Project proposals

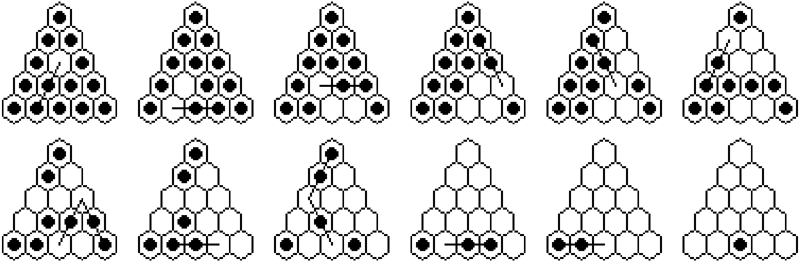
Luciano Serafini

FBK-IRST, Trento, Italy

June 3, 2019

Search using SAT

Triangular peg solitaire is a puzzle most commonly played on the 15-hole board shown on the left. To play start start from a position with one hole open and jump one peg over another, removing the peg that was jumped over. The goal is to finish at a board position with only one peg. Jumps are allowed along three directions parallel to the edges of the board. A possible solution of the puzzle is the following:

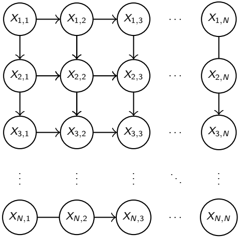


Search using SAT

- Using SAT, build a system in which you can specify the initial position of the hole, and the system is capable to generates a sequence of moves that leads to the solution.
- Generalize you system to any triangular peg puzzle with n elements on the edge.

Solving bayesian networks with Model Counting

Consider the following bayesian network: suppose that for every $X_{i,j}$ the CPT w.r.t., $X_{i-1,j}$ and $X_{i,j-1}$ is defined as shown in the table, where $0 \leq p_{ij}(x, y) \leq 1$.



$X_{i-1,j}$	$X_{i,j-1}$	$P(X_{ij} \mid X_{i-1,j}, X_{i,j-1})$
0	0	$p_{ij}(0,0)$
0	1	$p_{ij}(0,1)$
1	0	$p_{ij}(1,0)$
1	1	$p_{ij}(1,1)$

$X_{i-1,1}$	$P(X_{i,1} \mid X_{i-1,1})$
0	$p_{i1}(0)$
1	$p_{i1}(1)$

$X_{1,j-1}$	$P(X_{1,j} \mid X_{1,j-1})$
0	$p_{1j}(0)$
1	$p_{1j}(1)$

- implement a function that computes the value of $P(X_{N.N} \mid X_{1,1})$ for any value of $X_{1,1}$ and $X_{N,N}$ given the CPT. (N and p_{ij} are parameters).
- Implement a naive version of weighted model counting according to the DPLL algorithm described in class or
- download some platform that implements exact or approximated weighted model counting and:
- solve the conditional inference $P(X_{N,N} \mid X_{1,1})$ with your system.
- compare the two results.

Cost-Optimal Correlation Clustering

[1] Correlation clustering is a well-studied NP-hard clustering problem that finds important applications in various domains such as biosciences, and social network analysis and information retrieval.

The Problem

Given a symmetric similarity matrix W, the task is to find a function $cl : V \to \mathbb{N}$ minimizing the correlation clustering cost function:

$$\sum_{\substack{cl(v_i) \neq cl(v_j) \\ i<j}} \mathbb{1}_{0 < W_{ij} < \infty} \cdot W_{ij} - \sum_{\substack{cl(v_i) = cl(v_j) \\ i<j}} \mathbb{1}_{-\infty < W_{ij} < 0} \cdot W_{ij}$$

V	f_1	f_2	f_3	...
v_1	0.5	1	3	...
v_2	-3	0	-2	...
v_3	0.7	1	5	...
v_4	4	1	7	...
v_5	6	0	10	...

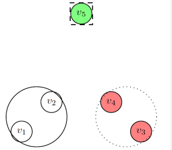
DATA

$$\Rightarrow W = \begin{bmatrix} 0 & 1 & 0.7 & 0 & 0.2 \\ 1 & 0 & 4 & -7 & -5 \\ 0.7 & 4 & 0 & \infty & 0 \\ 0 & -7 & \infty & 0 & -3 \\ 0.2 & -5 & 0 & -3 & 0 \end{bmatrix} \Rightarrow$$

SIMILARITY MATRIX

MAXSAT:
encoding
+
solving

$\Rightarrow$



SOLUTION CLUSTERING

[1] Jeremias Berg, Antti Hyttinen, and Matti Jarvisalo. "Applications of MaxSAT in Data Analysis". In: *Proceedings of Pragmatics of SAT 2015 and 2018*. Ed. by Daniel Le Berre and Matti Jarvisalo. Vol. 59. EPiC Series in

Clustering the Iris flower dataset

The data set consists of 50 samples from each of three species of Iris (Iris setosa, Iris virginica and Iris versicolor). Four features were measured from each sample: the length and the width of the sepals and petals, in centimeters.

Iris Setosa

Iris Versicolor

Iris Virginica

Use a MaxSat method to cluster the items of this dataset and compare the results with the groundtruth provided by the dataset.

Rule mining

Objective

Analyze the correlations between the votes of the 1984 United States Congressman, by using Apriori algorithm or using SAT based approach described in class.

Congressional Voting Records Data Set

It includes votes for each of the U.S. House of Representatives Congressmen on the 16 key votes. There are three type of votes, yea (y), nay (n), unknown disposition (?)

1. `Class Name: 2 (democrat, republican)`
2. `handicapped-infants: 2 (y,n)`
3. `water-project-cost-sharing: 2 (y,n)`
4. `adoption-of-the-budget-resolution: 2 (y,n)`
5. `...`
16. `duty-free-exports: 2 (y,n)`
17. `export-administration-act-south-africa: 2 (y,n)`

Rule mining

Task

Exetract rules of the form:

$$\texttt{democrat} \Rightarrow \texttt{vote}_1 \tag{1}$$

$$\texttt{republican} \Rightarrow \texttt{vote}_1 \tag{2}$$

$$\texttt{democrat}, \texttt{vote}_1 \Rightarrow \texttt{vote}_2 \tag{3}$$

$$\texttt{republican}, \texttt{vote}_1 \Rightarrow \texttt{vote}_2 \tag{4}$$

$$\texttt{vote}_1, \texttt{vote}_2 \rightarrow \texttt{democrat} \tag{5}$$

$$\texttt{vote}_1, \texttt{vote}_2 \rightarrow \texttt{republican} \tag{6}$$

$$\texttt{vote}_1 \rightarrow \texttt{vote}_2 \tag{7}$$

$$\texttt{vote}_1, \texttt{vote}_2 \rightarrow \texttt{vote}_3 \tag{8}$$

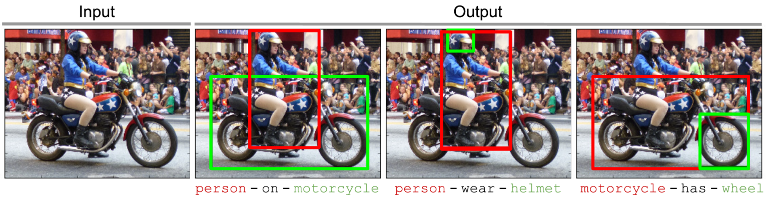
Visual Rule Mining (join project

Objective

Extract and use rules of co-occurrence of objects in a picture to improve object recognition. First: extract co-occurrence rules as for instance

- $car \rightarrow road, (89\%)$
- $bottle, glass \rightarrow plate(75\%)$

Dataset

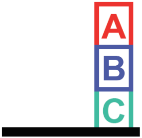


Visual Relationship Detection dataset

https://cs.stanford.edu/people/ranjaykrishna/vrd/

Reasoning in First Order Logic

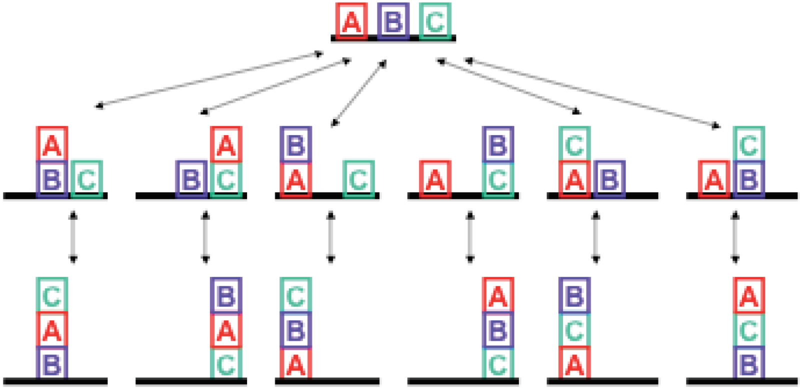
Initial state



Goal state

- **Initial state:** $On(A, table) \wedge On(B, table) \wedge On(C, table) \wedge Clear(A) \wedge Clear(B) \wedge Clear(C)$
- **Goal state:** $On(A, B) \wedge On(B, C)$
- **Actions:** $Move(b, x, y)$
    - **Precondition:** $On(b, x) \wedge Clear(b) \wedge Clear(y)$
    - **Effect:** $On(b, y) \wedge Clear(x) \wedge \neg On(b, x) \wedge \neg Clear(y)$
- **Action:** $MoveToTable(b, x)$
    - **Preconditions:** $On(b, x) \wedge Clear(b)$
    - **Effect:** $On(b, Table) \wedge Clear(x) \wedge \neg On(b, x)$

State Transitions in the Blocks World



Planning as first order theorem prooving

Task

planning problem of finding a plan (= sequence of actions) in first order logic and use Prover9 to find a plan for the initial state to the goal state.

Suggestion

- extend each predicate with a situation parameter

$$On(x, y) \qquad \Rightarrow \qquad On(x, y, s)$$

- actions are functions on states

$$MoveToTable(b, x) \qquad \Rightarrow \qquad MoveToTable(b, x, s)$$

  Intuitively $MoveToTable(b, x, s)$ is the state obtained after moving the block b on table aplied in the state s.

- Actions allows to pass from one state to the next state:

$$\begin{aligned} On(b, x, s) \wedge Clear(b, x) \rightarrow & On(b, Table, MoveToTable(b, x) \\ & \wedge Clear(x, MoveToTable(b, x)) \\ & \wedge \neg On(b, x, MoveToTable(b, x)) \end{aligned}$$

Improve character recognition with constraints

The MNIST dataset of handwritten digits

Task

Use LTN to recognize 3 digits at a time, knowing that one of them is the sum of the other. ()

Suggestion

- Define the predicates $Zero(x), One(x), Two(x) \ldots;$
- to each predicate attach the network that recognize digid (there are meny available in the main machine learning libraries (e.g. tensorflow, Keras))
- Encode the constraint in First order formula. E.g., $Six(x_1) \wedge Three(x_3) \rightarrow Nine(x_2)$