Entailment, Intensionality and Text Understanding

Dick Crouch, Cleo Condoravdi, Valeria de Paiva, Reinhard Stolle, Daniel G. Bobrow
PARC
3333 Coyote Hill Road
Palo Alto, CA, USA, 94304
rdc+@parc.com

Abstract

We argue that the detection of entailment and contradiction relations between texts is a minimal metric for the evaluation of text understanding systems. Intensionality, which is widespread in natural language, raises a number of detection issues that cannot be brushed aside. We describe a contexted clausal representation, derived from approaches in formal semantics, that permits an extended range of intensional entailments and contradictions to be tractably detected.

1 Introduction

What are the appropriate metrics for evaluating performance in text understanding? There is probably no one universal measure that suffices, leading to a collection of metrics for evaluating different facets of text understanding. This paper makes the case for the inclusion of one particular evaluation metric in this collection: namely the detection of entailment and contradiction relations between texts / portions of texts.

Relations of entailment and contradiction are the key data of semantics, as traditionally viewed as a branch of linguistics. The ability to recognize such semantic relations is clearly not a *sufficient* criterion for language understanding: there is more to language understanding than just being able to tell that one sentence follows from another. But we would argue that it is a minimal, *necessary* criterion. If you understand sentences (1) and (2), then you can recognize that they are contradictory.

(1) No civilians were killed in the Najaf suicide bombing.

(2) Two civilians died in the Najaf suicide bombing.

Conversely, if you fail to recognize the contradiction, then you cannot have understood (1) and (2).

In proposing an evaluation metric, the onus is on the proposer to do a number of things. First, to show that the metric measures something real and useful: in this case, that entailment and contradiction detection (ECD) measures an important facet of language understanding, and that it correlates with the ability to develop useful applications (section 2). Second, to indicate the range of technical challenges that the metric raises: section 3 emphasizes one of these — the need to deal with intensional entailments, and the wisdom of drawing on the large body of relevant work in formal semantics in attempting to do so. Third, to show that the metric is not impossibly difficult for current technologies to satisfy, so that it encourages technological progress rather than stunting it: section 4 discusses a prototype system (described more fully in (Crouch et al., 2002)) to argue that, with current technology, ECD is a realistic though challenging metric.

2 Entailment and Contradiction Metrics

2.1 Theoretical Justification

The ability to recognize entailment and contradiction relations is a consequence of language understanding, as examples (1)–(2) show. But before concluding that entailment and contradiction detection is a suitable evaluation metric for text understanding, two cautionary points should be addressed. First, it cannot be a sufficient metric, since there is more to understanding than entailment and contradiction, and we should ask what aspects of understanding it does not evaluate. Second, we need to be reasonably sure that it is a necessary metric, and does not measure some merely accidental manifestation of understanding. To give an analogy, clearing up spots is a consequence of curing infections like measles; but clearing spots is a poor metric, especially if success can be achieved by bleaching spots off the skin or covering them with make-up. A measles-cure metric should address the

presence of the infection, and not just its symptoms.

In terms of (in)sufficicency, we should note that understanding a text implies two abilities. (i) You can relate the text to the world, and know what the world would have to be like if the text were true or if you followed instructions contained in it.[1] (ii) You can relate the text to other texts, and can tell where texts agree or disagree in what they say. Clearly, entailment and contradiction detection directly measures only the second ability.

In terms of necessity, there are two points to be made. The first is simply an appeal to intuition. Given a pretheoretical grasp of what language understanding is, the ability to draw inferences and detect entailments and contradictions just does seem to be part of understanding, and not merely an accidental symptom of it. The second point is more technical. Suppose we assume the standard machinery of modern logic, linking proof theory and model theory. Then a proof-theoretic ability to detect entailments and contradictions between expressions is intrinsically linked to a model-theoretic ability to relate those expressions to (abstract) models of the world. In other words, the abilities to relate texts to texts and texts to the world are connected, and there are at least some approaches that show how success in the former feeds into success in the latter.

The reference to logic and in particular to model theory is deliberate. It provides an arsenal of tools for dealing with entailment and contradiction, and there is also a large body of work in formal semantics linking natural language to these tools. One should at least consider making use of these resources. However, it is important not to characterize entailment and contradiction so narrowly as to preclude other methods. There needs to be room for probabilistic / Bayesian notions of inference, e.g. (Pearl, 1991), as well as attempting to use corpus based methods to detect entailment / subsumption, e.g. the use of TF-IDF by (Monz and de Rijke, 2001). That is, one can agree on the importance of entailment and contradiction detection as an evaluation mertic, while disagreeing on the best methods for achieving success.

2.2 Practical Justification

Even if we grant that entailment and contradiction detection (ECD) measures a core aspect of language understanding, it does not follow that it measures a *useful* aspect of understanding. However, we can point to at least two application areas that directly demonstrate the utility of the metric.

The first is an application that we are actually working on, concerning quality maintenance for document collections. The Eureka system includes a large textual database containing engineer-authored documents (tips) about the repair and maintenance of printers and photocopiers. Over time, duplicate and inconsistent material builds up, undermining the utility of the database to field engineers. Human validators who maintain the quality of the document collection would benefit from ECD text analysis tools that locate points of contradiction and entailment between different but related tips in the database.

A second application building fairly directly on ECD would be yes-no question answering. Positive or negative answers to yes-no questions can be characterized as those that (respectively) entail or contradict a declarative form of the query. Yes-no question answering would be useful for autonomous systems that attempt to interpret and act on information acquired from textual sources, rather than merely pre-filtering it for human interpretation and action.

Despite its relevance to applications like the above, one of the advantages of ECD is a degree of task neutrality. Entailment and contradiction relations can be characterized independently of the use, if any, to which they are put. Many other reasonable metrics for language understanding are not so task neutral. For example, in a dialogue system one measure of understanding would be success in taking a (task) appropriate action or making an appropriate response. However, it can be non-trivial to determine how much of this success is due to language understanding and how much due to prior understanding of the task: a good, highly constraining task model can overcome many deficiencies in language processing.

Task neutrality is not the same as domain or genre neutrality. ECD can depend on domain knowledge. For example, if I do not know that *belladonna* and *deadly nightshade* name the same plant, I will not recognize that an instruction to uproot belladonna entails an instruction to uproot deadly nightshade. But this is arguably a failure of botanical knowledge, not a lapse in language understanding. We will return to the issue of domain dependence later. However, there are many instances where ECD does not depend on domain knowledge, e.g. (1)–(2) or (3)–(4) (taken, with simplifications, from the Eureka corpus).

(3) Corrosion caused intermittent electrical contact.

(4) Corrosion prevented continuous electrical contact.

One does not need to be an electrician to recognize the potential equivalence of (3) and (4); merely that *intermittent* means *non-continuous*, so that causing something to be intermittent can be the same as preventing it from being continuous. And even in cases where domain knowledge is required, ECD is still also reliant on linguistic knowledge of this kind.

The success of methods for ECD may also depend on genre. For newswire stories (Monz and de Rijke, 2001)

[1] Knowing what the world would be like if the text were true is not the same as being able to tell if the text is true. I know how things would have to be for it to be true that "There is no greatest pair of prime numbers, p_1 and p_2, such that $p_2 = p_1 + 2$." But I have no idea how to tell whether this is true or not.

reports that TF-IDF performs well in detecting subsumption (i.e. entailment) between texts. This may be a consequence of the way that newswires convey generally consistent information about particular individuals and events: reference to the same entities is highly correlated with subsumption in such a genre. The use of PLSA on the Eureka corpus (Brants and Stolle, 2002) was less successful: the corpus has less reference to concrete events and individuals, and contains conflicting diagnoses and recommendations for repair actions.

3 Intensionality

The detection of entailments and contradictions between pieces of text raises a number of technical challenges, including but not limited to the following. (a) Ambiguity is ubiquitous in natural language, and poses an especial problem for text processing, where longer sentences tend to increase grammatical ambiguity, and where it is not generally possible to enter into clarificatory dialogues with the text author. Ambiguity impacts ECD because semantic relations may hold under some interpretations but not under others. (b) Reference resolution in the broad sense of determining that two texts talk about the same things, rather than the narrower sense of intra-text pronoun resolution, is also crucial to ECD. Entailment and contradiction relations presuppose shared subject matter, and reference resolution plays a role in establishing this. (c) World/domain knowledge, as we noted before, can be involved in establishing entailment and contradiction relations. (d) Representations that enable ECD must be derived from texts. What should these representations be like, and how should they be derived? At a bare minimum some level of parsing to obtain predicate-argument structures seems necessary, but how much more than this is required?

We cannot address all of these issues in this paper, and so will focus on the last one. In particular, we want to point out that intensional constructions are commonplace in text, and that simple first-order predicate-argument structures are inadequate for detecting intensional entailments and contradictions. Within the formal semantics literature since at least Montague, the phenomena raised by intensionality are well known and extensively studied, though not always satisfactorily dealt with. Yet this has been poorly reflected in computational work relating language understanding and knowledge representation. Formal semanticists have the luxury of not having to perform automated inference on their semantic representations, and can trade tractability for expressiveness. Computational applications on the other hand have traded expressiveness for tractability, either by trying to shoe-horn everything into an ill-fitting first-order representation, or by coding up special purpose and not easily generalizable methods for dealing with particular intensional phenomena in special tasks and domains. None of these approaches are particularly satisfactory for the task of detecting substantial numbers of entailment and contradiction relations between texts. A more balanced trade-off is required, and we suggest at least one way in which machinery from formal semantics can be adapted to support this.

3.1 Intensionality is pervasive

Intensionality extends beyond the conventional examples of propositional attitudes (beliefs, desires etc) and formal semanticists seeking unicorns. Any predication that has a proposition, fact or property denoting argument introduces intensionality. Almost every lexical item that takes a clausal or predicative argument should be seen as intensional. As an anecdotal test of how common this is, inspection of 100 Eureka tips about the workaday world of printer and copier repair showed that 453 out of 1586 sentences contained at least one verb sub-categorizing for a clausal argument. Some randomly selected examples of intensional constructions are given in (5).

(5) a. When the rods are removed and replaced it is very easy to hit the glass tab and break it off.
b. The weight of the ejected sets is not sufficient to keep the exit switch depressed.
c. This is a workaround but also disables the ability to use the duplex tray after pressing the "Interrupt" button, which should be explained to the customer.
d. Machines using the defective toner may require repair or replacement of the Cleaner Assembly.

Nor is intensionality confined to lexical items taking clausal or predicative arguments, as sentences (3) and (4) demonstrate. Prevention and causation (of central importance within the Eureka domain) are inherently intensional notions. To say that "A prevented B" is to say that there was an occurrence of A and no occurrence of B, but that had A not occurred B would have occurred. Similarly, to say that "A caused B" is to say that there was an occurrence of both A and B, but that had there been no occurrence of A there would have been no occurrence of B. Both refer to things or events materialized in one context but not in another. It is plain that we cannot give a semantic analysis for (6a) along the lines of (6b)

(6) a. Corrosion prevented continuous contact.
b. $\exists x, y.\ corrosion(x)\ \&\ contact(y)\ \&\ continuous(y)\ \&\ prevent(x, y)$
c. $\exists x, y.\ corrosion(x)\ \&\ contact(y)\ \&\ continuous(y)\ \&\ prevent(x, y)\ \&\ exists(x)\ \&\ \neg exists(y)$

since this asserts the existence of the continuous contact that was prevented. In (Condoravdi et al., 2001) we argued at some length that preserving a first-order analysis along the lines suggested by (Hirst, 1991) — through

the introduction of explicit existence predicates (6c) — is at best a partial solution. Not only are identity criteria for non-existent entities problematic, but (6c) also fails to capture significant monotonicity entailments: Corrosion preventing continuous contact does not imply that corrosion prevents contact of any form; but first order inference allows one to drop the $continuous(y)$ conjunct from (6c), yielding the representation one would expect for *corrosion prevented contact*.

We do not completely rule out the possibility that some more sophisticated, ontologically promiscuous, first-order analysis (perhaps along the lines of (Hobbs, 1985)) might account for these kinds of monotonicity inferences. But a more overtly intensional analysis like (7) does not face this problem in the first place.

(7) $\exists x.\ corrosion(x)$
$\&\ prevent(x, [\exists y.\ contact(y)\ \&\ continuous(y)])$

In (7) we assume that $prevent$ carries a lexical entailment that its second, propositional, argument is false. Thus (7) rules out the existence of continuous contact, but does not rule out the existence of any form of contact. Hirst, however, points out that allowing quantification over individuals into intensional contexts brings in its wake other well known difficulties: what does it mean for the same individual to exist in different possible worlds? In some sense, this is the trans-world analogue of the problematic identity criteria for non-existent individuals.

In (Condoravdi et al., 2001) we proposed an alternative analysis, (8), based on viewing noun phrases as being concept denoting rather than individual denoting (Zimmermann, 1993).

(8) $\exists X, Y.\ subconcept(X,\ Corrosion)$
$\&\ subconcept(Y,\ Contact \sqcap Continuous)$
$\&\ prevent(X, Y)$

This says that there is some sub-type of corrosion, X, and some sub-type of continuous contact, Y, such that concept X prevents concept Y. This means, amongst other things, that there is some instance of X but no instance of Y. Of course, just because there are no instances of continuous contact, it does not follow that there are no instances of contact, and (8) predicts the correct monotonicity entailments. Moreover, since concepts are functions from possible worlds to their extensions (sets of individuals), the issue of the trans-world identity of concepts does not arise: any particular concept expresses a single function, regardless of possible world.[2]

[2]Uniform identity of concepts across possible worlds does not mean that substitution of concepts that are co-extensive in one world is always truth preserving. Thus our use of concepts is intensional in the philosophically traditional sense, which is a point of clarification requested by one of our anonymous reviewers.

3.2 Detecting an Intensional Entailment

In (Condoravdi et al., 2001) we went into greater depth about how an analysis like (8) formally predicts the right kinds of entailment. Our purpose here is not to repeat these arguments, still less to argue that ours is the only possible way of accounting for these facts. Rather, we want to show how this highly intensional analysis can be deployed for practical ECD.

As an example consider determining the possible mutual entailment between (3) and (4), repeated below.

(9) Corrosion caused intermittent electrical contact.

(10) Corrosion prevented continuous electrical contact.

The lexical semantics for $cause$ and $prevent$ can be stated as follows (where we use the term "context" instead of "possible world"):

(11) If $prevent(C_1, C_2)$ is true in context T then
(a) In context T the concept C_1 is instantiated and concept C_2 is uninstantiated, and
(b) There is a context F that is maximally similar to T with the exception that C_1 is uninstantiated in F, and in F concept C_2 is instantiated.

(12) If $cause(C_1, C_2)$ is true in context T then
(a) In context T the concept C_1 is instantiated and concept C_2 is also instantiated, and
(b) There is a context F that is maximally similar to T with the exception that C_1 is uninstantiated in F, and in F concept C_2 is also uninstantiated.

Applying these definitions to (3) and (4), on the assumption that both statements are true in some context T:

(13) If cause(corrosion, intermittent-contact) is true in T then
(a) In T there is an instance of corrosion and an instance of intermittent contact, and
(b) There is a context F that is maximally similar to T except that there is no instance of corrosion, where there is no instance of intermittent contact; hence either there is no contact at all, or contact in F is non-intermittent (i.e. continuous).

(14) If prevent(corrosion, continuous-contact) is true in T then
(a) In T there is an instance of corrosion but no instance of continuous contact; hence either there is no contact in T, or contact is non-continuous (i.e. intermittent).
(b) There is a context F that is maximally similar to T except that there is no instance of corrosion, where there is an instance of continuous contact.

Both (13) and (14) refer to a relation of maximal similarity between contexts, with respect to the instantiation of a particular concept. The nature of this relation has deliberately not been spelled out, as it is unnecessary to do so in order to detect the possible entailment relation between (13) and (14). Assuming that both are evaluated against the same initial context T, they both invoke counterfactual contexts F that are maximally similar to T with respect to the concept of corrosion. Moreover, provided we pick the right disjunctive alternatives for non-intermittent and non-continuous contact, we can see that T and F have the same contents in both cases. Thus, whatever maximal similarity might turn out to be, (3) and (4) can be analysed as introducing the same contexts related in the same ways: that is, mutual entailment.[3]

Before describing how this example can be generalized to a scheme for detecting certain classes of intensional entailments and contradictions, we want to emphasize one point. The example makes free use of the notion of one context/possible world being maximally similar to another, with respect to the instantiation of a particular concept. Relations of maximum similarity between worlds are standard fare within formal, model-theoretic semantics, and alternative definitions abound. It is probably fair to say that the notion is not yet well understood. Fortunately for our example, full understanding of maximal similarity is not required. We only need to know that the same relation applies to the same initial context (T) to pick out the same counterfactual contexts (F). Of course, other examples may necessitate spelling out the relation in more detail. For instance, suppose we had the statement that *rust caused intermittent contact*, where rust is a subtype of corrosion. This raises the question of how maximal similarity varies across the type hierarchy; i.e. how does a maximally similar context with no instance of rust compare to one with no instance of corrosion? To answer this, we still do not need to specify fully the maximal similarity relation; merely state some of its necessary properties. Ultimately, though, if we want to use such formal means to relate language to the world, then relations like maximal similarity will have to be fully spelled out. But this is not the task that ECD sets out to deal with.

3.3 A General Approach to Intensional Entailments

The example above points to a general, two stage strategy for ECD. First map texts to contexted clauses, showing what contexts there are, and what (atomic) facts hold in them. Then attempt to pair contexts between the two text representations, and use relatively limited inference to determine whether the facts in paired contexts entail or contradict. We will look at these two stages in turn.

Contexted Clauses A contexted atomic clause comprises an atomic fact, plus the context in which the fact is supposed to hold. Borrowing McCarthy's notation[4] we write *(ist <context> <fact>)* to state that some fact holds in some context. A list of flat contexted clauses is interpreted conjunctively. Consider the contexted clauses derived from (8)

(15) a. Corrosion prevented continuous contact.

b. (ist t (instantiated corrosion1))
(ist t (uninstantiated contact2))
(ist t (prevent corrosion1 contact2))
(maxsim prevent-context3 t corrosion1)
(ist prevent-context3 (uninstantiated corrosion1))
(ist prevent-context3 (instantiated contact2))
(subconcept corrosion1 corrosion)
(subconcept contact2 contact)
(subconcept contact2 continuous)

Here we have a number of facts about what holds in the initial context, t: that there is an instance of a sub-concept of corrosion but no instance of some sub-concept of (continuous) contact, and that the prevent relation holds between the corrosion and contact concepts. This relation also introduces a new context, prevent-context3, which is maximally similar to t with respect to corrosion. Within prevent-context3, alternative, counterfactual assertions about concept instantiations are made. Finally, and independently of any particular context, subconcept assertions are made. The first says that corrosion1 is some (unspecified) subconcept of the concept corrosion. This statement is not relativized to a context, since the concept hierarchy is assumed to be constant across all contexts (even though extensions of concepts can vary).

The 'flattening' of (8) to derive (15b) proceeds via skolemization, conversion to clausal form, the relativization of each conjunct to a context and canonicalization to introduce extra contextual structure that is only implicit in linguistic forms (the context prevent-context3, corresponding to the counterfactual state of affairs the lexical entailments of *prevent* make reference to), or domain knowledge.

The canonicalization process is both language and knowledge/ontology-driven, introducing a deeper level of semantic representation. Structures assembled by compositional semantics must thus be transformed to structures that are well-suited for making successive small,

[3] Note that if we pick the other disjunctive alternative for non-intermittent contact, i.e. no instance of contact at all, then (3) can be shown to contradict (4): (3) says that corrosion causes an intermittent short circuit, while in (4) it intermittently breaks a contact that should be present. We do not yet have anything very useful to say about preferences between such interpretations, though we have been exploring the use of evidential reasoning.

[4] Though not borrowing McCarthy's view of contexts as subsumption-ordered logical micro-theories.

automated inference steps. Performing comparison on canonicalized contexted representations reflects a computationally advantageous division of labor: highly directed use of world knowledge and inference in the service of creating meaning representations, followed by relatively lightweight inference procedures in the stage of determining inferential relations between texts. Further aspects of canonicalization to conceptual structure based on a linguistically independent knowledge representation are discussed in (Crouch et al., 2002), e.g. mapping word senses onto term in a domain appropriate concept hierarchy.

A more complex example of flattening and canonicalization is (16), which is ambiguous between it being the removal of the sleeve that prevents breakage, or making the cable flexible that prevents breakage. The initial logical form for the second interpretation is shown in (16), and the packed contexted representation for both parses is (partially) shown in (18).

(16) Removing a sleeve made the cable flexible, preventing breakage.

(17) ∃C. subconcept(C, cable) & def(C, ?A) &
prevent(make([∃s. sleeve(s) & ∃a. agent-pro(a)
& remove(a, s)]
flexible(C)),
^breakage))

(18) (ist t (uninstantiated breakage-type2))
(ist prevent-ctx5 (instantiated breakage-type2))
(ist make-ctx3 (make remove-ev4 flexible-ctx5))
(ist make-ctx3 (sleeve sleeve6(make-ctx3)))
(ist make-ctx3
(remove remove-ev4 agent7(make-ctx3)
sleeve6(make-ctx3))
(ist flexible-ctx5 (flexible cable1))
(subcontext make-ctx3 t)
(subcontext flexible-ctx5 make-ctx3)
(subconcept cable1 cable)
(concepteq cable1 part-12KE45)
(subconcept breakage-type2 breakage)
(parse 1
(ist t (prevent make-ctx3 breakage-type2)))
(parse 2
(ist t (prevent remove-ev4 breakage-type2)))
(parse 1
(maxsim prevent-ctx5 t make-ctx3))
(parse 2
(maxsim prevent-ctx5 t remove-ev4))

Amongst other things, note how the proposition argument $make(\ldots)$ is replaced by the new context name make-ctx3, and component clauses asserted within this new context. Note also how skolem functions like sleeve6(make-ctx3) take context terms as arguments, and how the hook for definite reference by "the cable", def(C, ?A), is canonicalized to a concept equality, where part-12KE45 is some recently mentioned machine part. Also, maxsim can be relativized either to an event type, remove-ev4, or a context, make-ctx3.

Context Matching Having obtained contexted representations for two texts, ECD proceeds in two stages. First, by assuming that both texts describe the same initial context, locate sub-contexts introduced by the two texts that have parallel relations to the initial context. Second, for the contexts thus paired identify local entailments and contradictions using first-order reasoning. Given our use of concepts, much of this can be done using T-box reasoning from description logics. At present, we only view identical context relations as parallel, and do not give much consideration to the inheritance of propositional content between related contexts. A deeper level of matching would be based on an algebra of contexts detailing different types of context relations and their inheritance properties.

4 Feasibility of ECD

The last section described one way of approaching intensionality in the setting of entailment and contradiction detection. Our intention has not been to claim that this is the "one true way" of dealing with intensional ECD. It is rather to demonstrate the claim that practical progress can be made in the area, and that formal model-theoretic semantics can make a contribution to this. However, the preceding discussion has arguably been at too abstract and theoretical a level to really demonstrate a claim of *practical* progress or feasibility. This section briefly discusses a prototype entailment and contradiction detection system (described at greater length in (Crouch et al., 2002)) in order to point out that current technology already makes it feasible to begin addressing ECD.

The system has been developed around the Eureka collection of printer and copier repair tips. The full collection contains 30–40,000 free text documents. We have been focusing on a development subset of some 1,300 of these documents, including 15 pairs that have been pulled out for closer scrutiny because of known entailments and contradictions between them. We do not as yet have any testing data separate from our development data.

The system maps each document into a set of contexted clauses, by means of full syntactic and semantic analysis followed by knowledge-based canonicalization. Document representations undergo (statistically filtered) pairwise comparison to identify sentences within document pairs related by contradiction or entailment. We will describe the mapping and the comparison in turn.

The first stage of mapping uses a broad coverage, hand coded Lexical Functional Grammar of English (Butt et al., 1998) and the parser from the Xerox Linguistic En-

vironment (XLE) (Maxwell and Kaplan, 1993) to parse the documents. Parsing is robust in the sense that every sentence receives a functional-structure analysis, encoding grammaticalized predicate-argument structure. In about 25% of cases the functional-structures are fragmentary, either because of coverage gaps in the grammar or because of poor spelling and punctuation (to which the technicians writing the tips are prone). Fragments comprise longest span structures for constituents such as S, NP or PP that have been successfully analysed by the grammar. Ambiguity management via packing (Maxwell and Kaplan, 1989) allows the parser to efficiently[5] find all possible analyses of each sentence according to the grammar, and represent the alternatives in a compact, structure-shared form. Evaluation of essentially the same grammar on a dependency annotated subset of section 23 of the UPenn Wall Street Journal gives the accuracy of best parses as 85%, increasing by another 4% for non-fragmentary analyses (Riezler et al., 2002). Stochastic selection of the most probable parse (not necessarily the best parse) gives an accuracy of 80%.

Initial semantic interpretation is via an implementation of "glue semantics", which uses linear logic deduction to assemble the meanings of words and phrases in a syntactically analysed sentence (Dalrymple, 1999). Semantic interpretation preserves the ambiguity packing in syntactic analysis (though currently not in an algorithmically optimal way), deals with such things as quantifier scoping, and incorporates lexico-semantic information not relevant to parsing.Semantic analysis is also robust, with about 65% of all sentences receiving full, non-fragmentary analyses (around 60% on WSJ-23).

Canonicalization starts with a systematic flattening of logical forms: skolemizing quantifiers, replacing intensional arguments by new context names, and expanding out the intensional arguments within their new contexts. Rewrite rules are then applied, with the assistance of a TMS-based evidential reasoner, to further refine the resulting contexted clauses. Some rules are domain independent simplifications of alternate linguistic constructions onto the same underlying form. Others exploit ontological information to map words onto appropriate word senses or to identify domain appropriate pronouns antecedents. Others introduce additional contextual structure, or eliminate irrelevant linguistically induced contexts. To promote domain-portability, care is being taken to write canonicalization rules in such a way as to distinguish between (a) domain independent rules, (b) general rules with an interface to domain dependent ontologies, and (c) domain specific hacks.

Comparison of representations starts with statistical pre-filtering. This uses probabilistic latent semantic analysis to identify, on the basis of word occurrences, which documents are likely to have some content overlap (Brants and Stolle, 2002). For candidate pairs of documents thus identified, we employ a charitable form of reference resolution: if it is possible to identify clauses or contexts occurring in different documents, then identity is assumed. The Structure Mapping Engine (SME) (Forbus et al., 1989) is used to match contexts. The SME is a graph matching algorithm developed for the recognition of analogy. In our case it is used to match up structurally similar context structures containing structurally similar clauses. Having paired the contextual structures, limited ontological inference is then used to detect contradictions or entailments between the contents of matched contexts.

In summary, the robust application of detailed, hand-coded rules to the syntactic and semantic analysis of open texts appears feasible, with syntax somewhat more advanced. Similar observations have been made by other researchers, e.g. (Siegel and Bender, 2002). Knowledge-based canonicalization is less well advanced. In part, progress depends on the construction of rules in many ways similar to the grammar rules and lexical entries of syntactic analysis. Progress also depends on the construction of appropriate ontologies.

5 Conclusion

We have argued that entailment and contradiction detection (ECD) should be included as one of a number of metrics for evaluating text understanding. Intensional constructions — predications with proposition- or property-denoting arguments — are a challenge for ECD. They occur commonly, but simple predicate-argument representations do not do justice to the variety of inferences they support. More sophisticated first-order accounts (Hirst, 1991; Hobbs, 1985) may be extendable to bear this load. But there is also a direct path building on results from possible-worlds semantics. We are developing contexted clausal representations to aim at a useful trade-off between tractability and expressivity. Other researchers are also building on insights from model-theoretic semantics in interesting ways, e.g. (Schubert and Hwang, 2000). Intensional ECD seems to presuppose deep and detailed syntactic and semantic analysis (though we have no arguments to rule out the possibility of shallower analysis). The current state of deep language processing technology suggests that ECD is a viable though challenging metric for open text in restricted domains.

One issue that we have not addressed is the best form for annotated evaluation material for ECD. Ideally, this should be raw texts, annotated only to link the sentences or clauses that have entailment or contradiction relations between them. This has the benefit of being an almost entirely theory-neutral annotation scheme. A mark-up

[5] It takes a morning to distribute the 1300 development documents across half a dozen workstations and perform syntactic and semantic analysis.

based around some form of semantic representation for texts (e.g. contexted clauses) would very likely impose an unfair penalty on alternative approaches. A limited precursor to raw-text mark-up for semantic evaluation was undertaken as part of the FraCaS project (Cooper and Colleagues, 1996). This was a semantic test suite of about 350 syllogisms, specifying entailment and contradiction relations, or the lack of them, e.g.

(19) The PC-6082 is faster than the ITEL-XZ.
The ITEL-XZ is fast.
Is the PC-6082 fast? **[Yes]**

Even for trivial, artificial examples like these two problems arose. (i) The premises or conclusions can be ambiguous, where entailments of contradictions follow under one set of interpretations but not under another. There is no obvious way of marking the intended interpretations. (ii) It is extraordinarily hard to construct examples where inference relations do not in part depend on world knowledge. By taking texts rather than sentences as the units of annotation, the intended interpretation is generally much clearer (to human annotators). With regard to domain dependence, one just has to accept that ECD quality will decline without world knowledge.

References

Thorsten Brants and Reinhard Stolle. 2002. Finding similar documents in document collections. In *Using Semantics for Information Retrieval and Filtering: State of the Art and Future Research. Workshop at LREC-2002*, Las Palmas, Spain.

M. Butt, Tracy King, M. Niño, and F. Segond. 1998. *A Grammar Writer's Cookbook*. CSLI Lecture Notes. CSLI, Stanford.

Cleo Condoravdi, Richard Crouch, Martin van den Berg, John O. Everett, Valeria de Paiva, Reinhard Stolle, and Daniel G. Bobrow. 2001. Preventing existence. In C. Welty and B. Smith, editors, *Formal Ontology in Information Systems: Proc. FOIS-2001, Ogunquit, Maine*, pages 162–173. ACM Press, New York.

Robin Cooper and Colleagues. 1996. Fracas deliverable d16: Using the framework. Technical Report LRE 62-051 D16, HCRC, University of Edinburgh. www.cogsci.ed.ac.uk/fracas.

Richard Crouch, Cleo Condoravdi, Reinhard Stolle, Tracy King, Valeria de Paiva, John O. Everett, and Daniel G. Bobrow. 2002. Scalability of redundancy detection in focused document collections. In *Proceedings First International Workshop on Scalable Natural Language Understanding (SCANALU-2002)*, Heidelberg, Germany.

Mary Dalrymple, editor. 1999. *Semantics and Syntax in Lexical Functional Grammar: The Resource Logic Approach*. MIT Press, Cambridge, MA.

Kenneth D. Forbus, Brian Falkenhainer, and Dedre Gentner. 1989. The structure mapping engine: Algorithm and examples. *Artificial Intelligence*, 41(1):1–63.

Graeme Hirst. 1991. Existence assumptions in knowledge representation. *Artificial Intelligence*, 49:199–242, May.

Jerry R. Hobbs. 1985. Ontological promiscuity. In *Proc. ACL-1985*, pages 61–69, Chicago, IL.

John T. Maxwell and Ronald M. Kaplan. 1989. An overview of disjunctive constraint satisfaction. In *Proceedings of the International Workshop on Parsing Technologies*, pages 18–27.

John Maxwell and Ronald M. Kaplan. 1993. The interface between phrasal and functional constraints. *Computational Linguistics*, 19:571–589.

Christof Monz and Maarten de Rijke. 2001. Lightweight inference for computational semantics. In *Proceedings of 3rd International Conference on Inference in Computational Semantics*, pages 59–72, Sienna, Italy.

Judea Pearl. 1991. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann.

Stefan Riezler, Ronald Kaplan, Tracy King, Mark Johnson, Richard Crouch, and John Maxwell. 2002. Parsing the Wall Street Journal using a lexical functional grammar and discriminative estimation techniques. In *Proc. ACL-2002*. To appear.

L. K. Schubert and C. H. Hwang. 2000. Episodic logic meets little red riding hood: A comprehensive, natural representation for language understanding. In L. M. Iwanska and S. C. Shapiro, editors, *Natural Language Processing and Knowledge Representation*. MIT Press.

Melanie Siegel and Emily Bender. 2002. Efficient deep processing of Japanese. In *Proc 3rd Workshop on Asian Language Resources and International Standardization*, Taipei, Taiwan.

Ede Zimmermann. 1993. On the proper treatment of opacity in certain verbs. *Natural Language Semantics*, 1:149–179.