CS584: Machine Learning

Implementation of Result Prediction by Analyzing Data in DotA2

Presenter: Xin Su, Shiyang Li

Instructor: Gady Agam

Department of Computer Science

Illinois Institute of Technology

Outlines

- Introduction
- Data Collection
- Algorithm
  - Decision Tree
  - Random Forest (Majority voting / AdaBoost voting)
- Performance
- Result and Discussion
- Conclusion

Introduction

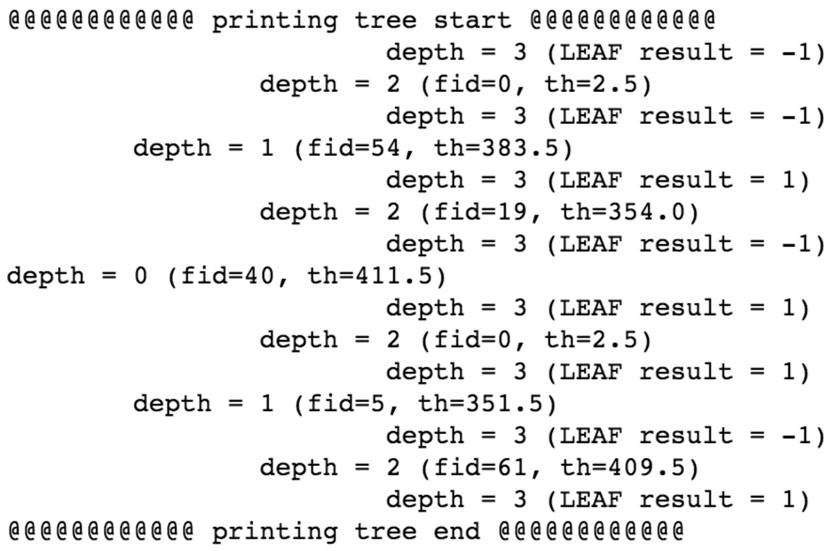
- DotA2
  - Multi-player online battle game: 5 vs 5

Data Collection

- Filtered WebAPI
- Game Duration > 18 min
- Features
  - Each player owns 7 features:
    - 'kills', 'deaths', 'assists’
    - 'last_hits', 'denies’
    - 'gold_per_min', 'xp_per_min’
  - Total 70 features for one game
- Total 1908 lines of data

Algorithm - Greedy Decision Tree

- Feature split selection
  - Least classification error
- Stopping condition
  - Data completed separated
  - No features
  - Minimum data amount
  - Maximum tree depth



```
@@@@@@@@@@@@ printing tree start @@@@@@@@@@@@
                        depth = 3 (LEAF result = -1)
                depth = 2 (fid=0, th=2.5)
                        depth = 3 (LEAF result = -1)
        depth = 1 (fid=54, th=383.5)
                        depth = 3 (LEAF result = 1)
                depth = 2 (fid=19, th=354.0)
                        depth = 3 (LEAF result = -1)
depth = 0 (fid=40, th=411.5)
                        depth = 3 (LEAF result = 1)
                depth = 2 (fid=0, th=2.5)
                        depth = 3 (LEAF result = 1)
        depth = 1 (fid=5, th=351.5)
                        depth = 3 (LEAF result = -1)
                depth = 2 (fid=61, th=409.5)
                        depth = 3 (LEAF result = 1)
@@@@@@@@@@@@ printing tree end @@@@@@@@@@@@
```

Algorithm - Random Forest

- Data & Feature Split
  - Bootstrap sampling data
  - Random m features << total M features
- Voting Machine
  - Majority voting
  - AdaBoost voting

Algorithm - AdaBoost

- T weak models $f_1(X)..f_T(X)$, m datasets, n features
- Start same weight for all data: $\alpha_i = 1/m$, i = 1..m
- for (t in 1 .. T) :

  Learn $f_t(X)$ with data weights α_i

  Compute model weight w_i

  - $w_t = ½ \ln((1- weight\_error(f_t)) / weight\_error(f_t))$

  Recompute data weights α_i

  - $\alpha_i = \alpha_i \exp(-w_t)$, if $f_t(x_i) = y_i$
  - $\alpha_i = \alpha_i \exp(w_i)$, if $f_t(x_i)$!= y_i
- Final model predict by
  - $ÿ = sign(sum_{t=1..T}(w_t f_t(X)))$

Performance

Model	Parameters	Accuracy	Duration

Decision Tree	max_depth=8
	min_amt=5
		0.99	<= 1s

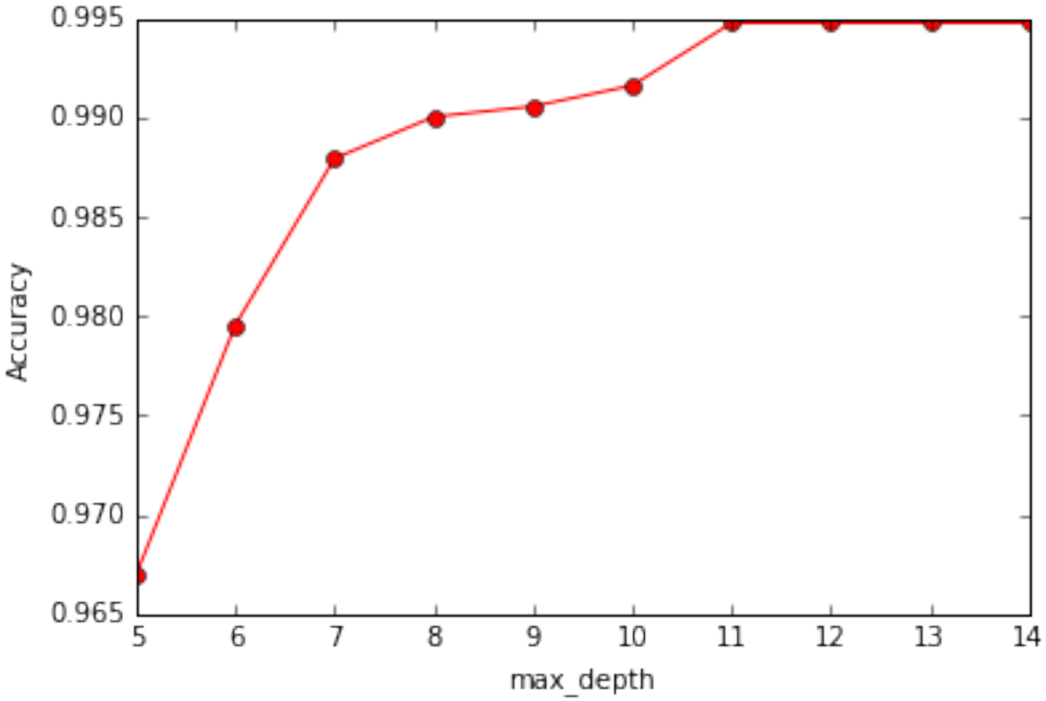
Random Forest (Majority voting)	min_amt = 5
	max_depth = 3
	tree_amt = 20
	feature_size = 0.1
		0.984	<= 16s

Random Forest (AdaBoost voting)	min_amt = 5
	max_depth = 3
	tree_amt = 20
	feature_size = 0.1
		0.982	<= 16s

DT	sklearn	0.79	<= 0.5s
RF	sklearn	0.985	<= 0.5s
SVM	sklearn	0.99	<= 0.1s
Naive Bayes	sklearn	0.99	<= 0.1s

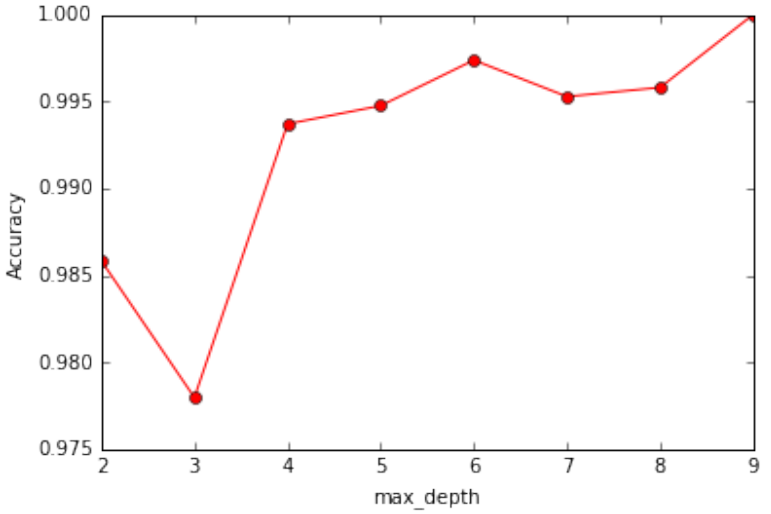
Parameter Exploration

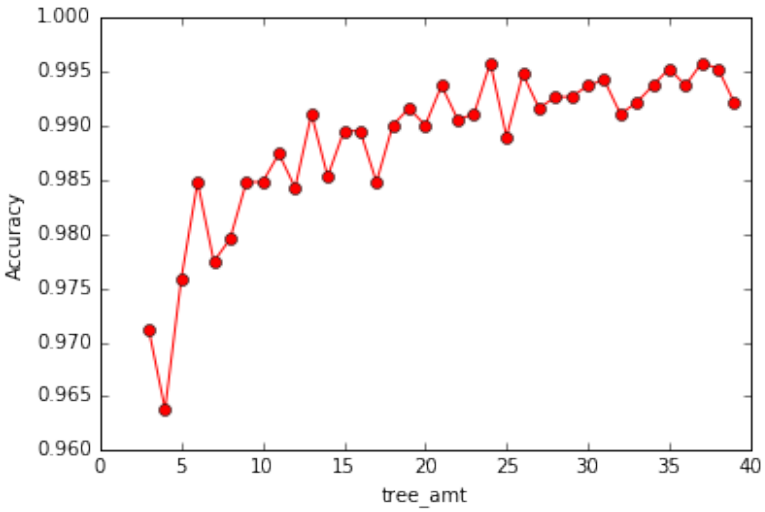
- Decision Tree
  - max_depth

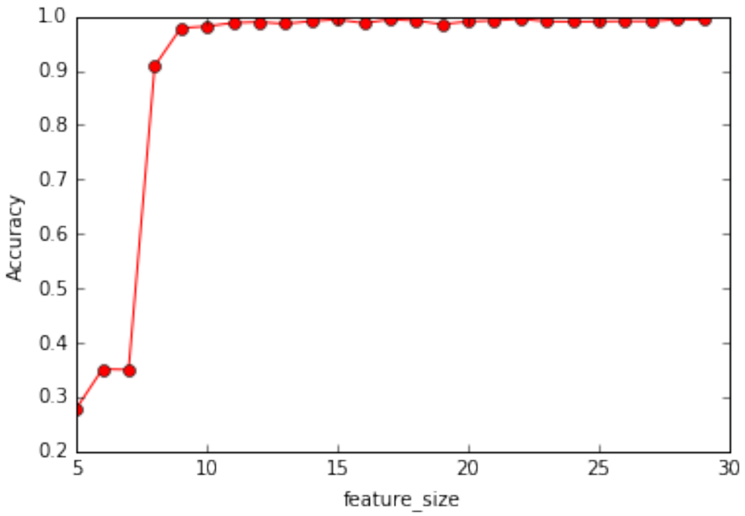


Parameter Exploration

- Random Forest
  - max_depth
  - tree_amt
  - feature_size







Algorithm Comparison

- Decision Tree
  - High Accuracy
  - Fast
- Random Forest
  - High Accuracy
  - Fast (a little bit slower)
- Random Forest (AdaBoost):
  - High Accuracy
  - Fast (a little bit slower)
  - Not contribute too much in this case

Conclusion

- Future Work
  - Predict based on real game timeline
  - Classification of player’s role

Reference

- [1] F. Johansson and J. Wikstr¨om, “Result prediction by mining replays in dota 2,” Master’s thesis, Blekinge Institute of Technology, SE-371 79 Karlskrona, Sweden, 2015.
- [2] “The WebAPI Website,” 2012. http://dev.dota2.com/showthread.php?t=58317 [Online; accessed 10-Mar-2016].
- [3] D. P. Kevin Conley, “A Recommendation Engine for Picking Heroes in Dota2,” 2013.
- [4] C. G. Emily Fox, “Classification: A machine learning perspective,” University of Washington, Coursera, 2016.
- [5] M. P. Atish Agarwala, “Learning Dota 2 Team Compositions,” 2014.
- [6] C. M. Kuangyan Song, Tianyi Zhang, “Predicting the winning side of DotA2,” 2015.