The Pragmatics of Word Meaning

Alex Lascarides
Centre for Cognitive Science,
University of Edinburgh
alex@cogsci.ed.ac.uk

1 Introduction

A parsimonious lexicon must encode generalisations (e.g., [9]). One then needs to reason about when these apply. A general consensus is that an operation known as *default inheritance* is useful for this ([2, 4, 8, 10, 11], and others). A frequent motivation for using it, is to capture the overriding of regularities by subregularities in a computationally efficient manner. Information need only be stated once, instead of many times in each separate word, and default inheritance ensures that words inherit the right information.

But there's a problem with this. Many lexical generalisations are of the sort where there are exceptions to the rules, which are triggered by information which resides outside the lexicon. In particular, pragmatic knowledge can trigger exceptions and default inheritance doesn't communicate properly with pragmatics to encode this.

In this paper, we'll consider three examples where this occurs: logical metonymy (e.g., *enjoy the book* means *enjoy reading the book*), adjectives (e.g., the interpretation of *fast* in *fast car*, *fast motorway*, *fast typist* etc.), and noun-verb agreement. We'll argue for a new version of default inheritance, which allows default results of lexical generalisations to *persist* as default beyond the lexicon. We'll show that this persistence can be exploited by the pragmatic component, to reason about when generalisations encoded in the lexicon survive in a discourse context. We'll represent the link between the lexicon and pragmatics via two axioms. These will predict the pragmatic exceptions to lexical generalisations that arise in a discourse context. We thereby explain how words are interpreted in discourse, in a way that neither the lexicon nor pragmatics could achieve on their own.

2 Generalisations with Exceptions

Briscoe *et al* [4] and Copestake and Briscoe [7], among others, show how to encode certain aspects of metonymy (e.g., (1a) means (1b)) via default unification; one of the popular methods for implementing default inheritance.

(1) a. John enjoyed the book.
b. John enjoyed reading the book.

Here, the lexical generalisation is: when *enjoy* takes an artifact as an object, then the event that *enjoy* predicates over is determined by the default telic role, or purpose, of that artifact. Telic roles are conventionalised in the lexicon, as part of the *qualia* for the lexical entry. The *qualia* represents properties of the artifact, such as what it's made of, what one does with it, and so on. So the lexical entry for *book* includes a path QUALIA:TELIC:PRED:**read**, because the default telic role of books is to be read. Therefore, when the generalisation concerning *enjoy* is specified in the lexicon, default inheritance predicts that *enjoy the book* means *enjoy reading the book*. Default inheritance also predicts that the same entry for *enjoy* in *enjoy the film* yields *enjoy seeing the film*, because the default telic role of films is to be seen.

Briscoe *et al* [4, 7] argue for conventionalising certain aspects of metonymy (but see [16] for an alternative view): for example, (2) is strange, even if the doorstop is a book:

(2) ?John enjoyed the doorstop.

But the above generalisation about *enjoy* has exceptions which are triggered by pragmatic knowledge, such as information about the domain. (3a) means (3b).

(3) a. The goat enjoyed the book.
b. The goat enjoyed eating the book.
c. The goat enjoyed reading the book.

But if one encodes metonymy using only default unification in the lexicon, then one predicts that (3a), like (1a), means (3c). This is because the operation cannot see information—like domain knowledge that *goats don't read*—that resides outside the lexicon. Nor does it communicate to the other components of the grammar which generalisations can have exceptions, and which can't. So the pragmatic component won't be able to override the result of default unification in (3a), because by this stage, it's not marked as defeasible.

There are three ways in which one might preserve the existing default unification account of logical metonymy, while interpreting (3a) correctly. First, one could assume

selectional restrictions on read; if the agents of reading events must be human, then default unification would detect the conflict between the agent *the goat*, and the expansion of the metonymy via the telic role of books. However, it's not viable to assume that this selectional restriction is non-default: sentences such as *The goat read the book* aren't ungrammatical. So one would have to make the selectional restriction defeasible. But then, the non-default information that *the goat* is the agent would override this. Thus the selectional restriction is rendered impotent, and (3a) still means (3c).

A second way of bypassing the problem, would be to encode every bit of information that can affect lexical generalisations in the lexicon itself. So the fact that goats don't read would be part of the lexical entry *goat*. Then default unification would in principle produce the right interpretation of (3a). But it would be unreasonable to assume that domain knowledge such as *goats don't read* is conventionalised. Representing all domain knowledge in the lexicon would make it unwieldy.

A final strategy would be to encode in the pragmatic component, that *all* results of default unification are overridable. But this doesn't do justice to the fact that some lexical generalisations are not default, while others are. The pragmatic component must be aware of these differences. And so default unification must communicate this information to pragmatics; in other words, default results of default unification must persist as default, beyond the boundaries of the lexicon.

Similar problems arise with adjectives. Pustejovsky [19] and others have argued against distinct lexical entries for *fast*, for each of its senses in *fast car*, *fast typist*, *fast motorway* and so on. Rather, it is possible to assume just a single lexical entry entry for *fast*, where its different senses arise from the process of syntagmatic co-composition. Copestake and Briscoe [7] show how this can be coded with default unification. The lexical generalisation is much like that for *enjoy*: an adjective like *fast* predicates over the telic role of the artifact. So default unification predicts that *fast car* means *a car which goes fast*, and *fast typist* means *a typist who types fast*, via the same entry for *fast*.

But as before, there are exceptions to this generalisation, which are triggered by the discourse context. In (4), *fast typist* means *typist who runs fast*, and not *typist who types fast*.

(4) a. All the office personnel took part in the company sports day last week.
b. One of the typists was a good athlete, but the other was struggling to finish the courses.
c. The fast typist came first in the 100m.

This creates problems with the default unification accounts of lexical orgnanisation, where the default results don't persist as such beyond the lexicon. The pragmatic component is unaware that interpreting *fast typist* as *a typist who types fast* is a default.

A third example concerns agreement.[1] Group nouns such as *committee* can take singular or plural agreement. The agreement used can have semantic effects: singular agreement indicates that the property denoted by the verb phrase applies to the group as a whole; whereas plural agreement indicates that it applies to *members* of the group. This generalisation predicts the semantic differences between (5a) and (5b); (5c) is unacceptable because of the agreement constraints imposed by the word *each*:

(5) a. The committee gets £20,000 per annum. *The whole group gets one lump sum.*
b. The committee get £20,000 per annum. *Each member gets £20,000.*
c. ?The committee gets £20,000 per annum each.

Copestake [6] shows how one can predict these semantics effects of agreement via default unification. She demonstrates that the 'distributive' sense of *committee* can be coerced from the 'collective' sense. The constraints on agreement for these entries then predict which entry of *committee* should be used in building logical form. And because of the different semantic components of the two senses, the semantic effects are accounted for.

But there are exceptions to the above generalisations, which are triggered by the discourse context. Copestake [6] points out that sports commentators tend to use plural agreement, even when assigning the verbal property to the 'collective' sense:

(6) Forfar are a good side. LOB *corpus*.

And uttering (7) in reference to Thatcher's term as Prime Minister is a joke, rather than ungrammatical:

(7) The government was a grandmother.

3 Default Unification Tailored for Pragmatics

It would be useful to modify existing lexical tools so that the recent ground gained in lexical productivity is not lost, but instead improved, through providing a communication link between the lexicon and other components. If the lexicon tells the other components of the system which generalisations can be overridden, then one could improve the interpretation of words in a discourse context, over what either component can do alone.

But the versions of default unification in [3, 5, 6] don't permit defaults to persist beyond the lexicon. So the other components don't know what's overridable. This means that pragmatics can't treat (1a) vs (3a) and (5b) vs (6) differently; nor can it predict when *fast typist*

[1] Thanks to Ann Copestake for pointing me to this data.

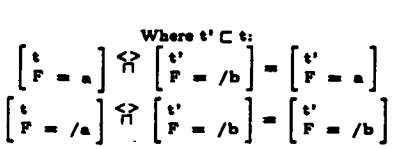


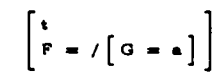
Figure 1: Some examples of PDU

means *typist who runs fast*, rather than *typist who types fast*. A second problem is that default unification, with the exception of [20], is order dependent. But in a discourse situation, one cannot always predict which pieces of information are to be unified, *in advance* of starting the discourse parsing process. So providing an interface between discourse processing and order dependent lexical processing would have to take into account the order in which the lexical operations are done, and this immensely complicates the reasoning task.

Lascarides *et al.* [13] have defined an order independent form of default unification over typed feature structures (TFSs). This solves the above two problems. Defaults in the lexicon *persist* under this operation, in the sense that one can distinguish in the semantic form that is sent to the pragmatic component, which parts are default. Because of this, the operation is known as Persistent Default Unification (PDU).

Copestake and Briscoe [7] show how PDU encodes lexical generalisations in a similar manner to previous default unification accounts. But the link between PDU and pragmatic reasoning hasn't been investigated; we present some preliminary results here.

PDU uses a slashed notation for partially defeasible FSs, where values to the left of the slash are indefeasible and those to the right defeasible (*indefeasible/defeasible*). We abbreviate this to */defeasible* where the indefeasible value is ⊤, and omit the slash when the defeasible and indefeasible values are the same. So for example, the FS (8) states that the value on the feature F is by default G:a, although that there is a value on F is nondefault:

(8) 

When a default value survives PDU (notated $\overleftrightarrow{\sqcap}$), it does so with the slash notation. The details of PDU don't concern us here; they're in [13]. Some of the results of PDU are given in Figure 1, however. These indicate that PDU validates defeat of Defeasible Modus Ponens (DMP), and Specificity (i.e., defeasible values on more specific types override defeasible ones on more general types).

PDU can form the basis of an inheritance account of lexical organisation. Lascarides *et al* [13] show how to encode the inheritance of telic roles in PDU (Figure 2); the inheritance is default because the telic role of "literature" will be *read*, but for the subclass of reference books it's *refer-to*.

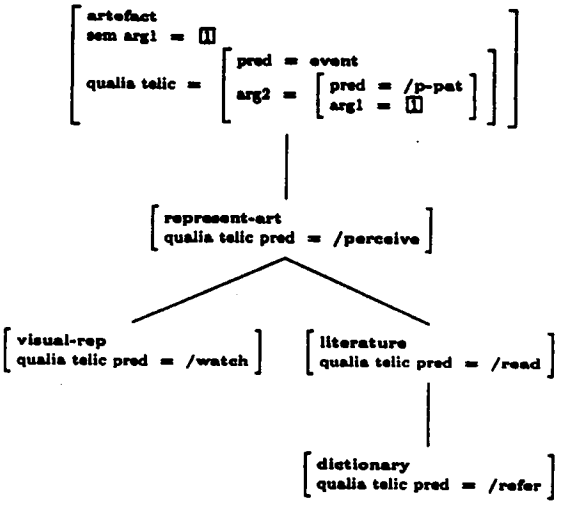


Figure 2: The Telic Roles of Artifacts

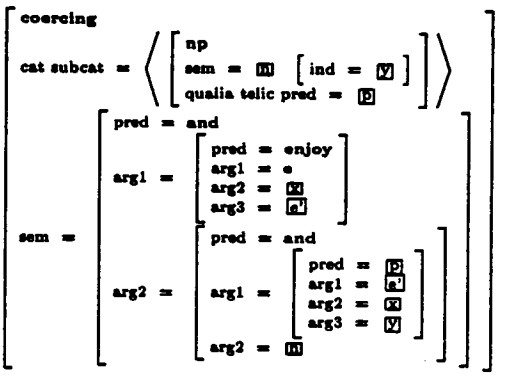


Figure 3: The Generalisations for Verbs like *Enjoy*

Having encoded generalisations about telic roles via PDU, Copestake and Briscoe [7] show how to state the lexical generalisation concerning *enjoy*, which exploits telic roles. As mentioned before, it is: *enjoy* predicates over the telic role of the artifact. This is captured in Figure 3: the coercion of *enjoy*, for taking artifacts as opposed to events as objects, is represented as internal to the verb semantics (cf. [12]). When *enjoy* takes an artifact denoting object, the event that is enjoyed is instantiated via the telic role, as indicated by the reentrancy 3 in Figure 3. Reentrancies always survive in PDU. Therefore, when PDU is used to build the TFS representing the phrase *enjoy the book*, the result is SEM:ARG2:ARG1:PRED:/read, thereby yielding the interpretation *enjoy reading the book*. PDU works in a simlar fashion on *begin the beer*, *enjoy the film*, and so on. The important thing to note is that the slash notation has survived in PDU: the fact that *enjoy the book* means *enjoy reading the book* is thus marked as default. This is in contrast to other inheritance accounts.

We can exploit this when building the compositional semantic representation. We'll use DRT, since it underlies

the pragmatic component DICE [14, 15], that we'll link the lexicon to. We assume that DRS-conditions that arise from elements on the RHS of the slash notation are embedded in an operator *, and this will affect their truth conditional status. So the logical form of (1a) derived via PDU is (1a'):[2]

(1) a. John enjoyed the book.

a'.

e, e', x, y, t
$john(x)$
$enjoy(e, x, e')$
$book(y)$
$hold(e, t)$
$t \prec now$
$*read(e', x, y)$

We now have the task of assigning DRS-conditions of the form $*\phi$ a model theoretic semantics, which must reflect that they're derived by defaults. PDU is formalised in a conditional logic. So the way defaults behave in PDU is determined by constraints on a function *, from worlds and propositions to propositions. $*(w, p)$ encodes what according to w, normally follows from p. So, let K be DRS, and let K^- be the DRS K with all the DRS-conditions of the from $*\psi$ removed. Then we can define the semantics of $*\phi$ as follows:

- $M, w \models_f *\phi$ in DRS K just in case for all w' in $*(w, [K^-])$, there is a $g \supseteq f$ such that $M, w' \models_g \phi$.

DRS conditions of the form $*\phi$ aren't asserted to be true in the actual world w. So in (1a'), the logical semantics *doesn't* entail that the event that was enjoyed was a reading; however, it does entail that an event was enjoyed by John. Thus we have utilised the fact that defaults persist, by assigning default results of PDU a different truth conditional status in logical semantics, than the indefeasible results. It is now up to the pragmatic component, to see whether *read* should be pragmatically inferred. We'll come to this shortly.

Copestake and Briscoe [7] treat *fast* in exactly the same way as *enjoy*. They argue that the telic role of *typist* is $[x][type(e, x)]$, where x is coindexed with the 'normal' variable. But this is defeasible: it's on the RHS of the slash. We can now encode the truth conditional effects of this: the DRS for *fast typist* involves:

(9) $[x][typist(x) \wedge fast(e) \wedge *type(e, x)]$

4 Linking The Lexicon to Pragmatics

We'll link PDU in the lexicon to a theory of pragmatics: specifically DICE (Discourse in Commonsense Entailment, [14, 15]). This is a model of discourse interpretation which encodes domain knowledge like *goats don't read*, and the information used to compute the rhetorical links between the segments of discourse. DICE uses the default logic Commonsense Entailment (CE) [1] to integrate the various knowledge resources. This logic exploits conditions of the form: $A > B$, which mean *If A then normally B*. So one could represent *goats don't read* as the schema:

- Goats Don't Read: $goat(x) > \neg read(e, x, y)$

The nonmonotonic validity ($\mid\!\sim$) has several nice properties. There are only three that are relevant here: first, it validates DMP: if one default applies and its consequent is consistent with the KB, then it's nonmonotonically inferred. Second, it validates the Penguin Principle: if conflicting defaults have their antecedents verified, then the consequent of the default with the most specific antecedent is preferred. Finally, for each deduction $A \mid\!\sim B$ there is a corresponding embedded default in the object language, $\exists(A, B)$. So $\exists(A, B)$ means $A \mid\!\sim B$.

To link the PDU treatment of lexical productivity to pragmatic knowledge, we add two axioms to DICE. First, Defaults Survive captures the intuition that defaults in the lexicon normally survive at the discourse level:

- Defaults Survive: $*\phi > \phi$

Second, we need an axiom that ensures that when the consequents of discourse processing and lexical processing conflict, the discourse processing wins. This is what happens in (3a), for example. The PDU prediction that the event enjoyed was a reading, is overridden by the conflicting pragmatic information stipulated in the >-rule Goats Don't Read. Let KB_h be obtained from the knowledge base KB, by removing all the DRS conditions of the form $*\phi$ (h stands for "hard information"). Then Discourse Wins states: when this KB yields a nonmonotonic conclusion ψ, then normally this survives the KB with conditions like $*\phi$ added to it:

- Discourse Wins: $(*\phi \wedge \exists(KB_h, \psi)) > \psi$

This rule is called Discourse Wins, because by the Penguin Principle with Defaults Survive, if ψ conflicts with ϕ, then ψ is nonmonotonically inferred and ϕ is not, even if $*\phi$ was in the KB: in other words, the clues from discourse context, if there are any, override conflicting results of PDU.

Let's now investigate how this affects the interpretation of the above examples. First, consider (1a). There are no >-rules which give information about the kinds of things that John enjoys. Consequently, the only >-rule that applies in DICE is Defaults Survive, with $*read(e', x, y)$ substituted in the antecedent. So by DMP, one pragmatically infers that John enjoyed reading the book. One could revise the DRS (1a') accordingly by replacing $*read(e', x, y)$ with $read(e', x, y)$, but we gloss over this. Now compare this with (3a). DMP with respect to KB_h on Goats Don't Read yields $\neg read(e', x, y)$.

[2] One could use a Parsons' like notation [17] to further refine this, so that the predicate *read* is marked as default, whereas the fact the agent of e' is John and the object is the book is non-default. We gloss over this here. Moreover for ease of presentation, we have ommitted WFFs of the form $*\phi$ when ϕ also holds.

So $\exists(KB_h, \neg read(e', x, y))$ holds. So both Defaults Survive and Discourse Wins have their antecedents verified. These rules conflict (the consequent of the former is $read(e', x, y)$ and the consequent of the latter is its negation). So by the Penguin Principle, $\neg read(e', x, y)$ is inferred. We would need more >-rules to infer that the event enjoyed is an eating. But this does show that we haven't obtained an unintuitive interpretation of (3a), in contrast to the purely lexical account of metonymy.

This account provides further motivation for conventionalising some aspects of metonymy. For suppose we were to compute metonymy *solely* within pragmatics. Then we would need to replace the information in Figures 2 and 3 with >-rules in DICE. Such a strategy is technically possible, but representation would be trickier. For example, to interpret (3a) correctly, the domain knowledge that goats don't read must win over the >-rules concerning generalisations about *enjoy* on telic roles. This means that the antecedent of this rule would have to be more specific, otherwise the logic won't resolve the conflict in the right way. So Goats Don't Read would have to be replaced with something like (10):

(10) $(enjoy(e, e') \wedge agent(e', x) \wedge goat(x) \wedge object(e', y) \wedge literature(y)) > \neg read(e')$

Spreading the load between pragmatics and the lexicon, and having communication links between them, allows us to 'loosen up' how we represent information.

Now consider (4). In this discourse context, *fast typist* means *typist who runs fast*, rather than *typist who types fast*. The above two axioms in DICE can capture this. DICE is equipped with knowledge which allows one to compute the rhetorical connections—such as *Elaboration*, *Narration* and *Contrast*, among others—that connect the meanings of segments of text together [14, 15]. So assume that the rules in DICE encode the intuitive attachment of (4c) to (4a,b). Then the definite NP must successfully refer to a unique referent from (4a,b). Since there are two typists, who have been differentiated only on the grounds of their athletic ability, accommodating the uniqueness condition is possible only if *fast* is equated with athletic ability. Thus $\exists(KB_h, fast(e') \wedge run(e', x))$ holds (where $typist(x) \in KB_h$). So Defaults Survive and Discourse Wins both apply, and they have the consequents $type(e', x)$ and $run(e', x)$ respectively. Assuming that e' can't be both a typing and a running, these rules conflict. And so by the Penguin Principle, $run(e', x)$ is nonmonotonically inferred. In contrast, in 'neutral' discourse contexts, DMP on Defaults Survive will yield that *fast typist* means *typist who types fast*.

Now we return back to the semantic effects of agreement on collective nouns. We can use PDU to code the lexical generalisation that a collective noun with plural agreement normally means that the verbal property applies to the members of the group, whereas with singular agreement it applies to the group as a whole ((5a) vs (5b)). The details can't be given here for reasons of space. But sentences like (6) and (7) indicate that this lexical generalisation must at best be defeasible. One shouldn't loosen the constraints between agreement and semantic effects completely in the lexicon. For this would forfeit the explanation of why (5a) is different from (5b). We would have to replace the lexical account of this difference with a pragmatic one. But intuitively, the pragmatic context is neutral in this case (unlike (6), where the verbal property is one where it only makes sense to apply it to the group as a whole; similarly for (7)).

PDU and DICE offer an alternative to loosening the connection betwen agreement and semantics completely in the lexicon. Let the semantic components of the distributive and collective senses of group nouns be *defeasible*. Given the above link to DICE, this defeasible constraint is then in principle overridable by the discourse context. So in principle, a group noun could take plural agreement and yet the verbal property be assigned to the group as a whole; or it could take singular agreement,. and yet the verbal property be assigned to the members of the group. Defaults Survive and Discourse Wins would tell us exactly when this happens. If the discourse context is neutral as to whether the PDU prediction on semantics should survive or not, then via DMP on Defaults Survive it survives. This is what happens in (5a) vs (5b); here the pragmatic context is neutral about what the verbal property should be assigned to (i.e., from a pragmatic point of view, getting $£20,000$ per annum could apply either to a group or individuals). So (5a) and (5b) are interpreted as intuitions dictate.

On the other hand in (6), the pragmatic context isn't neutral: *being a good side* normally applies to teams rather than individuals. If this is encoded in DICE, then via the Penguin Principle, the PDU prediction that the verbal property applies to the members of the group is overridden by this discourse information. We thus correctly predict that (6) is about the team as a whole, rather than the individual memebers. The PDU results are similarly overridden in (7).

Pollard and Sag [18] propose that agreement is largely pragmatic. Through using persistent defaults in the lexicon, we are able to conventionalise some aspects of agreement, without ruling out the exceptions that are triggered by pragmatic information. In essence, PDU conventionalises what happens when the discourse context is 'neutral', without forfeiting the impact of pragmatics.

Briscoe *et al* [4] claim that the lexical generalisations are only cancelled in discourse contexts that are informationally rich. We have illuminated in a formal setting exactly what this means. According to Defaults Survive and Discourse Wins, a lexical generalisation $*\phi$ can be cancelled only if $\exists(KB_h, \neg\phi)$. So a discourse context is 'informationally rich' if, independently of all default lexical generalisations, there are discourse clues which enable one to nonmonotonically conclude the exception.

So KB_h can't be 'neutral' about the proposition ϕ, if it is to block the lexical generalisation $*\phi$ from surviving in the discourse context.

5 Conclusion

Many lexical generalisations are of the sort where there are exceptions to the rules, which are triggered by information outside the lexicon. This poses a challenge to default unification accounts of the lexicon.

Using an account of lexical organisation involving *persistent* default unification [13, 7], we showed that links to a pragmatic component were possible with just two axioms: the first ensures that lexical generalisations normally apply in a discourse context, while the second ensures that normally, discourse information about how a word should be interpreted—if there is any—wins over defaults from the lexicon. This accounted for exceptions to lexical generalisations in a discourse context in three areas: logical metonymy, adjectives and agreement. Moreover, the axioms clarified in a formal setting the claim in [4], that exceptions to lexical generalisations can only be triggered by discourse contexts which are informationally rich.

This is just a first step towards linking lexical and pragmatic knowledge. Much more needs to be done, to achieve a robust theory of lexical interpretation in a discourse context. Nevertheless, these first results indicate the kinds of operations that one needs in both components for them to communicate properly. First in the lexicon: persistent defaults are useful. Second in pragmatics: the Penguin Principle, and representing information such as $A \not\approx B$ in the object language itself are useful.

References

[1] Asher, N. and Morreau, M. [1991] Commonsense Entailment: A Modal Theory of Nonmonotonic Reasoning. In *Proceedings of the 12th International Joint Conference on Artificial Intelligence*, Sydney, Australia, August 1991.

[2] Boguraev, B., and Pustejovsky, J. [1990] Lexical Ambiguity and the Role of Knowledge Representation in Lexicon Design, *Proceedings of the 13th International Conference on Computational Linguistics (COLING90)*, pp36–42, Helsinki.

[3] Bouma, G. [1992] Feature Structures and Nonmonotonicity, *Computational Linguistics*, 18 2, pp183–204.

[4] Briscoe, E. J., Copestake, A., and Boguraev, B. [1990] Enjoy the Paper: lexical semantics via lexicology, *Proceedings of the 13th International Conference on Computational Linguistics (COLING90)*, Helsinki, pp42–47.

[5] Carpenter, R. [1993] Skeptical and Credulous Default Unification with Application to Templates and Inheritance, in Briscoe, E. J., Copestake, A. and de Paiva, V. (eds.) *Inheritance, Defaults and the Lexicon*, Cambridge University Press, Cambridge, England, pp13–37.

[6] Copestake, A. [1993] Defaults in lexical representation, in Briscoe E. J., Copestake, A., and de Paiva, V. (eds.) *Inheritance, Defaults and the Lexicon*, pp223–245, Cambridge University Press, Cambridge, England.

[7] Copestake, A. and Briscoe, E. J. [1994] Semi-productive Polysemy and Sense Extension, to appear in *Journal of Semantics*, December 1994.

[8] Daelemans, W. [1987] A tool for the automatic creation, extension and updating of lexical knowledge bases, *Proceedings of the 3rd Conference of the European Chapter of the Association for Computational Linguistics (EACL-87)*, pp70–74, Copenhagen.

[9] Daelemans, W., De Smedt, K. and Gazdar, G. (1992) Inheritance in Natural Language Processing, *Computational Linguistics*, 18 2, 205–219.

[10] Evans, R. and Gazdar, G. [1989a] Inference in DATR, *Proceedings of the 4th Conference of the European Chapter of the Association for Computational Linguistics (EACL89)*, Manchester, England,pp66–71.

[11] Flickinger, D. *Lexical Rules in the Hierarchical Lexicon* PhD thesis, Stanford University.

[12] Godard, D. and Jayez, J. [1993] Towards a Proper Treatment of Coercion Phenomena, *Proceedings of the Sixth Conference of the European Chapter of the Association for Computational Linguistics (EACL93)*, Utrecht, The Netherlands, pp168–177.

[13] Lascarides, A., Briscoe, E. J., Asher, N., and Copestake, A. [1994] Persistent Order Independent Typed Default Unification, to appear in *Linguistics & Philosophy*.

[14] Lascarides, A. and Asher, N. [1991] Discourse Relations and Defeasible Knowledge, in *Proceedings of the 29th Annual Meeting of Computational Linguistics*, 55–63, Berkeley California, USA, June 1991.

[15] Lascarides, A. and Asher, N. [1993] Temporal Interpretation, Discourse Relations and Commonsense Entailment, *Linguistics and Philosophy*, 16 number 5, pp437–493, Kluwer Academic Publishers, Dordrecht.

[16] Nunberg, G. [1979] The Non-Uniqueness of Semantic Solutions: Polysemy, *Linguistics & Philosophy* 3, pp145–184, Kluwer Academic Publishers.

[17] Parsons, T. [1990] *Events in the Semantics of English*, MIT Press.

[18] Pollard, C. and Sag, I. [1994] *Head Driven Phrase Structure Grammar*, CSLI Lecture Notes Series, University of Chicago Press.

[19] Pustejovsky, J. [1991] The Generative Lexicon, *Computational Linguistics*, 17 4, pp409–441.

[20] Young, M., and Rounds, W. [1993] A Logical Semantics for Nonmonotonic Sorts, *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics (ACL93)*, Columbus, Ohio, pp209–215.