

Presupposed Content and Entailments in Natural Language Inference

David Clausen

Department of Linguistics
Stanford University

clausend@stanford.edu

Christopher D. Manning

Departments of Computer Science and Linguistics
Stanford University

manning@cs.stanford.edu

Abstract

Previous work has presented an accurate *natural logic* model for natural language inference. Other work has demonstrated the effectiveness of computing presuppositions for solving natural language inference problems. We extend this work to create a system for correctly computing lexical presuppositions and their interactions within the *natural logic* framework. The combination allows our system to properly handle presupposition projection from the lexical to the sentential level while taking advantage of the accuracy and coverage of the *natural logic* system. To solve an inference problem, our system computes a sequence of edits from premise to hypothesis. For each edit the system computes an entailment relation and a presupposition entailment relation. The relations are then separately composed according to a syntactic tree and the semantic properties of its nodes. Presuppositions are projected based on the properties of their syntactic and semantic environment. The edits are then composed and the resulting entailment relations are combined with the presupposition relation to yield an answer to the inference problem.

1 Introduction

Various approaches to the task of Natural Language Inference (NLI) have demonstrated distinct areas of expertise. Systems based on full semantic interpretation in first order logic are highly accurate but lack broad coverage, requiring large amounts of background knowledge to do open-domain NLI (Bos and Markert, 2006). Other systems based on statistical classifiers and machine learning achieve broad coverage but sacrifice accuracy by using shallow semantic representations (MacCartney et al., 2006). Natural logic was developed as a compromise between these two extremes (MacCartney and Manning, 2009). It makes use of rich semantic features while using syntactic representations closely related to the natural language surface strings to achieve broad coverage. Other work

has demonstrated the effectiveness of lexically triggered inferences and presuppositions to the task of natural language entailment and contradiction detection (Nairn et al, 2006; Hickl et al., 2006).

The natural logic model attempted to integrate these insights but recognized the difficulty of treating presuppositions within their current framework. Natural logic models negation, monotonicity, lexical relations and implicatures together as part of a sentence’s asserted content allowing them to be treated through a single projection mechanism. Presuppositions notoriously do not interact with these features although they do interact with other semantic features requiring a separate projection mechanism. We present a model for presupposition detection and computation separate from asserted content. We extend the natural logic model to compute lexically triggered presuppositions covered by Nairn et al. We then integrate this information to produce improved coverage for the NLI task.

2 Presuppositions

Presuppositions are propositions that are taken to be true as a prerequisite for uttering a sentence. The set of phenomena often grouped as presuppositions are diverse, although they are frequently systematically related to certain lexical items in a sentence, in which case they are said to be lexically triggered. Lexically triggered presuppositions like (1c) from (1a) can be used by an NLI system to expand the information available for solving a particular problem without full semantic interpretation.

- (1a) Bush knew that Gore won the election.
- (1b) Bush did not know that Gore won the election.
- (1c) Gore won the election.
- (1d) If Gore won the election, Bush knew that Gore won the election.

In (1a) the factive verb ‘knew’ triggers the local factive presupposition that the sentential complement ‘Gore won the election’ is true. (1a) is a simple sentence so the sentence as a whole

presupposes (1c) and we can make use of this information for an NLI problem. A defining feature of presuppositions is their invariance under negation so we have (1b) also entailing (1c). The factive presupposition is said to project through negation to become a presupposition of the entire sentence. In other cases such as the consequent of a conditional, the presupposition sometimes does not project so sentence (1d) does not presuppose (1c). Whether or not a lexically triggered local presupposition becomes a presupposition of the entire sentence is known as the problem of presupposition projection.

A complete treatment of the projection problem for all types of presupposition triggers is outside the bounds of current NLI systems but for most purposes we can compute presupposition projections based on a simple model first outlined by Karttunen (1973). The model categorizes lexical items as either filters, plugs or holes and uses these properties to determine how local presuppositions project upwards through a syntactic tree to become presuppositions of the entire sentence. Lexical items are categorized according to their effect on presuppositions they dominate syntactically. The verb ‘realize’ is a hole, and projects the presuppositions of its complement unchanged so (2a) has a sentential presupposition of (2c). The verb ‘pretend’ is a plug and projects none of the presuppositions of its complement so (2b) does not entail (2c). The conditional is a filter and will sometimes project the presuppositions of its antecedent and consequent based on the entailment relation that holds between the two. In the case of (1d) the antecedent entails the presupposition of the consequent so the presupposition of the consequent is not projected and it does not entail (1c).

(2a) Rehnquist realized Bush knew that Gore won the election

(2b) Rehnquist pretended Bush knew that Gore won the election

(2c) Gore won the election

The verbs ‘realize’ and ‘pretend’ represent two modest size classes of verbs and nouns called factives and antifactives. The sentential presuppositions for any given factive or antifactive operator depend on its position in the sentence’s syntactic tree and the number and type of holes, plugs or filters that dominate it.

To implement this theory we model the local factivity presuppositions triggered by various sentential complement taking operators. We

then calculate the presuppositions of the entire sentence by projecting the local presuppositions according to Karttunen’s theory. For each operator our system traverses the sentence’s syntactic tree from operator node to root calculating how the local factivity presuppositions project through the various holes, plugs and filters. The result is a set of sentential level presuppositions that can be used to determine inference relations to other sentences.

3 Presupposition in NatLog

The NatLog system of MacCartney and Manning (2008; 2009) is a multi-stage NLI system that decomposes the NLI task into 5 stages: (1) linguistic analysis, (2) alignment, (3) lexical entailment classification, (4) entailment projection, and (5) entailment composition. The NatLog architecture and the theory of presupposition projection outlined in section 2 reflect two parallel methods for computing entailment relations between premise and hypothesis. We augment the NatLog system at steps (1), (4) and (5) to compute entailment relations and presuppositions in parallel. The result is two separate entailment relations which are combined to form an answer to an NLI problem. At stage (1) we calculate the lexically triggered factivity presuppositions for a given sentence. At stage (4) we project the presuppositions to determine the effective factivity according to the theory outlined in section 2. In stage (5) we compose the presuppositions across the alignment between premise and hypothesis to determine the presupposition entailment relation. Finally we combine the presupposition entailment relation with the entailment relation generated from the standard NatLog system to produce a more informed inference.

3.1 Lexical Factivity Presuppositions

Lexical factivity presuppositions are detected by regular expressions over lemmatized lexical items taken from the classes of factive and antifactive verbs and nouns. Figure 1 gives example entries for two operators. A sentence is analyzed for factivity operators by matching the regular expressions to the tree structure and when one is detected its terminal projection is marked as a factive operator with the appropriate factivity. The sentential complement of the operator is marked as being in the scope of a factive operator of the appropriate type.

```

Operator: know
Pattern: VP<(/^VB/</^know$/)
Scope: /SBAR|S$/
Factivity: FACT

Operator: pretend
Pattern:
VP<(/^VB/</^pretend$/)
Scope: /SBAR|S$/
Factivity: ANTI

```

Figure 1: A factive and antifactive operator

3.2 Presupposition Projection

For any given constituent of a sentence we can calculate its effective factivity presupposition by determining the number and type of factivity operators which dominate it. This is analogous to computing the projected presuppositions for a sentence but instead stores the information locally on the representation of the sentence. Let’s compute the factivity of ‘Gore won the election’ in (2b). First we look for the immediately dominating factivity operator and find that it is dominated by the factive operator ‘know’ which assigns the local factivity FACT. We then traverse up the tree and find the operator ‘pretend’, which assigns the local factivity ANTI and dominates the constituent and the operator ‘know’. We then compose the two according to table 1. to determine the effective factivity for the constituent is ANTI. If the sentence included more factive or antifactive operators we would continue to calculate the effective factivity recursively using the effective factivity output at each level as the dominated input for the next level.

Dominated	Dominating	Effective
ANTI	ANTI	ANTI
ANTI	FACT	ANTI
FACT	ANTI	ANTI
FACT	FACT	FACT

Table 1: The effective factivity for any pair of dominated and dominating factivity assignments.

The result tells us that the sentence in (2b) has an antifactive presupposition that ‘Gore won the election’. This is equivalent to the presupposition that ‘Gore did not win the election’. This contradicts (2c) and we can conclude that (2b) does not entail (2c). Detecting that the presuppositions of a premise are incompatible with the

hypothesis is achieved in step (5) presupposition composition.

3.3 Presupposition Composition

The NatLog model for NLI computes a sequence of atomic edits from premise to hypothesis. The entailment relation between each atomic edit is computed and then composed across the sequence of edits to determine the entailment relation that holds between premise and hypothesis. An atomic edit consists of an insertion (INS), deletion (DELN) or substitution (SUB) operation. To compose the presuppositions calculated in step (4) we compare the factivity presuppositions before and after each atomic edit. In our simplified model the only edits that can change the factivity presuppositions are INS, DELN or SUB of factive or antifactive operators. Using table 2 we compute an atomic presupposition entailment relation between each atomic edit based on the edit type, local factivity and effective factivity. We then compose the atomic presupposition entailment relations to produce the presupposition entailment relation that holds between the premise and the conclusion. Finally we combine the presupposition entailment relation with the entailment relation generated by the standard NatLog architecture to yield the answer to the NLI problem. Atomic presuppositions are computed according to table 2.

Operator	DEL	INS
ANTI	Alternation	Alternation
FACT	Forward	Reverse

Table 2: Operator effective factivity and the resulting atomic presupposition entailment relation for DEL and INS edits.

The sequence of atomic edits converting the premise (2b) to the hypothesis (2c) involves DEL of one antifactive operator ‘pretend’ and one factive operator ‘know’. The first DEL of ‘pretend’ results in an atomic presupposition entailment relation of Alternation. The second DEL of ‘know’ results in an atomic presupposition entailment relation of Forward, together yielding a presupposition entailment relation between the premise and hypothesis of Alternation. This allows our system to correctly predict (2c) is incompatible with and a contradiction of (2b).

4 Improvements

Previous implementations of the NatLog system were unable to handle NLI problems with (1b) as the premise and (1c) as the hypothesis because atomic presupposition entailment relations were treated together with normal entailment relations. The sequence of atomic edits from (1b) to (1c) would involve the DEL of ‘know’ resulting in an atomic entailment relation of Forward while DEL of ‘not’ would result in an atomic entailment relation of Negation together yielding Alternation instead of Forward. Our augmented system handles these types of inferences by separating presupposition entailment relations from normal entailment relations. In our augmented system only the DEL edit of ‘know’ produces an atomic presupposition entailment relation of Forward. Since no other operators in (1b) produce atomic presupposition entailment relations the resulting presupposition entailment relation between (1b) and (1c) is the correct Forward entailment.

Evaluating on a set of 3-way entailment NLI test problems developed at PARC by the authors of (Nairn et al. 2006) the Augmented NatLog system achieved an accuracy of 60.53% compared to the original NatLog system accuracy of 53.95% by correctly treating problems like (3) where (3b) should be inferred from (3a).

(3a) Bush didn’t realize that Afghanistan is landlocked.

(3b) Afghanistan is landlocked.

With further development we expect to extend these results to other NLI test sets.

5 Conclusion

Our system extends the coverage of the NatLog system to correctly handle factive presuppositions. By computing entailments based on semantic containment and exclusion separately from those based on presupposition we avoid unwanted interaction between the two dimensions of meaning while leveraging the information contained in presuppositions to improve NLI performance. Although they are invariant under negation, presuppositions do not uniformly project. Projection is determined by a myriad of complex factors which ultimately require logical formalisms much more complex than predicate logic to compute (Beaver 2001). Our treatment does not currently take into account other types of presuppositions including those based on as-

pectual relations, (Mary has/hasn’t stopped beating her boyfriend \Rightarrow Mary has been beating her boyfriend), definitive descriptions, (The king of France is/isn’t bald \Rightarrow There is a king of France), or iteratives, (The boy cried/didn’t cry wolf again \Rightarrow The boy cried wolf before). We have, however, provided a framework that can be extended to compute many types of lexically triggered presupposition and their projections. This work continues the theme of MacCartney and Manning in asserting “open-domain NLI is likely to require combining disparate reasoners”. By augmenting NatLog with a reasoner based on factive presuppositions we take one step closer to the goal of achieving open-domain NLI.

References

- Beaver, David I. 2001. *Presupposition and assertion in dynamic semantics*. Stanford: CSLI.
- Bos, Johan and Katja Markert. 2006. When logical inference helps determining textual entailment (and when it doesn’t). In *Proceedings of the Second PASCAL Challenges Workshop on Recognizing Textual Entailment*.
- Hickl, Andrew, John Williams, Jeremy Bensley, Kirk Roberts, Bryan Rink, and Ying Shi. 2006. Recognizing textual entailment with LCC’s GROUND-HOG system. In *Proceedings of the Second PASCAL Challenges Workshop on Recognizing Textual Entailment*.
- Karttunen, Lauri. 1973. Presuppositions of compound sentences. *Linguistic Inquiry* 4: 169-93.
- MacCartney Bill, Trond Grenager, Marie-Catherine de Marneffe, Daniel Cer, and Christopher D. Manning. 2006. Learning to recognize features of valid textual entailments. In *Proceedings of the North American Association of Computational Linguistics (NAACL-06)*.
- MacCartney, Bill and Christopher D. Manning. 2007. Natural logic for textual inference. In *ACL-07 Workshop on Textual Entailment and Paraphrasing*, Prague.
- MacCartney, Bill and Christopher D. Manning. 2008. Modeling semantic containment and exclusion in natural language inference. In *Proceedings of Coling-08*.
- MacCartney, Bill and Christopher D. Manning. 2009. An extended model of natural logic. In *The Eight International Conference on Computational Semantics (IWCS-8)*, Tilburg, Netherlands, January 2009
- Nairn, Rowan, Cleo Condoravdi, and Lauri Karttunen. 2006. Computing relative polarity for textual inference. In *Proceedings of ICoS-5 (Inference in Computational Semantics)*, Buxton, UK.