Surface Reasoning

Lecture 5: Beyond Monotonicity

Thomas Icard

June 18-22, 2012

- So far we focused on two sorts of functions, and corresponding functional expressions: monotone and antitone.

- The basic insight behind the Monotonicity Calculus was that type domains for most functional expressions inherit a preorder from that associated with the basic truth type domain $(2, \leq)$.

- In fact, this ordered set has a lot more structure: it is also the smallest Boolean lattice: $(2, +, \cdot, 0, 1)$. The pre-ordering $\leq$ is then defined so that $x \leq y$ if and only if $x + y = y$. Moreover this structure is inherited by any domain for a type that ends in t:

Proposition

If $B = (B, \vee, \wedge, 0, 1)$ is a Boolean lattice and A is any set, the set of functions $f : A \longrightarrow B$ forms a Boolean lattice, in which $f \vee g(a) = f(a) \vee g(a)$, $f \wedge g(a) = f(a) \wedge g(a)$, and $\mathbf{0}$ and $\mathbf{1}$ are the constant functions sending all $a \in A$ to 0 and 1, respectively.

- Already in the case of predicates this gives us a whole host of new relations. Not only can we compare X and Y by inclusion, $X \subseteq Y$, but we can also talk about exclusion relations: $X \cap Y = \varnothing$, etc.

- Satisfyingly, these relations are projected by various functional expressions in predictable ways, just like inclusion relations.

- In this lecture we explore reasoning about inclusion and exclusion together, as a modest extension of the Monotonicity Calculus. One of the central points is that by incorporating exclusion relations we can actually derive new instances of inclusion.

- The fundamental insight behind this idea, as well as the practical applications to be discussed later, are due to Bill MacCartney [2, 3]. The formalization in the style of Monotonicity Calculus, including the type marking system and the function classes, and the connection to strong NPIs, is from a forthcoming paper of mine [1].

- Consider the following intuitively valid pattern:

$$\frac{\text{Every occupation that involves a giant squid is hazardous.}}{\text{Not every occupation that involves a cephalopod is safe.}}$$

- This involves one case of inclusion.

$$[\![\text{giant squid}]\!] \subseteq [\![\text{cephalopod}]\!].$$

- But it also involves two cases of exclusion. Informally,

$$[\![\text{hazardous}]\!] \cap [\![\text{safe}]\!] = \varnothing.$$

$$[\![\text{every}]\!] \cap [\![\text{not every}]\!] = \varnothing \text{ and } [\![\text{every}]\!] \cup [\![\text{not every}]\!] = \text{the universe}$$

$$\text{i.e., } \overline{[\![\text{every}]\!]} = [\![\text{not every}]\!],$$

- Can an example like this be captured with monotonicity reasoning?

$$\frac{\text{Every occupation that involves a giant squid is hazardous.}}{\text{Not every occupation that involves a cephalopod is safe.}}$$

- It seems these particular exclusion relations can be written as (boolean combinations of) inclusion relations:

$$[\![\text{hazardous}]\!] \subseteq \overline{[\![\text{safe}]\!]},$$

$$\overline{[\![\text{every}]\!]} \subseteq [\![\text{not every}]\!] \;\&\; [\![\text{every}]\!] \subseteq \overline{[\![\text{not every}]\!]}$$

- The problem is that we have not seen any special rules that allow us to substitute such terms in a validity preserving way. We will see shortly that we need more information about the quantifiers than just their monotonicity properties.

Definition (The Set $\mathcal{R}$ of Relations)

For any bounded, distributive lattice we define:

$$\begin{array}{lll} x \sqsubseteq y & x \wedge y = x & (x \leq y) \\ x \sqsupseteq y & x \vee y = x & (x \geq y) \\ x \mid y & x \wedge y = 0 & \\ x \smile y & x \vee y = 1 & \end{array}$$

We write $x \equiv y$ if both $x \sqsubseteq y$ and $x \sqsupseteq y$; write $x \curlywedge y$ if both $x \mid y$ and $x \smile y$; and write $x \# y$ for the universal (uninformative) relation. Thus we define the set $\mathcal{R}$ of relations to be:

$$\equiv, \sqsubseteq, \sqsupseteq, \curlywedge, \mid, \smile, \#.$$

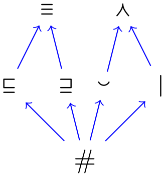
Examples:

- hazardous $\mid$ safe
- animate object $\smile$ non-human
- with $\curlywedge$ without
- juggles $\#$ pacifist

$(\mathcal{R}, \ll)$

The relations in $\mathcal{R}$ can be ordered according to:

$$R' \ll R \text{ just in case, whenever } xRy \text{, also } xR'y.$$



Lemma

In any bounded distributive lattice, if x and y are distinct from 0 and 1, there is a unique $\ll$-maximal $R \in \mathcal{R}$ such that xRy.

- For the relations $\sqsubseteq$ and $\sqsupseteq$, we have already studied classes of functions that project these in predictable ways: monotonic functions project $\sqsubseteq$ as $\sqsubseteq$ and $\sqsupseteq$ as $\sqsupseteq$, while antitonic functions reverse them.
- What about for the rest of the relations? Can we refine the class of functions usefully, beyond monotonic/antitonic/non-monotonic?
- The answer is positive. First recall the following characterizations of monotonic and antitonic functions:

Lemma

The following are (each) equivalent to f being monotone:

- $f(x) \vee f(y) \leq f(x \vee y)$;
- $f(x \wedge y) \leq f(x) \wedge f(y)$.

Lemma

The following are (each) equivalent to f being antitone:

- $f(x \vee y) \leq f(x) \wedge f(y)$;
- $f(x) \vee f(y) \leq f(x \wedge y)$.

- Our refined function classes result simply from turning each one of these '$\leq$' signs into an '$=$' sign.

Definition

1. f is *additive* if $f(x \vee y) = f(x) \vee f(y)$.
2. f is *multiplicative* if $f(x \wedge y) = f(x) \wedge f(y)$.
3. f is *anti-additive* if $f(x \vee y) = f(x) \wedge f(y)$.
4. f is *anti-multiplicative* if $f(x \wedge y) = f(x) \vee f(y)$.

- These function classes made an appearance in semantics through early work of Hoeksema and Zwarts.

Lemma

1. *The set of anti-additive functions from A to B is equal to the set of additive functions from A to B^{op}.*

2. *The set of anti-multiplicative functions from A to B is equal to the set of multiplicative functions from A to B^{op}.*

- To obtain function classes that project the relations in $\mathcal{R}$ in useful ways we need one extra property in each case:

Definition

1. f is *completely additive* if it is additive and $f(1) = 1$.

2. f is *completely multiplicative* if it is multiplicative and $f(0) = 0$.

3. f is *completely anti-additive* if it is anti-additive and $f(1) = 0$.

4. f is *completely anti-multiplicative* if it is anti-mult. and $f(0) = 1$.

- From here on, by X we mean *completely* X.

- For quantifiers, for example, this will amount to assuming non-triviality of predicate extensions: either $A \neq \varnothing$ or $A \neq \mathcal{P}(E)$.

* ‘Few’ fails all of these properties in its first argument.
* ‘At least two’ is (merely) monotone in both arguments.
* ‘If’ is (merely) antitone in its first argument.
* ‘Some’ is additive in both arguments.
* ‘No’ is anti-additive in both arguments.
* ‘Most’ is multiplicative in its second argument.
* ‘Not every’ is anti-multiplicative in its second argument.
* ‘Is’ is additive and multiplicative.
* ‘Not’ is anti-additive and anti-multiplicative.

- Since we have natural language realizations of all possible combinations of these function properties, we correspondingly introduce new type markings for each. Σ is the set of markings:

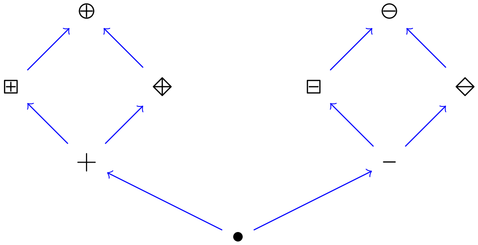
$$+, -, \diamond\!\!\!+, \diamond\!\!\!-, \boxplus, \boxminus, \oplus, \ominus, \bullet.$$

$+$: monotonic	$-$: antitonic
$\diamond\!\!\!+$: additive	$\diamond\!\!\!-$: anti-additive
$\boxplus$: multiplicative	$\boxminus$: anti-multiplicative
$\oplus$: additive and multiplicative	$\ominus$: anti-additive and anti-multiplicative
$\bullet$: non-monotonic	

$(\Sigma, \preceq)$

The set Σ of signatures also has a natural ordering:

$\psi \preceq \varphi$ just in case any φ-function is also a ψ-function.



Definition (Projection)

The *projection of* R *under* φ is the $\ll$-maximal $R^* \in \mathcal{R}$ for which:

Whenever xRy *and* f *is a* φ*-function*, $f(x)R^*f(y)$.

We write $[R]^\varphi$ for the projection of R under φ.

[ ]	$\sqsubseteq$	$\sqsupseteq$	$\curlywedge$	$\mid$	$\smile$
$+$	$\sqsubseteq$	$\sqsupseteq$	#	#	#
$\diamondplus$	$\sqsubseteq$	$\sqsupseteq$	$\smile$	#	$\smile$
$\boxplus$	$\sqsubseteq$	$\sqsupseteq$	$\mid$	$\mid$	#
$\oplus$	$\sqsubseteq$	$\sqsupseteq$	$\curlywedge$	$\mid$	$\smile$

[ ]	$\sqsubseteq$	$\sqsupseteq$	$\curlywedge$	$\mid$	$\smile$
$-$	$\sqsupseteq$	$\sqsubseteq$	#	#	#
$\diamondminus$	$\sqsupseteq$	$\sqsubseteq$	$\mid$	#	$\mid$
$\boxminus$	$\sqsupseteq$	$\sqsubseteq$	$\smile$	$\smile$	#
$\ominus$	$\sqsupseteq$	$\sqsubseteq$	$\curlywedge$	$\smile$	$\mid$

- Do we now have enough information to derive our example?

$$\frac{s_1 = \text{Every occupation that involves a giant squid is hazardous.}}{t = \text{Not every occupation that involves a cephalopod is safe.}}$$

We need to show $s_1 \sqsubseteq t$.

- We can assume 'every' has now has marked type $p \xrightarrow{\diamondminus} (p \xrightarrow{\boxplus} t)$. (Recall p is an abbreviation for $e \rightarrow t$.)

- Given the assumption that 'hazardous | safe', by one substitution, and using our projectivity chart from the last slide, we can conclude:

$$s_1 \mid s_2$$

$$s_2 = \text{Every occupation that involves a giant squid is safe.}$$

- Take now s_2 and consider s_3:

  $s_3 =$ Not every occupation that involves a giant squid is safe.

- If we know that 'not every $\curlywedge$ every', then our table shows:

$$s_2 \curlywedge s_3$$

- The expression 'not every' has type $p \xrightarrow{\oplus} (p \xrightarrow{\boxminus} t)$, so its first argument is in an additive position. Hence given that 'giant squid $\sqsubseteq$ cephalopod', we can further conclude that:

$$s_3 \sqsubseteq t$$

$$t = \text{Not every occupation that involves a cephalopod is safe.}$$

- As a result, we have:

$$s_1 \mid s_2 \curlywedge s_3 \sqsubseteq t$$

  So what?

Joining Relations

- To complete the derivation we need to know what follows from a sequence of relations between expressions. In general, if xRy and $yR'z$, for $R, R' \in \mathcal{R}$, what can we conclude about x and z?

Definition

The join of R and R', denoted $R \bowtie R'$, is the $\ll$-maximal relation $R^* \in \mathcal{R}$ such that, if xRy and $yR'z$ then xR^*z.

$\bowtie$	$\sqsubseteq$	$\sqsupseteq$	$\wedge$	$\mid$	$\smile$
$\sqsubseteq$	$\sqsubseteq$	#	$\mid$	$\mid$	#
$\sqsupseteq$	#	$\sqsupseteq$	$\smile$	#	$\smile$
$\wedge$	$\smile$	$\mid$	$\equiv$	$\sqsupseteq$	$\sqsubseteq$
$\mid$	#	$\mid$	$\sqsubseteq$	#	$\sqsubseteq$
$\smile$	$\smile$	#	$\sqsupseteq$	$\sqsupseteq$	#

- Returning to the previous example, we have:

$$((\mid \bowtie \wedge) \bowtie \sqsubseteq) = (\sqsubseteq \bowtie \sqsubseteq) = \sqsubseteq$$

Hence, $s_1 \sqsubseteq t$.

Signature Arithmetic

- The final missing ingredient is to understand the behavior of φ-functions within the scope of ψ-functions, for $\varphi, \psi \in \Sigma$.

- We saw that with $+$ and $-$, this is simply captured by positive/negative arithmetic. What about for the full set Σ?

$\cdot$	$+$	$-$	$\diamondplus$	$\diamondminus$	$\boxplus$	$\boxminus$	$\ominus$
$+$	$+$	$-$	$+$	$-$	$+$	$-$	$-$
$-$	$-$	$+$	$-$	$+$	$-$	$+$	$+$
$\diamondplus$	$+$	$-$	$\diamondplus$	$-$	$+$	$\boxminus$	$\boxminus$
$\diamondminus$	$-$	$+$	$\diamondminus$	$-$	$-$	$\boxplus$	$\boxplus$
$\boxplus$	$+$	$-$	$+$	$\diamondminus$	$\boxplus$	$-$	$\diamondminus$
$\boxminus$	$-$	$+$	$-$	$\diamondplus$	$-$	$+$	$\diamondplus$
$\ominus$	$-$	$+$	$\diamondminus$	$\diamondplus$	$\boxminus$	$\diamondplus$	$\oplus$

Lemma

$(\Sigma, \cdot, \oplus)$ *is a monoid.*

A Simple Relational Calculus

$$\frac{}{t \sqsubseteq t}$$

$$\frac{t \sqsubseteq t'}{t' \sqsupseteq t} \qquad \frac{t \sqsupseteq t'}{t' \sqsubseteq t} \qquad \frac{t \mid t'}{t' \mid t} \qquad \frac{t \smile t'}{t' \smile t}$$

$$\frac{t \mid t}{sRs'} \qquad \frac{tRu \qquad uR'v}{t(R \bowtie R')v}$$

Finally, where $o(s, t)$ denotes the projectivity of the occurrence of s in t:

$$\frac{sRs'}{t[R]^{\varphi}t^{s' \leftarrow s}}$$

provided $o(s, t) = \varphi$.

Example

- s_1 = Every occupation that involves a giant squid is $(\text{hazardous})^{\boxplus}$.
- s_2 = $(\text{Every})^{\oplus}$ occupation that involves a giant squid is safe.
- s_3 = Not every occupation that involves a $(\text{giant squid})^{\diamondplus}$ is safe.
- t = Not every occupation that involves a cephalopod is safe.

$$\dfrac{\dfrac{\dfrac{\text{hazardous} \mid \text{safe}}{s_1 \mid s_2} \qquad \dfrac{\text{every} \curlywedge \text{not every}}{s_2 \curlywedge s_3}}{s_1 \sqsubseteq s_3} \qquad \dfrac{\text{giant squid} \sqsubseteq \text{cephalopod}}{s_3 \sqsubseteq t}}{s_1 \sqsubseteq t}$$

- Within the domain of monotonicity reasoning, we said that our marked types were in some sense genuinely syntactic. Now having incorporated exclusion relations are we going deeper into semantics and moving beyond what could be called syntax?

- Arguably no. Early on, Zwarts, and later van der Wouden, showed that there are in fact different classes of NPIs that require different strengths of negation. Some, such as ‘yet’ or ‘any’, require mere antitonicity. Others seem to require anti-additivity, e.g. ‘in ages’. Yet others, such as ‘a bit’, seem to require both anti-additivity and anti-multiplicativity, i.e. full negation.

1. If you make any progress, let me know.

2. * If you have written about that in years, you should write a book.
3. * Not everyone has written about it in years.
4. No one has written about it in years.

5. * No one was a bit happy with the service.
6. Theodore was not a bit happy with the service.

Puzzle: What could possibly explain this connection?

- Much of what we have discussed in this course is of a more theoretical nature. One might wonder how much of this could actually be implemented in concrete applications.
- In this last part of the course, we will discuss work by Bill MacCartney [2], and by MacCartney and Manning [3], that demonstrates its viability.
- The setting is the PASCAL RTE Challenge, a contest that was organized for a number of years that involves detecting whether a hypothesis follows from a premise, contradicts the premise, or is consistent with the premise but is not entailed by it.
- Problem premises come from real sources and are typically rather long, while hypotheses are devised for the specific premise and tend to be shorter.

Example Problems

p: Sharon warns Arafat could be targeted for assassination

h: prime minister targeted for assassination

p: Twenty-five of the dead were members of the law enforcement agencies and the rest of the 67 were civilians

h: 25 of the dead were civilians

p: Mitsubishi Motors Corp.'s new vehicle sales in the US fell 46 percent in June

h: Mitsubishi sales rose 46 percent

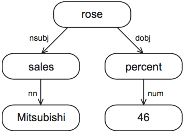
p: The main race track in Qatar is located in Shahaniya, on the Dukhan Road

h: Qatar is located in Shahaniya

Stanford RTE System

- Before summarizing MacCartney and Manning's NatLog system, which uses the ideas we have covered, we first give a quick overview of the Stanford RTE system, a closely related system.

- The system is split up into three stages:
  1. Generate a typed dependency graph for the premise, and one for the hypothesis. For example:



  2. Compute a partial alignment between the two graphs.
  3. Determine whether an entailment holds using a MaxEnt classifier with features learned from pairs (p, h).

- A number of the features used in the classifier are in large part motivated by semantics and semantic theory. E.g.:
  - Polarity: the presence of antitone contexts
  - Antinomy: whether aligned words are antonyms, and if so, whether they are in similar semantic contexts
  - Factivity: the presence of factive or non-factive verbs
- On the RTE Challenge the Stanford System achieve 59% accuracy and 65% confidence-weighted accuracy, which at the time (2006) was state-of-the-art.
- The motivation behind NatLog was to achieve yet higher precision, even if at the cost of recall.

NatLog

The NatLog algorithm is split up into five stages:

1. Linguistic analysis:
   - tokenization, lemmatization, parsing, NER, etc.
   - **monotonicity/projectivity marking**
2. Alignment: many-to-one with four edit types: EQ, SUB, DEL, INS.
3. Lexical entailment classification: for different possible edits, determine the right relation.
   - Most cases are learned with a classifier from hand labeled examples.
   - Some special cases are hand-coded, e.g. SUB(all, none) $= |$.
   - This is analogous to our lexical assumptions in the earlier discussion.
4. Using stages 1 and 3, determine the overall relation that holds between each edit. Roughly, and in the vocabulary of our earlier discussion, this involves projecting the relation through the functions that take it as argument.
5. Join the sequence of edits together using the join table as we presented earlier.

- On the RTE Challenge, NatLog was significantly higher than the Stanford System on precision (70% versus 61%), lower on recall (36% versus 60%), and roughly comparable on overall accuracy.
- The interesting result, however, is that the combination of the Stanford System and NatLog in a *hybrid* system outperformed both.
- The Stanford System assigns a score to a pair and chooses based on a learned threshold. The hybrid system adjusts this score by $+\delta$ or $-\delta$, depending on the prediction given by NatLog.
- The value of δ is learned, and the resulting system achieves a 4% gain in accuracy, leading to 64% accuracy overall.

Summary

- The guiding idea of this course has been that much of the logic of natural language can be captured by appealing merely to 'surface level' information. We saw many argument patterns that can be nicely captured using deductive rules operating directly over parsed natural language sentences, with no translation to a separate system.

- The striking parallels between these argument patterns and the distribution of NPIs suggests that much of this logic is closely related to syntax. Understanding this situation better would be worthwhile.

- We also saw that many of these inference patterns are quite easy and efficient for humans and computers alike. It seems that computational linguistics and psychology of language can benefit from clear-headed theoretical proposals. But also *vice versa*, theoretical work should be appropriately sensitive to, and perhaps even motivated by, psychological and computational considerations.

- T. Icard. 'Exclusion and Containment in Natural Language', forthcoming in *Studia Logica*, 2012.
- B. MacCartney. *Natural Language Inference*, Ph.D. Dissertation, Stanford University, 2009.
- B. MacCartney and C. Manning. 'An Extended Model of Natural Logic', in *The Eighth International Conference on Computational Semantics (IWCS-8)*, 2009
- L.S. Moss. *Logics for Natural Language Inference*, ESSLLI course notes, 2010.