CSE 513T Theory of Artificial Intelligence & Machine Learning Oct 25, 2016

Lecture 16

Lecturer: Brendan Juba *Scribe: Shitianyu Pan*

1 Review

Last time we saw resolution proofs capture when SAT-solvers detect unsatisfiability. (If the algorithm terminates quickly, then there is a small resolution refutation.)

2 What we want

If there is a small proof then our algorithm finds one quickly. Trick, on HW3, space S tree like resolution proofs are those with graphs that are rank $s-1$ trees. Search for the smaller branch first,

3 Algorithm

$Space - RES(s, \varphi)$

if φ *has* $\perp$ *or* $\varphi = 0$ **then**
  return 1
end
if φ *= 1* **then**
  return fail
else if $s > 1$ **then**
  for *each literal* l *appearing in* φ **do**
    if $Space - RES(s-1, \varphi|_{l=1}) = \pi^l (\neq Fail)$ **then**
      if $Space - RES(s, \varphi|_{l=0}) = \pi^l (\neq Fail)$ **then**
        return corresponding proof for π^l, $\pi^{\neg l}, \perp$
      end
      return Fail
    end
  end
return Fail

- **Claim**: $Space - RES(s, \varphi)$ runs in time $O(|\varphi| n^{2s-1})$ and return a space S treelike refutation if one exists
- **Pf**: We use the fact that a resolution proof is treelike and uses space s iff its graph is a rank $s-1$ tree. We?ll argue by induction on S and n that the algorithm is correct.
- **Base case**: $s = 1$ or no variables, (rank 0 $\Leftrightarrow$ single node), the only possible refutation is $\perp$.
- **Induction step**: since our two recursive calls to Space-RES use space $s-1$ or have space s and $n-1$ variables, by IH, the algorithms runs respectively in time $\leqslant C|\varphi|(n-1)^{2(s-1)-1}$ for the space $s-1$ calls, and in time $\leqslant C|\varphi|(n-1)^{2s-1}$ for a space s call, and correctly reports whether or not there is a refutation of $\varphi|_{l=0}$, so if there is a space s refutation of φ, when we consider the literal l such that the branch of the proof in which we derive $\neg l$,
- Claim: Suppose that φ has a space s treelike resolution proof, and p is any partial assignment, then $\varphi|_p$ also has a space s treelike resolution refutation. Moreover applying the restriction p to each step of the proof for φ gives a proof for $\varphi|_p$ so therefore, under the restriction $l = 1, \neg l|_{l=1} \perp$

- so for the rank $s-2$ (space $s=1$) branch of the proof , Space-RES will return a refutation of space $s-1$, similarly considering the larger branch that derives l, $l|_{l=0} = \perp$ there is a space s refutation of $\varphi|_{l=0}$, so Space-RES must find and return a refutation of space s.
- The corresponding derivation on the classes of φ either derives $\perp$ in one of π^l or $\pi^{\neg l}$, or else π^l or $\pi^{\neg l}$ derives l and $\neg l$. So in either case, we can conclude the proof.
- Note that for any l such that Space-RES finds a space $s-1$ refutation we know that φ was unsatisfiable by any assignment in which $l=1$, so φ overall is unsatisfiable iff $\varphi|_{l=0}$, is unsatisfiable in particular, if we take our original refutation of φ of space s, we can still obtain a space s refutation of $\varphi|_{l=0}$ by our claim (from HW4). In this case, still Space-RES is guaranteed by IH to find a refutation of $\varphi|_{l=0}$, of space s, If we then combine our space $s-1$ derivation of $\neg l$ and our space s derivation of l, then since these proofs had graphs of rank $s-2$ and $s-1$, respectively the proof we return has a graph of $rank \leqslant s-1$, hence using space $\leqslant s$, as needed.
- For the running time, we have $\leqslant 2n$ recursive calls of space $s-1$, and $\leqslant 1$ recursive calls for space s our time spent on recursive calls is

  $\leqslant C|\varphi|(n-1)^{2s-1} + 2nC|\varphi|(n-1)^{2(s-1)-1}$

  (Noting that $s \geqslant 2$ and $n \geqslant 2$, $2(s-1)-1 \geqslant 1$, $2s-1 \geqslant 3$)

  $\leqslant \frac{n-1}{n}\frac{n-1}{n}\frac{n-1}{n}C|\varphi|n^{2s-1} + 2s\frac{n-1}{n}C|\varphi|n^{2s-3}$

  $\leqslant (1-\frac{1}{n})[(1-\frac{2}{n}+\frac{1}{n^2})C|\varphi|n^{2s-1} + \frac{2}{n}(|\varphi|n^{2s-1})]$

  $\leqslant (1-\frac{1}{n^2})(1-\frac{1}{n})C|\varphi|n^{2s-1}$

  $= (1-\frac{1}{n}+\frac{1}{n^2}-\frac{1}{n^3})C|\varphi|n^{2s-1}$

  as long as my work per-node is $\leqslant (1-\frac{1}{n}+\frac{1}{n^2})C|\varphi|n^{2(s-1)}$

  notice in particular, this is at least $\frac{3}{4}C|\varphi|n^2$ for large enough C, we can afford to prepare the $2n$ formulas $\varphi|_{l=b}$ (each should only take $O(|\varphi|)$, so the back keeping is not hard.
- **Corollary**: There is a quasipolynomial time algorithm for finding treelike resolution proofs.
- Note a resolution refutation never needs to apply the cut rule more than once on the same variable on any branch. (otherwise we can keep the cut rule closer to the root and omit the one closer to the leaves) So, if we label each step of the proof with the variable we eliminated to reach it, we end up with a tree (of rank $s-1$ if there proof used space s) with each node labeled by distinct variables, so by HW2.1
  - For a proof to have l steps, it must use space $\geqslant log(\frac{l+1}{2})+1$
  - Proofs using space s have size $\leqslant n^{s-1}+1$
- In particular, if there is a length-l proof, it uses space at most $logl+1$, Space-RES runs in time $O(|\varphi|n^{2log(l+1)-1})$ and finds a treelike refutation of φ if one exists
- Open question: Can this be improved to polynomial times?

4 Reasoning with Examples

Returning to the setting of PAC learning, we suppose that we have access to examples drawn independently from an unknown distribution D over χ

- **Classical reasoning**: Given a query φ, some KB such that $D \subseteq KB$, decide whether or not φ is valid for KB.
- **PAC semantics** for reasoning given φ, decide whether $Pr_{x \in D}[\varphi(x)] \geqslant 1 - \epsilon$, i.e. whether φ is $(1-\epsilon)$-valid for D.
- **HW3.2**: Show that if KB is $(1-\epsilon)$-valid and $KB \models \varphi$, then φ is also $(1-\epsilon)$-valid respect to D.
- Given direct access to the examples, this becomes even easier. Let's estimate $Pr_{x \in D}[\varphi(x)]$ using $\frac{1}{m}\sum_{i=1}^{m} \varphi(x^{(i)})$, (heavily using complete information)
- By the (additive) Chernoff bound, that using $m \geqslant \frac{1}{2\gamma^2} ln \frac{2}{\delta}$ examples, then with probability $\geqslant 1-\delta$. $\frac{1}{m}\sum_{i=1}^{m} \varphi(x^{(i)})$ is within an additive γ of $Pr_{x \in D}[\varphi(x)]$
- So we can easily approximately decide $(1-\epsilon)$-valid
- Moreover, we can use Occam's Razor style bounds to guarantee that a small set of examples suffices to decide a large class of queries.
- e.g. 3CNFs. These can be represented using one bit for each of the clauses of size $O(n^3)$ bits. So $O(\frac{1}{\gamma^2}(n^3 + log\frac{1}{\delta}))$ examples suffice to answer all 3 CNF queries to within an additive γ, even thought it is classically NP-hard.