Stronger Practical Game-Playing AI for *Duelyst II*

Bryan McKenney
Advisors: Wheeler Ruml and Alexander Dockhorn (Leibniz University Hannover)
Department of Computer Science, University of New Hampshire, Durham NH 03824

Problem

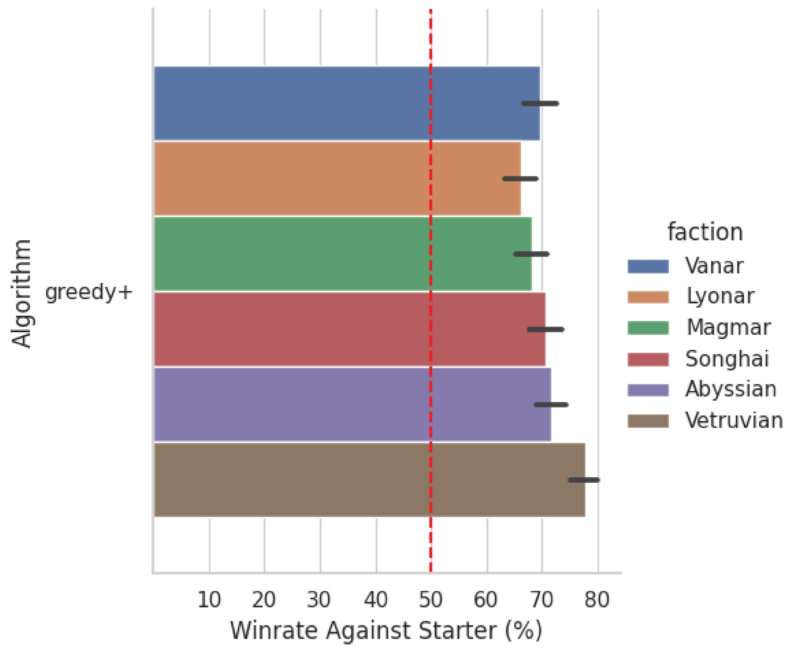
Duelyst II

challenges:
- partially observable (opponent hand + deck hidden)
- stochastic (drawing cards, card abilities)
- adversarial
- large state and action spaces
- perform several actions in 90-sec turn
- frequent balance changes and new cards
- **slow successor generation**

New SotA: Greedy+

- **one-step lookahead using static evaluator**
- rules for mulligan, replace, and lethal checking
- beat previous expert-rule-based AI and enabled boss battles
- **pros**
  - can play with/against any deck
  - flexible to changes in game
- **cons**
  - still weak against humans
  - manually weighting different aspects of state is difficult

problem: impossible to search more due to time constraints
proposed solution: enhance one-step lookahead with ML



Approach

Abstractions

abstract state
- state features:
  - global features like max mana
  - hand card features like type and cost
  - board unit features like Attack and Health

635 state features
10 scripts

abstract action — script
- script: state -> action
- does **one-step lookahead using criterion**, such as:
  - max diff in General Health
  - min # of unplayable cards

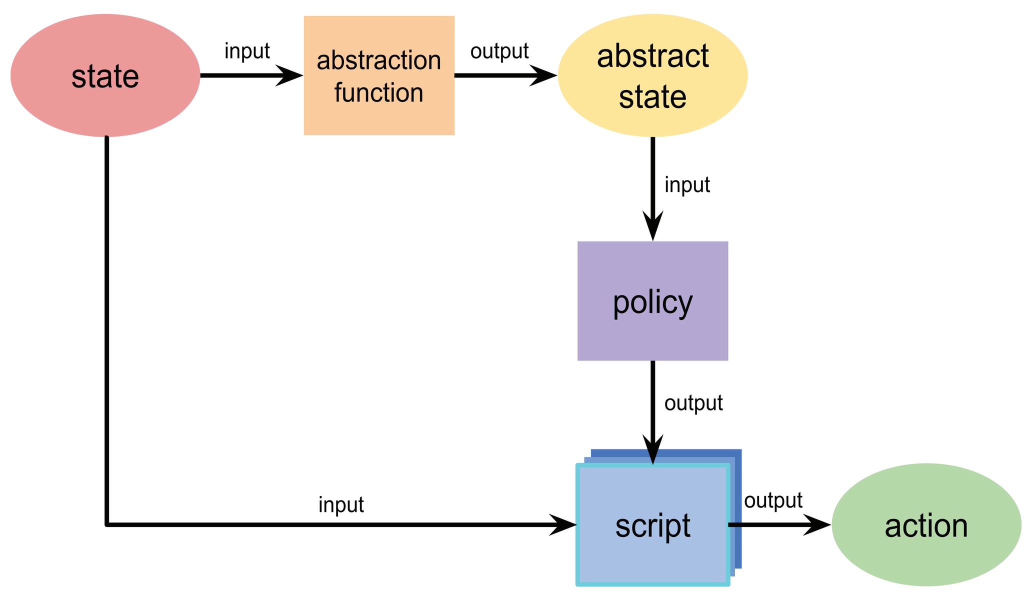
Learning from Data

1. **download game logs of pro players**
2. **abstract game data**
   a. state -> abstract state
   b. player action -> script
3. **generalize to unseen abstract states**
   a. CNN + merge model (features split into global, hand, and board)
   b. 30% training/validation/test accuracy

5,411 games = 250,991 data points

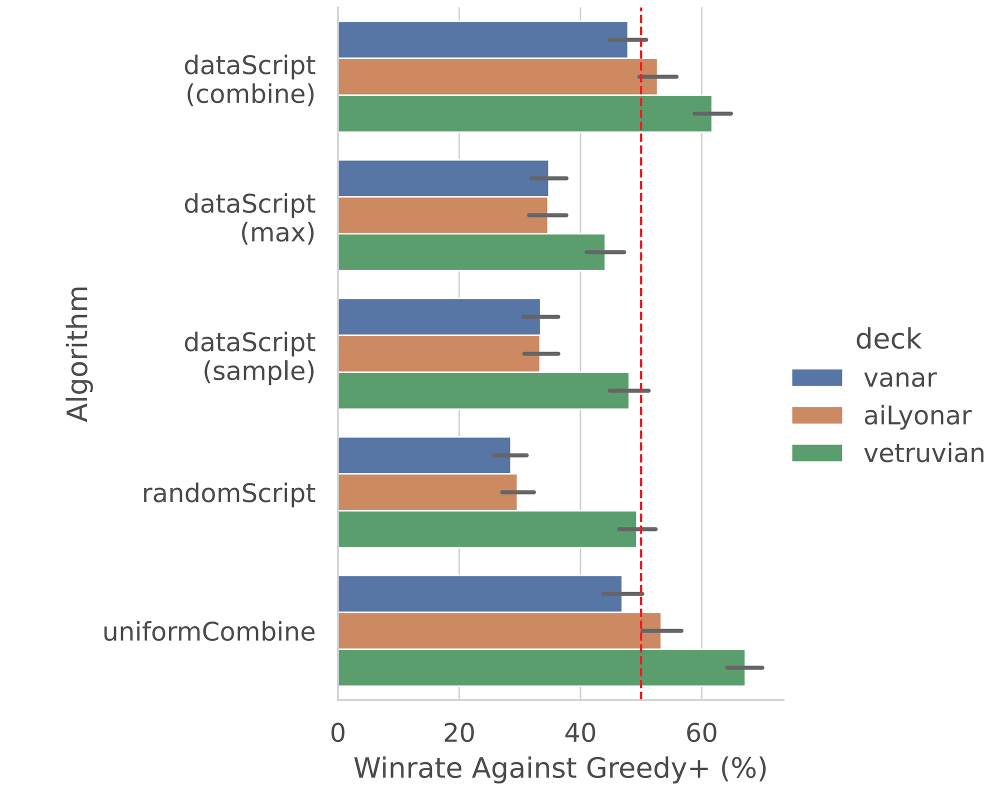
Algorithms

1. **DataScript**: converts state to abstract state, then predicts probability for each script
   A. **(Combine)**: sum normalized action-values of all scripts weighted by their probs and pick highest-value action
   B. **(Max)**: execute script with highest prob
   C. **(Sample)**: execute script sampled from prob distribution
2. **RandomScript**: execute random* script
3. **UniformCombine**: sum normalized action-values of all scripts and pick highest-value action



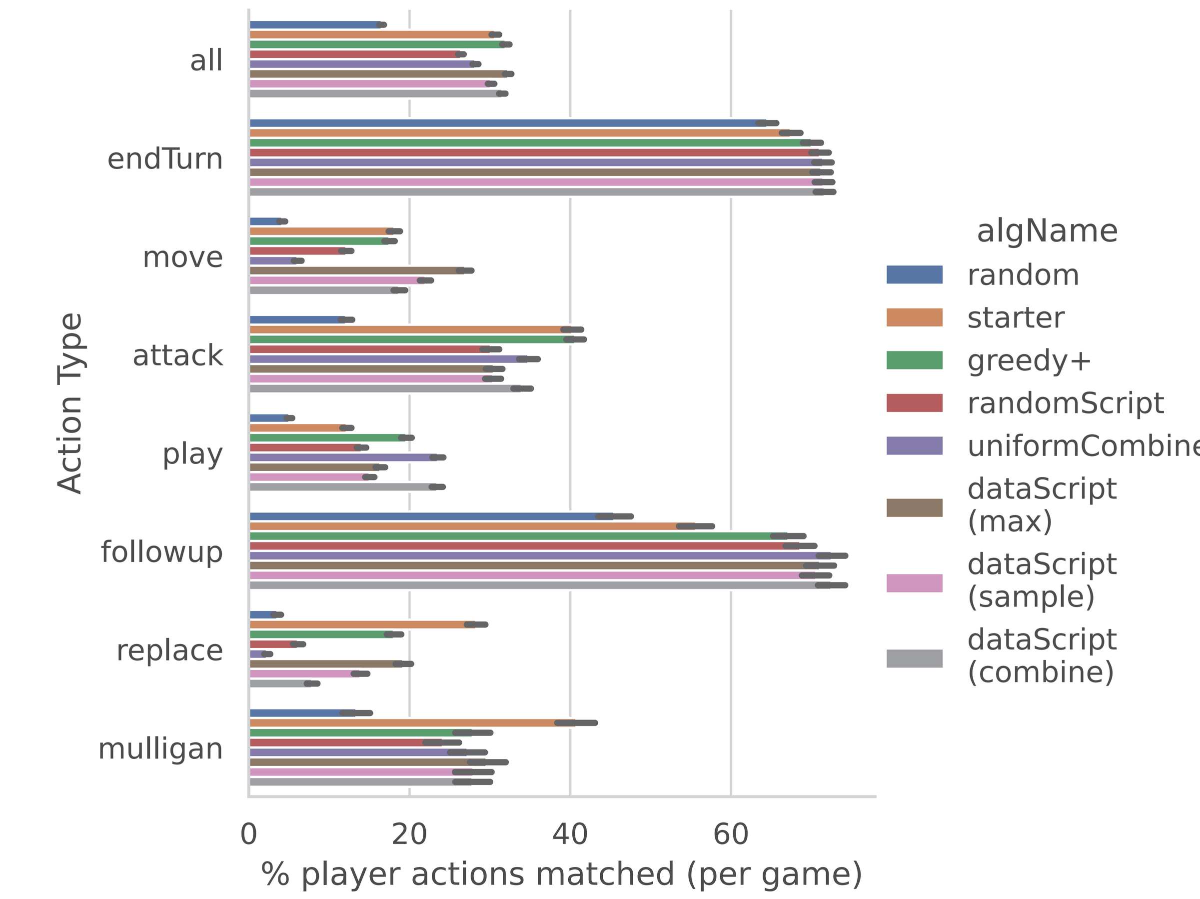
Results

Algorithms vs Greedy+



- each bar is the average from 1,000 games (w/ 95% conf)
- no evidence that learning affects winrate
- **DataScript (Combine) and UniformCombine beat Greedy+ with 2/3 decks**

Algorithms vs Human Data



- each bar is the average from ~800 games (w/ 95% conf)
- Play and Followup actions seem to matter more than others, since Starter < Greedy+ < DataScript (Combine) = UniformCombine there, matching winrates
- **learning does help, but not with the most important action types, while combining scripts helps with these action types**

Acknowledgements

Thanks to:
- **Wheeler Ruml** and **Alexander Dockhorn** for being invaluable advisors
- **Dream Sloth Games** for letting me work on *Duelyst II*
- **IEEE CIS**, the **UNH Grad School**, and the **UNH CS Department** for IEEE Conference on Games presentation funding

References

my published Duelyst work:
- McKenney, B. 2023. General-Purpose Planning Algorithms In Partially-Observable Stochastic Games. In *UNH Honors Theses and Capstones*. https://scholars.unh.edu/honors/773/.
- McKenney, B. and Ruml, W. 2023. General-purpose planning algorithms in the card game Duelyst II. In *Proceedings of CoG-23*.

related work:
- Lelis, L. 2020. Planning Algorithms for Zero-Sum Games with Exponential Action Spaces:
- A Unifying Perspective. In *Proceedings of IJCAI-20*.
- Zhang, S. and Buro, M. 2017. Improving Hearthstone AI by learning high-level rollout policies and bucketing chance node events. In *Proceedings of CIG-17*.