Logics and Statistics for Language Modeling

Carlos Areces

areces@loria.fr
http://www.loria.fr/~areces/ls
INRIA Nancy Grand Est
Nancy, France

2009/2010

The Course

- Two Topics - Two Parts - Two Teachers

What?	When?	Who?
Logic	Monday	Carlos Areces
Statistics	Tuesday	Kamel Smaili

For Logic

- Course in English.
- Site of the course http://www.loria.fr/~areces/ls
  Slides will be available there.
- Evaluation.
  - Exercises will be handed out at each lecture and must be returned the following week. They should be solved in teams of 2 or 3 students. They will be graded. Exercises that are not returned on the due date will be graded as 0.
  - There will be a written exam at the end of the course.
  - Grades will be calculated as follows:

Class participation:	5 %
Exercises:	25 %
Final Exam:	70 %

Today's Program

- Quick Poll: What do you know about logic?
- Propositional Logics
  - Syntax and Semantics
  - Simple Application
  - More Interesting Applications (Graph Coloring)
  - Decision Methods

Computational Logics: Logic in Action!

- Logic was born as part of philosophy:
  - originally meant to model human reasoning processes
  - and to help making correct correct inferences
- With the advent of computer science, things changed
  - logic actually played a fundamental part in the development of computers (logic circuits)
  - but nowadays computer science fuels logic
- In this course we will take a computational view on logical calculi. Discussing in particular how logic could be used in linguistic applications.

Why do we Need Computers?

- Why do we need computers?
  - well, after all we are lazy and we don't want to do we work if somebody else can do it for us
  - even if we could overcome our laziness, we wouldn't be able to do the task ourselves
- Some of the inference tasks we want to tackle are simply too difficult to perform without the help of computers
  - sometimes billions of possibilities need to be checked to verify that a system satisfies a certain property we want to enforce
  - and even using computers we need to be clever, or all the time till the end of the universe won't be enough.

Propositional Logic

- The language of propositional logic (PL) is very easy:
  Some propositional symbols: $p_1, p_2, p_3, \ldots$
  Two logical symbols: $\neg, \vee$
  Two syntactic symbols: (,)
- What is a formula?

Every propositional symbol p_i is a formula in PL
If φ is a formula in PL then $\neg(\varphi)$ is a formula in PL
If φ and ψ are formulas in PL then $(\varphi \vee \psi)$ is a formula in PL

Propositional Logic

- The semantics is also straightforward:

$$\begin{array}{rcl} \neg(\phi) \text{ is true} & \text{iff} & \phi \text{ is false} \\ \neg(\phi) \text{ is false} & \text{iff} & \phi \text{ is true} \\ (\phi \vee \psi) \text{ is true} & \text{iff} & \text{either } \phi \text{ or } \psi \text{ are true} \\ (\phi \vee \psi) \text{ is false} & \text{iff} & \text{both } \phi \text{ and } \psi \text{ are false} \end{array}$$

- Given an assignment V of true or false to the propositional symbols we can determine for any formula whether it is true or false under V.
- We can define additional logical symbols in terms of $\neg$ and $\vee$

$$\begin{array}{rcl} (\varphi \& \psi) & \equiv & \neg(\neg(\varphi) \vee \neg(\psi)) \\ (\varphi \rightarrow \psi) & \equiv & (\neg(\varphi) \vee \psi) \end{array}$$

A Diplomatic Problem

You are chief of protocol for the embassy ball. The crown prince instructs you either to invite Peru or to exclude Qatar. The queen asks you to invite either Qatar or Romania or both. The king, in a spiteful mood, wants to snub either Romania or Peru or both. Who do you invite?

- Can we model this using just propositional logic?
- And what do we gain by doing that?
- What kind of questions can we "ask" our model?

Formalizing the Diplomatic Problem

- Three propositional symbols

P	$\equiv$	invite Peru	$\neg P$	$\equiv$	exclude Peru
Q	$\equiv$	invite Qatar	$\neg Q$	$\equiv$	exclude Qatar
R	$\equiv$	invite Romania	$\neg R$	$\equiv$	exclude Romania

- The problem can be formalized as

prince: $P \vee \neg Q$	$\equiv$	invite Peru or exclude Qatar
queen: $Q \vee R$	$\equiv$	invite Qatar or Romania or both
king: $\neg R \vee \neg P$	$\equiv$	snub Romania or Peru or both

- $\Sigma = (P \vee \neg Q) \ \& \ (Q \vee R) \ \& \ (\neg R \vee \neg P)$

Solving the Diplomatic Problem

- What can we deduce from Σ?

$$\frac{\text{prince: } P \vee \neg Q \qquad \text{queen: } Q \vee R}{P \vee R}$$

hence from Σ it follows that $P \vee R$ (that is, if Σ is true then $P \vee R$ is true.)

- One consequence of satisfying the prince and the queen is that we must invite Peru or Romania (or both).
- Is Σ satisfiable? 2 out of 8 possible truth assignments satisfy Σ

P = true Q = true R = false
P = false Q = false R = true

So either invite Peru and Qatar and not Romania or invite Romania and not Peru and Qatar

Solving the Diplomatic Problem 2

- How can we compute this solution?
- We can use truth tables

P	Q	R	$P \vee \neg Q$	$Q \vee R$	$\neg R \vee \neg P$	Σ
T	T	T	T	T	F	F
T	T	F	T	T	T	T
T	F	T	T	T	F	F
T	F	F	T	F	T	F
F	T	T	F	T	T	F
F	T	F	F	T	T	F
F	F	T	T	T	T	T
F	F	F	T	F	T	F

- But this uses exponential space... while the complexity of the SAT problem for PL is NP

Some of the Techniques for SAT Solving

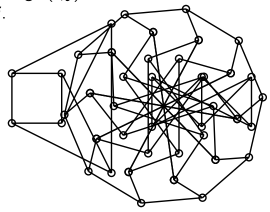
- Complete methods
  - resolution
  - tableaux
  - Davis-Putman
  - map into linear equations
- Approximation procedures
  - flip the value of a variable in an unsatisfied clause
  - genetic algorithms
  - hill-climbing

More than Diplomacy

- We just saw a simple use of propositional logic in our diplomatic problem
- But the expressive power of PL is enough for doing many more interesting things:
  - graph coloring
  - constraint satisfaction problems (e.g., Sudoku)
  - hardware problems
  - planning
  - scheduling

Applications: Graph Coloring

- The Problem: Given a graph $G = \langle N, E \rangle$ where N is a set of nodes and E a set of edges, and a fixed number k of colors. Decide if we can assing colors to nodes in N s.t.
  - All nodes are colored with one of the k colors.
  - For every edge $(i, j) \in N$, the color of i is different from the color of j.
- An Example:



Graph Coloring: The Nitty-Gritty Details

- We will use $n \times k$ propositional symbols that we write p_{ij} (n is the number of nodes in N, k the number of colors)
- We will read p_{ij} as node i has color j
- We have to say that
  1. Each node has (at least) one color.
  2. Each node has no more than one color.
  3. Related nodes have different colors.

1. Each node has one color: $p_{i1} \vee \ldots \vee p_{ik}$, for $1 \leq i \leq n$
2. Each node has no more than one color: $\neg p_{il} \vee \neg p_{im}$, for $1 \leq i \leq n$, and $1 \leq l < m \leq k$
3. Neighboring nodes have different colors. $\neg p_{il} \vee \neg p_{jl}$, for i and j neighboring nodes, and $1 \leq l \leq k$

Applications: Graph Coloring 3

- Results:
  - the GSAT and WalkSAT algorithms seem to be competitive with specialized graph coloring algorithms
- An application to algebra:
  - problems dealing with quasigroups can be viewed as specialized coloring problems
  - some open problems on quasigroups have been encoded in this way and solved automatically by propositional provers. E.g. is there a quasigroup satisfying the equations

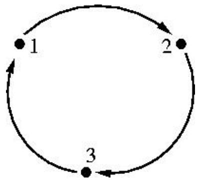
$$\forall a.(a \cdot a) = a$$
$$\forall a, b.((b \cdot a) \cdot b) \cdot b = a$$

Decision Methods for PL

- The most traditional methods for solving the SAT problem for propositional logics (PL-SAT) behave as follows. For any input formula φ
  - They always answer SATISFIABLE or UNSATISFIABLE.
  - After a finite time.
  - They always answer correctly.
- The best known complete methods probably are
  - truth tables
  - tableaux
  - axiomatics, Gentzen calculi, natural deduction, resolution
  - Davis-Putnam

Exercises

Use the graph coloring coding in the following graph, using 2 colors (R and B):



Show the first three lines of a truth table for the formula obtained in the codification.