Synthesizing Chess Tactics from Player Games

Abhijeet Krishnan[1]
Chris Martens[2]

[1] North Carolina State University
[2] Northeastern University

AIIDE 2022 Strategy Games Workshop
2022/10/25

POEM
Principles of Expressive Machines

NC STATE UNIVERSITY

Motivation

- Competitive games are big business
- People want to get better at them
- Superhuman AI exist for many games
  - *Starcraft 2* (AlphaStar)
  - *Dota 2* (OpenAI 5)
  - Chess (Stockfish 14)
  - Go (Leela Zero)
  - Poker (DeepStack)
  - ...

DeepMind

OpenAI

Stockfish

DeepStack

Motivation

- Could inform how humans play

"In essence I have become a very different player in terms of style than I was a bit earlier, and it has been a great ride."

- Magnus Carlsen, 5x world chess champion on AlphaZero's influence on him [6]


[5]

Motivation

Could explanations of superhuman agents for games improve human play?

Related Work

1. Strategy synthesis for games
2. Explainable RL

Related Work: Strategy Synthesis

- evolutionary approaches to learn rule-based agents for games like
  - *Neverwinter Nights* (Spronck, Sprinkhuizen-Kuyper, and Postma 2004)
  - Hanabi (Canaan et. al. 2018)
  - μRTS (Mariño et. al. 2021)
  - ...

Related Work: Explainable RL (XRL)

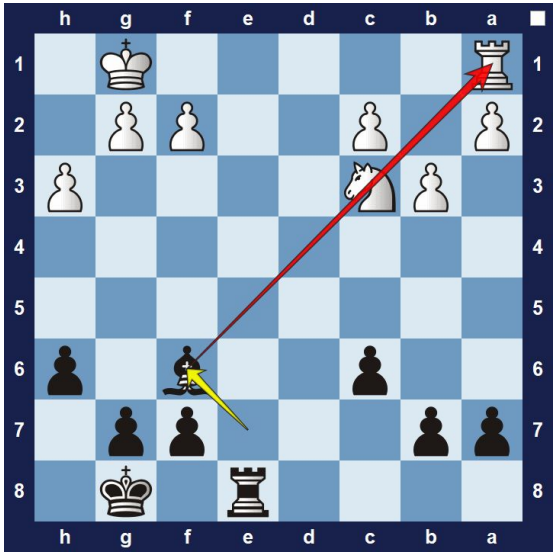
- Explainability using surrogate model
  - decision trees (Bastani, Pu, and Solar-Lezama 2018)
  - programmatic policies (Verma et. al. 2019)
  - ...
- Less attention on game environments or explanations for players

Methodology

1. Chess Tactic as Strategy Model
2. Learning via ILP
3. Tactic Utility Metrics

Methodology: Chess Tactic

- *A chess tactic is a maneuver that takes advantage of short-term opportunities* (Seirawan 2005)
- E.g., fork, pin, skewer, x-ray, windmill, deflection
- Important concept in chess training and education[8]



Example of a pin in chess [7]

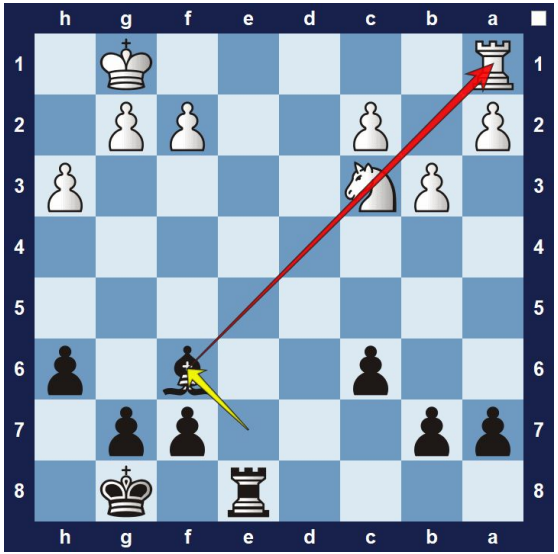
Methodology: Chess Tactic Model

- *Strategy*: a policy defined on a *subset* of the state space for a game
- Model chess strategy as a chess tactic
- Implemented as a first-order logic rule in *Prolog*
- Uses a *predicate vocabulary* to express features

```
tactic(Position, From, To) ←
        feature_1(···),
        feature_2(···),
        ⋮
        feature_n(···)
```

Prolog-like representation of our chess tactic model

Methodology: Chess Tactic vs. Model



Example of a pin in chess [7]

```
pin(Pos, From, To) :-
    make_move(From, To, Pos, NewPos),
    behind(To, Pinned, Target, NewPos),
    attacks(To, Pinned, NewPos),
    attacks(To, Target, NewPos).
```

The pin tactic model

Methodology: Learning using ILP

- Tactic model learned using *inductive logic programming* (ILP)
- symbolic ML technique
- (position, move) pair in gameplay trace used as positive example

$$E \cup B \rightarrow T \ (induce)$$

E	B	T
parent(Mary,Vicky). parent(Mary,Andre). parent(Carrey,Vicky). parent(Carrey,Andy).	mother(Mary,Vicky). mother(Mary,Andy). father(Carrey,Vicky). father(Carrey,Andy).	parent(X,Y) :- mother(X,Y). parent(X,Y) :- father(X,Y).

Example of ILP [11]

Methodology: ILP via Popper

- Popper (Cropper and Morel 2021) to learn chess tactic model
- Learns in three stages - generate, test, constrain
- Generate stage modified to only generate tactics that produce legal moves
- Constrain stage modified to prevent *specializations* of a tactic that do not apply to any position in the training set

Methodology: Tactic Utility Metrics

- *Coverage* = fraction of positions to which a tactic is applicable
- High coverage -> tactic is more general
- Low coverage -> tactic is more situational

$$P_{\mathrm{A}} \doteq P \cap \mathrm{A}_{\varsigma}$$

$$\mathrm{Coverage}(\varsigma, P) \doteq \frac{|P_{\mathrm{A}}|}{|P|}$$

Formula for Coverage

- *Divergence* = average engine eval. error between tactic and ground move
- High divergence -> tactic and ground moves are different
- Low divergence -> tactic and ground moves are similar

$$\mathrm{Divergence}_E(\varsigma(\cdot), P) \doteq \frac{1}{|P_{\mathrm{A}}|} \sum_{(s,a_1)\in P_{\mathrm{A}}} \sum_{a_2 \in \mathcal{A}(s)} \varsigma(a_2|s) d_E(s, a_1, a_2)$$

Formula for Divergence

Evaluation

- Dataset
- Training
- Performance Metrics

Evaluation: Dataset

- Used collection of online chess games played by human players on Lichess
- Generated (position, move) pairs from games
- Selected positions beginning at move 12 to avoid standard opening moves

- Training Set
  - Drawn from January 2013 archive
  - Avg. ELO = 1601 ± 289
  - Num. examples = 100
- Testing Set
  - Drawn from February 2013 archive
  - Avg. ELO = 1595 ± 298
  - Num. examples = 10

Evaluation: Training

- Empirically limit hypothesis size to 1 clause, 5 variables, 5 body literals to strike balance between learning time and solution quality
- Run modified Popper until no more solutions found
- Obtained 837 tactics

Evaluation: Performance Metrics

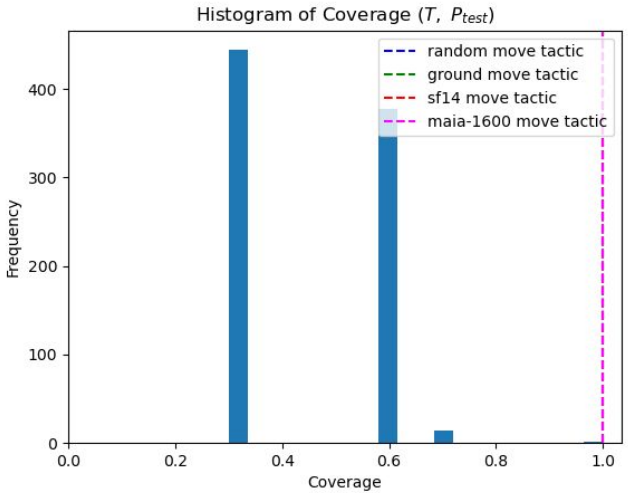
- Measure *coverage*, *accuracy* and *divergence* for tactics on test data
- Use *Maia-1600* [11] and *Stockfish 14* to calculate divergence
- Limit engine search depth to 1 node
- four baseline tactics -
  - *random*: make a random move
  - *ground*: make the move played in the actual example
  - *maia-1600*: make the best move suggested by Maia-1600
  - *sf14*: make the best move suggested by Stockfish 14

Results

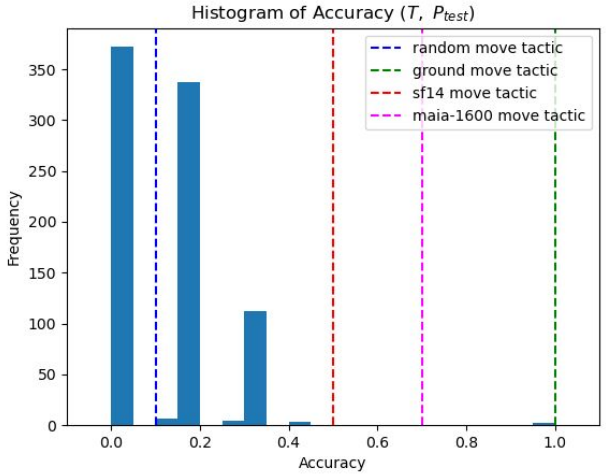
- Performance Metrics
- Qualitative Analysis

Results: Performance Metrics

Coverage

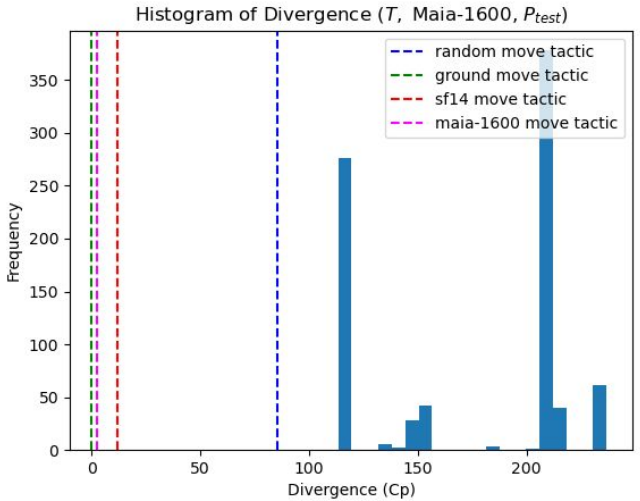


Accuracy

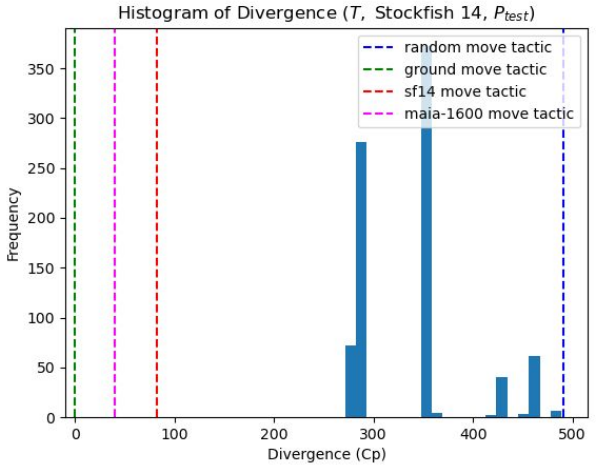


Results: Performance Metrics (contd.)

Divergence (Maia-1600)



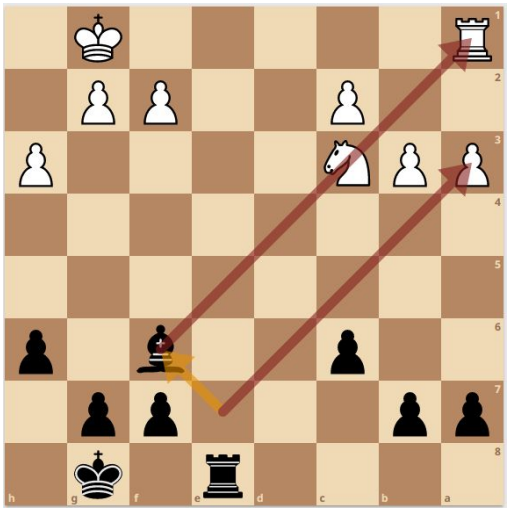
Divergence (Stockfish 14)



Results: Qualitative Analysis (1/2)

```
f(Position,From,To) ←
    legal_move(From, To, Position),
    attacks(From, Square1, Position),
    behind(To, Square1, Square2, Position),
    different_pos(Square1, Square2).
```

A learned tactic with low divergence from ground



Author interpretation

Results: Qualitative Analysis (2/2)

```
f(Position,From,To) ←
    legal_move(From, To, Position),
    behind(From, To, Square1, Position),
    behind(From, Square2, Square3, Position),
    behind(From, Square3, Square2, Position),
    behind(From, Square2, To, Position).
```

A learned tactic with meaningless variable reshuffling

- *Author interpretation*: ???
- Points to opportunities to add constraints to hypothesis space

Limitations + Future Work

- *Interpretability not verified with user study* - measure ease of learning and applying the tactics in real games
- *Tactics diverge from target policy* - add more features to BK; use different metrics
- *Learn tactics based on player skill* - use logic-based player models (Krishnan, Williams and Martens 2020)

Conclusion

- Formulated *tactic synthesis problem* of learning chess strategies as first-order logic rules
- Presented a learning system that *uses ILP* to solve the tactic synthesis problem
- Introduced *divergence* as a metric to evaluate chess tactics
- Showed that learned tactics can *outperform a random baseline* in approximating a human beginner

Sources

[1] https://www.thesprucecrafts.com/chess-strategy-tactics-4102130

[2] https://franz.cgsociety.org/pqpw/personal-robot-04

[3] https://liquipedia.net/starcraft2/TLO

[4] https://deepmind.com/blog/article/alphastar-mastering-real-time-strategy-game-starcraft-ii

[5] https://svw.no/en/chess-blog/dubai-expo-fide-world-championship-match/

[6] https://www.newinchess.com/media/wysiwyg/product_pdf/872.pdf

[7] https://chessfox.com/pins/

[8] Szabo, A. 1984. Computer chess tactics and strategy. Ph.D. thesis, University of British Columbia.

[9] Gobet, F.; and Jansen, P. J. 2006. Training in chess: A scientific approach. Education and chess

[10] Zhang, Y.; Tiˇno, P.; Leonardis, A.; and Tang, K. 2021. A survey on neural network interpretability. *IEEE Transactions on Emerging Topics in Computational Intelligence*.

[11] https://wiki.ubc.ca/Course:CPSC522/Inductive_Logic_Programming

[12] McIlroy-Young, Reid, et al. "Aligning superhuman ai with human behavior: Chess as a model system." *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2020.

Thank You!

Reach out at -

- akrish13@ncsu.edu
- abhijeetkrishnan.me