Basics of AI and Machine Learning

Propositional Logic: DPLL Algorithm

Daniel Gnad

Linköping University

Motivation

Propositional Logic: Motivation

- Propositional logic allows for the representation of knowledge and for deriving conclusions based on this knowledge.
- many practical applications can be directly encoded, e.g.
  - constraint satisfaction problems of all kinds
  - circuit design and verification
- many problems contain logic as ingredient, e.g.
  - automated planning
  - general game playing
  - description logic queries (semantic web)

Propositional Logic: Algorithmic Problems

main problems:

- reasoning ($\Theta \models \varphi$?):
  Does the formula φ logically follow from the formulas Θ?
- equivalence ($\varphi \equiv \psi$):
  Are the formulas φ and ψ logically equivalent?
- satisfiability (SAT):
  Is formula φ satisfiable? If yes, find a model.

The Satisfiability Problem

The Satisfiability Problem (SAT)

given:

propositional formula in conjunctive normal form (CNF)

usually represented as pair $\langle V, \Delta \rangle$:

- V set of propositional variables (propositions)
- Δ set of clauses over V
  (clause $=$ set of literals v or $\neg v$ with $v \in V$)

find:

- satisfying interpretation (model)
- or proof that no model exists

SAT is a famous NP-complete problem (Cook 1971; Levin 1973).

Relevance of SAT

- The name "SAT" is often used for the satisfiability problem for general propositional formulas (instead of restriction to CNF).
- General SAT can be reduced to CNF.
- All previously mentioned problems can be reduced to SAT.

$\leadsto$ SAT algorithms important and intensively studied

this chapter: SAT algorithms

Systematic Search: DPLL

The DPLL Algorithm: Pseudo-Code

function DPLL(Δ, I):

if $\Box \in \Delta$: [empty clause exists $\rightsquigarrow$ unsatisfiable]
  return unsatisfiable
else if $\Delta = \emptyset$: [no clauses left $\rightsquigarrow$ interpretation I satisfies formula]
  return I
else if there exists a unit clause $\{v\}$ or $\{\neg v\}$ in Δ: [unit propagation]
  Let v be such a variable, d the truth value that satisfies the clause.
  $\Delta' := \text{simplify}(\Delta, v, d)$
  return DPLL($\Delta', I \cup \{v \mapsto d\}$)
else: [splitting rule]
  Select some variable v which occurs in Δ.
  for each $d \in \{\mathbf{F}, \mathbf{T}\}$ in some order:
    $\Delta' := \text{simplify}(\Delta, v, d)$
    $I' := \text{DPLL}(\Delta', I \cup \{v \mapsto d\})$
    if $I' \neq$ **unsatisfiable**
      return I'
  return unsatisfiable

The DPLL Algorithm: simplify

function simplify(Δ, v, d)

Let ℓ be the literal for v that is satisfied by $v \mapsto d$.
$\Delta' := \{C \mid C \in \Delta \text{ such that } \ell \notin C\}$
$\Delta'' := \{C \setminus \{\bar{\ell}\} \mid C \in \Delta'\}$
return Δ''

Example (1)

$$\Delta = \{\{X, Y, \neg Z\}, \{\neg X, \neg Y\}, \{Z\}, \{X, \neg Y\}\}$$

Example (1)

$$\Delta = \{\{X, Y, \neg Z\}, \{\neg X, \neg Y\}, \{Z\}, \{X, \neg Y\}\}$$

1. unit propagation: $Z \mapsto \mathbf{T}$

Example (1)

$$\Delta = \{\{X, Y, \neg Z\}, \{\neg X, \neg Y\}, \{Z\}, \{X, \neg Y\}\}$$

1. unit propagation: $Z \mapsto \mathbf{T}$
   $\{\{X, Y\}, \{\neg X, \neg Y\}, \{X, \neg Y\}\}$

Example (1)

$$\Delta = \{\{X, Y, \neg Z\}, \{\neg X, \neg Y\}, \{Z\}, \{X, \neg Y\}\}$$

1. unit propagation: $Z \mapsto \mathbf{T}$
   $\{\{X, Y\}, \{\neg X, \neg Y\}, \{X, \neg Y\}\}$
2. splitting rule:

Example (1)

$$\Delta = \{\{X, Y, \neg Z\}, \{\neg X, \neg Y\}, \{Z\}, \{X, \neg Y\}\}$$

1. unit propagation: $Z \mapsto \mathbf{T}$
   $\{\{X, Y\}, \{\neg X, \neg Y\}, \{X, \neg Y\}\}$
2. splitting rule:

2a. $X \mapsto \mathbf{F}$
$\{\{Y\}, \{\neg Y\}\}$

Example (1)

$$\Delta = \{\{X, Y, \neg Z\}, \{\neg X, \neg Y\}, \{Z\}, \{X, \neg Y\}\}$$

1. unit propagation: $Z \mapsto \mathbf{T}$
   $\{\{X, Y\}, \{\neg X, \neg Y\}, \{X, \neg Y\}\}$
2. splitting rule:

2a. $X \mapsto \mathbf{F}$
$\{\{Y\}, \{\neg Y\}\}$

3a. unit propagation: $Y \mapsto \mathbf{T}$
$\{\square\}$

Example (1)

$$\Delta = \{\{X, Y, \neg Z\}, \{\neg X, \neg Y\}, \{Z\}, \{X, \neg Y\}\}$$

1. unit propagation: $Z \mapsto \mathbf{T}$
   $\{\{X, Y\}, \{\neg X, \neg Y\}, \{X, \neg Y\}\}$
2. splitting rule:

2a. $X \mapsto \mathbf{F}$
$\{\{Y\}, \{\neg Y\}\}$

3a. unit propagation: $Y \mapsto \mathbf{T}$
$\{\Box\}$

2b. $X \mapsto \mathbf{T}$
$\{\{\neg Y\}\}$

Example (1)

$$\Delta = \{\{X, Y, \neg Z\}, \{\neg X, \neg Y\}, \{Z\}, \{X, \neg Y\}\}$$

1. unit propagation: $Z \mapsto \mathbf{T}$
   $\{\{X, Y\}, \{\neg X, \neg Y\}, \{X, \neg Y\}\}$
2. splitting rule:

2a. $X \mapsto \mathbf{F}$
$\{\{Y\}, \{\neg Y\}\}$

3a. unit propagation: $Y \mapsto \mathbf{T}$
$\{\square\}$

2b. $X \mapsto \mathbf{T}$
$\{\{\neg Y\}\}$

3b. unit propagation: $Y \mapsto \mathbf{F}$
$\{\}$

Example (1)

$$\Delta = \{\{X, Y, \neg Z\}, \{\neg X, \neg Y\}, \{Z\}, \{X, \neg Y\}\}$$

1. unit propagation: $Z \mapsto \mathbf{T}$
   $\{\{X, Y\}, \{\neg X, \neg Y\}, \{X, \neg Y\}\}$
2. splitting rule:

2a. $X \mapsto \mathbf{F}$
$\{\{Y\}, \{\neg Y\}\}$

3a. unit propagation: $Y \mapsto \mathbf{T}$
$\{\Box\}$

2b. $X \mapsto \mathbf{T}$
$\{\{\neg Y\}\}$

3b. unit propagation: $Y \mapsto \mathbf{F}$
$\{\}$

Example (2)

$$\Delta = \{\{W, \neg X, \neg Y, \neg Z\}, \{X, \neg Z\}, \{Y, \neg Z\}, \{Z\}\}$$

Example (2)

$$\Delta = \{\{W, \neg X, \neg Y, \neg Z\}, \{X, \neg Z\}, \{Y, \neg Z\}, \{Z\}\}$$

1. unit propagation: $Z \mapsto \mathbf{T}$

Example (2)

$$\Delta = \{\{W, \neg X, \neg Y, \neg Z\}, \{X, \neg Z\}, \{Y, \neg Z\}, \{Z\}\}$$

1. unit propagation: $Z \mapsto \mathbf{T}$
   $\{\{W, \neg X, \neg Y\}, \{X\}, \{Y\}\}$

Example (2)

$$\Delta = \{\{W, \neg X, \neg Y, \neg Z\}, \{X, \neg Z\}, \{Y, \neg Z\}, \{Z\}\}$$

1. unit propagation: $Z \mapsto \mathbf{T}$
   $\{\{W, \neg X, \neg Y\}, \{X\}, \{Y\}\}$
2. unit propagation: $X \mapsto \mathbf{T}$
   $\{\{W, \neg Y\}, \{Y\}\}$

Example (2)

$$\Delta = \{\{W, \neg X, \neg Y, \neg Z\}, \{X, \neg Z\}, \{Y, \neg Z\}, \{Z\}\}$$

1. unit propagation: $Z \mapsto \mathbf{T}$
   $\{\{W, \neg X, \neg Y\}, \{X\}, \{Y\}\}$
2. unit propagation: $X \mapsto \mathbf{T}$
   $\{\{W, \neg Y\}, \{Y\}\}$
3. unit propagation: $Y \mapsto \mathbf{T}$
   $\{\{W\}\}$

Example (2)

$$\Delta = \{\{W, \neg X, \neg Y, \neg Z\}, \{X, \neg Z\}, \{Y, \neg Z\}, \{Z\}\}$$

1. unit propagation: $Z \mapsto \mathbf{T}$
   $\{\{W, \neg X, \neg Y\}, \{X\}, \{Y\}\}$
2. unit propagation: $X \mapsto \mathbf{T}$
   $\{\{W, \neg Y\}, \{Y\}\}$
3. unit propagation: $Y \mapsto \mathbf{T}$
   $\{\{W\}\}$
4. unit propagation: $W \mapsto \mathbf{T}$
   $\{\}$

Example (2)

$$\Delta = \{\{W, \neg X, \neg Y, \neg Z\}, \{X, \neg Z\}, \{Y, \neg Z\}, \{Z\}\}$$

1. unit propagation: $Z \mapsto \mathbf{T}$
   $\{\{W, \neg X, \neg Y\}, \{X\}, \{Y\}\}$
2. unit propagation: $X \mapsto \mathbf{T}$
   $\{\{W, \neg Y\}, \{Y\}\}$
3. unit propagation: $Y \mapsto \mathbf{T}$
   $\{\{W\}\}$
4. unit propagation: $W \mapsto \mathbf{T}$
   $\{\}$

Properties of DPLL

- DPLL is sound and complete.
- DPLL computes a model if a model exists.
  - Some variables possibly remain unassigned in the solution I; their values can be chosen arbitrarily.
- time complexity in general exponential

$\rightsquigarrow$ important in practice: good variable order and additional inference methods (in particular clause learning)

- Best known SAT algorithms are based on DPLL.

Summary

Summary

- satisfiability basic problem in propositional logic to which other problems can be reduced
- here: satisfiability for CNF formulas
- Davis-Putnam-Logemann-Loveland procedure (DPLL): systematic backtracking search with unit propagation as inference method
- DPLL successful in practice, in particular when combined with other ideas such as clause learning