrictions (we abstract from possible granularity phenomena).

[15] When we sloppily refer to w as a 'definite noun phrase' in the following discussion, we actually mean the lexical head noun of that phrase in the metonymic case.

[16] If $\tilde{w}$ is used metonymically, then its conceptual correlate $\tilde{w}.con$ has to be replaced by the conceptual class of its underlying referent in the following discussion.

2. w corresponds to a *predicative metonymy*: [17]

(17) "10 minutes before the *notebook* switches off, it starts beeping. The clock frequency of **the computer** is reduced to 8 MHz."

The literal denotation of "*computer*" does not fulfill all selectional restrictions, as "*clock frequency*" is a property attributed to a processor, not to a computer as such. Thus, "*computer*" must be recognized as a metonymy for its CPU (cf. Example (12)). However, the literal denotation is available for anaphora resolution—"*computer*" ($= w$) resolves to "*notebook*" ($= \tilde{w}$). So $\tilde{w}.con \sqsubseteq_{\mathcal{F}} w.con$ holds as for case 1 above. Predicative metonymies do not cause a reference change so that no significant interaction between anaphora and metonymy resolution need take place.

3. w corresponds to a *referential metonymy*:

(18) "We also tested *the printer Epson EPL-5600*. I liked **the laser**, as its printouts were excellent." [18]

(19) "We also tested *the printer Epson EPL-5600*. I liked **the laser**."

In both examples, anaphora resolution of "*the laser*" ($= w$) to "*Epson EPL-5600*" ($= \tilde{w}$) is fully dependent on the resolution of the `Part-for-Whole` metonymy "*laser*" for "*laser printer*". The required conceptual generalization relation between $\tilde{w}.con$ (= EPSON-EPL-5600) and $w.con$ (= LASER) does not hold, although w and $\tilde{w}$ are coreferent. It does hold, however, between the antecedent and the *inferentially recoverable* referent of the anaphoric metonymy. So, if w is metonymic for an instance of the conceptual class B (e.g., $w =$ "*laser*" being metonymic for $B =$ LASER-PRINTER), then $\tilde{w}.con$ (= EPSON-EPL-5600) $\sqsubseteq_{\mathcal{F}} B$ holds.

One may argue that in Example (18) metonymy resolution can be achieved without information about the possible anaphoric antecedent of "*the laser*", as the analysis of the entire sentence reveals a semantic violation (the combination of "*the laser*" with "*its printouts*" will fail). This is yet another case of what we call *indirect* violations (cf. also Example (5)), the recognition of which requires the resolution of the possessive pronoun "*its*", as well as a full syntactic analysis to precede metonymy resolution. Considering an incremental approach, anaphora resolution for "*the laser*" would be triggered *before* the information about the violation was available so that the problem is reduced, in effect, to cases like Example (19).

Example (19) illustrates the benefits metonymy resolution gets from possible anaphoric antecedents. Firstly, this information may help with choosing amongst several metonymic readings—excluded are those readings that do not allow for anaphora resolution (e.g., the competing metonymic reading "*laser*" for "*light*") (see also Examples (8) and (9)). Secondly, only the information about possible antecedents can help triggering a metonymy resolution at all, since no semantic constraints are violated (cf. also our remarks about soft selectional restrictions in Section 2.1 and Examples (3) and (4)).

[17] Stallard [53] and Nunberg [42] discern cases of *predicative* and *referential* metonymy, depending on whether the literal or the inferentially recoverable referent, respectively, is available for subsequent pronominal reference.

II. w is not anaphoric

1. w is *literal*.
   The literal denotation of w fulfills all intrasentential semantic restrictions. Although w is not anaphoric, it is still possible to establish referential cohesion via the resolution of a textual ellipsis:

   (20) "We also tested *the printer Epson EPL-5600*. I did not like *the paper tray*."

   In Example (20), a conceptual relation (*Part-of*) between the conceptual correlates of "*paper tray*" and "*Epson EPL-5600*" can be inferred. The parallel structure of Examples (19) and (20) indicates that this criterion can also be met by anaphoric noun phrases w that are referential metonymies. Thus, we may determine either truly ambiguous readings or, at least, readings which cannot be distinguished without further lexical or pragmatic information. We incorporate *aptness* constraints in order to make informed choices among the alternatives.
2. w is *metonymic*.
   There can be a variety of idiosyncratic non-anaphoric definite metonymies, e.g., metonymic reference to an entity not mentioned in the text before (e.g., "*The White House*" for the US-President). These phenomena are very uncommon in our corpus, at least. A phenomenon which happens to come up more often in our texts, however, are predicative metonymies that are textual ellipses such as the following example:

   (21) "10 minutes before *the notebook* switches off, it starts beeping. **The processor** is throttled to 8 MHz."

   As only the frequency of a processor can be throttled, not the processor itself, "*the processor*" is metonymic for its frequency. However, the literal denotation of "*processor*" is available for a textual ellipsis, linking the conceptual correlates of "*processor*" and "*notebook*" via a *Part-of* relation. Here, metonymy resolution and textual ellipsis resolution are not interdependent as the metonymy can be recognized by an SRV and the literal denotation is available for textual ellipsis resolution (see the related issue for predicative metonymy and anaphora in Example (17)).

4.2. Determining referential readings

In order to determine referential readings, we will formalize the observations from Section 4.1 and incorporate them into our path finding paradigm. For this purpose, we now abstract from processing considerations (for the incremental extension of the basic metonymy resolution algorithm, cf. Section 4.4).

Up to now, we have only considered whether *one* particular path between two concepts (corresponding to one possible conceptual interpretation) can be classified as literal or metonymic. We now introduce a two-way extension. First, we will be considering *sets* of paths which have been computed by the `Path Finder` between the conceptual correlates of two lexical items. Since we are not subscribing to the view that literal readings are always preferred, we cannot exclude nonliteral paths from this path set a priori. Second,

Table 4
Conditions for literal readings

w can be interpreted as literal $\Leftrightarrow$
$\forall w' \in U_i(w)\ (\exists p \in P(w.con, w'.con) : literal_p(w))$

Table 5
Conditions for metonymic readings

w can be interpreted as a metonymy for (an instance of the concept class) $B \Leftrightarrow$
$\exists w' \in U_i(w)\ ((\exists p \in P(w.con, w'.con) : metonymic_p(w, B))$
$\wedge\ (\forall \hat{w} \in U_i(w), \hat{w} \neq w', (\exists \hat{p} \in P(w.con, \hat{w}.con) : (metonymic_{\hat{p}}(w, B) \vee literal_{\hat{p}}(w)))))$

we will consider cases where one lexical item can be syntactically related to *several* other lexical items. As a consequence, we then have to account for *sets of sets* of paths when resolving metonymies.

Let w be a definite noun phrase in the utterance U_i. We will call the set of all $w' \in U_i$ that are syntactically related to w as $U_i(w)$—the construction of this set in a lexicalized dependency grammar is described in detail in Romacker et al. [50]. All words that impose selectional constraints on w are elements of $U_i(w)$.

If w is literal, it must either fulfill all selectional restrictions or SRVs can be explained by granularity considerations. [19] In our path finding paradigm, this means that *any* syntactic relation between w and another word $w' \in U_i$ can be interpreted by a path p, which yields a literal reading of w, i.e., a literal path p between w'.CON and w.CON, or a metonymic path p which yields a metonymic interpretation of w' and a literal one of w (see also Definition 3.5). When a path p yields a literal interpretation of w, we will write $literal_p(w)$. The criteria under which w can receive a literal interpretation are summarized in Table 4.

If w is metonymic for an instance of the concept B, then there is at least one $w' \in U_i(w)$, so that a metonymic path p between w'.CON and w.CON exists with $metonymic_p(w, B)$. As w cannot be metonymic for two different concepts all other paths from w.CON to any $\hat{w}$.con, $\hat{w} \in U_i(w)$, must allow for the same metonymic interpretation or for a literal one. [20] The criteria for w under a metonymic interpretation are summarized in Table 5.

We may now accomodate these definitions to incorporate anaphora as well. Since the identity of the antecedent is not known in advance, we have to consider a list of all possible antecedents of w. Based on a functional centering framework [56], each utterance U_{i-1} is assigned a partially ordered set of forward-looking centers $C_f(U_{i-1})$ containing the possible antecedents for an anaphor in U_i. In our framework, this centering list consists of

[19] We do not consider cases where selectional restrictions are violated due to metaphoric readings of the verb.

[20] The co-occurrence of literal and metonymic interpretations of a word w regarding different words w' in U_i is possible. In the phrase "*The clock-frequency of the new computer*", e.g., "*computer*" stands metonymically for its processor with regard to the constraint imposed by "*clock-frequency*", but in a literal relationship to "*new*". Alternatively, we may also argue that "*new computer*" stands metonymically for a processor of the new computer. For a deeper treatment of such compositionality issues, cf. Stallard [53] and Nunberg [42].

Table 6
Literal and predicative metonymic anaphora

w can be interpreted as a literal anaphor $\Leftrightarrow$
w can be interpreted as literal $\wedge\, \exists\, c \in C_f(U_{i-1}) : \mathit{class}(c) \sqsubseteq_{\mathcal{F}} w.\mathrm{CON}$
w can be interpreted as a predicative metonymy for B and as anaphoric $\Leftrightarrow$
w can be interpreted as a metonymy for B
$\wedge\ \exists\, c \in C_f(U_{i-1}) : \mathit{class}(c) \sqsubseteq_{\mathcal{F}} w.\mathrm{CON}$

Table 7
Referential metonymic anaphora

w can be interpreted as a referential metonymy for B and as an anaphor $\Leftrightarrow$
w can be interpreted as a metonymy for $B \wedge \exists\, c \in C_f(U_{i-1}) : \mathit{class}(c) \sqsubseteq_{\mathcal{F}} B$

Table 8
Conditions for textual ellipsis

w can be interpreted as a literal textual ellipsis $\Leftrightarrow$
w can be interpreted as literal $\wedge\, \exists\, c \in C_f(U_{i-1}) : P(\mathit{class}(c), w.con) \neq \emptyset$

instances in the knowledge base, corresponding to the referents of words or phrases in the text.

If w is a literal or a predicative metonymic anaphor, then $w.\mathrm{CON}$ allows for anaphora resolution. Hence, the centering list contains an element whose class is a subconcept of $w.\mathrm{CON}$. These conventions are summarized in Table 6.

If w is a referential metonymy for an instance of B and an anaphor, too, then B (instead of $w.\mathrm{CON}$) allows for anaphora resolution (cf. Table 7).

If w is a literal textual ellipsis, then $w.\mathrm{CON}$ can be linked via a non-generalization conceptual relation (i.e., a well-formed conceptual path) to its antecedent (cf. Table 8).[21]

As the following discussion will show, these predicates are not mutually exclusive and lead to ambiguous readings.

Example (16). With $w =$ "*computer*", $U_2(w) =$ ("*display*") and $C_f(U_1) =$ (`Notebook-2`, `Minute-1`) hold. The (only) well-formed path between DISPLAY und COMPUTER-SYSTEM found by the `Path Finder` is the literal path (DISPLAY, *Visual-Device-Of*, NOTEBOOK). Therefore only a literal interpretation of "*computer*" is allowed (cf. Table 4). Since *class*(`Notebook-2`) $=$ NOTEBOOK $\sqsubseteq_{\mathcal{F}}$ COMPUTER-SYSTEM ($=$ $w.\mathrm{CON}$) holds, an interpretation of w as a literal anaphor to "*notebook*" is determined (cf. Table 6).

[21] This predicate can be generalised to textual ellipsis that are predicative metonymies (see Example (21)).

Example (17). With $w =$ "*computer*", $U_2(w) =$ ("*clock-frequency*") and $C_f(U_1) =$ (`Notebook-2, Minute-1`) hold. The `Path Finder` finds only metonymic paths between CLOCK-FREQUENCY and COMPUTER-SYSTEM, expressing that "*computer*" is metonymic for an instance of the concept class CPU ($= B$) (see also Example (12)). Therefore only a metonymic reading is possible (cf. Table 5). According to Table 6 an interpretation as a predicative metonymy allows for anaphora resolution. An interpretation as a referential one does not allow for anaphora resolution (cf. Table 7).

Examples (18) and (19). $C_f(U_1) =$ (`Test-Team, Epson-EPL-5600-2`) holds in both cases,[22] as well as $w =$ "*laser*" and $U_2(w) =$ ("*liked*"). The `Path Finder` finds the following paths between LASER and LIKE.

1. (LIKE, *Like-Patient*, OBJECT) is a *literal* path expressing that "*laser*" is meant literally. An anaphoric literal interpretation is not possible, as EPSON-EPL-5600 $\sqsubseteq_{\mathcal{F}}$ LASER does not hold (cf. Table 6). An interpretation as a literal textual ellipsis is possible with $p =$ (EPSON-EPL-5600, *Has-Laser*, LASER) as the proper conceptual relation (cf. Table 8). This reading means that we really talk about an instance of the class LASER, which is *Part-of* the LASER-PRINTER mentioned before.
2. (LIKE, *Like-Patient*, OBJECT, $\sqsupseteq_{\mathcal{F}}$, EPSON-EPL-5600, *Has-Laser*, LASER) is a metonymic path expressing a `Part-for-Whole` metonymy where "*laser*" stands for an instance of the class EPSON-EPL-5600 $=: B$ (cf. Table 5 and Definition 3.5). Regarding anaphora resolution, the following two hypotheses are valid.
   (a) "*Laser*" is a predicative metonymy. Then anaphora resolution is precluded (cf. Table 6).
   (b) "*Laser*" is a referential metonymy. Then anaphora resolution to "*Epson-EPL-5600*" is possible (cf. Table 7).

Example (20). Similar results as for Examples (18) and (19) hold for Example (20). The `Path Finder` finds the two paths (*Like-Patient*) und (*Like-Patient, Has-Paper-Tray*), which allow for the same kinds of readings as Example (19). The ambiguity between a literal textual ellipsis reading and a referential metonymic anaphoric reading is especially frequent.

Example (5). At this point it is worthwhile to revisit Example (5), which we have already analyzed from the `Path Finder` perspective. Here, $w =$ "*Quantum*", $U_2(w) =$ ("*achieves*") and $C_f(U_1) =$ (`LPS105-2, Performance-1`) hold. Therefore, a literal reading of "*Quantum*" does not allow for anaphora resolution as neither PERFORMANCE $\sqsubseteq_{\mathcal{F}}$ QUANTUM nor LPS-105 $\sqsubseteq_{\mathcal{F}}$ QUANTUM holds. However, if we interpret "*Quantum*" metonymically for an instance of the concept HARD-DISK-DRIVE, then anaphora resolution to "LPS 105" is possible, assuming the path $p :=$ (ACHIEVE, *Achieve-Agent*, $\sqsupseteq_{\mathcal{F}}$, HARD-DISK-DRIVE, *Produced-by*, PRODUCER) can be determined.

[22] Given the discourse context of a typical IT test report, and technical writers acting simultaneously as test personnel, "*We*" has TEST-TEAM as a conceptual correlate. This item is, nevertheless, ruled out as a potential antecedent for "*the laser*" due to number disagreement.

Table 9
Principle of Referential Success for literal and metonymic readings

literal anaphor $>_{ref}$ other literal readings
metonymic anaphor $>_{ref}$ other metonymic readings

Table 10
Principle of the Direct Trigger for anaphoric readings

(literal anaphor $\vee$ predicative metonymic anaphor) $>_{ref}$ referential metonymic anaphor
literal anaphor $>_{ref}$ textual ellipsis

4.3. Ranking alternative readings

A crucial problem arising from the non-determinacy of the criteria proposed so far is their potential for creating alternative interpretations. In this section we will introduce two general disambiguation heuristics and integrate them into the metonymy resolution procedure. The first of these is inspired by Crain and Steedman [14], who propose a *Principle of Referential Success* for syntactic ambiguity resolution. Applied to metonymy resolution (cf. also Principle 3 in Section 2.3), it implements the preference for readings that establish referential cohesion over ones that do not (always assuming that both readings are compatible with intrasentential and knowledge-based constraints). Its formulation in Table 9 combines the preference assumptions (indicated by "$>$") for literal and metonymic readings.

Nunberg [41] claimed that anaphoric readings which are directly triggered (i.e., by equality or by a generalization relation between anaphor and antecedent, like literal anaphora) should be preferred over other anaphoric constructions (e.g., textual ellipsis or referential metonymic anaphora). This is reflected in the *Principle of the Direct Trigger* (Table 10).

These two basic heuristics can be combined in different ways. One choice is to assume a strict preference of literal over figurative readings—such a *Literal-Meaning-First* (LMF) approach (cf. Table 11) has been adopted by the majority of researchers in the field of metonymy processing.

Here, the *Principle of Referential Success* is used only *within* literal or metonymic readings. When a metonymy does not violate intrasentential constraints, a competing literal reading always exists, which according to Table 11 will then always be preferred over the metonymic one. This disregards the possibility that discourse constraints might nevertheless favor the metonymic reading. We therefore subscribe to the disambiguation heuristics expressed in Table 12. This ranking incorporates the *Principle of Referential Success* (cf. Table 9) and the *Principle of the Direct Trigger* (cf. Table 10), but avoids preferring literal readings over metonymic ones in all circumstances. In addition, it incorporates a preference for conventional readings (literal or schematized metonymic ones) over unconventional ones (unclassified readings) (cf. Principle 4 in Section 2.3).

Table 11
Literal-Meaning-First approach

literal anaphor $>_{ref\text{-}LMF}$
(literal) textual ellipsis $>_{ref\text{-}LMF}$
other literal readings $>_{ref\text{-}LMF}$
predicative metonymic anaphor $>_{ref\text{-}LMF}$
referential metonymic anaphor $>_{ref\text{-}LMF}$
other metonymic readings

Table 12
Disambiguation scheme for literal and metonymic readings

literal anaphor $>_{ref}$
predicative metonymic anaphor $>_{ref}$
(referential metonymic anaphor $\vee$ literal non-anaphoric readings, including textual ellipsis) $>_{ref}$
non-anaphoric metonymic readings $>_{ref}$
unclassified readings

Our disambiguation heuristics have also been evaluated against the heuristics in Table 11. They led to an increase in metonymy resolution from 69% of all anaphoric metonymies to 84.5%. The data are illustrated in detail in Section 5.2.3.

We may now formulate a (non-incremental) algorithm for the interpretation of a definite noun phrase w in U_i:

1. Determine *all possible* literal and metonymic readings of w that are compatible with the intrasentential selectional restrictions. In our framework this can be done as follows:
   (a) Determine $U_i(w)$.
   (b) Determine all well-formed conceptual paths between w'.CON and w.CON for all $w' \in U_i(w)$ using the `Path Finder`.
   (c) Derive all literal and metonymic readings from these paths, using the `Path Classifier` and the predicates in Tables 4 and 5.
2. Determine the possible anaphoric readings for the ensuing literal and metonymic readings with the predicates in Tables 6, 7 and 8, using $C_f(U_{i-1})$, the list of forward looking centers of U_{i-1}.
3. Disambiguate between the ensuing readings using the preference ranking in Table 12.

Our algorithm does not affect the resolution of literal anaphora. For Example (16) only a literal reading of "*computer*" is possible. We prefer to interpret this reading as an anaphor to "*notebook*" (instead of assuming that a new computer is introduced) (see line 1 in Table 12). Example (17) shows that metonymies with SRVs are handled as in all other approaches, since no competing literal interpretation exists. Thus, "*computer*" in Example (17) is treated as a `Whole-for-Part` metonymy. As the literal meaning of "*computer*" allows for anaphora resolution we treat the metonymy as a predicative one (line 2 of Table 12).

Our algorithm allows for the resolution of anaphoric metonymies that are *not* marked by an SRV. So, in Example (19), "*the laser*" can be resolved to "*Epson EPL-5600*" in at least one reading (line 3 of Table 12). The previous discourse also rejects unplausible metonymic readings that might have been licensed from an intrasentential perspective only, but become void in the discourse context as, e.g., a "laser" for "light" reading (see also Examples (8) and (9)). The computation of such readings is usually blocked by our notion of extended connectivity (cf. Definition 3.4), which takes the discourse context into account. In addition, any such reading is ranked low in Table 12 (line 4). We also do not depend on information about indirect semantic constraints that may not be available when metonymy or anaphora resolution is called for and which are hard to recognize (see Examples (18) and (5), which are handled in the same way as Example (19)). Our approach therefore clearly extends both metonymy and anaphora resolution.

A crucial problem of our approach lies in the disjunction in line 3 of Table 12, which is a continuous source of ambiguity. Whereas it leads correctly to an ambiguity in Example (19), it also leads to the same kind of ambiguity in Example (20) where this is clearly not wished for. In the latter example, "*paper tray*" should only be interpreted literally as the paper tray of the laser printer, i.e., as a case of textual ellipsis, and not have an additional reading as an anaphoric reference for "*the printer Epson EPL-5600*" via a `Part-for-Whole` metonymy. The algorithm as proposed so far does not yield any criteria to prevent this kind of overgeneration.

This is by no means an artificial problem created by our algorithm . Rather, it mirrors the fact that the resolution of metonymies is not fully constrained by metonymic patterns like `Part-for-Whole` or `Producer-for-Product`. One may ask why "*laser*" can easily stand for a laser printer, whereas "*paper tray*" cannot? This question is more or less independent of applying an SRV-based approach or not. Although in this particular example the lexematic similarity between the words *laser* and *laser printer* might play a role, other examples show that conceptual reasons play a bigger role. Why is it that "*screen*" can easily be used metonymically for a TV, whereas "*transistor*" is much harder? Or that "*head*" is often used metonymically for a person, whereas "*kidneys*" are not? We hypothesize that the typicality and visibility of parts play a crucial role for this metonymic usage (so, e.g., a laser is a more *typical* part for a laser printer than the paper tray, as it distinguishes a laser printer from other kinds of printers; the head is more visible than the kidneys, etc.). Hahn and Markert [25] formalise three aptness heuristics that capture these intuitions and further constrain coercion, so that the ambiguity in line 3 of Table 12 does only arise for apt metonymies (see also Principle 5). Another reason might be related to the functionality of components: a laser carries the main part of the functionality of a laser printer whereas a paper tray does not. This latter point was raised by one anonymous reviewer.

Still a hard case to deal with are non-definite metonymic noun phrases, which we can only handle when they violate selectional restrictions. In general, the 'Principle of Referential Success' is not applicable to them. So we fall back on preferring literal over metonymic readings if two or more readings fulfill intrasentential selectional restrictions, although we will show in Section 5.1 that this is not satisfactory (see also Example (3)).

Table 13
Local conditions for literal readings

w can locally be interpreted as literal $\Leftrightarrow$
$\exists p \in P(w.con, w'.con) : literal_p(w)$

Table 14
Local conditions for metonymic readings

w can locally be interpreted as a metonymy
for (an instance of the concept class) $B \Leftrightarrow$
$\exists p \in P(w.con, w'.con) : metonymic_p(w, B)$

4.4. Incremental metonymy resolution

The algorithm described above and the predicates in the Tables 4, 5, 6 and 7 are non-incremental as they depend on using *all* w' in $U_i(w)$. In an incremental algorithm not all such w' are known when metonymy resolution or anaphora resolution actually takes place. We can adapt the original algorithm to an incremental processing mode under the assumption that metonymy resolution and anaphora resolution are triggered when a particular conceptual relationship between w.CON and another w'.CON is being determined (as an interpretation of a syntactic relation). [23]

1. Determine all possible literal and metonymic readings of w that are compatible with the selectional restriction mediated by w'. In our framework this can be done as follows:
   (a) Determine all well-formed conceptual paths between w'.CON and w.CON using the `Path Finder`.
   (b) Derive all literal and metonymic readings from these paths, using the `Path Classifier` and local equivalents of the predicates in Tables 4 and 5 as given in Tables 13 and 14, respectively.
2. Determine the possible anaphoric readings for the ensuing literal and metonymic readings with local equivalents of the predicates in Tables 6, 7 and 8, using $C_f(U_{i-1})$, the list of forward looking centers of U_{i-1}.
3. Disambiguate between the ensuing readings using the preference ranking from Table 12.

The incremental version of the algorithm needs only little specification changes compared with its non-incremental version. For determining literal and metonymic readings we use only *one* $w' \in U_i(w)$ and the `Path Finder`, which operates locally on two concepts. The ensuing paths will be classified as usual, since the `Path Classifier` operates on single paths anyway. The two predicates for literal and metonymic readings in their 'local' version are defined in Tables 13 and 14, assuming w' as an arbitrary constant. Anaphoric interpretation is not affected at all by incrementality, since the centering list is given at the end of the previous sentence already, and is therefore available for determining anaphoric readings no matter when anaphora resolution is carried out. One example would be the local interpretation of Table 6 in Table 15. Also, the disambiguation scheme (Table 12) remains stable for different processing strategies.

[23] One possibility is to proceed from left to right and use the relationship to the left-most w' to be established. Another possibility, which we will evaluate in Section 5, is to base disambiguation on a particular prominent type of dependency relation, e.g., between w and its head. This relationship is especially important as every word has exactly one head in a dependency grammar, whereas it can have several modifiers.

Table 15
Literalness and predicative metonymies as anaphors—the incremental interpretation

w can locally be interpreted as literal and as anaphoric $\Leftrightarrow$
w can be locally interpreted as literal
$\wedge\ \exists\, c \in C_f(U_{i-1}) : \mathit{class}(c) \sqsubseteq_{\mathcal{F}} w.\mathrm{CON}$
w can be locally interpreted as pred. metonymy for B and as anaphor $\Leftrightarrow$
w can be locally interpreted as metonymy for B
$\wedge\ \exists\, c \in C_f(U_{i-1}) : \mathit{class}(c) \sqsubseteq_{\mathcal{F}} w.\mathrm{CON}$

As the decision on metonymic or literal readings and the disambiguation predicate have only a local basis, one might expect that they are more prone to errors. Assume a locally chosen reading that turns out not to be compatible with the selectional restrictions which surface during the processing of the rest of the sentence. Then backtracking is needed, which we have not incorporated yet. In our experiments it turned out, however, that local disambiguation is surprisingly accurate for our texts (at least, if one chooses w's head as w')—the reason being that one of the main influences on disambiguation is the previous discourse, which does not change as more selectional restrictions are incorporated. Coming back, e.g., to Example (5)—if anaphora and metonymy resolution are triggered when establishing the relation between "*Quantum*" and "*achieves*", we already establish the literal *and* the metonymic reading due to the discourse context (line 3 in Table 12). Later inconsistencies for the literal reading signalled by an indirect semantic constraint violation might lead to discarding it, but the correct metonymic interpretation remains. (Note that in the examples discussed so far $U_i(w)$ contains only one element, so that the non-incremental and incremental algorithm make no difference to the final readings.)

5. Evaluation

We will now give an evaluation of both our theoretical claims, regarding the interactions of metonymies with selectional restrictions and with nominal anaphora (Section 5.1), and of the incremental metonymy resolution algorithm described in Section 4.4 (Section 5.2).

We analyzed a sample of 16 German-language product reviews from information technology magazines. The texts contained a total of 622 sentences.[24] The gold standard for comparison was determined manually by considering all nouns (including proper names) in the sample. Other phenomena (e.g., pronouns or verbs) were discarded from analysis because they are rarely metonymic. The test set contained 2719 nouns.

[24] This evaluation is an extension of the one presented in Markert and Hahn [38]. In the latter paper we refer to 26 texts, with one text being a compilation of 12 printer reviews. We here count this compilation as only one single text for reasons of comparison with other evaluation data. We enhance our previous evaluation by including farther developed annotation principles for metonymies, as well as a more fine-grained distinction of semantic violations.

5.1. Occurrence rates

5.1.1. Metonymies

For every noun we decided whether it was metonymic and, if so, resolved the metonymy. This way, we manually annotated the texts with pairs of the form (metonymic expression—resolved metonymy). The annotation built on a stringent theoretical distinction between literal and figurative language [26], which forms the basis for simple *replacement tests* for deciding on metonymy. These tests are primarily based on metonymic schemata—for the `Producer-for-Product` schema, e.g., the following rule was used, which is applicable to all words w in the lexicon mapped to a concept in the knowledge base that is a subconcept of PRODUCER (e.g., all producer names such as "*Quantum*", the noun "*company*", etc.): If w occurs in an utterance U, then replace w, on the one hand, by the expression "*producer* w" (literal reading) and, on the other hand, by the expression "*product by* w" (metonymic reading). Choose the replacement which captures the sense of the utterance most adequately in the given discourse context. Similar rules were developed for the other schemata we employ.

Our test set also contained non-schematic metonymies. If neither the literal reading nor the schematic metonymic readings fulfilled the replacement test, we looked for other metonymic readings. In this case, we relied mainly on our intuition. If we determined w to be metonymic for an instance of the concept B, the replacement test with B was carried out to check that intuition.

If both a literal and a metonymic reading (or neither) fulfilled the replacement test, the example was seen as literal to avoid including metonymies into our test set where the decision was not clear cut.

Given these conventions, we identified 106 metonymies in the 622 sentences. This means that 17% of all sentences contained a metonymy and that semantic analysis would fail for a wide range of sentences without metonymy resolution.

5.1.2. Interaction of anaphors and metonymies

We also annotated nominal anaphors (anaphoric expression—antecedent) and textual ellipses (elliptic expression—antecedent) in the sample. For most of the 16 texts (451 from 622 sentences) this annotation was carried out independently of the authors by one of our colleagues who used the same corpus as evaluation data for a centering algorithm [56]. The remainder was annotated by the authors themselves, using the same method. This independent annotation of metonymies and anaphora is crucial for the exclusion of biases in metonymy annotation, e.g., to avoid concentrating on anaphoric metonymies only. We found 303 nominal anaphors and 355 textual ellipses in the sample.

When comparing the two annotations, we found the following interactions. From the perspective of metonymies, 78 (73.6%) of them were (heads of) definite noun phrases [25] and 58 (54.7% of all metonymies; 74.4% of all definite metonymies) were nominal anaphors. This high co-occurrence rate can be compared with the general occurrence rate

[25] Definite noun phrases in German are ones which contain a definite article, either "*der*", "*die*", "*das*", or one of their inflectional variants, contractions of prepositions with these definite articles (e.g., "*zur*", "*zum*") and deictic expressions incorporating the basic form "*dies*". All the other remaining noun phrases are here called *non-definite*.

of anaphora. Of all 2719 noun phrases in all the texts 1287 (47.33%) were definite noun phrases and only 303 (11.14% of all noun phrases; 23.54% of all definite noun phrases) were nominal anaphors. [26] Therefore, we conclude that metonymies have a tendency to be anaphoric.

From the perspective of anaphora, the following picture emerges. 58 from 303 nominal anaphora (19.1%) are metonymic. From these, 50 (16.5% of all nominal anaphora) are referential metonymies. Note that these anaphora are not resolvable with the standard semantic criteria, unless metonymy resolution is incorporated (cf. Section 4.1). This number is rather high and might indicate some degree of genre specificity (e.g., in the product reviews one encounters a lot of `Producer-for-Product` metonymies, which are always referential and mostly anaphoric). Interestingly enough, in a follow-up study using our methodology, Harabagiu [29] was able to replicate this data for another text genre and the English language as well. In particular, she points out that 16.3% of the nominal anaphora that appeared in 20 texts from the *New York Times* can only be dealt with by including a dedicated metonymy resolution mechanism.

This data supports our view that the interaction of metonymies and anphora is especially frequent and its incorporation into a metonymy resolution algorithm will eventually result in a significant performance improvement of both anaphora and metonymy resolution algorithms.

5.1.3. Semantic violations

Up to now we have illustrated our claim why SRVs are inadequate as a triggering condition for metonymy resolution with examples only. Since explicit data has been lacking so far in the discussion of metonymy resolution algorithms, we will substantiate our claims with quantitative data from our test set.

Identifying a direct SRV is a rather straightforward task. As noted in Section 3.1, selectional restrictions are incorporated in *range* restrictions of concept roles in the knowledge base (e.g., *Agent* and *Patient* restrictions for a verb). Whenever a syntactic relation between a word w and some $w' \in U_i(w)$ is going to be established and its conceptual correlate $w.\textsc{con}$ does not belong to the range of the corresponding conceptual role of $w'.\textsc{con}$, an SRV is encountered.

In our study of SRVs we have included the syntactic relations listed in Table 16. Verb roles encompass direct and indirect objects as well as the subject. Logical metonymies are included here as well. PP-attachment encompasses attachments to nouns, verbs and adjectives. Approaches which handle such direct SRVs only (e.g., Fass [17] and Amghar et al. [1]) are systematically limited to 49 (46.23%) of all metonymies.

Identifying an indirect semantic violation is much harder than a direct SRV. In many cases there is no clear-cut dividing line between readings that cause semantic violations and readings that are semantically well-formed, but rather odd due to default assumptions about lexemes or the real world, or readings that cause conceptual or pragmatic violations. As a first approximation, we identified metonymies with semantic violations as metonymies

[26] Comparable occurrence rates are reported by Fraurud [20], who counts 36% of definite noun phrases to be nominal anaphora in a Swedish corpus. The remaining difference may be due to different text genres or language-specific properties of the texts.

Table 16
Types and occurrence data of direct SRVs

Type of SRV	Example	Frequency
Verb roles	"*der **Prozessor** wird zurückgefahren*" ("***the processor** is throttled down*") (`Processor-for-Frequency`)	31
Genitives	"*Der Kontrast des **LCD***" ("*The contrast of **the LCD***") (`LCD-for-Picture`)	10
Adjective-Noun modifications	"*das helle **Display***" ("*the bright **display***") (`LCD-for-Picture`)	5
PP-attachments	"*ist er mit **2 MByte** bestückt*" ("*it [the computer] is equipped with **2 MByte***") (`MByte-for-Storage`)	3
$\sum$ frequency		**49**

exhibiting purely sentence-internal violations that do not need additional conceptual inference, default reasoning or discourse for recognition.

Four subtypes of indirect semantic violations appear in Table 17. Type I (Verb frame) has already been mentioned by Fass [17] and Iverson and Helmreich [32] (cf. also Example (5)). Two arguments fulfil the restrictions of a verb when considered in isolation, but fail to do so when both restrictions are combined. This holds especially between the subject and direct or indirect object of a verb.

Type II are restrictions brought about by relative clauses of the type "w *that/who/which performs an action*" or "w *that/who/which an action is performed on*". In such a construction the literal reading of w violates an indirect semantic restriction if it violates a (direct) SRV in the transformations "w *performs an action*" or "*an action is performed on* w". Thus, "***the laser** that prints*" violates an indirect semantic restriction as "***the laser** prints*" violates a direct one.

Type III are semantic violations which result from resolving a comparative or a comparison between two objects, where one is referred to metonymically, e.g., "*The hard disk is as good as/better than **the Seagate***". Comparisons can be handled by determining candidate correlates between which the comparison is carried out (here, e.g., between "*hard disk*" and "*Seagate*"). Depending on the restrictions one puts on correlate types, we can only determine the comparison if we interpret "*Seagate*" metonymically. Note that this is the easiest case of the interaction between metonymies and comparatives (for a more detailed study including discourse phenomena, cf. Staab and Hahn [52]). For reasons of simplicity, we regard all comparatives where one of the correlates is metonymic as instances of intrasentential violations.

Type IV are semantic violations that presume the resolution of anaphors that are not identical to w. These anaphora resolutions do not interact with metonymy resolution, but only help with discovering an intrasentential violation. Discourse constraints on the metonymy itself are not seen as intrasentential semantic violations.

Table 17
Types and occurrence data of indirect semantic violations

Type of violation	Example	Frequency
Verb frames	"*Grafik wie Fotos bringt der **Laser**.*" ("***the laser** yields graphics and photos*") (`Laser-for-Laser-Printer`)	18
Relative clauses	"***Vieldrucker**, die jeden Tag viele Seiten ausgeben*" ("***people** who print many pages every day*") (`User-for-Printer`)	3
Comparatives or comparisons	"*ein Laufwerk, das genauso geringe Aussenmasse besitzt wie die **Maxtor***" ("*a hard disk that is as small as the **Maxtor***") (`Producer-for-Product`)	7
Previous anaphors	"*sie [Quantum-Festplatte] lag in etwa gleichauf mit der **Seagate** auf Platz 1*" ("*it [the Quantum hard disk] is on level pegging with the **Seagate** at rank 1*") (`Producer-for-Product`)	2
$\sum$ frequency		**30**

We also regard all combinations of these semantic violations as a semantic violation. So, in the example "*It is on level pegging with the **Seagate***", the discovery of the violation involves the resolution of a pronominal anaphor that does not interact with metonymy resolution *and* the discovery of an indirect verb frame violation. In many of these examples information about anaphoric readings of the *metonymy* can give clues to metonymic readings before the violation is encountered. It is also often easier to identify and does not need the interaction of several semantic processes. So, in the example described above, "*the Seagate*" can be anaphorically resolved to "*a hard disk produced by Seagate*" in the metonymic reading. This gives evidence for a metonymy following the 'Principle of Referential Success' without heavy inferencing. As discovering indirect violations is usually very hard, almost none of the previous approaches to metonymy resolution describe what kinds of violations they cover (apart from direct SRVs).

Finally, we found 27 (25.47% of all metonymies) examples without any intrasentential violations—a substantial number. Here we included metonymies that can only be recognized by discourse constraints or conceptual inferences. Examples for the first type have been given throughout the paper (see Section 4.1). They are especially common with evaluative phrases ("***The Quantum** achieves good results*"), but can also be present in descriptions ("***The laser** is in the secretary's office*"). Dealing with the second type requires a wide variety of inference types, which we do not provide as yet—temporal reasoning for Example (3), reasoning about the likelihood of people being on shelves for Example (22) (cf. Nunberg [42] for similar examples).

(22) "**Pynchon** is on the top shelf."

Table 18
Occurrence data for metonymies

	Direct SRV		Ind. sem. violation		No sem. violation		Frequency of occurrence	
Anaphora	18	(17.0%)	22	(20.8%)	18	(17.0%)	**58**	**(54.7%)**
Other definite NPs	10	(9.4%)	7	(6.6%)	3	(2.8%)	20	(18.9%)
Non-definite NPs	21	(19.8%)	1	(0.9%)	6	(5.7%)	28	(26.4%)
$\sum$	**49**	**(46.2%)**	30	(28.3%)	**27**	**(25.5%)**	106	(100.0%)

The occurrence data for metonymies is summarized in Table 18 (percentages are rounded to the nearest tenth of a percent). In the table "ind." stands for "indirect" and "sem." for "semantic".

5.1.4. Schematized versus non-schematized metonymies

Our texts contained mainly metonymies that belonged to one of the 5 schemata we described in Section 3.3. Of the 106 metonymies we found, 92 (86.79%) belonged to one of the given schemata, 6 were logical metonymies and only the remaining 8 were non-schematized metonymies. Therefore our data on unconventional metonymies is scarce. Even of these 8 non-schematized metonymies, some were not really unconventional metonymies, but belonged to a schema that we have not included yet. So, in the example "***The processor** is throttled down*" we encounter a `Processor-for-Frequency` metonymy, which is basically a `Object-for-Attribute` metonymy. This is not covered by our schemata, but is not really unconventional either, being similar to `Whole-for-Part` metonymies. Thus, truly innovative metonymies similar to Example (1) were limited to one or two examples in our corpus. These were highly context-dependent examples: so, e.g., a previously mentioned cursor was referred to anaphorically as "the arrow" in one text.

5.2. Resolution rates

5.2.1. Method

From an architectural point of view, the metonymy resolution process interacts with the parsing and anaphora resolution process. However, in order to avoid error chaining, i.e., carrying over or interacting with the erroneous results from these co-processes, the input from these components was simulated manually. In particular, the two concepts with which the `Path Finder` and the `Path Classifier` are fed and, for definite noun phrases, the list of possible anaphoric antecedents from the previous utterance contained in the centering list were manually extracted from the texts. With this data supplied, the metonymy resolution algorithm, composed of the `Path Finder` and `Path Classifier`, as well as the disambiguation on the basis of metonymic schemata and discourse information were carried out fully automatically.

Thus, for every metonymy w we extracted its syntactic head w' and ran the `Path Finder` and `Path Classifier` with w.CON and w'.CON, respectively. For definite noun phrases, we also supplied the centering list of the previous utterance. For Example

(23), e.g., we supplied the `Path Finder` and `Path Classifier` with the concepts TOSHIBA and COME-UP-WITH.

(23) "**Toshiba** hat sich mehr einfallen lassen."
"**Toshiba** came up with more ideas."
(`Organisation-for-Members` metonymy, a subtype of `Whole-for-Part` metonymies; cf. Lakoff and Johnson [35])

For non-definite metonymies, an LMF disambiguation (preferring literal over metonymic readings) was carried out. For definite metonymies we used the disambiguation scheme in Table 12.

A schematic metonymy w for an instance of concept B (as determined in the annotation phase) is *correctly resolved* if one of the computed paths p incorporates the correct metonymic reading $\textit{metonymic}_p(w, B)$ (see Definition 3.5) *and* is ranked first by the disambiguation criteria (other readings might be ranked equally high as well). For metonymies that are not schematic, w is considered as correctly resolved, if an unclassified path passing B was ranked first. In cases of an anaphoric metonymy we also required the correct antecedent to be found.

5.2.2. Non-definite metonymies

From a total of 106 metonymies, we encountered 28 non-definite metonymies of which six occurred without semantic violations (cf. Table 18). For these six, the literal reading is incorrectly ranked highest by our disambiguation predicate, since for non-definite NPs we follow a strict LMF policy. One single metonymy appeared with an indirect semantic violation. As it occurred within a comparative it could be recognized and interpreted correctly due to the interaction between comparatives and metonymies [52].

21 non-definite metonymies violated SRVs. 4 of them were logical metonymies, which we cannot resolve due to the failure of establishing a syntactic link, which we require to hold before metonymy resolution is started at all. From the remaining 17 non-definite metonymies with SRVs, 12 could be correctly resolved, i.e., the `Path Finder` found the correct path and it was ranked highest. In these cases the result was also unambiguous. So, Example (23) was correctly interpreted as a `Whole-for-Part` metonymy (for the employees of Toshiba). In the other five cases, two metonymies could not be resolved as the corresponding concepts were not specified in the knowledge base. Three other ones remained unresolved because the `Path Finder` was unable to find the correct paths. Therefore 13 (46.4%) from 28 non-definite metonymies could be correctly resolved. The main reasons for this low rate are the choice of the LMF approach for disambiguation and the inability to deal with logical metonymies.

5.2.3. Definite metonymies

From the 20 non-anaphoric definite metonymies, the three without semantic violations could not be resolved due to lacking specifications of the head w' in the knowledge base. Regarding the seven with indirect semantic violations, five could be resolved because of the interaction with comparatives. The other two could not be resolved—the `Path Finder` did not find any path between $w.\textsc{con}$ and $w'.\textsc{con}$ as we excluded global specialization during the search process (cf. Definition 3.4). From the ten metonymies with direct violations, six were resolved correctly and unambiguously; one was a logical metonymy which we cannot account for; in three cases the `Path Finder` did not find the valid paths, again, due to excluding a global specialization in the search process. Thus, only 11 (55.0%) of the definite non-anaphoric metonymies could be resolved.

Our discourse predicates directly refer to anaphoric metonymies and, therefore, could not be used for non-definite metonymies and the analysis of non-anaphoric definite metonymies. When discourse knowledge is made accessible a direct positive effect on the resolution quality can be expected. Indeed, from 58 anaphoric metonymies only three were not resolved because of lacking knowledge specifications. In four cases the `Path Finder` found only paths which did not express the correct metonymic schema. In all other 51 cases the `Path Finder` identified the correct path (among others).

Our disambiguation schema in Table 12 ranked the metonymic reading correctly on first place in 49 of the 51 cases, although in 34 cases literal readings were found as well and ranked equally high (line 3 of Table 12). This was only the case when no SRV at the head w' was encountered. In cases without any semantic violations it is correct that the ambiguity between literal and metonymic readings remains (cf. Example (19)). In cases with an indirect semantic violation, our evaluation method leads to an ambiguity in many cases as we used only local constraints for w'—in an actual parsing setting it is more likely that the literal reading is subsequently discarded when the indirect semantic violation is encountered, thus increasing the precision of the procedure. We currently have no data on this issue.

On the whole 49 (84.5%) of the 58 anaphoric metonymies could be resolved. This high rate lends some credit to our disambiguation ranking. Especially important is the fact that 86.4% of all anaphoric metonymies marked by an indirect semantic violation and 88.9% without a violation can be resolved—with local information only. Approaches relying on semantic violations only are systematically restricted to 40 (69.0%) from all 58 anaphoric metonymies (18 direct SRV and 22 indirect violations; see Table 18). We conclude by summarizing all resolution rates for metonymies in Table 19. In the Table "ind." stands for "indirect".

5.3. Discussion

Our data indicates that metonymies and anaphora interact heavily and that any approach accounting for these interactions will be beneficial to the resolution rate for both anaphors and metonymies. In addition, it seems that SRVs are not as frequent with metonymies as commonly believed—even metonymies without any semantic violation occur quite frequently.

Table 19
Resolution data for metonymies

	Direct SRV		Ind. violation		No violation		Overall
Anaphora	(14/18)	77.8%	(19/22)	86.4%	(16/18)	88.9%	84.5%
Other definite NPs	(6/10)	60.0%	(5/7)	71.4%	(0/3)	0.0%	55.0%
Non-definites	(12/21)	57.1%	(1/1)	100.0%	(0/6)	0.0%	46.4%
Overall	(32/49)	65.3%	(25/30)	83.3%	(16/27)	59.3%	68.9%

Our approach copes successfully with anaphoric metonymies because it takes into account the explicit interaction of metonymy and anaphora resolution, which leads to a greater independence from SRVs. This holds even for an incremental approach. Our resolution rates for non-anaphoric metonymies are certainly not satisfactory—the main reason being that we are then fully dependent on SRVs and that the `Path Finder` has no access to discourse information.

We frankly admit that this evaluation is but one step towards more sophisticated experiments (both in method and scope). Nevertheless, it is one of the first evaluations at all that deal with metonymy resolution for real-world texts. Although Stallard [53] and Harabagiu [29] evaluate their approaches, too, they do not handle the issue of occurrences of semantic violations. In addition, they do not give general data on how many metonymies are resolved correctly by their approach (Harabagiu only gives these data for anaphoric metonymies; Stallard only mentions that semantic analysis is hugely improved by metonymy resolution, but does not give explicit numbers on how many metonymies are resolved correctly). Our evaluation, however, shares the following drawbacks with the other evaluations:

- All evaluations are highly domain- and/or genre-dependent as well as small in scope. Investigations across corpora are still missing, too.
- The methodology of determining metonymies and their correct readings is dependent on the intuitions of the researchers. No generally accepted annotation scheme exists as yet. This issue is normally not even discussed nor are any tests for detecting metonymies offered apart from detecting SRVs (which only cover less than half of all metonymies). In this respect, we have offered a way forward via replacement tests starting from accepted schemata.
- The issue of false positives is normally not touched upon at all. We have given a small evaluation of false positives in Hahn and Markert [25].

6. Related work

Research on metonymy resolution has mainly been pursued in two communities. One group of researchers rely on 'deep' conceptual, language-independent knowledge repositories, typically a sophisticated knowledge base with powerful inferencing capabilities—we refer to this work as *knowledge-based paradigm* (cf. Section 6.1). Another major stream of work has its roots in linguistics. Many of the foundational descriptive insights into the nature of figurative language, e.g., by Stern [54], Apresjan [2], Lakoff and Johnson [35], Bonhomme [3], lack computational orientation, however. Those with a more technical background assume that the knowledge required for metonymy interpretation is best captured in the lexicon as metonymies also have many language-dependent properties—we refer to this approach as the *lexicon-based paradigm* (cf. Section 6.2).

6.1. Knowledge-based approaches to metonymy interpretation

Knowledge-based approaches to metonymy interpretation assume that, given some sort of knowledge representation format (e.g., predicate calculus formulae, semantic networks, frames), a conceptual relation can be established between the metonymic expression and its underlying referent as part of the process of semantic or conceptual interpretation. Such a relation is usually identified by some mechanism searching the underlying knowledge base. Aside from these general commonalities relating to Principle 1, our approach can be contrasted with earlier work in the knowledge-based paradigm in the following way:

- All other knowledge-based approaches treat only metonymies that *violate* sentence-internal syntactic or semantic well-formedness *constraints*. Problems emerging from such an approach have been detailed in Section 2.1.
- Procedures for metonymy resolution usually do not take the *discourse context* of metonymies into account. This is, by definition, true for studies that are intentionally limited to the understanding of single utterances [1,4,18,24,32,53]. It is, nevertheless, also true for systems that clearly aim at text understanding [30,40]. As metonymies and different kinds of anaphors are dealt with separately, they cannot account for the systematic *interactions* between these phenomena we have been arguing for in Sections 2.2 and 4.1 (for an exception, cf. Harabagiu [29]).
- Most often the underlying parsing and semantic interpretation components operate in a *non-incremental* way. Metonymy resolution then requires a complete syntactic analysis of the entire utterance [4,24,29,30,32,40] and/or some kind of semantic annotation or preprocessing [1,30,40]. The main exception here is the work by Dan Fass [16–18].
- Few researchers provide an *empirical evaluation* of their metonymy resolution procedures. Bouaud et al. [4] give some concrete data with respect to the over-all semantic interpretation rate of their system, but do not supply specific data on the correctness of metonymy resolution proper. A similar criticism applies to Stallard [53]. A case study of the contribution of metonymy resolution to the task of reference resolution was conducted by Harabagiu [29], building on our anaphora and metonymy interaction model [38]. The data indicate that metonymies occur at a rate of 16.3% for anaphoric nominal expressions. The precision (76%) and recall data (83%) reported are hard to interpret but seem to indicate a success rate of the resolution of *anaphora* based on previous metonymy resolution, not of metonymy resolution in general.
- The issue of *knowledge base granularity* is usually not recognized as an important problem at all. An exception is the work by Bouaud et al. [4], though they do not distinguish between problems arising from granularity and the analysis of metonymies proper.
- Unlike our own efforts, some researchers have already successfully integrated the resolution of *metaphors* into their approaches [18,32,40].

In the following, we supplement these general observations with discussions of specific approaches to metonymy resolution.

TEAM. The TEAM system [24], a domain-independent natural language interface to data base systems, can be considered as one of the earliest approaches that deals with metonymy resolution in a computational setting. TEAM does not supply general metonymic schemata (apart from the `Name-for-Object` schema). Instead, the knowledge engineer who customizes TEAM to a particular domain decides whether specific instances of concept names in the relation tables of the underlying databases may stand metonymically for other specific concepts. So, given a data base for cars, a typical question addressed to the knowledge engineer might be: "*Will you be able to say* **Fords** *to* *mean* **cars whose car-manufacturer is Ford**?" (example from Section 3.4.2 in [24]). Only those specific concept pairs acknowledged by the knowledge engineer are allowed for subsequent metonymic interpretation. With this domain-specific acquisition mode, highly domain-dependent metonymies can be accounted for. However, just enumerating special metonymies lacks generality and does not explain the phenomenon. Not surprisingly, the TEAM system is unable to cope with any form of recursion or innovation in metonymy use, as well as any effects due to the context in which an utterance is made.

Metonymy recognition is triggered in TEAM only upon the occurrence of syntactic deviances (proper names occurring with an indefinite article) or SRVs. Within the knowledge-based paradigm, the TEAM approach is unique in the sense that it explicitly exploits syntactic deviances for metonymy recognition. How metonymy interpretation is actually achieved, and how TEAM deals with ambiguous metonymic readings is, unfortunately, not discussed.

met and Metallel.* The research of Dan Fass [16–18] constitutes one of the classics in metonymy and metaphor resolution. The structure of his resolution algorithm is as follows: After the syntactic component has detected a subject-verb, object-verb, or adjective-noun construction, the MET* algorithm for conceptual interpretation is activated—hence, we are dealing here with an incremental method of metonymy interpretation. The conceptual correlates of the lexical items encountered (so-called sense frames) are tried for conceptual relatability. The system, first, tries to find a literal interpretation that obeys all required conceptual constraints. Only if no such reading can be determined, metonymy resolution is triggered. Hence, the resolution of metonymies is made dependent on the recognition of SRVs. Their range is highly constrained, since only selectional restrictions of subjects and objects for verbs and noun-modifying adjectives are taken into consideration. Furthermore, possible problems of an incremental approach such as those resulting from indirect semantic violations are discussed but not covered by the algorithm (see Section 6 in [17]). Metonymy interpretation attempts to repair SRVs by (possibly recursively) applying metonymic inference rules (corresponding to what we call metonymic schemata). If the violations cannot be remedied by assuming a schematized metonymy, metaphor interpretation takes over. Therefore non-schematized metonymies are disregarded.

The METALLEL system [32] extends MET* by incorporating metonymies and metaphors indicated by *indirect* semantic violations. The intention is to generate a globally coherent interpretation of the entire utterance on the basis of conceptual interpretations of individual syntactic relationships, which have paths in the corresponding knowledge base as their semantic counterparts. This approach diverges from Fass's original work in two ways. First, the incrementality of MET* is sacrificed in favor of parallel local computations (interpretations of syntactic relationships), whose mutual combinability gets evaluated in a subsequent interpretation step at the end of the utterance, i.e., no local disambiguations occur. Second, the (direct) SRV hypothesis is abandoned. *All* local literal, metonymic and metaphoric interpretations are computed *in parallel* and distinguished via different path patterns.

While the latter property of the METALLEL system is similar to our approach, there are nonetheless considerable differences regarding tractability and disambiguation strategies. In the spreading-activation-based path search algorithm [32] propose, almost no search restrictions (apart from syntactic preferences) are incorporated. Furthermore, some specific concept pairs are *manually* excluded from being metonymic for each other (see p. 488 in [32], where the authors also concede that this approach "can at times seem arbitrary"). In addition, pursual of subsumption relations with unconditioned specialization is granted. Thus, in the case of metaphor they discuss (cf. Example (24)), a path between CAR and DRINK is computed, which (in a longer form) expresses the following relations (other paths including the correct one are found as well): CAR is a subconcept of PHYSICAL-OBJECT, which is, in turn, a superconcept of ANIMAL, which can be an *Agent* of DRINKing. While one might already call in question the *prima facie* plausibility of such a path pattern, what is really strange is that the distinction criteria [32] propose for such path patterns lead them to claim that this reading is actually a *literal* one. Since postulating the existence of a literal reading is contrary to semantic evidence, we think that their path patterns might not be restrictive enough.

(24) "The car **drank** gasoline."

The disambiguation between alternative readings depends on similarity considerations of the determined paths with already existing knowledge structures in the underlying knowledge base (so-called *grounding*). The criteria which lead to grounding are not made fully explicit. In the above example, *both* the described literal and the correct metaphorical path license grounding (the authors argue, p. 486), which leads us to suspect that grounding is not a very restrictive disambiguation method. Additional *ad hoc* measures (such as path length considerations) are introduced for disambiguation purposes. Thus, in Example (24) the literal reading is eventually discarded because its path length is '9', whereas the metaphoric one has length '8'. This method is, however, highly dependent on knowledge base granularity.

Conceptual graphs. Inspired by early work of John Sowa [51], several procedures for metonymy interpretation have been proposed within the *Conceptual Graph* approach, e.g., Amghar et al. [1], whose approach is very similar to Fass's approach. We here focus on the one developed within the MENELAS project [4], basically a syntax-first approach—the parsing component produces a syntactic parse tree which gets transformed into a Conceptual Graph tree during semantic interpretation. The replacement of syntactic relations by conceptual ones relies on computations of a path search algorithm, which distinguishes between so-called *Concept Fusion*, *Concept Inclusion* and *Model Join* paths. The latter include cases of metonymy, although they are not systematically distinguished from other paths in *Model Join*. Search criteria for paths incorporate syntactic preferences and a preference for conceptual specificity in the identified paths. Disambiguation seemingly relies upon disregarding metonymic paths when paths like *Concept Fusion* or *Concept Inclusion* (corresponding approximately to literal paths) exist (hence, an SRV approach is adopted). Disambiguation is extended by grammatical preferences and path length conditions. Unlike other researchers, Bouaud et al. [4] are aware of granularity effects on metonymy resolution rooted in the encoding of particular knowledge bases. They do not distinguish, however, granularity phenomena from metonymies in their algorithmic schema.

Faustus. The FAUSTUS system focuses on inference-based text understanding [39,40], its inferencing mode being based on marker passing. Marker collisions indicate possible inferences. The six different inference types being proposed include reference resolution and the so-called view application, which is crucial for the analysis of metonymies and metaphors (therefore handling a wider range of problems than we do here). The different inference types are distinguished by formal path patterns (mainly distinguishing between subsumption links and other relations), which direct the search through the underlying semantic network. Although Norvig argues passionately for the use of path patterns (instead of marker energy) to guide the search in a knowledge base, the definitions of path patterns he gives are not restrictive enough. Additional rules for coping with combinatorial search problems (e.g., an antipromiscuity rule excluding a path passing concepts that subsume too many concepts) still have to be supplied and the choice of particular numerical thresholds to cut off computation processes is not made explicit. The path patterns we propose are complemented by structural criteria which do not rely upon numerical restrictions in any way. The cyclicity criterion (Definition 3.3), e.g., increases the degree of path length (and thus granularity) independence of the search procedure.

Norvig exploits two criteria for the resolution of metonymies. First, the concepts involved *must* be related by one of the specific, prespecified *view links*, corresponding approximately to schemata. This obviously prevents FAUSTUS from dealing with innovative metonymies. Second, a figurative interpretation is only triggered when an SRV has been identified [40, p. 609f.]. After all possible inferences have been computed their assessment is started in order to determine the most plausible one. This evaluation process is internal to each particular type of inference, e.g., choosing amongst all possible figurative interpretations of a word or between all possible antecedents of an anaphor. This precludes an analysis of utterances where, e.g., anaphoric and metonymic readings interact with each other, or only an inference of one type (e.g., a metonymic reading) allows to draw an inference of another type (e.g., an anaphoric reading).

Tacitus. The TACITUS text understanding system [30] determines a logical representation for natural language input and then proves this representation via abductive reasoning. Reasoning by abduction tries to generate the "best" explanation relative to what is *a priori* known in the underlying knowledge base. Hence, different interpretations or proofs, including those that derive from metonymy resolution, usually compete for being superior. The best explanation is always achieved by a proof that requires the least effort, the *minimal proof*. The expenditures for a proof are measured in terms of a cost model, assigning each single proof step a numerical cost factor.

The encoding of the logical representations in TACITUS allow literal language to be considered a special case of metonymic language. However, as metonymy is not the main emphasis of the approach, it is handled in a fairly restrictive manner: metonymy resolution is only triggered when a semantic violation has been identified [30, p. 79].[27] In addition, metonymy resolution is limited to a set of specific relations, e.g., of the *Part-of* type.

In addition, metonymy resolution in TACITUS does not take the discourse context into account. In effect, TACITUS implements a clear priority regime such that reference resolution always precedes metonymy resolution [30, p. 128], which precludes referential

[27] The range of violations being covered is not made explicit in the article. From the richness of the logical representation used one may assume that, at least, all direct SRVs are covered.

metonymic anaphors from being resolved at all (they require metonymy resolution to precede anaphora resolution, cf. Section 4.1).

It is interesting to explore in what respect the SRV assumption is inherent in the approach and the principles of its cost model. As an example, consider the principle of *Factoring*, which targets at redundancies in the logical form. In a sentence such as "(*I*) *liked the laser*" a *Patient* of "*like*" has to be proved and the existence of an object of the type "*laser*" has to be proved, as well. Factoring states that it is less costly to assume these two objects to be identical (as no type conflict occurs) and only prove the existence of one of them. Therefore, a metonymy is never under consideration, if the actual argument already fulfils the selectional restrictions imposed by the predicate. Otherwise, the existence of the argument *and* the one of a non-identical underlying referent had to be proved. Factoring is not explicitly applied to an example of metonymy resolution in the paper, but *is* stated as a general, overall principle—our reasoning that applies it to metonymy resolution seems to show that it might be too restrictive for some inference types. We do not doubt the general usefulness of the abduction principle for metonymy resolution but rather focus on inadequate minimality principles for choosing amongst alternative abductive proofs.

WordNet. All the approaches discussed up to now use a conceptually rich knowledge base with limited coverage. An interesting idea to pursue is the use of large-scale lexico-semantic repositories for metonymy interpretation. Sanda Harabagiu [29] has developed a methodology for deriving metonymic coercions from WORDNET that combines its lexico-semantic relations with semantic information derived from conceptual definitions (the WORDNET glosses). Three phases are distinguished: (1) the approximation of selectional restrictions that need to be satisfied during the interpretation of nominal expressions, (2) the retrieval of related knowledge that complies with the identified selectional restrictions, (3) the validation of metonymic expressions against anaphoric ones from the sentence *following* the processed sentence. The first two phases rely on knowledge extracted from WORDNET and its automatic translation into a logical representation structure. In phase (2), a search through the WORDNET database is conducted that checks all retrievable synonyms, hypernyms and genera (in glosses) of nouns that can be related to the particular syntactic role of the currently considered verb. If the search succeeds, a literal reading for the noun under scrutiny is encountered. Metonymic readings are identified, if a literal one is not possible and if a relation pattern can be found that incorporates at least one occurrence of *is-part*, *is-member* or *is-stuff*. Hence, this work follows a strict SRV approach.

Harabagiu replicates our main claim related to the *discourse embedding* of metonymy resolution (based on an earlier version in [38]) by additional empirical data. The case study for validating her approach was conducted within the coreference task for the MUC-7 competition. A brief evaluation is reported that is based on the analysis of twenty NYT articles. Harabagiu's data indicate that 23% from 1261 nominal expressions in these documents are anaphoric, with literal meaning assigned to 74.1% and metonymic readings assigned to 16.8% of these anaphors. The contribution of metonymy resolution to the task of reference resolution is on the order of 16.3%.

6.2. Lexicon-based approaches to metonymy interpretation

There are reasonable arguments that a purely conceptual account of metonymies might not be fully adequate. Firstly, knowledge-based inference types (often incorporating metonymic schemata) are often *semi-productive* [13]. Let us consider *productivity* as the ratio of the number of lexical items to which a schema is applicable, in principle, and the number of lexical items the schemata is actually applied to [5,11]. Then, e.g., the `Producer-for-Product` schema is highly productive as almost all names of producers can stand for one of their products. This is certainly not true for `Part-for-Whole` metonymies (see also Section 2.3). Many of these "gaps" in the exploitation of schemata do not easily fit a conceptual constraint model but are instead language-specific. Nunberg [42], e.g., notes that whole subdomains can be excluded from schemata and that these excluded domains *vary from language to language* for the *same* schema. We already noted in Section 2.3 schema exceptions due to "blocking" [7]. Secondly, interactions of metonymic readings with morphological properties of the lexical item or its phrasal context also occur. So, we may encounter a metonymic use of a lexical item with the definite article, while the metonymic use with another determiner is precluded as illustrated by an example of a logical metonymy taken from Verspoor [59]:

(25) "John began **the cheese**" versus *"John began **some cheese**."

Other forms of interaction are changes in the use of the article or the plural (e.g., the indefinite article or plural may indicate a metonymic interpretation of a proper name as in "*a Quantum*" or "*two BMWs*" [3]), or changes in countability (e.g., "*I like **wine***" versus "*I like **a wine***" [12]). Interactions of this sort are especially frequent for conventional metonymies.

These examples stress the relevance of linguistic criteria as an additional set of constraints. Nevertheless, we rank these constraints as secondary and as supplementing the primary conceptual ones, since the main prerequisite for understanding a metonymic utterance is *always* the determination of a conceptual/semantic relation between the items involved, no matter whether we are dealing with conventionalized or innovative metonymies (cf. also Nunberg [42, p. 119] for supporting this view). A conceptual approach provides the adequate representational setting for the formulation of the major operational and pragmatic principles underlying metonymies. It allows us to formulate language-independent constraints to explain and resolve metonymies. Postulating a *language-independent* representation level for the basic regularities underlying metonymy interpretation has also a relieving effect on the lexicon, which then houses the *language-dependent* rules or even entirely idiosyncratic exceptions (as cases of blocking). This way, a clear division of the workload is achieved. In addition, some aspects of semi-productivity can be explained by conceptual aptness conditions [25]. Nunberg [41] also introduces a pragmatic criterion for assessing the plausibility of a metonymic reading that bears some resemblance to the uniqueness condition discussed by Hahn and Markert [25].

In the following, we will discuss *the* dominating lexicon-based approach in more detail, *viz.* Pustejovsky's *generative lexicon* [44–47]. Related lexical approaches by other researchers, especially by Copestake, Briscoe and Lascarides, which share the fundamental idea of working with a lexicon enhanced with semantic roles, will be discussed as far as they offer extensions interesting for metonymy resolution.

In a generative lexicon, lexical entries not only contain morphosyntactic features but are organized in type hierarchies and are composed of *qualia structures*, which specify four aspects of their meaning: the relation between the lexical entry and its constituent parts (the constitutive role), that which distinguishes it within a larger domain, its physical characteristics (the formal role), its purpose and function (the telic role), and whatever brings it about (its agentive role) [44, p. 418]. So, the qualia structure for a *novel* specifies that it has the purpose of being read (in the telic role) and that it comes about by writing (in the agentive role).

Lexical rules (corresponding to schemata) allow the generation of metonymic meanings of a lexical item based on its roles in the qualia structure. If a metonymy is encountered such as in Example (15) (rephrased below as Example (26)), the qualia structure (especially the telic and agentive role) of the lexical entry "*novel*" is checked for substituting "*novel*" such that it fit the requirements by the verb "*began*". Substitutions of this kind may lead to an ambiguous replacement of the form "*John began reading the novel*" relating to information from its telic role, or "*John began writing the novel*" relating to information from its agentive role.

(26) "John began **the novel**."

So far, this approach shares a lot of similarities with a knowledge-based approach: Metonymic readings are not enumerated, but derived from a canonical representation structure (here contained in the lexicon, in our approach in the knowledge base) via a search procedure (here operating on the qualia structures in the lexicon, in our approach on the concepts and their relations in the knowledge base). Fundamental differences to our approach are considered below. Some are due to a lexicon-based approach and a focus on restricting overgeneration, which sets them apart from all knowledge-based approaches, in general; others, especially the lack of discourse integration and the restriction to metonymies violating SRVs, set them apart from our approach, in particular.

1. The focus of most lexicon-based approaches to metonymy interpretation is on *declarative* specification issues (lexicon and rule design), usually without a deeper concern for *processing* considerations. With the exception of Verspoor [58] and Lascarides and Copestake [37], no algorithms are presented for sentence or text analysis, which account for metonymic phenomena.
2. Rule ordering and the disambiguation between literal and metonymic or several metonymic readings are dealt with only sketchily in many approaches. This holds especially for the original work by Pustejovsky:

   > "It is not the role of a lexical semantic theory to say what readings are preferred, but rather which are available."
   >
   > (Pustejovsky [44, p. 430])

   In most papers, only one criterion for rule application and disambiguation is mentioned, namely that metonymic readings should only be generated if the literal one violates syntactic constraints or exhibits SRVs (cf., e.g., Pustejovsky [44, p. 425] or Copestake and Briscoe [12, p. 117]). Therefore, they suffer from all the problems brought up in Section 2.1.

3. An interesting proposal for dealing with the problem of productivity of lexical rules, which, if realized for metonymies, has also consequences for disambiguation and rule application is due to Briscoe and Copestake [5]. They propose the use of *probabilistic lexical rules* in order to estimate the probability of single derivable meanings of a lexical item based on relative frequencies of these meanings in a given text corpus. More frequent meanings of a lexical item are then preferred over less frequent ones, if the more frequent ones do not violate intrasentential constraints. Since the literal meaning might not necessarily be the most frequent one, metonymic readings can then be preferred over literal ones. Also, more productive lexical rules are preferred over less productive ones, leading to a disambiguation strategy for several metonymic readings. In addition, this model easily incorporates cases of morphological blocking, since "blocked" meanings do not or only rarely occur in the corpus. Though compelling from a methodological point of view, Briscoe and Copestake's work suffers from a lack of evaluation and we are not aware that it has been applied to metonymy resolution in practice up to now. Furthermore, the proposed probability criterion takes only the a priori frequency of single lexical items into account and disregards contextual factors.
4. Most of the research concentrates on conventional metonymies, since they can be reconstructed by recourse to the qualia structures. Innovative metonymies can therefore usually not be handled. [28] Briscoe et al. [6] and Lascarides and Copestake [37], however, note that discourse considerations may overrule conventional metonymic readings originating from the lexical component. Consider the following example, adapted from Lascarides and Copestake [37]:

   (27) "John is my pet goat. He loves eating things. He enjoyed **the book**."

   In this case the *writing/reading* interpretation derived from the qualia structure of "*book*" should be discarded in favor of the *eating* reading. Verspoor [58] and Lascarides and Copestake [37] present an algorithm to deal with such phenomena for logical metonymies. After discovering a syntactic or semantic constraint violation the conventional *reading/writing* metonymic reading is derived from the lexicon. After anaphora resolution (*he = John, the goat*), the *writing/reading* assumption is void, since goats cannot read or write. The *eating* hypothesis is then strongly supported by the discourse context, allowing cohesion by establishing a rhetorical relation [36], whereas other hypotheses only lead to what the authors call "weak" coherence. While favoring one reading over another on account of its support for discourse coherence is similar to our approach, many differences remain:

   - Metonymy resolution in Verspoor [58] and Lascarides and Copestake [37] builds on a logical representation of the entire utterance. Hence, it is non-incremental.

[28] Verspoor [59] and Briscoe et al. [6] give data for logical metonymies indicating that conventional metonymies are much more frequent than innovative ones, so that the coverage for logical metonymies, at least, is not greatly impaired by not covering innovative metonymies.

- The resolution of metonymies in Verspoor [58] and Lascarides and Copestake [37] is still dependent on SRVs or syntactic violations. Discourse criteria only come into play for choosing among different metonymic readings, not between literal and metonymic one(s). Metonymies whose recognition is fully dependent on the discourse context cannot be treated this way.
- The interaction mode we have outlined above relates to metonymies and rhetorical relations, not to metonymies and anaphors, thereby making the two approaches complementary. This is evident from the observation that the recognition of the innovative metonymy in Example (27) requires anaphora resolution between "*he*" and "*John*" to precede metonymy interpretation. Referential metonymic anaphors that require metonymy resolution prior to anaphor resolution, therefore, are not handled by this approach.
- This approach is restricted to logical metonymies and some similar metonymies within adjective-noun constructions. Whereas the interaction between anaphora and metonymies that we use is very frequent (see Section 5.1) metonymies such as Example (27) are very rare—so only 4.9% of the logical metonymies in the BNC by the verb "*begin*" are determined by the context [59]. As we, on the other hand, cannot handle logical metonymies, the two approaches are—again—complementary.

5. A major concern of lexicon-based approaches are specifications intended to prevent overgeneration by systematically constraining lexical rules, an issue almost neglected in knowledge-based approaches. Single studies then deal with very specific lexical examples in a very detailed way, often at the price of a reasonable coverage and methodological generalization. For instance, Godard and Jayer [23] discuss various constraints on logical metonymies for the French verb "*commencer*", Pustejovsky and Bouillon [47] consider aspectual verbs and their associated restrictions on logical metonymies.

We, finally, recognize a fundamental difference of research goals for both approaches rendering them complementary. Lexicon-based approaches aim primarlily at questions of the organization of the lexicon, the representation of lexical rules and language-specific constraints they are associated with. The focus of knowledge-based approaches, ours in particular, is more on adequacy criteria for metonymies that are grounded in discourse or domain knowledge. The latter are much coarser than the constraints provided for lexicons but they are far more general. Hence, we may refine these specifications by lexically based ones in order to add discriminative power to the constraint system underlying our metonymy resolution process.

7. Conclusions and outlook

We have presented an approach to metonymy resolution that is rooted in conceptual representation structures of a domain knowledge base. From a processing perspective, we do not consider literal language as being prioritized relative to (conventional) figurative language. Rather we propose a computation model in which literal and metonymic readings are determined independently of each other. Thus, metonymic readings need not violate semantic restrictions in order to be computed at all. Another major aspect of our work concerns the incorporation of the discourse context, in particular, the heavy interaction of metonymies and anaphors. These two design decisions together extend the coverage of metonymies by about 15% for anaphoric metonymies as well as increase the anaphora resolution capacity on the order of about 16%. In addition, they allow metonymy resolution to be incorporated into an incremental processing model (as well as into a serial one) without an excessive need for backtracking on the one hand or an excessive proliferation of ambiguities on the other hand.

Despite its increased coverage our approach still has problems with two kinds of metonymies. Firstly, it cannot recognise non-anaphoric metonymies without SRVs (see Example (3)). In these cases we are able to compute literal as well as metonymic readings via the `Path Finder` but our disambiguation strategy then follows the line common to previous approaches, namely of discarding the metonymic interpretation. Our evaluation has shown this to be wrong in many cases (e.g., 7 (25%) of 28 non-definite metonymies in our data were not marked by an SRV). Non-definite metonymic noun phrases without SRVs seem to be substantially different, though, from the definite metonymic noun phrases without SRVs. Whereas the definite ones were mostly anaphoric and therefore recognisable by incorporating discourse constraints, the non-definite ones mostly require extensive pragmatic reasoning (see p. 38), so that our disambiguation method cannot be readily extended towards non-definite metonymies.

Secondly, our algorithm cannot handle cases of metonymies where syntactic or morphological analysis already fails, e.g., variants of logical metonymy. In those cases the semantic interpretation mechanism including metonymy resolution is not triggered, although semantic interpretation as such is not precluded from handling such phenomena. This problem needs a refinement of the language-oriented specifications of our system, viz. the grammar and lexicon, not of the conceptual and semantic mechanisms we have detailed in this paper. Thus we should also trigger metonymy interpretation in some specific cases of failure of the syntactic analysis. Obviously this class of phenomena has to be defined very restrictively as syntactic analysis is an important filter *before* conceptual interpretation in most cases. We suggest the following cases:

- Syntactic linkage between a number of verbs, e.g., "begin, enjoy, finish" and their direct objects, if these are nouns.
- Some morphological phenomena, e.g., the plural of proper names that are normally restricted to the singular ("2 *BMWs*").

In theory, disregarding language-specific metonymy constraints can also lead to overgeneration as, e.g., cases of blocking cannot be recognised. In practice, this has not been a problem and we have actually not encountered any case in our text analysis yet, where we produced a metonymic interpretation, which should not be licensed because of language-specific properties. [29] We expect this is due to the fact that we deal with the *analysis* of natural language texts, which normally do not contain non-licensed metonymies anyway. The problem is probably much more serious for natural language generation.

[29] We have of course had cases of false positives. But these were mainly due to overgeneration in the case of bridging phenomena, not to language-specific idiosyncracies (see Section 4).

Regarding evaluation, our approach has been evaluated by empirical data on a limited scale. Further work on annotation schemes for metonymies is badly needed to run evaluations on a larger scale. Ideally, one might want to set up ‘metonymy banks’ similar to parse tree banks—nothing like that exists so far, leading to a lack of principled evaluations for metonymy resolution. The authors are at the moment extracting such a metonymy bank from the British National Corpus.

Regarding prerequisites, rich knowledge structures—concepts and their conceptual relations—are needed for successful metonymy resolution. While our resolution mechanism does not depend on the relation hierarchies we introduced into our work, there is some dependence on the granularity of the knowledge base. We tried to minimize granularity issues by several abstraction mechanisms. While the quest for rich domain knowledge is certainly a bottleneck from a knowledge engineering view, it does not constitute a principled argument against our approach when compared with lexicon-oriented work. The specification of sufficiently expressive qualia structures is likely to consume the same amount of work than that required for the specification of knowledge bases. Less work is faced when one commits to shallower knowledge sources such as WORDNET. In such a case, however, constraint knowledge is much more restricted, though the lexical coverage is increased. Even a ‘knowledge-free’ approach to metonymy resolution can be thought of, with no costs involved to build up a knowledge base. Thus, Utiyama et al. [57] propose a statistical approach for Japanese metonymy *interpretation*, which is based on co-occurrence data from a Japanese corpus. However, it is hard to compare their results with ours because of three reasons. Firstly, their approach is evaluated on textbook examples only—the evaluation is also flawed as the authors count an interpretation as correct, if “it makes sense in some context”. Thus, for the metonymy “*Dave drank* ***the glasses***” they count the reading “*Dave drank coke*” as correct, although it is clearly overspecific and may be inconsistent with the discourse context in real-world texts. The necessary abstraction processes to a reading like “*Dave drank the liquid in the glasses*”, would probably again involve at least a semantic hierarchy. It is in addition not clear how well the method works in context. Secondly, the authors handle subject-verb and verb-object combinations only, exhibiting SRVs and verbs which put fairly strong selectional restrictions on their objects/subjects. Thus, the very problems we address (discourse-specific readings and metonymies without direct violations) are excluded. Thirdly, the authors do not handle metonymy recognition yet, which they take to be the harder problem (Utiyama, p.c.).

Regarding usability in different system contexts, our algorithm profits from its inherent modularity (see also Section 4.3 for the algorithm description). It is not necessary to subscribe entirely to the various decisions we have been arguing for. The algorithm structure makes no prescriptions related to the format and structure of the background knowledge or the particular discourse theory for determining anaphoric relationships although we have made specific decisions for our version of the algorithm. So, it is possible to determine literal and metonymic readings by using lexical structures instead of knowledge-based ones or by using conceptual graphs instead of description logics and then still use our disambiguation and discourse predicates on top of these.

An interesting question is how far our algorithm can be applied to other figurative language phenomena, notably metaphor. The computation of metaphoric readings is inherently different from the computation of metonymic readings as metaphor relies more on similarities between two concepts or instances than on other relationships between them as computed by the `Path Finder`. In addition, it is more often than metonymy not only a local phenomenon but spans a whole utterance. Thus, our model of computation cannot easily be transferred to metaphor resolution. We still have two contributions to make regarding disambiguation. Firstly, our approach to disambiguation between literal and metonymic language can be adopted in a straightforward way to referring metaphoric noun phrases (e.g., referring to a man as "the lion"). Here, also, a metaphoric reading allowing for anaphora resolution might be preferrable over a literal reading not allowing for anaphora resolution. Secondly, the *Principle of Referential Success* might also give clues for the disambiguation between metonymic and metaphoric readings. Thus, in Example (7) a metonymic reading of the noun *shirt* competes with a metaphoric reading of the verb *wait* (interpreting the noun literally at the same time). As the latter reading allows for anaphora resolution whereas the former does not, a metaphoric reading of the verb can be preferred. Incorporating this principle is in contrast to the work of Fass [18] who seemingly prefers metonymic readings strictly over metaphoric ones, leading to problems in the example mentioned.

Acknowledgements

K. Markert was funded by a Graduate Program *Human and Machine Intelligence* of the DFG when part of the research was carried out. We would also like to thank Michael Strube for his cooperation, as well as Bonnie Webber and two anonymous reviewers for their comments.

References

[1] T. Amghar, F. Gayrat, B. Levrat, Table 10 left without paying the bill!—A good reason to treat metonymy with conceptual graphs, in: G. Ellis, R.A. Levinson, W. Rich, J.F. Sowa (Eds.), Conceptual Structures: Applications, Implementation and Theory. Proc. 3rd Internat. Conference on Conceptual Structures (ICCS-95), Santa Cruz, CA, August 14–18, 1995, Lecture Notes in Artificial Intelligence, Vol. 954, Springer, Berlin, 1995, pp. 129–143.

[2] J. Apresjan, Regular polysemy, Linguistics 142 (1973) 5–32.

[3] M. Bonhomme, Linguistique de la Métonymie, Peter Lang, Bern, 1987.

[4] J. Bouaud, B. Bachimont, P. Zweigenbaum, Processing metonymy: A domain-model heuristic graph traversal approach, in: Proc. 16th Internat. Conference on Computational Linguistics (COLING-96), Copenhagen, Denmark, August 5–9, 1996, Vol. 1, 1996, pp. 137–142.

[5] T. Briscoe, A. Copestake, Lexical rules in constraint-based grammar, Comput. Linguistics 25 (4) (1999) 487–526.

[6] T. Briscoe, A. Copestake, B. Boguraev, Enjoy the paper: Lexical semantics via lexicology, in: Proc. 13th Internat. Conference on Computational Linguistics on the Occasion of the 25th Anniversary of COLING and the 350th Anniversary of Helsinki University (COLING-90), Helsinki, Finland, August 1990, Vol. 2, 1990, pp. 42–47.

[7] T. Briscoe, A. Copestake, A. Lascarides, Blocking, in: P. Dizier, E. Viegas (Eds.), Computational Lexical Semantics, Cambridge University Press, Cambridge, 1995.

[8] R. Chaffin, The concept of a semantic relation, in: A. Lehrer, E.F. Kittay (Eds.), Frames, Fields and Contrasts, Lawrence Erlbaum, Hillsdale, NJ, 1992, pp. 253–288.

[9] R. Chaffin, D.J. Herrmann, M. Winston, An empirical taxonomy of part-whole relations: Effects of part-whole relation type on relation identification, Language and Cognitive Processes 3 (1) (1988) 17–48.
[10] H.H. Clark, Bridging, in: R. Schank, B. Nash-Webber (Eds.), Proc. Conference on Theoretical Issues in Natural Language Processing, Cambridge, MA, 1975, pp. 169–174.
[11] T.C. Clausner, W. Croft, Productivity and schematicity in metaphors, Cognitive Sci. 21 (3) (1997) 247–282.
[12] A. Copestake, T. Briscoe, Lexical operations in a unification-based framework, in: J. Pustejovsky, S. Bergler (Eds.), Lexical Semantics and Knowledge Representation, Lecture Notes in Artificial Intelligence, Vol. 627, Springer, Berlin, 1992, pp. 101–119.
[13] A. Copestake, T. Briscoe, Semi-productive polysemy and sense extension, J. Semantics 12 (1) (1995) 15–67.
[14] S. Crain, M. Steedman, On not being led up the garden path: The use of context by the psychological syntax processor, in: D.R. Dowty, L. Karttunen, A.M. Zwicky (Eds.), Natural Language Parsing. Psychological, Computational, and Theoretical Perspectives, Cambridge University Press, Cambridge, 1985, pp. 320–358.
[15] D.A. Cruse, On the transitivity of the part-whole relation, J. Linguistics 15 (1979) 29–38.
[16] D.C. Fass, An account of coherence semantic relations, metonymy, and lexical ambiguity resolution, in: S.L. Small, G.W. Cottrell, M.K. Tanenhaus (Eds.), Lexical Ambiguity Resolution. Perspectives from Psycholinguistics, Neuropsychology and Artificial Intelligence, Morgan Kaufmann, San Mateo, CA, 1988, pp. 151–177.
[17] D.C. Fass, met*: A method for discriminating metonymy and metaphor by computer, Comput. Linguistics 17 (1) (1991) 49–90.
[18] D.C. Fass, Processing Metonymy and Metaphor, Contemporary Studies in Cognitive Science and Technology, Vol. 1, Ablex Publishing, Westport, CT, 1997.
[19] C. Fellbaum (Ed.), WORDNET: An Electronic Lexical Database, MIT Press, Cambridge, MA, 1998.
[20] K. Fraurud, Definiteness and the processing of noun phrases in natural discourse, J. Semantics 7 (1990) 395–433.
[21] R. Gibbs, The Poetics of Mind. Figurative Thought, Language and Understanding, Cambridge University Press, Cambridge, 1994.
[22] R. Gibbs, The fight over metaphor in thought and language, in: A.N. Katz, C. Cacciari, R.W. Gibbs, M. Turner (Eds.), Figurative Language and Thought, Oxford University Press, Oxford, 1998, pp. 88–118.
[23] D. Godard, J. Jayez, Towards a proper treatment of coercion phenomena, in: Proc. 5th Conference of the European Chapter of the Association for Computational Linguistics, Berlin, Germany, 1991, pp. 168–177.
[24] B.J. Grosz, D.E. Appelt, P.A. Martin, F.C.N. Pereira, TEAM: An experiment in the design of transportable natural-language interfaces, Artificial Intelligence 32 (2) (1987) 173–243.
[25] U. Hahn, K. Markert, In support of the equal rights movement for literal and figurative language: A parallel search and preferential choice model, in: M. Shafto, P. Langley (Eds.), Proc. 19th Internat. Conference on Cognitive Science (CogSci-97), Stanford, CA, Lawrence Erlbaum, Mahwah, NJ, 1997, pp. 289–294.
[26] U. Hahn, K. Markert, On the formal distinction between literal and figurative language, in: P. Barahona, J.J. Alferes (Eds.), Progress in Artificial Intelligence. Proc. 9th Portuguese Conference on Artificial Intelligence (EPIA-99), Evora, Portugal, September 21–24, 1999, Lecture Notes in Artificial Intelligence, Vol. 1695, Springer, Berlin, 1999, pp. 133–147.
[27] U. Hahn, K. Markert, M. Strube, A conceptual reasoning approach to textual ellipsis, in: W. Wahlster (Ed.), Proc. 12th European Conference on Artifical Intelligence (ECAI-96), Budapest, Hungary, August 11–16, 1996, John Wiley, Chichester, 1996, pp. 572–576.
[28] U. Hahn, M. Romacker, Content management in the SYNDIKATE system: How technical documents are automatically transformed to text knowledge bases, Data Knowledge Engrg. 35 (2) (2000) 137–159.
[29] S.M. Harabagiu, Deriving metonymic coercions from WORDNET, in: Proc. COLING-ACL-98 Workshop 'Usage of WordNet in Natural Language Processing Systems', Montreal, Quebec, August 16, 1998, Association for Computational Linguistics, New Brunswick, NJ, 1998, pp. 142–148.
[30] J.R. Hobbs, M.E. Stickel, D.E. Appelt, P. Martin, Interpretation as abduction, Artificial Intelligence 63 (1993) 69–142.
[31] M.N. Huhns, L.M. Stephens, Plausible inferencing using extended composition, in: Proc. IJCAI-89, Detroit, MI, Morgan Kaufmann, San Mateo, CA, 1989, pp. 1420–1425.
[32] E. Iverson, S. Helmreich, METALLEL: An integrated approach to non-literal phrase interpretation, Comput. Intelligence 8 (3) (1992) 477–493.

[33] S. Kamei, T. Wakao, Metonymy: Reassessment, survey of acceptability, and its treatment in a machine translation system, in: Proc. 30th Annual Meeting of the Association for Computational Linguistics, Newark, DE, 1992, pp. 309–311.

[34] G. Lakoff, Women, Fire, and Dangerous Things. What Categories Reveal about the Mind, University of Chicago Press, Chicago, IL, 1987.

[35] G. Lakoff, M. Johnson, Metaphors We Live By, University of Chicago Press, Chicago, IL, 1980.

[36] A. Lascarides, N. Asher, Discourse relations and defeasible knowledge, in: Proc. 29th Annual Meeting of the Association for Computational Linguistics, Berkeley, CA, 1991, pp. 55–62.

[37] A. Lascarides, A. Copestake, The pragmatics of word meaning, J. Linguistics 34 (2) (1998) 387–414.

[38] K. Markert, U. Hahn, On the interaction of metonymies and anaphora, in: Proc. IJCAI-97, Nagoya, Japan, Morgan Kaufmann, San Francisco, CA, 1997, pp. 1010–1015.

[39] P. Norvig, Inference in text understanding, in: Proc. AAAI-87, Seattle, WA, Morgan Kaufmann, Los Altos, CA, 1987, pp. 561–565.

[40] P. Norvig, Marker passing as a weak method for inferencing, Cognitive Sci. 13 (4) (1989) 569–620.

[41] G. Nunberg, The pragmatics of reference, PhD Thesis, City University of New York, 1978.

[42] G. Nunberg, Transfers of meaning, J. Semantics 12 (1995) 109–132.

[43] A. Ortony, Some psycholinguistic aspects of metaphor, in: R. Honeck, R. Hoffman (Eds.), Cognition and Figurative Language, Erlbaum Associates, 1980, pp. 69–83.

[44] J. Pustejovsky, The generative lexicon, Comput. Linguistics 17 (4) (1991) 409–441.

[45] J. Pustejovsky, The Generative Lexicon, MIT Press, Cambridge, MA, 1995.

[46] J. Pustejovsky, B. Boguraev, Lexical knowledge representation and natural language processing, Artificial Intelligence 63 (1993) 193–223.

[47] J. Pustejovsky, P. Bouillon, Aspectual coercion and logical polysemy, J. Semantics 12 (1995) 133–162.

[48] P. Resnik, Using information content to evaluate semantic similarity in a taxonomy, in: Proc. IJCAI-95, Montreal, Quebec, Morgan Kaufmann, San Mateo, CA, 1995, pp. 448–453.

[49] P. Resnik, Selectional preference and sense disambiguation, in: Proc. ACL/SIGLEX Workshop on 'Tagging Text with Lexical Semantics: Why, What and How', Washington, DC, 1997.

[50] M. Romacker, K. Markert, U. Hahn, Lean semantic interpretation, in: Proc. IJCAI-99, Stockholm, Sweden, Morgan Kaufmann, San Francisco, CA, 1999, pp. 868–875.

[51] J.F. Sowa, Logical structures in the lexicon, in: J. Pustejovsky, S. Bergler (Eds.), Lexical Semantics and Knowledge Representation, Lecture Notes in Artificial Intelligence, Vol. 627, Springer, Berlin, 1992, pp. 39–60.

[52] S. Staab, U. Hahn, Comparatives in context, in: Proc. AAAI-97/IAAI-97, Providence, RI, AAAI Press, Menlo Park, CA/MIT Press, Cambridge, MA, 1997, pp. 616–621.

[53] D. Stallard, Two kinds of metonymy, in: Proc. 31st Annual Meeting of the Association for Computational Linguistics, Columbus, OH, 1993, pp. 87–94.

[54] G. Stern (Ed.), Meaning and Change of Meaning, Wettergren & Kerbers Förlag, Göteborg, 1931.

[55] M. Strube, U. Hahn, PARSETALK about sentence- and text-level anaphora, in: Proc. 7th Conference of the European Chapter of the Association for Computational Linguistics (EACL-95), Dublin, Ireland, 1995, pp. 237–244.

[56] M. Strube, U. Hahn, Functional centering: Grounding referential coherence in information structure, Comput. Linguistics 25 (3) (1999) 309–344.

[57] M. Utiyama, M. Murata, H. Isahara, A statistical approach to the processing of metonymy, in: Proc. 18th Internat. Conference on Computational Linguistics (COLING-2000), Saarbrücken, Germany, Morgan Kaufmann, San Francisco, CA, 2000, pp. 885–891.

[58] C. Verspoor, Lexical limits on the influence of context, in: G.W. Cottrell (Ed.), Proc. 18th Internat. Conference on Cognitive Science (CogSci-96), San Diego, La Jolla, CA, Lawrence Erlbaum, Mahwah, NJ, 1996, pp. 116–120.

[59] C. Verspoor, Conventionality-governed logical metonymy, in: H. Bunt, L. Kievit, R. Muskens, N. Verlinden (Eds.), Proc. 2nd Internat. Workshop on Computational Semantics (IWCS II), Tilburg, The Netherlands, 1997, pp. 300–312.

[60] W.A. Woods, J.G. Schmolze, The KL-ONE family, Comput. Math. Appl. 23 (1992) 133–177.