Introduction to Computational Semantics for Natural Language

©2015, Ted Briscoe
Computer Laboratory
University of Cambridge

January 26, 2015

Abstract

This handout builds on *Introduction to Formal Semantics for Natural Language*. The handout is not meant to replace textbooks – see the course syllabus and the sections below for readings, and the references herein. Please read each section in advance of the session, attempt the exercises, and be prepared to ask and answer questions on the material covered.

Contents

1 Generalized Categorial Grammars

1.1 Categorial Grammar

Classic (AB) categorial grammar consists of atomic categories of the form: N, NP, S, etc., and functor categories of the form S/N, (S\ NP)/NP, etc. constructed by combining atomic categories with slash and backslash with the functor leftmost and the 'outer' argument rightmost (see Wood, 1993 for a textbook introduction to CG and its generalisations).

Functors and arguments are combined by directional rules of (function-

argument) application as in Figure 2 below. CGs of this form are weakly equivalent to CFGs but assign a fully binary branching structure. So ditransitive verb complements, whose categories will be ((S\ NP)/PP)/NP, will be assigned a different structure than in a standard CF PSG approach. Figure 1 shows the CG derivation for a simple example. One feature of

Kim	loves	Sandy
NP	(S\NP)/NP	NP
kim'	λ y,x [love$'$(x y)]	sandy$'$
	---------- FA	
	S\NP	
	λ x [love$'$(x sandy$'$)]	
---------- BA		
S		
love$'$(kim$'$ sandy$'$)		

Figure 1: CG Derivation for *Kim loves Sandy*

CG is that syntax and semantics can be more closely associated than in a standard 'rule-to-rule' framework as function application in the syntax can correspond to function application (beta reduction) in the lambda calculus (regardless of directionality). This framework is 'radically lexical' since now there are just two rules of syntactic combination (FA,BA) and one rule of semantic application. Everything else must be captured in terms of the lexical categories. For example, modifiers cannot be dealt with in terms of separate rules and instead must be characterised lexically as functor arguments which yield categories of the same type (X/X, X\X) e.g. N/N or (S\NP)/(S\NP) – can you see what classes of word these categories would be appropriate for?

1.2 Generalized Categorial Grammar

The main interest in exploring CGs is that various extensions of classic AB CG (with just function application) have been proposed in recent years. These deal well with phenomena like non-constituent coordination and mostly extend the generative capacity of the grammar to 'mild context-sensitivity' / indexed languages. The specific extension I will outline adds rules of composition, permutation and type raising to AB CG as in Figure 2. These license derivations involving non-standard constituency such as Figure 3. Each of

Forward Application:	
X/Y Y ⇒ X	λ y [X(y)] (y) ⇒ X(y)
Backward Application:	
Y X\Y ⇒ X	λ y [X(y)] (y) ⇒ X(y)
Forward Composition:	
X/Y Y/Z ⇒ X/Z	λ y [X(y)] λ z [Y(z)] ⇒ λ z [X(Y(z))]
Backward Composition:	
Y\Z X\Y ⇒ X\Z	λ z [Y(z)] λ y [X(y)] ⇒ λ z [X(Y(z))]
(Generalized Weak) Permutation:	
$(X\|Y_1)\ldots\|Y_n \Rightarrow (X\|Y_n)\|Y_1\ldots$	λ $y_n\ldots,y_1$ [$X(y_1\ldots,y_n)$] ⇒ λ $y_1,y_n\ldots$ [$X(y_1\ldots,y_n)$]
Type Raising:	
(X ⇒ T/(T\X)	a ⇒ λ T [T a]
(X ⇒ T\(T/X)	a ⇒ λ T [T a]

Figure 2: GCG Rule Schemata

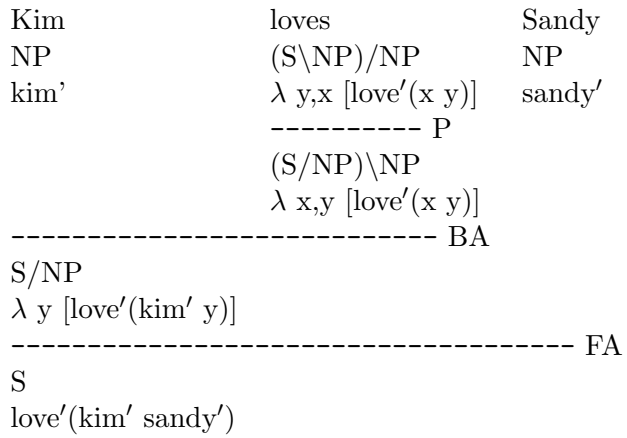

Figure 3: GCG Derivation for *Kim loves Sandy*

the rule schema come with a corresponding semantic operation defined in terms of the lambda calculus, illustrated in the sample derivations. What semantic type is being associated with NPs in the derivations shown below? How does typing work in this framework (i.e. given the X,Y and T labels in the rule schemata, how do we know what types are being combined in specific derivations and that these types are compatible)? Neither function composition nor permutation change the semantics associated with a given sentence, rather they introduce 'spurious' ambiguity in that they allow the same semantics to be recovered in different ways. This can be exploited to deal with non-constituent coordination (Figure 5), unbounded dependencies (Figure 4), and the relationship between intonation, focus and semantics. (See Wood, 1993 or Steedman, 1996, 2000, 2012 for fuller treatments of closely related approaches. CCG is like my GCG but without permutation.)

There are polynomial parsing algorithms (n^6) for some types of generalized CGs of this form (so long as rules such as type raising are constrained to apply finitely. Because of the 'spurious' ambiguity of GCGs some effort has been devoted to defining parsing algorithms which only find a single derivation in the equivalence class of derivations defining the same logical form. Steedman (esp. 2000) argues instead that the ambiguity is not spurious at all but rather correlates with different prosodies conveying different information structure (give-new, theme-rheme, focus – see Discourse Processing course).

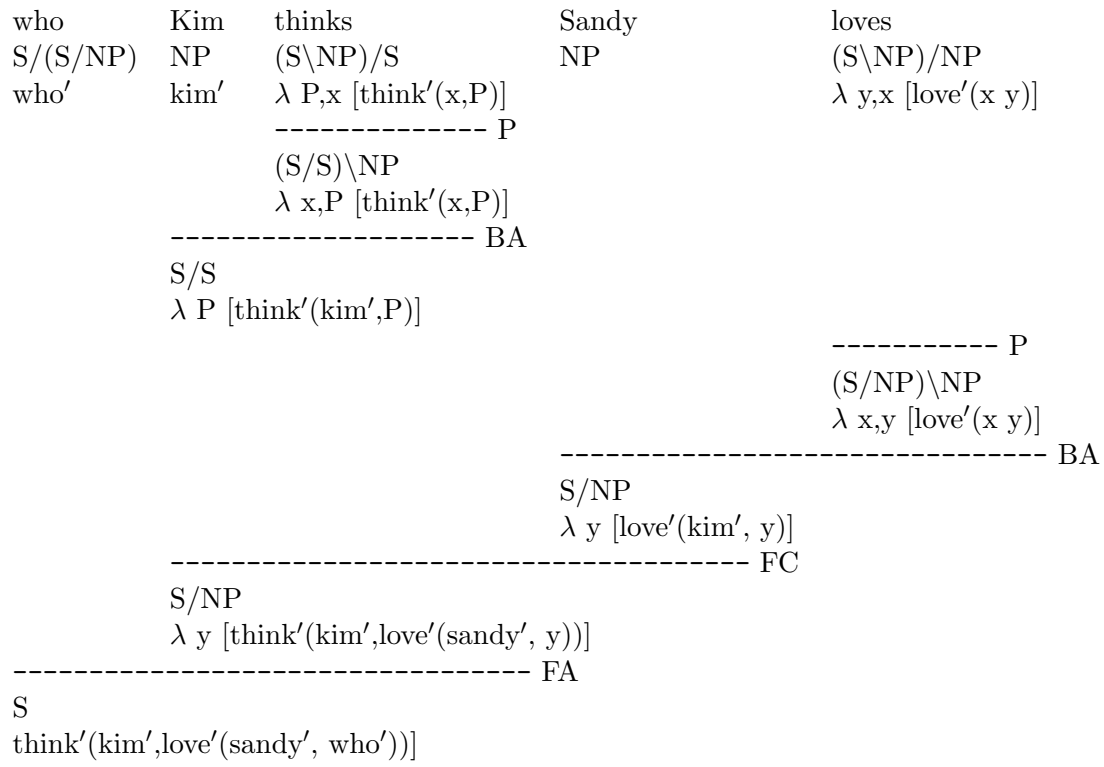

Figure 4: GCG Derivation for *who Kim thinks Sandy loves*

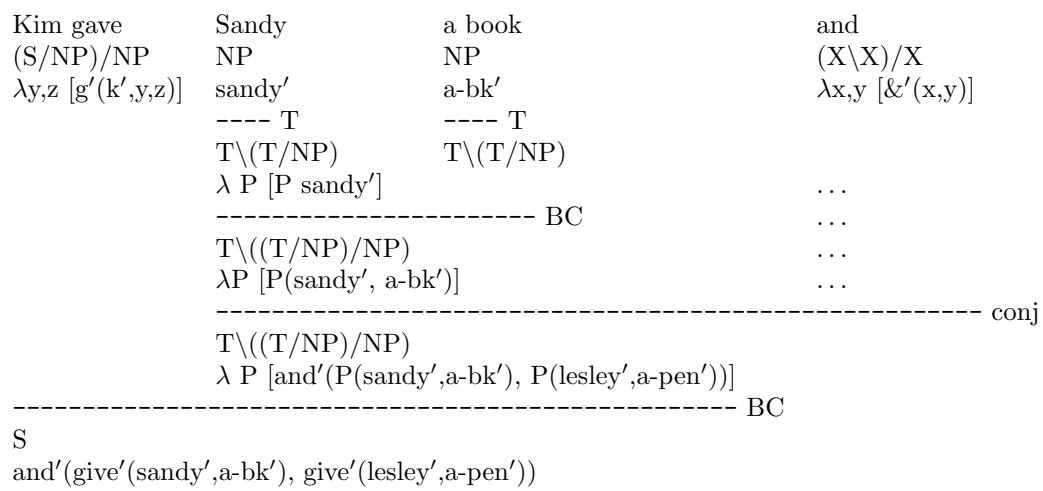

Figure 5: GCG Derivation for *Kim gave Sandy a book and Lesley a pen*

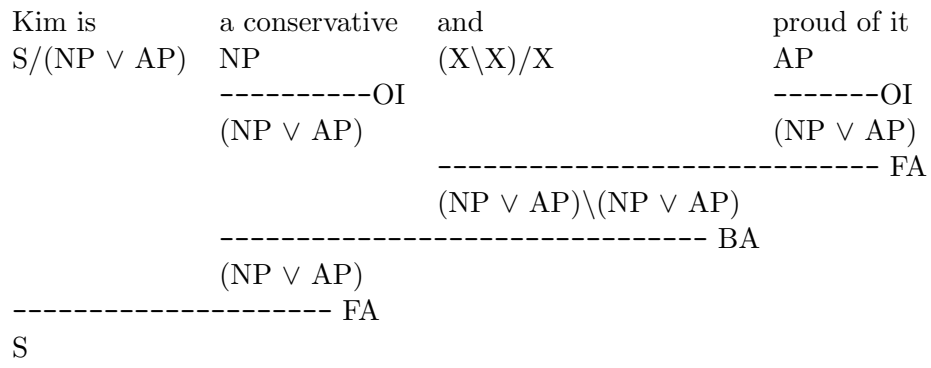

Figure 6: GCG Derivation for *Kim is a conservative and proud of it*

Bayer (1996) draws on another tradition in GCG research which emphasises the connection between CG and substructural or resource logics (see e.g. Carpenter, 1997; Morrill, 1994). This tradition has concentrated on demonstrating that the rules of GCGs can be shown to implement sound deductive systems, rather than on implementation via unification operations. Thus the slash operator is a form of (linear) implication: from X/Y, infer an X given a Y.

From this perspective, it makes sense to introduce rules (of inference) like $\wedge$-elimination (AE): given X $\wedge$ Y, infer Y, and $\vee$-introduction (OI): given Y, infer X $\vee$ Y. Bayer defines his GCG category set as the closure of the atomic category under the operators: / \, $\wedge$, and $\vee$. He assigns *and* the usual polymorphic category (X\X)/X and *be* the category (S\NP)/(NP $\vee$ AP). This along with the rule of OI is enough to license coordinations of unlike categories when the verb allows different complement types, as in Fig 6 This approach can be generalised to featural mismatches 'within' the same cateogory simply by allowing disjunctive feature values and aplying OI and AE to these values (e.g. CASE: **acc** $\vee$ **dat**) (feature neutralisation in German).

Although the use of unrestricted disjunction operators with complex unification-based feature systems is known to lead to computational inefficiency, if not intractability, it is not clear that this move would be so problematic in the context of the 'logicist' approach to GCG, as the features would be restricted to finite-valued morphosyntactic ones.

The sample derivations above illustrate (succinctly!) how GCG/CCG can

handle constructions such as unbounded dependencies and non-constituent coordination, both syntactically and semantically, which would be problematic for a CFG + LC approach like that introduced in the first handout.

1.3 Exercises

1. Work out another derivation for *Kim loves Sandy* to that shown in Fig 3 and show that it yields the same semantic interpretation

2. Suppose we assign NPs semantic type `<<e t> t>` so that we can handle quantifiers properly, but continue to assign verbs types like `<e t>`, `<e <e t>>` etc. Can you show that the derivation of the interpretation of *Max snores* using the syntactic and semantic definition of BA given in Figure 2 still works using appropriate lambda expressions for the NP and V(P) (i.e. S\NP)?

3. Work out the interpretation for *Every man snores* by assigning an appropriate semantic type and lambda expression to *every*.

4. There are two derivations for *Every man loves some woman*, the more complex involving P, BA and then FA. Can you see how to make this second derivation less 'spurious' by building the more marked interpretation in which *some* outscopes *every* using this derivation?

5. In F2 introduced in the last handout we argued that *every* and *some* should be treated like the quantifiers $\forall$ and $\exists$ in FOL. We could plausibly do the same for *all*, *a* an *one*, but what about other 'quantifiers' like *most* or *more than one third (of)*? Read about Generalized Quantifiers in Blackburn and Bos or on the web in Wikipedia or the Stanford Encyclopedia of Philosophy and see if you can work out how to assign a formula and interpretation to *Most men snore*.

1.4 CCG / GCG References

Bayer, S. 'The coordination of unlike categories', *Language* 72.3, 579–616, 1996.
Carpenter, R. *Type-Logical Semantics*, MIT Press, 1997.
Morrill, G. *Type-logical Grammar*, Kluwer, 1994.
Steedman, M. *Surface Structure and Interpretation*, MIT Press, 1996.
Steedman, M. *The Syntactic Process*, MIT Press, 2000.

Steedman, M. *Taking Scope*, MIT Press, 2012.
Wood, M. *Categorial Grammar*, Routledge, 1993

2 (Neo-)Davidsonian Semantics

We've seen that some quite simple constructions (e.g. those involving adjectives or adverbs) create problems for FOL representations of natural language semantics. The obvious interpretation of an adverb is that it modifies a verbal predicate or proposition, but this isn't possible in FOL. We've extended FOL with LC but so far only used LC as a means to compositionally construct FOL semantics for sentences in a syntax-guided fashion. We want to avoid higher-order logics such as modal, intensional or possible world semantics in computational semantics to keep theorem proving / inference tractable. One way to do so is to reify events (and worlds) in FOL:

(1) a Kim kissed Sandy passionately
b passionate1(kiss1(kim1, sandy1))
c $\exists$ e kiss1(e, kim1, sandy1) $\wedge$ passionate1(e)

(2) a Possibly Kim kissed Sandy
b possible1(kiss1(kim1, sandy1))
c $\exists$ e kiss1(e, kim1, sandy1)) $\wedge$ possible1(e)

Davidson was the first to suggest that we could replace the b) semantics with the c) semantics by reifying events, i.e. including event individuals/entities in the corresponding FOL model. We will write them, e, $e1$, etc to indicate that events are of a different *sort* to other entities. A *sort* being just like a *type* but applying only to the space of individuals/entities in the FOL model. States like *Kim weighs too much* are also reified, so some prefer to talk about 'eventualities' rather than events. Actually, this move doesn't quite work for *possibly* because there is a difference in meaning between b) and c) below – can you see it?

(3) a Possibly every unicorn is white
b possible1($\forall$ x unicorn1(x) $\rightarrow$ white1(x))
c ($\forall$ x unicorn1(x) $\rightarrow$ possible1(white1(x)))

(The problem is very similar to that discussed for propositional attitude verbs in the semantics and related to the *de re /de dicto*, sense vs. reference distinctions. (see e.g. L95 *Theories of Syntax, Semantics and Discourse Interpretation for Natural Language*, section 2.4).)

Parsons took this a stage further by proposing that arguments to predicates become binary relations between event variables and entities:

(4) a Kim kissed Sandy passionately
b $\exists$ e kiss1(e) $\wedge$ agent(e,kim1) $\wedge$ patient(e,sandy1) $\wedge$ passionate1(e)
c $\exists$ e kiss1(e) $\wedge$ arg1(e,kim1) $\wedge$ arg2(e,sandy1) $\wedge$ passionate1(e)

The problem with relations like 'agent' and 'patient' is determining exactly what they entail which is constant across all verbs (e.g. is max1 the agent in *Max enjoys films*?), so we generally prefer to use more semantically-neutral relations, as in c). The advantage of this neo-Davidsonian, Parsons-style representation is that it makes it easy to handle argument optionality. For example, the nominalization of (4i)s:

(5) a Kim's / The kissing of Sandy (was passionate).
b $\exists$ e kiss1(e) $\wedge$ arg1(e,kim1) $\wedge$ arg2(e,sandy1) $\wedge$ passionate1(e)
c $\exists$ e kiss1(e) $\wedge$ arg2(e,sandy1) $\wedge$ passionate1(e)

and we don't have to specify the agent, so c) is a reasonable semantics for this case. Some other advantages of this representation are that we can handle tense naturally, and PP adjectival and adverbial modifiers looks more similar:

(6) a $\exists$ e kiss1(e) $\wedge$ arg1(e,kim1) $\wedge$ arg2(e,sandy1) $\wedge$ passionate1(e) $\wedge$ past(e)
b $\exists$ e,x kiss1(e) $\wedge$ arg2(e,sandy1) $\wedge$ passionate1(e) $\wedge$ past(e) $\wedge$ in1(e,x) $\wedge$ bar(x)
c $\exists$ e,x kiss1(e) $\wedge$ arg1(e,kim1) $\wedge$ arg2(e,y) $\wedge$ passionate1(e) $\wedge$ past(e) $\wedge$ in1(y,x) $\wedge$ bar(x) $\wedge$ person1(y)

2.1 Exercises

1. Can you provide English sentences that match the semantics of the examples in (6?) Can you work out how to build these representations compositionally for sentences using CCG + LC?

2. Assign an event-based semantics to the following examples:

   (7) a Most men snore loudly or snuffle.
   b Each man (who) I met snored.
   c Kim's examination was hard.
   d Few students at Cambridge think they are geniuses.

3. Suppose we want to capture the semantics of *probably* or *may* properly using a similar approach. Can you work out how to reify worlds in FOL and give a semantics to these words in sentences like *Max may snore* or *Probably Max snores*? (See Blackburn & Bos book/papers or http://plato.stanford.edu/entries/possible-worlds/semantics-extensionality.html)

2.2 References

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

2.3 Wide-coverage Event-based Semantics for CCG

Bos *et al.* (2004) show how to derive FOL neo-Davidsonian representations from CCG derivations using the lambda calculus by assigning lambda functions to complex CCG categories (e.g. (S\NP)/NP λ P,y,x [P(x y)]) and defining decomposed function application schemes to associate with combinatory and type changing rules. (The decomposition operator, (@), circumvents some of the complexities of using the lambda calculus for syntax-directed semantic composition.). The paper is the first to show that it is possible to derive a logical semantic representation compositionally from a wide-coverage state-of-the-art parser applied to real data, and to evaluate the well-formedness of the represention produced. However, the resulting semantics doesn't handle scope underspecification or integrate with generalized quantifiers, it introduces argument relations like *agent* and *patient* which lack a coherent semantics (see discussion in Copestake RMRS draft

and above), and it doesn't handle 'construction-specific semantics' (e.g. a noun compound such as *steel warehouse* can mean warehouse *for* steel or warehouse *of* steel, so the N/N + N forward application (FA) rule needs to be sensitive to whether it is forming a compound or combining an adjective and noun, because for the compound an adequate semantics will introduce an additional underspecified relation: steel(x) $\wedge$ warehouse(y) $\wedge$ R(x,y)).

2.4 Underspecified (Robust) Minimal Recursion Semantics

MRS: An Introduction (see References below) goes over the motivations for underspecification, describes in detail an approach which is compatible with the generalized quantifier approach to natural language quantification, and outlines a preliminary theory of MRS composition. What follows is based on Copestake (2007) (see References) which develops the theory of (R)MRS underspecification and composition so that it is applicable to any syntactic framework and degree of syntactic information, in principle. The paper shows how a very underspecified RMRS representation can be extracted from a PoS tagger, whilst a more specified one can be extracted from a parser like RASP which returns syntactic trees but doesn't utilize a lexicon of complex categories / supertags like CCG which encode subcategorisation or predicate valency information.

To extract MRS representations for CCG we start like Bos *et al.* by assuming that (complex) lexical categories are asociated with elementary predications and any arguments encoded in the category (e.g. *kiss* : (S\NP)/NP : l1,a1,kiss(e1), l2,arg1(a1,x1), l3,arg2(a1,x2) where *lN* is a label and aN is an anchor (see discussion of Fig 6 in Copestake, 2007 for the need for anchors as well as labels). Closed-class vocabulary, such as quantifiers, negation etc, is assigned a lexical semantics as in standard (R)MRS, and the combinatory and unary (type-changing) rules must be coupled with semantic operations which handle different types of constructions (e.g. FA must be able to build the appropriate semantics for NP/N + N and for (S\NP)/NP + NP, in MRS terms scopal combination, op_{spec} and op_{obj} respectively). In other words, we have an even worse construction-specific semantic problem than Bos *et al.* do because we no longer have access to a relatively generic notion of function-argument application within the typed lambda calculus to associate with combinatory rules, and are instead relying on composing our semantics by binding variables in a construction-specific way.

To date, no-one has worked out such a semantics in detail, however, below I sketch a MRS semantics based on a CCG derivation which I think combines the best of MRS with the best of CCG syntax without complicating either unnecessarily. It is close to Copestake's (2007) approach to CFG+MRS as exemplified in Fig3 of that paper because it exploits the fact that CCG complex categories encode information about their arguments, and thus represent the same local constructional information as a CFG PS rule.

A semantic derivation for *A person kisssed Kim*

	Hooks	**Slots**	**Rels**	**(Q)Eqs**
a	l1,x1	l2,x1$_{spec}$	l3 a(x1)	h2 $=_q$ l2
			l3 rstr(h2)	
			l3 body(h3)	
person	l4,x2		l4 person(x2)	
NP/N+N	l1,x1			l2=l4
op$_{spec}$				x1=x2
kissed	l5,e1$_{past}$		l5 kiss(e1)	
		l6,x3$_{arg1}$	l5 arg1(e1,x3)	
		l7,x4$_{arg2}$	l5 arg2(e1,x4)	
Kim	l8,x5		l8 kim(x5)	
(S\NP)/NP+NP	l5,a3,e1			l7=l8
op$_{arg2}$				x4=x5
(S\NP)+NP	l5,a3,e1			l2=l6
op$_{arg1}$				x1=x3

Given this approach, the combinatory and unary rules do not need to be associated with a semantics because the semantics is pushed onto the (complex) categories associated with lexical items. By adding features to syntactic categories we can ensure we associate the right construction semantics with subtypes (e.g. for noun compounds N/N$_{nc}$ $\mapsto$ op$_{nc}$ as opposed to adjectives N/N$_{adj}$, etc).

We have seen that it may be possible to construct an underspecified semantic representation of sentence meaning compositionally in (R)MRS. However, although much of this representation is motivated by work on formal semantics (e.g. generalized quantifiers), (R)MRS itself is not a logic with proof and model theory. Rather it describes sets of trees of well-formed formulas in a neo-Davidsonian version of FOL extended with generalized quantifiers.

This implies that if you want to do inference and actual interpretation then it is still necessary to expand out the set of formulas and work with these. For instance, given the input (8a), a parser should produce a mostly resolved (R)MRS like (8b).

(8) a Every man loves some woman

b l1:every(x, h1, h2), l2:man(x), l3:love(e), l3:arg(e, x), l3:arg2(e, y), l4: some(y, h3 h4), l5:woman(y), h2=$_q$ l3

c every(x man(x), (some y, woman(y), love(e), arg1(e, x), arg2(e, y)))

d some(y, woman(y), every(x man(x), love(e), arg1(e, x), arg2(e, y)))

From (8b) we can create two fully specified formuli (8c) or (8d). Given an appropriate model and theorem prover we can then compute truth-values or reason that (8d) entails (8c), etc. However, we can't do this directly with (8b). For some tasks this may not matter; e.g. for (S)MT we might be able to generate directly from (8b) into another language which also underspecifies quantifier scope morphosyntactically (most do).

2.4.1 Exercise

If you want to follow up on (R)MRS and the idea of underspecification, then follow up the Copestake references and then work out the derivation of *Most men probably snore.*

2.5 Boxer

Bos (2005, 2008) has developed the approach to obtaining a wide-coverage FOL semantics from CCG to support reasoning. Firstly, he uses (underspecified) Discourse Representation Theory, (u)DRT, as his semantic representation. This is very similar to (R)MRS (see Copestake paper and handout) in that it is a neo-Davidsonian FOL and a similar approach to conjunction of formuli which was historically developed to handle anaphora better, rather than to support (more) underspecification; e.g. in (9a) and (9b), the pronouns function semantically like bound variables within the scope of *every* and *a*:

(9) a Every farmer who owns a donkey beats it.
b Every farmer owns a donkey. He beats it.
c every(x, farmer(x), some(y, donkey(y), own(x y), beat(x y)))

That is the (simplified) semantics of these examples is captured by (9c). For (9b) it is fairly easy to see that syntax-guided translation of sentences into FOL will lead to problems as the translation of the first sentence will 'close off' the scope of the quantifiers before the pronouns are reached. Something similar happens in (9a), at least in classical Montague-style semantics (as in Cann's *Formal Semantics* book). Bos & Blackburn (2004, *Working with DRT*) discuss (u)DRT and pronouns in detail.

Although, uDRT provides a technical solution that allows something similar to elementary predications being inserted into an implictly conjunctive semantic representation within the scope of quantifiers (i.e. to fill a hole / link to a hook in MRS terms), this doesn't really solve the problem of choosing the right antecedent for a pronoun. So Bos (2008) extends Boxer with a simple anaphora resolution system and Bos (2005) extends it with meaning postulates for lexical entailments derived from WordNet (see next section).

At this point, Boxer is able to output a resolved semantics for quite a large fragment of English. This can (often) be converted to FOL / Horn Clauses and fed to a theorem prover to perform inference and to a (minimal) model builder to check for consistency between meaning postulates and Boxer's output. Bos' papers give examples of inferences that are supported by the system and discuss where the system makes mistakes. The inferences mostly involve comparatively simple hyponymy or synonymy relations and the mistakes mostly involve discourse interpretation (pronouns, presuppositions). The off-the-shelf theorem proving technology that he uses also means that natural, generalized quantifiers can't be handled unless they translate into FOL quantifiers. Nevertheless, the coverage of real data is unprecedented and impressive.

2.6 Boxer / Underspecified Semantics References

Wide-Coverage Semantic Representations from a CCG Parser. Johan Bos, Stephen Clark, Mark Steedman, James R. Curran and Julia Hockenmaier. Proceedings of COLING-04, pp.1240-1246, Geneva, Switzerland, 2004.
http://www.aclweb.org/anthology/C/C04/C04-1180.pdf

Wide Coverage Semantic Analysis with Boxer, Johan Bos, 2008, STEP
Towards Wide Coverage Semantic Interpretation, Johan Bos, 2005, IWCS-6
http://www.let.rug.nl/bos/publications.html
Sections 1–4 from Ann Copestake *et al.*, 'Minimal Recursion Semantics: An Introduction' http://www.cl.cam.ac.uk/users/aac10/papers/mrs.pdf
Ann Copestake. Semantic composition with (Robust) Minimal Recursion Semantics. In: Proceedings of the ACL-07 workshop on Deep Linguistic Processing, pages 73-80. Prague, 2007.
http://www.aclweb.org/anthology/W/W07/W07-1210.pdf

2.7 Software

Boxer: http://svn.ask.it.usyd.edu.au/trac/candc/wiki/boxer
(Including on-line demo)

2.7.1 Exercise

If you want to explore what Boxer can and can't do, try running some examples through the on-line demo. You can use the sentences from previous exercises and examples on this and the previous handout, if you need inspiration for things to try out (e.g. the 'donkey' sentences and variants like *Every farmer who owns a donkey beats it. It hurts.*

3 Computational Approaches to Plausible Inference and Word Meaning

3.1 Word Meaning

Formal semantics has largely ignored word meaning except to point out that in logical formuli we need to replace a word form or lemma by an appropriate word sense (usually denoted as bold face lemma prime, lemma-number, etc (*loved*, **love′** / **love1**). We also need to know what follows from a word sense and this is usually encoded in terms of (FOL) meaning postulates:

(10) a $\forall x, y$ love′(x, y) $\rightarrow$ like′(x, y)
b $\forall x, y$ love′(x, y) $\rightarrow \neg$ hate′(x, y)
c $\neg\forall x, y$ desire′(x, y) $\rightarrow$ love′(x, y)

Although this is conceptually and representationally straightforward enough, there are at least three major issues:

1. How to get this information?
2. How to ensure it is consistent?
3. How to choose the right sense?

Bos solves 1) by pulling lexical facts from WordNet (nouns) and VerbNet – these are manually created databases (derived in part from dictionaries) which are certainly not complete and probably inconsistent. The information they contain is specific to senses of the words defined, so is only applicable in context to a word sense, so Bos simply assumes the most frequent sense (sense 1, given Wordnet) is appropriate. If the background theory built via WordNet/VerbNet is overall inconsistent, because the data is inconsistent, the algorithm for extracting relevant meaning postulates doesn't work perfectly, or a word sense is wrong, then the theorem prover cannot be used or will produce useless inferences.

There has been a lot of work on learning word meaning from text using distributional models of meaning (see Turney and Pantel, 2010 or Baroni *et al.* 2012a in references below for a review and/or Word Meaning and Discourse Understanding Module). These models represent words by their contexts of occurrence using approaches which are extensions of techniques used from information retrieval and document clustering, where a document is represented as a bag-of-words and retrieved via keywords indexed to documents, a word-document matrix is reduced so that documents are clustered, or document similarity is computed using the distribution of words (or clustered 'topics' or latent semantic dimensions) in each document.

If our interest is in word meaning then the same techniques can be used to represent, compare and cluster words. First we need to choose a representation of the context of occurrence for words (e.g. other words in a document or in a more local window around the target word, or sets of words to which the target is linked by grammatical relations). Second we

obtain word-context frequency counts from texts and turn these counts into association measures using e.g. PMI or TF/IDF. The resulting multidimensional word vectors can then be compared using a similarity measure, such as cosine. Instead of reducing the word-context matrix using SVD/LSA to yield clusters of word 'topics' / latent semantic dimensions, a more useful next step may be to use an (a)symmetric measure of similarity which computes the 'inclusion' of one word's context in that of another to infer synonymy/hyponymy relations (e.g. *dog* / *canine* is-a *animal*)).

Distributional semantics / semantic 'space' models provide a general notion of word similarity where word senses are 'blended' into a single vector of contexts. To obtain a representation of word senses identified by contexts, we need to do clustering over context vectors built at the first stage. This should reveal that a noun like *bank* has two homonyms bank1 (financial institution) and bank2 (earth mound) identified by two distinct 'sense' clusters. It is less clear that clustering contexts will handle polysemy where a word has multiple related senses (e.g. *the hammer struck the nail / he struck the match / an idea struck him*). However, it is also less clear that we want to treat these cases as independent senses rather than as a contextually-driven refinement of sense which is a side effect of semantic composition (see below).

There are many association and similarity measures and many clustering techniques. One approach to clustering that is popular, conceptually quite clear and results in a conditional probability distribution of word senses given a word is Latent Dirichlet Allocation (LDA) (as described e.g. in lecture 8 of the ML4LP Module). However, these models are built, they provide a more motivated way of picking a word sense to associate with a word occurrence in context than Bos' and so goes some way to solving 3) above, but it isn't obvious how to integrate them with a logical approach to (compositional) semantics.

3.2 Probabilistic Theorem Proving

Machine learning offers many models for classification (i.e. plausible propositional inference of the form:

$\forall x\ p(x) \wedge\ q(x) \rightarrow\ C(x)$

Probabilistic / statistical relational inference of the form, e.g:

$\forall x, y\ P(x, y) \wedge\ Q(x, y) \rightarrow\ R(x, y)$

is far less advanced. Recently, some progress has been made which is beginning to influence NLP and semantic interpretation.

Markov Logic Networks (MLNs, Richardson & Domingos, 2006) extend theorem proving to plausible probabilistic reasoning with finite (small) first-order models in a theoretically-motivated and representationally convenient way, and thus open up the possibility of reasoning in the face of partial knowledge, uncertainty and even inconsistency. Some of the inspiration for MLNs comes from NLP work on statistical parsing as the approach basically applies a maximum entropy model to FOL. Garrette *et al.*, give a succinct introduction to MLNs and then explore how they can be used in conjunction with Boxer to (partially) resolve issues 1), 2) and 3) above. They demonstrate that word vectors can be used to resolve word sense ambiguity so that the theorem prover is guided towards the most plausible contextually-appropiate interpretation. However, their approach to integrating distributional and compositional semantics is complicated and computationally expensive.

3.3 Weighted Abduction

Abduction is somewhat like (minimal) model building in that it allows the introduction of supporting premises to aid (discourse) interpretation. Intuitively, abduction is reasoning from consequent to antecedent. For example, knowing (11a), (11b) and (11c), we might conclude (11d) on the basis that drunkenness is more common than fever (weights: w1 > w2).

(11) a w1, $\forall$ x drunk(x) $\rightarrow$ stagger(x)
b w2, $\forall$ x fever(x) $\rightarrow$ stagger(x)
c stagger(kim)
d drunk(kim)

This inference is not deductively valid, because we need to assume the antecedent in order to prove the consequent. In the '80s and early '90s Hobbs and colleagues developed a theory of language interpretation based around weighted abduction. (See section 3.1 of Handout 2 from Intro to NLP for an example.) Weights were asigned manually and weight combination, especially when combining multiple sources of informaton – it is Saturday night in a club or it is Monday morning in a doctor's surgery – was difficult.

Scaling the approach would require a great deal of background knowledge.

Blythe *et al.* (2011) show how the output of Boxer/uDRT can be used for weighted abduction using MLNs. This solves the weight combination problem at least and is, in principle, compatible with Bos's (2008) integration of WordNet and other resources for background knowledge with Boxer. Their system manages to make 18/22 inferences necessary to interpret a small test set of cases requiring abduction / plausible inference.

3.4 Quantifier Scope Resolution

Underspecifying quantifier scope is all very well but for at least some language interpretation tasks choosing the most likely scoping is necessary. Manshadi and Allen (2011) present a dataset with scopes resolved and use supervised classification to assign narrow/wide or no scope classes to pairs of quantified NPs. This allows them to scope the quantifiers in test data using features such as quantifier type, order, head of the NP, etc. They achieve an accuracy of 78% which is better than the human annotators managed on this task.

The approach is crude compared to the techniques for computing scoped logical forms from (R)MRS and from uDRT and isn't guaranteed to produce a globally consistent set of scopings. It also doesn't model the scope interactions between quantifiers and logical operators (e.g. negation), so there is a need for more sophsticated integration of scope resolution and underspecified semantic representations.

An interesting and radically different approach to scope and its resolution is presented in Steedman (2012), which is also a recent book length review of issues in formal/computational semantics developing an account compatible with CCG. Steedman argues for a 'natural logic' approach in which instead of shoehorning natural language into a known formal logic, we develop a semantics close to (English) syntax and provide a model and proof theory for the resultant representation. He suggests that the only true quantifier in English is the universal which is needed to capture the meaning of words such as *every* and *each*. Words typically represented as existential or generalized quantifiers, such as *a*, *the*, *one*, *at least two*, *most*, etc, should be treated as Skolem functions, i.e. referential functions which optionally can refer dependent on univerally-bound variables in their environments (see section 4 of L95 *Introduction to Formal Semantics for Natural Language* handout

for a quick introduction to referential functions in FOL).

This theory has the nice consequence that skolemisation of such quantifiers is a necessary step to produce Horn Clauses from FOL formuli for automated theorem proving anyway (see e.g. L95 *Theories of Syntax, Semantics and Discourse Interpretation for Natural Language*, section 2.6). It also has the useful property that scope ambiguity reduces to non-deterministic skolemisation of referring NPs during interpretation, so that syntax-guided interpretation can represent meanings compactly. For example, the meaning of (12a)) can be represented by b) instead of c) or d).

(12) a A representative saw a sample
b see$'$(sk_1representative$'$, sk_2sample$'$)
c $\exists x, \exists y$ representative$'$(x) $\wedge$ sample$'$(y) $\wedge$ see$'$(x,y)
d $\exists y, \exists x$ representative$'$(x) $\wedge$ sample$'$(y) $\wedge$ see$'$(x,y)

c) and d) are truth-conditionally equivalent but the alternative quantifier orderings are significant in (13,) so need to be enumerated here too in any syntax-guided approach to interpretation. On the other hand, the Skolem function *sk* in b) is not ambiguous because there are no variables in the context in which skolemisation takes place.

(13) a Every representative saw a sample
b $\forall$x see$'$(representative$'$(x), sk_1sample$'$)
c $\forall$x see$'$(representative$'$(x), sk_1sample$'$(x))
d $\forall$x, $\exists$y representative$'$(x) $\wedge$ sample$'$(y) $\wedge$ see$'$(x,y)
e $\exists$y, $\forall$x representative$'$(x) $\wedge$ sample$'$(y) $\wedge$ see$'$(x,y)

In (13a)) *every* introduces a true quantifier in b) and c) which does provide a bound variable in the environment for skolemisation, so there are two possible ways of specifying the Skolem function here one of which, c), makes reference dependent on this bound variable. Unlike d) and e), these two representations can be derived in a structurally-identical fashion if skolemisation is a non-deterministic operation which occurs at the point in the derivation when the NP is composed with the semantic 'template' introduced by the universal quantifier. We can ensure this using translations into LC like:
a, an, some,... : $\lambda P\ \lambda Q.Q(sk_x P)$

every, each,... : $\lambda P\ \lambda Q\ \forall x P(x) \rightarrow\ Q(x)$
Essentially, skolemisation is non-deterministically dependent on any subset of bound variables available in the environment and the 'scope' of the NP will be increasingly 'wide' as more of these variables are ignored. This approach ensures that (14h)as only two readings – one in which every girl and boy are talking about one specific celebrity and one in which they are both talking about at least one possibly distinct celebrity.

(14) a Every boy likes and every girl dislikes some celebrity
b $\exists$z $\forall$x,y boy$'$(x) $\wedge$ girl$'$(y) $\rightarrow$ like$'$(x,z) $\wedge$ dislike$'$(y,z)
c $\forall$x,y boy$'$(x) $\wedge$ girl$'$(y) $\exists$z $\rightarrow$ like$'$(x,z) $\wedge$ dislike$'$(y,z)

So-called 'mixed' readings in which e.g. there is one celebrity liked by all boys and at least one possibly distinct one disliked by every girl are not available although specifiable in FOL and thus made available by theories like uDRT and (R)MRS. The rest of Steedman's long and complex book is devoted to the ramifications of correctly treating (14a)) correctly for every other aspects of formal semantics!

3.5 Question-Answering

A number of studies have used supervised machine learning techniques to learn 'semantic parsers' that map from text to logical representations, but these require training data matching sentences to logical forms which can only be produced by experts. Liang *et al.* (2011) develop a system which, given a question returns the correct answer. It learns appropriate logical representations for questions to compute answers using a domain knowledge base from a training dataset of question-answer pairs. They develop a simpler dependency-based semantics which is similar to the dependency tree representation of RMRS. This representation is convenient as it can be learnt more simply as a mapping from the output of a parser which returns a syntactic dependency representation. They demonstrate that the resulting system is able to scope quantifiers and negation, handling ambiguities in (elliptical) comparative constructions, by learning the most likely logical forms on the basis of optimising performance on the training data.

This is an impressive peice of work and probably the resulting system is the most robust and sophisticated semantic interpreter extant. However, the

approach seems limited to QA for now.

3.6 Compositional Distributional Semantics

Baroni *et al.* (2012) provides an excellent if long introduction to recent work extending distributional semantic models of word meaning to compositional models. This has typically involved combining word vectors by vector multiplication or addition or by some supervised weighted version of these operations to project word into phrase meanings. This potentially allows inference of synonymy or hyponymy to be extended to phrases and sentences, but doesn't look much like the notion of logical compositionality. More recently, this strand of work has got closer to logical semantics because several groups of researchers have suggested that distributional word representations be extended to include matrices and, in general, multidimensional arrays (i.e. tensors) which correspond more closely to their logical semantic types. For instance, an intransitive verb or an adjective would be represented by a matrix (second-order tensor) which when combined by vector multiplication with a vector representing a noun (phrase) would yield a new vector representing a nominal phrase (sentence) – see also Clark (2012).

This approach to compositionality is still a long way from being able, for instance, to express how generalized quantifiers alter truth-conditional meaning. Quantifiers are about the cardinality of sets, whilst vector/tensor operations capture associations / similarity. However, it does open up intriguing possibilities, such as individuating propositions in an appropriately fine-grained way – instead of propositions denoting a set of truth-values (as in e.g. possible world semantics) where tautologies and contradictions all have the same denotation and thus 'sense', they would denote a vector of derived 'contexts'. In this vein, Copestake and Herbelot (2012) make the interesting suggestion that word contexts should be represented as RMRS formuli rather than as words or words and grammatical relations. The advantage of this view is that now the denotation of a word is a linguistic semantic object amenable to various forms of inference – for instance, true synonymy equates to identical context sets of formuli, hyponymy only requires a subset relationship between elementary predications and not identity down to the level of variable bindings, and identity of reference does not require identity of sense (*The morning/evening star* can have different context sets even if they both refer to Venus). In this framework many compositional operations can be represented as set intersection between the words' or phrases' con-

text sets (e.g. the meaning of *red car*). This approach also seems better able to integrate, the complex functional meanings associated with closed-class vocabulary like generalized quantifiers with the context sets associated with open class words. However, whilst this proposal has a lot of interesting ideas about how to integrate lexical semantics with formal semantics, it isn't clear (to me at least) how it might be integrated with the tensor-based approach to compositional distributional semantics outlined above.

4 Conclusions

It is hard to know where computational semantics / language interpretation will be in a few years' time. After languishing from the early 90s 'til recently (whilst the field pursued statistical / machine learning approaches, and ignored compositional semantics) suddenly logical semantics is back in fashion, partly because of recent advances in probabilistic logic/inference. We are still some way from robust wide-coverage language interpretation, but I expect to see fast progress over the next few years, because many of the key pieces needed to build a system are in place: wide-coverage compositional semantics, distributional semantic space models of word meaning, large knowledge bases (WordNet, FrameNet, FreeBase, Yago, etc), and better theorem provers, model builders, and probabilistic inference engines (Church, Alchemy, Tuffy). Meanwhile compositional distributional semantics is also very fashionable and may soon yield frameworks capable of going beyond computing word, phrase or sentence similarity and of handling (some) logical entailments as well. How it will all fit together is anyone's guess and indeed may partially be fashioned by you.

Homework
Look at (some of) the readings below and come prepared to ask and answer questions.

Readings

Baroni, M., Bernardi, Zamparelli, Frege in Space: A program for compositional distributional semantics, 2012
clic.cimec.unitn.it/composes/materials/frege-in-space.pdf
Blythe J., Hobbs, J. *et al.*, Implementing weighted abduction in Markov

logic, Int. Wkshp on Computational Semantics, 2011
aclweb.org/anthology-new/W/W11/W11-0107.pdf
Clark, S., Vector Space Models of Lexical Meaning, ms. 2012
http://www.cl.cam.ac.uk/s̃c609/pubs/sem_handbook.pdf
Copestake, A., Herbelot, Lexicalised Compositionality 2012
http://www.cl.cam.ac.uk/ãac10/papers/lc1-0web.pdf
Garrette, D., K. Erk & R. Mooney, Integrating logical representations with probabilistic information using Markov logic, Int. Wkshp on Computational Semantics, 2011
aclweb.org/anthology-new/W/W11/W11-0112.pdf
Liang, P., Jordan, M. and Klein, D., Learning dependency-based compositional semantics, ACL 2011
aclweb.org/anthology-new/P/P11/P11-1060.pdf
Manshadi, M. and Allen, J. Unrestricted quantifier scope disambiguation, ACL Textgraphs Wkshp, 2011
aclweb.org/anthology-new/W/W11/W11-1108.pdf

Optional More Background

Turney P. & P. Pantel, From frequency to meaning: vector space models of semantics, JAIR, 37, 141–188, 2010
arxiv.org/pdf/1003/1141
Richardson, M. & P. Domingos, Markov logic networks, ML 62, 107–136, 2006
citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.170.7952.pdf
Steedman, M. *Taking Scope*, MIT Press, 2012.