CS 181AG
Lecture 13

Packet Classification Cont.

Arthi Padmanabhan

Oct 12, 2022

Net Neutrality Discussion

Packet Classification Recap

- Grid of Tries
  - Backtracking algorithm -> high lookup time
  - All possible sources kept in source trie -> high memory
  - Switch pointers
- Geometrical View
- Bitmaps
- Today:
  - Finish bitmaps
  - Cross producting
  - Decision trees
  - Recent paper on using neural networks for packet classification

Divide and Conquer Algorithms

- Solve best match by field, then efficiently put results together

Using Bitmaps

- Key observation: operations on bitmaps can be sped up by hardware

Rule	Destination	Source	Dest. Port	Src Port	Flags
R1	D1 = 0*	S1 = 10*	25	*	*
R2	D2 = 0*	S2 = 01*	25	123	*
R3	D3 = 00*	S3 = 11*	53	*	UDP
R4	D4 = 00*	S4 = 1*	*	*	*
R5	D5 = 0*	S5 = 1*	20	*	*
R6	D6 = 10*	S6 = 1*	*	*	TCP_ack
R7	D7 = *	S7 = 00*	*	*	UDP
R8	D8 = *	S8 = *	*	*	*

Using Bitmaps

- Find least cost rule using bitmaps for the following (for simplicity, IPs will be 4 bits):
  - (0001, 0011, 53, 35, UDP)
  - (0111, 1011, 25, 30, -)
  - (0011, 1000, 20, 30, UDP)

Flags	Bitmap
UDP	
TCP_ack	
*	

Dst	Bitmap
0*	
00*	
10*	
*	

Src	Bitmap
10*	
01*	
11*	
1*	
00*	
*	

Dst Port	Bitmap
25	
53	
20	
*	

Src Port	Bitmap
123	
*	

Using Bitmaps

- Find least cost rule using bitmaps for the following (for simplicity, IPs will be 4 bits):
  - (0001, 0011, 53, 35, UDP) – R7
  - (0111, 1011, 25, 30, -) – R1
  - (0011, 1000, 20, 30, UDP) – R4

Flags	Bitmap
UDP	11111011
TCP_ack	11011101
*	11011001

Dst	Bitmap
0*	11001011
00*	11111011
10*	00000111
*	00000011

Src	Bitmap
10*	10011101
01*	01000001
11*	00111101
1*	00011101
00*	00000011
*	00000001

Dst Port	Bitmap
25	11010111
53	00110111
20	00011111
*	00010111

Src Port	Bitmap
123	11111111
*	10111111

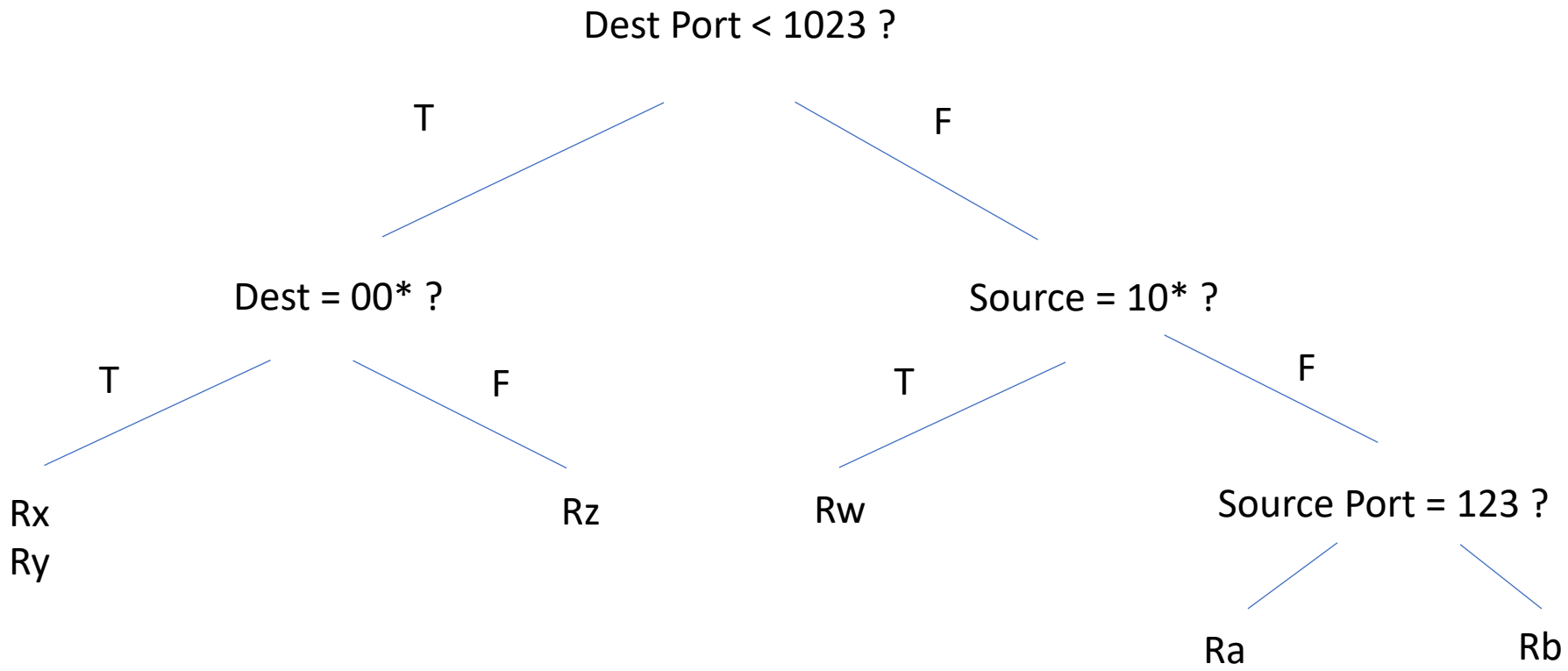
Using Bitmaps

- Does require linear search
  - Summary bits can help, but still requires linear search of summary bits

Rule	Destination	Source	Dest. Port	Src Port	Flags
R1	D1 = 0*	S1 = 10*	25	*	*
R2	D2 = 0*	S2 = 01*	25	123	*
R3	D3 = 00*	S3 = 11*	53	*	UDP
R4	D4 = 00*	S4 = 1*	*	*	*
R5	D5 = 0*	S5 = 1*	20	*	*
R6	D6 = 10*	S6 = 1*	*	*	TCP_ack
R7	D7 = *	S7 = 00*	*	*	UDP
R8	D8 = *	S8 = *	*	*	*

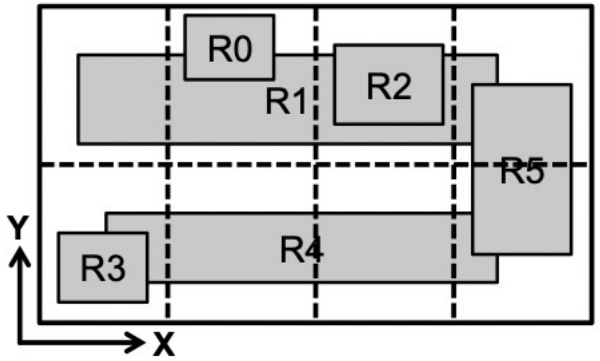
Decision Trees

- Consider grid of tries. Why look at all bits of one field before looking at any others?



Decision Trees

- Consider grid of tries. Why look at all bits of one field before looking at any others?



NeuroCuts (SIGCOMM 2019)

- Reinforcement Learning approach to packet classification

Problem

- What problem does the paper solve?
  - Current approaches to constructing decision trees rely on hand-tuned heuristics, which doesn’t scale well

NN a good fit?

- To do classification?
- To build decision trees?

Reinforcement Learning

- Concerned with how intelligent agents should take action to maximize cumulative reward
- Agent sees environment and current reward
- Well-suited for problems where there is a long-term payoff over a series of actions

Reinforcement Learning for Packet Classification

- Environment: current decision tree
- Reward: classification time, memory footprint, or combination

Why is RL a good fit?

- Rewards are delayed – we don’t know if it’s a good decision tree until we finish building it
- There are clear metrics to maximize
- Can cheaply build huge number of samples (use software, run in parallel)

NeuroCuts Design

- Possible decisions are to cut the search space along a particular axis (e.g., source, destination port, etc.)
- All possible decisions are advertised to agent
- Agent starts with a naïve policy and learns through multiple rollouts (multiple iterations of building tree and seeing reward) and updates its policy over time until it is satisfied with the reward

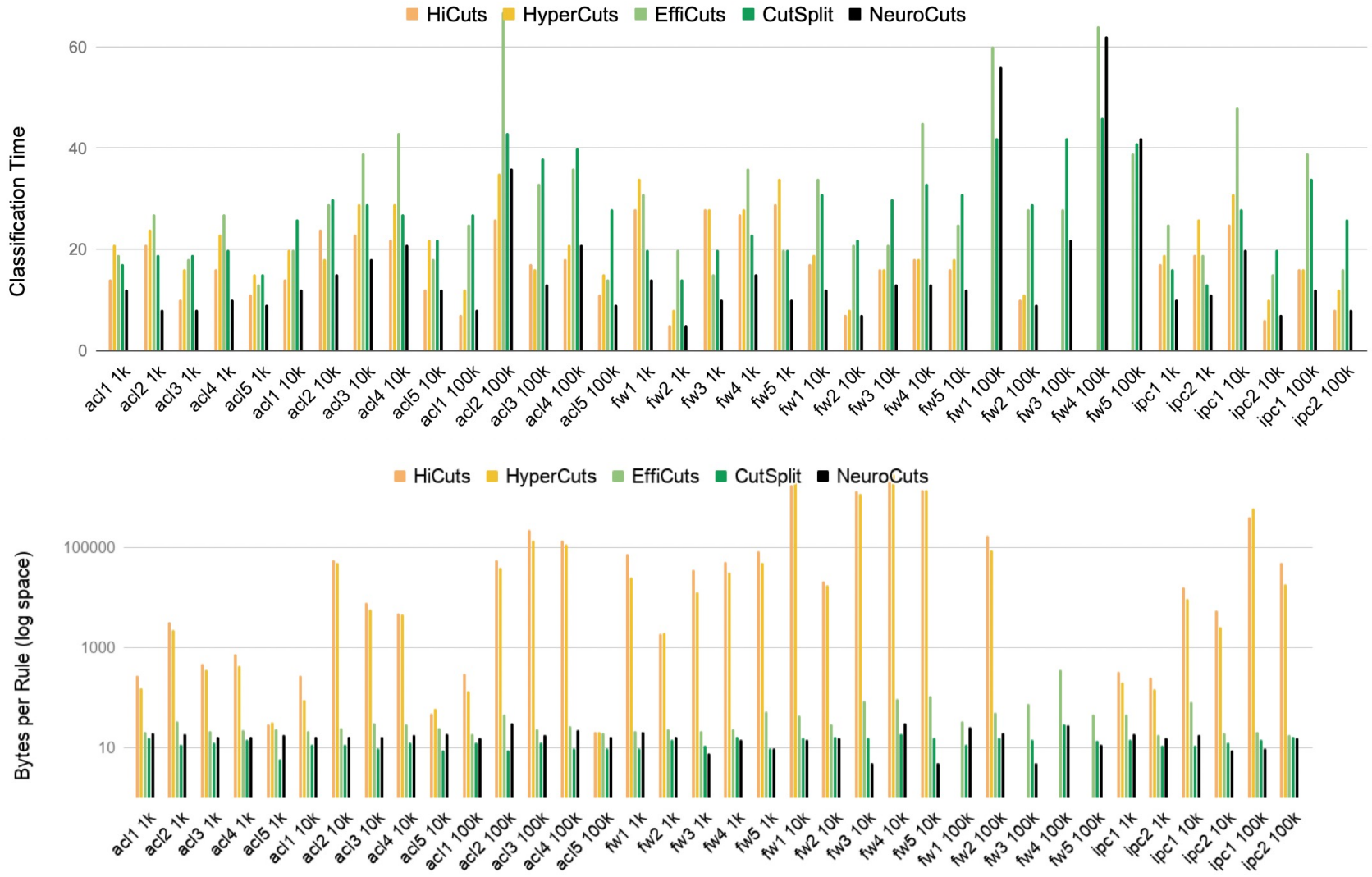
Details

- When making a decision at a particular node, state only involves that node, rather than entire tree
- By induction, if each subtree is optimized, the whole tree is optimized

Training

- Usually converges within a few hundred rollouts
- Larger rule set = larger time per rollout, but not larger number of rollouts

Evaluation


HiCuts
HyperCuts
EffiCuts
CutSplit
NeuroCuts
Classification Time
0
20
40
60
Bytes per Rule (log space)
10
1000
100000
acl1 1k
acl2 1k
acl3 1k
acl4 1k
acl5 1k
acl1 10k
acl2 10k
acl3 10k
acl4 10k
acl5 10k
acl1 100k
acl2 100k
acl3 100k
acl4 100k
acl5 100k
fw1 1k
fw2 1k
fw3 1k
fw4 1k
fw5 1k
fw1 10k
fw2 10k
fw3 10k
fw4 10k
fw5 10k
fw1 100k
fw2 100k
fw3 100k
fw4 100k
fw5 100k
ipc1 1k
ipc2 1k
ipc1 10k
ipc2 10k
ipc1 100k
ipc2 100k

Reminder: Upcoming Schedule

- 10/10, 10/12: Finish packet classification
- 10/17-10/18: Fall break!
  - No assignment due week of fall break
- 10/19: Start new topic (switching)
- 10/24: Midterm Review in class
  - No assignment due week of midterm
- 10/25 – 10/30: Midterm (no class 10/26)
  - No assignment due right after midterm
- Assignment 7 due Nov. 7 (Monday)