# Combining Natural Logic and Natural Deduction for Inference Over Natural Language

Ghadah Binhadba
Supervisor: Prof. Allan Ramsay
Advisor: Dr. Antoniu Pop

October 2017

## 1 Introduction

An inference is "a conclusion drawn from one or more assumptions–premise(s)" [1] (p.10). Traditionally, natural language inference (NLI) requires either defining a pairwise entailment relations between a premise and a conclusion texts as in Example 1, where "a dog" $\vdash$ [1] "an animal", or applying a set of rules to chain over the multi-premises (this is often called argumentation [2]) as in Example 2 where the rule in Example 3 is used to infer its conclusion.

Automatic reasoning about natural language (NL) requires computers to transform the input text into a data structure to which inference rules can be applied–a *meaning representation*. Over decades, such representations have ranged between shallow syntactic analysis such as parse trees to sophisticated logical expressions such as first-order logical (FOL) formulas.

*p:* Fido is a dog

---

*c:* Fido is an animal

**(Example 1: pairwise entailment)**

*p1:* Fido is animal

*p2:* All animals are mortals

---

*c:* Fido is mortal

**(Example 2: 2 premises argument)**

$A$ is $B$

all $Bs$ are $Cs$

---

$A$ is $C$

**(Example 3: Inference rule)**

The remainder of this paper will be as follows: Section 2 states the problems of translating NL texts into FOL expressions as representation and why we need to merge pairwise entailment relations and chaining inference rules in the process of drawing a conclusion. Section 3 will explain the key features of the inference techniques used in our research. Section 4 will show how our proposed NLI system works and outlines the proposed inference methods. Finally, Section 6 demonstrates the progress made so far and what is planned for the remaining period of study.

## 2 Research Problem

FOL expressions have been used for decades as ***a meaning representation***, thanks to its expres-

[1] ($\vdash$) is the entailment symbol

siveness and inferential effectiveness [3]. To use FOL as a meaning representation, NL sentences first have to be translated into logical expressions, and this translation has proven difficult for two reasons. First, logical forms are created on top of a syntactic analysis of the language–parse trees–which has proved difficult to do using rule-based parsers. Second, natural language contains too many anomalous constructions that do not fit within any rule-based grammar.

Besides the difficulty of translating into logic, FOL is inadequate for proper treatments of important linguistic issues. One example concerns *default* sentences, such as 'Most birds fly' and 'Most swans are white'. Reasoning about defaults is non-monotonic, and for that reason, FOL, as monotonic logic, cannot handle it properly. Another issue is related to *intentionality*, where in referential context, FOL fails to distinguish between what is designated by a term (*extension*) and what it really means (*intention*). For instance, in FOL, 'I saw the Morning star' is not a logical consequent of 'I saw the Evening star', even though they both refer to the planet Venus. In non-referential contexts, sentences of propositional attitudes that include verbs such as 'believe', 'know' and 'claim', can be treated as modalities by accepting the 'possible worlds' account of the semantics of modal logic and embodying this by reifying the worlds in which some sentence is true; however, Barr [4] explained that this treatment could lead to mistaken inference.

As for the ***NLI tradition*** to be applied, there are many examples that occur in NL for which neither pairwise entailment nor argumentation alone can permit inferring a conclusion, as in Example 4.

| | |
|---|---|
| *p1:* | Fido is a dog |
| *p2:* | Every animal is a mortal |
| *c:* | Fido is a mortal |
| | **(Example 4: Mixed inference)** |

Therefore, bringing these two NLI traditions together and constructing a suitable meaning representation for NL texts are the key issues for enabling proper reasoning.

# 3 A Bit of a Background

Before going through the details of our proposed NLI system, this section will give a brief description of the used inference methods which are the following:

## 3.1 Natural Deduction

Natural deduction (ND) is a proof system that tries to prove valid reasoning by repeatedly applying simple basic inference rules, the rules of logical constants–conjunction, disjunction, implication, universal and existential quantifier, and possibly introducing valid assumptions with some constraints for discharging them afterwards[5]. Some of the ND rules can be combined together practically during the course of proof. An important case of such rules combination is what leads to the notion of *unification*[2] in proofs, see Figure 1.

## 3.2 Natural Logic

Natural logic is "a logic for natural language that could categorise valid inferences by working on syntactic forms of a language that are

[2] "A unifier of two expressions–terms or clauses–is a substitution that when applied to the expressions makes them equal." [6]

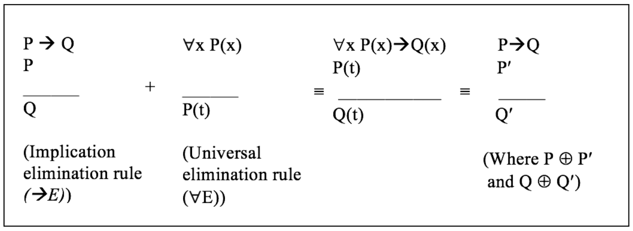


Figure 1: Unification operator (⊕).

close to their surface forms" [7].It defines entailment relations between a pair of sentences based on an approximate unification ($\sqsubseteq$)–also called *containment relation*– of their constituents. Approximate unification can be seen as sequences of edits–*substitutions, insertions* and *deletions*– that turn one sentence to another [8]. The type of these edits are determined by the monotonicity features–polarity–of the sentential constituents. Upward-monotone (positive) expressions can be expanded by deleting modifiers and/or substituting a term by more general one as in Example 5, whereas, downward-monotone (negative) expressions can be contracted by inserting modifiers and/or substituting a term by a more specific one as in Example 6. Expressions could be non-monotone as well which means that they cannot be expanded nor contracted to another expression but can only be matched with identical expressions, for example 'prettiest butterfly' $\nvdash$'prettiest insect' [8].

Polarities are context-based features i.e., not something fixed to lexical items. For example, **'man'** in Example 5 has a positive polarity, while in Example 6 its polarity is negative because of the negation word **'not'**.Therefore, *polarity marking* is an important step to be made before starting the natural logic entailment process. This marking is compositional; the polarity of a lexical item is calculated with consideration to the surrounding polarities using the compositional operator in Figure 2. It, also, requires having a pre-defined table of lexical expressions that have polarity effects on other expressions within a sentence such as 'not' and 'no'.

*p:* He saw **an old man**

*c:* He saw **a human**

**(Example 5: upward-monotone entailment)**

*p:* He is not **a human**

*c:* He is not **an old man**

**(Example 6: downward-monotone entailment)**

| | | | |
|---|---|---|---|
| *No polarity* $(\cdot)$: | $(\cdot) \circ (+\vert-) =$ | $(+\vert-) \circ (\cdot) =$ | $(\cdot)$ |
| *Positive polarity* $(+)$: | $(+) \circ (+) =$ | $(-) \circ (-) =$ | $(+)$ |
| *Negative polarity* $(-)$: | $(+) \circ (-) =$ | $(-) \circ (+) =$ | $(-)$ |

Figure 2: Polarity comositional operator ($\circ$).

### 3.3 Normal forming and Transformations

In logic, normal forming is "the reduction of logical formulas to standardized (normalized) forms that are more appropriate for automated theorem proving" [9]. These forms enable the inference engine to concentrate on key features of logical forms [10].

There are different kinds of normal forms (NFs), such as conjunctive normal form (CNF), negation normal form(NNF) and Skolem normal form(SNF). Different theorem provers want different NFs. However, obtaining them, basically, is by applying a set of procedural steps called *transformations.* These transformations usually involve applying a collection of inference rules forward; typically, the rules of logical equivalences such as *double negation rule*, *distributive law* and *De Morgan's laws.* However, some NFs could require more than equivalence rules like the case of SNFs were the conversion preserves satisfiability rather than strict equivalence.

# 4 Proposed NLI System

## 4.1 Research goal

To investigate the use of dependency trees as a basis for reasoning about natural language.

## 4.2 Research Questions

To what extent inferences over dependency trees for English sentences can be carried out?

## 4.3 The system

For the reasons stated in Section 2 we have proposed a new NLI system(see Figure 3) that takes one or more premises(s) and a question as an input and returns *Yes*, *No* or *Unknown* as an answer and it has the following parts:

- **Parse Trees:** we proposed using dependency trees as a meaning representation. In our project we are using an already existing parser; we will not develop a new one. Moreover, the choice of dependency parser is not the center of our project and it does not affect the main purpose and the outcome of this project as a matter of fact changing the parser only means re-visitng and updating the pre-processing part of the system.
- **SATCHMO+:** is the core part of our system; the theorem prover. It is basically an adaptation for the FOL theorem prover SATCHMO [11] which works on FOL expressions and does proofs by contradiction i.e, it proves the validity of a proposition such $P \rightarrow Q$ by proving that $P \wedge \neg Q$ is not valid. SATCHMO+, on the other hand, accepts dependency trees as text representation and carries out its proofs constructively [12], i.e. assuming $P$ and trying to derive $Q$ from it, using the natural deduction rules. Moreover, instead of the straight unification used in SATCHMO, SATCHMO+ uses *natural logic approximate unification (NLAUnification)* that, as explained in Section 3.2, allows asymmetric matching. The integration of the NLAUnification and SATCHMO+ is what we believe will enable us to solve mixed inference problems as Example 3, where SATCHMO+ rules will do the chaining over multi premises and the approximate unification will do the pairwise entailment bit.
- **Pre-processing:** Dependency trees will not be used as they are by the theorem prover. First, they need to be marked with polarity information so that NLAUnification could decide on the direction of unification. Then, the marked trees will go through a sequence of steps to become in normalized forms that SATCHMO+ could operate on.
  1. **Polarity marking:** each node on the tree will be given either a '+' for *upward-monoton*, '-' for *downward-monoton* or '.' for *non-monotone.* This process is carried out by scanning the trees in a top-down manner and compositionally assigning marks to nodes by making use of the pre-

constructed polarity table.

2. **Normal forming and transformations:** like any other theorem prover SATCHMO+ needs a certain form of the meaning representation to work with. Basically, each of the premises is to be turned into either a rule of the form $A \rightarrow B$ or a fact of the form $A(x)$ and then stored as part of the Knowledgebase. The question is also to be normalized so that SATCHMO+ can use the stored rules and facts to draw an answer for it.

- **Polarity table:** is a table of lexical items that have a polarity effect on other items, mainly their arguments. Each entry of the table will state an item and how many arguments does it affect and how the arguments will be affected. For example, the word *'No'* in the sentence 'No man loves a woman', takes two arguments, a noun and a verb phrase, and has a negative polarity effect on both of them, thus, they should be marked with the '-' sign.

- **Knowledge base:** besides the rules and facts that are stored temporarily during the course of the proof, permanently, we have a WordNet[3] driven lexical relations– synonyms and hyponyms– that are used by the NLAUnification to decide whether an expression can be contained by/or contain another. For example, 'man' is a hyponym of 'human',thus, in a positive context $man \sqsubseteq human$.

[3] WordNet "is a large electronic lexical database of English" [13] (P. 231). It was manually constructed in 1986 at Princeton University and continually maintained and updated.

# 5 Research Contribution and Evaluation

The intended contribution of this research is a demonstration that inference over NL can be carried out without constructing an intermediate meaning representation. In order to demonstrate that, we will test our program on two kinds of data; pairwise relations from RTE test suites [14, 15, 16, 17, 18, 19, 20, 21] and syllogistic inferences from FraCaS [22]. We plan to compare our results with the results of the NLI systems in the literature that tested their systems on either one of the test sets or both, especially Maccartney and Manning's [8, 23, 24] *NatLog system* as their work has achieved great results on both FraCas and RTE sets.

# 6 Progress and Future Planes

**System Implementation:** This year, several important things were completed. First, in terms of pre-processing, dependency trees were transformed into the intended normal forms but not marked with polarity yet. The steps for achieving these forms are not finalised as we believe that testing the transformations steps on real data could lead to enhancing the current transformation steps and/or adding new ones. Second, the inference engine, SATCHMO+, has been developed and tested on a few examples. The engine is in its basic version and we are planning to expand it to cover challenging linguistic issues such as 'defaults' and 'intensionality'. Third, the NLAUnification has been developed and successfully integrated with SATCHMO+. Although trees are not marked with polarity yet, tables of WordNet lexical relations–hypernyms and synonyms–have been built and are used by the NLAUnification. Missing the polarity information means that the direction of unification cannot be determined, therefore, in addition to building the polarity table, a mechanism of marking the trees should be designed and implemented.

**Writing Thesis:** Although the focus on the first two years were on reading the literature and implementing the proposed system, sections of the thesis have been written. The final thesis is expected to be delivered by the end of the coming year and its preliminary structure is given below.

1. **Introduction**
   - 1.1. Natural Language (NL)
   - 1.2. Natural Language Inference (NLI)
     - 1.2.1. Pairwise Entailment
     - 1.2.2. Multi-Premises Inference (Argumentation)
   - 1.3. Meaning Representation
   - 1.4. Proposed NLI System
     - 1.4.1. Research Problem
     - 1.4.2. Research Goal
     - 1.4.3. Research Question
     - 1.4.4. Research Tasks
     - 1.4.5. Research Contribution
2. **Background**
   - 2.1. Natural Logic
   - 2.2. Natural Deduction
   - 2.3. SATCHMO
   - 2.4. Normal Forming and Transformations
   - 2.5. Depdendnecy Trees
   - 2.6. Bacground Knowledge
3. **Litrature Review**
4. **System Design**
   - 4.1. Parser
   - 4.2. Bacground Knowledge
     - 4.2.1. Polarity Table
     - 4.2.2. WordNet Lexical Relations
   - 4.3. Pre-Processing
     - 4.3.1. Normal Forming and Transformations
     - 4.3.2. Polarity Marking
   - 4.4. Natural Logic Approximate Unification (NLAUnification)
   - 4.5. SATCHMO+
5. **Experiments and Evaluation**
   - 5.1. Test Sets
     - 5.1.1. FraCaS
     - 5.1.2. RTE
   - 5.2. Experimental Design
   - 5.3. Rsults and Discussion
6. **Conclusion and Future Work**

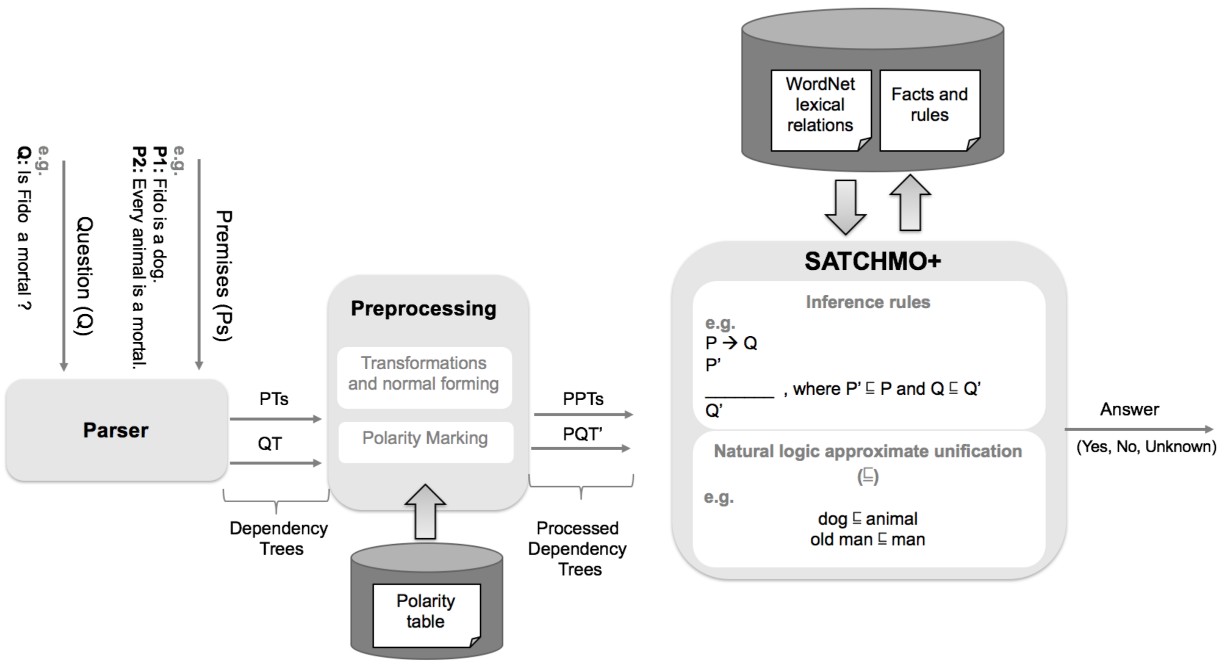


Figure 3: System designe.

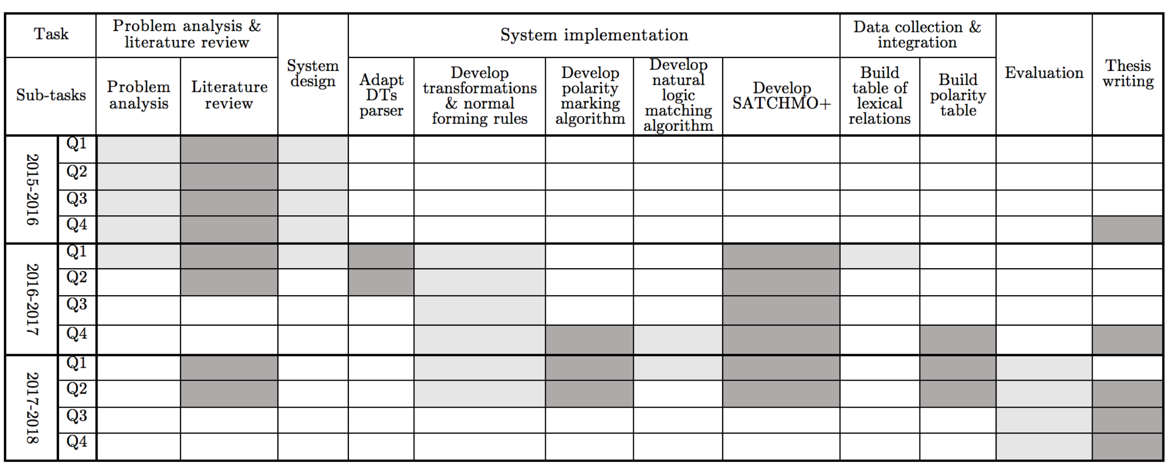


Figure 4: Plans timeline.

## References


[1] Keith Allan. Natural language semantics. 2001.

[2] Patrick Blackburn and Johan Bos. Representation and inference for natural language. *A first course in computational semantics. CSLI*, 2005.

[3] Johan Bos. A survey of computational semantics: Representation, inference and knowledge in wide-coverage text understanding. *Language and Linguistics Compass*, 5(6):336–366, 2011.

[4] Avron Barr. Natural language understanding. *AI Magazine*, 1(1):5, 1980.

[5] Dag Prawitz. Natural deduction: a proof-theoretical study. *Almquist and Wiksell*, 1965.

[6] Stanford Encyclopedia of Philosophy. Automated reasoning, 2014.

[7] George Lakoff. Linguistics and natural logic. In *Semantics of natural language*, pages 545–665. Springer, 1972.

[8] Bill MacCartney and Christopher D Manning. Natural logic for textual inference. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 193–200. Association for Computational Linguistics, 2007.

[9] Ovidiu Bagdasar. *Normal Forms, Proof and Argument*, pages 35–43. Springer International Publishing, Cham, 2013.

[10] Grigoris Antoniou, David Billington, Guido Governatori, and Michael J. Maher. Representation results for defeasible logic. *ACM Trans. Comput. Logic*, 2(2):255–287, April 2001.

[11] Rainer Manthey and François Bry. Satchmo: a theorem prover implemented in prolog. In *International Conference on Automated Deduction*, pages 415–434. Springer, 1988.

[12] Allan Ramsay. Theorem proving for untyped constructive $\lambda$-calculus: implementation and application. *Logic Journal of IGPL*, 9(1):83–100, 2001.

[13] Christiane Fellbaum. Wordnet. In *Theory and applications of ontology: computer applications*, pages 231–243. Springer, 2010.

[14] Ido Dagan, Oren Glickman, and Bernardo Magnini. The pascal recognising textual entailment challenge. In *Machine learning challenges. evaluating predictive uncertainty, visual object classification, and recognising tectual entailment*, pages 177–190. Springer, 2006.

[15] Roy Bar-Haim, Ido Dagan, Bill Dolan, Lisa Ferro, Danilo Giampiccolo, Bernardo Magnini, and Idan Szpektor. The second pascal recognising textual entailment challenge. In *Proceedings of the second PASCAL challenges workshop on recognising textual entailment*, volume 6, pages 6–4, 2006.

[16] Danilo Giampiccolo, Bernardo Magnini, Ido Dagan, and Bill Dolan. The third pascal recognizing textual entailment challenge. In *Proceedings of the ACL-PASCAL workshop on textual entailment and paraphrasing*, pages 1–9. Association for Computational Linguistics, 2007.

[17] Danilo Giampiccolo, Hoa Trang Dang, Bernardo Magnini, Ido Dagan, Elena Cabrio, and Bill Dolan. The fourth pascal recognizing textual entailment challenge. *TAC 2008 Proceedings*, 2008.

[18] Luisa Bentivogli, Ido Dagan, Hoa Trang Dang, Danilo Giampiccolo, and Bernardo Magnini. The fifth pascal recognizing textual entailment challenge. *Proceedings of TAC*, 9:14–24, 2009.

[19] Luisa Bentivogli, Ido Dagan, Hoa Trang Dang, Danilo Giampiccolo, and Bernardo Magnini. The sixth pascal recognizing textual entailment challenge. *Proceedings of TAC*, 2010.

[20] Luisa Bentivogli, Peter Clark, Ido Dagan, Hoa Dang, and Danilo Giampiccolo. The seventh pascal recognizing textual entailment challenge. *Proceedings of TAC*, 2011, 2011.

[21] Myroslava O Dzikovska, Rodney D Nielsen, Chris Brew, Claudia Leacock, Danilo Giampiccolo, Luisa Bentivogli, Peter Clark, Ido Dagan, and Hoa T Dang. Semeval-2013 task 7: The joint student response analysis and 8th recognizing textual entailment challenge. Technical report, DTIC Document, 2013.

[22] Robin Cooper, Richard Crouch, Jan van Eijck, Chris Fox, Josef van Genabith, Jan Jaspers, Hans Kamp, Manfred Pinkal, Massimo Poesio, Stephen Pulman, et al. Fracas: A framework for computational semantics. *Deliverable*, 8:62–051, 1994.

[23] Bill MacCartney and Christopher D Manning. Modeling semantic containment and exclusion in natural language inference. In *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*, pages 521–528. Association for Computational Linguistics, 2008.

[24] Bill MacCartney and Christopher D Manning. An extended model of natural logic. In *Proceedings of the eighth international conference on computational semantics*, pages 140–156. Association for Computational Linguistics, 2009.