Automatic Segmentation of Go Game Phases via N+1 State Classification

Pei-Cheng Tsao[†] Tzu-Chun Kao[†1] Tzu-Cheng Tseng[†1] Shi-Jim Yen[†1]

Abstract: Segmenting Go games into opening, middle-game, and endgame phases is vital for phase-conditioned research, yet manual annotation remains labor-intensive and inconsistent among experts. We propose a pipeline leveraging length-adaptive equidistant resampling to transform limited expert-labeled transitions into a class-balanced dataset. Using the 2023 AI CUP champion backbone, we train an N+1 state classifier paired with two heuristics to derive monotonic transition points (T1 and T2) from per-move probabilities. Although move-level accuracy reaches 87 to 89 percent on held-out data, validation against independent annotators reveals a critical discrepancy: T1's correlation with ground truth is near zero, and T2's is only weak-to-moderate. Furthermore, predicted transition steps are two to three times less variable than expert judgments. This confirms a conservative, low-variance bias, demonstrating that while the model effectively captures population-level phase tendencies, it cannot reliably pinpoint individual transition points.

Keywords: computer Go, game phase segmentation, sequence classification, transition-point detection, deep learning.

1. Introduction

Go analysis conventionally divides a game into the opening, middle game, and endgame; these phases engage measurably different cognitive processes [5]. An automatic way to locate phase boundaries would let downstream research condition on phase without slow, imperfect manual annotation (Section 3).

This paper contributes: (1) a preprocessing pipeline building a class-balanced training set from few expert-labeled transitions; (2) an N+1 classifier with heuristics converting per-move probabilities into two transition points; and (3) an evaluation against independent annotators and aggregate statistics revealing a conservative, low-variance bias invisible to move-level accuracy.

2. Related Work

Automatic phase/state recognition from game records has been studied in other domains: global state evaluation and macro-management prediction from StarCraft replays [1,2], and player skill/difficulty prediction from raw play traces [3]. Within computer Go, prior work predicts player strength and style [6] and aims to make engine evaluations more interpretable [4]. To our knowledge, none directly targets automatic three-phase segmentation together with a quantitative check of how precisely a predicted transition matches an independent human expert's judgment — the gap this paper addresses.

3. Data Preprocessing

Game records are sourced from the go4go archive. A matching expert-annotated transition dataset (opening-end move and endgame-start move) was provided by the supervising research group, planned for release in a future AI CUP competition. Annotations carry roughly 10% inter-expert disagreement. The raw corpus has only 197 annotated games — just 394 labeled moves total, too few for a robust three-way per-move classifier. Since phase transitions are inherently sequential, we expand the training set via equidistant resampling with a length-adaptive interval K: K=3 ($<$200 moves), K=6 (200–300), K=10 ($>$300). This keeps sampling density comparable across game lengths and produces a larger, class-balanced (1:1:1) training set.

4. The N+1 State Classification Model

We frame phase labeling as an N+1 state classification problem: given a move and its N preceding moves, predict Stage 0 (opening), 1 (middle game), or 2 (endgame). The backbone reuses, unmodified, the residual tower from the 2023 AI CUP champion model (pre-activation residual blocks, RepVGG-style parallel convolutions in place of SE units, Batch Renormalization, Mish activation). Boards use the original $17\times19\times19$ tensor encoding (stone color, liberty-count buckets, previous four moves, ko point); we drop the original Focal-Loss/Lion-optimizer choices, since our phase-classification task is already class-balanced by construction.

The model reaches 98.12% $\pm$ 0.19% / 87.17% $\pm$ 1.34% / 89.00% $\pm$ 1.25% accuracy on train / validation / test ($\pm$ is the 95% half-width). Stage 1 (middle game) is consistently hardest (F1 0.97 / 0.80 / 0.84), bounded by two fuzzy transitions. Misclassification occurs essentially only between adjacent stages — no game ever jumps directly from opening to endgame — despite no explicit ordering constraint.

Since per-move class probabilities are noisy near true transitions and can flip back and forth between classes, raw argmax trajectories can't locate T1/T2 directly. In the per-move argmax over the 1,561 evaluation games of Section 5.1, 5.3% $\pm$ 0.2% of moves return from the middle game to the opening and 6.2% $\pm$ 0.3% from the endgame to the middle game (95% half-widths, resampling games); none return from the endgame to the opening. Returns are therefore uncommon, but the phases are not strictly one-directional. Two heuristics enforce monotonicity: T1 is fixed at the first move where opening-class probability drops below 0.55 and is never revisited; a T2 candidate (Class 2 probability exceeding Class 1) is confirmed only if the average trend over the next five moves also favors Stage 2, filtering out short-lived spikes.

[1]Department of Computer Science and Information Engineering, National Dong Hwa University, Hualien, Taiwan

© 2026 Information Processing Society of Japan

5. Experiments and Results

5.1 Human-Model Alignment Analysis

We validated T1 and T2 predictions against two independent annotators who were uninvolved in creating the training labels (pooled dataset N = 1561; split into Annotator 1 with N = 1257 and Annotator 2 with N = 305). Table 1 reports Pearson's correlation coefficient (r).

Table 1. GT-prediction correlation, pooled vs. annotator.

Set	N	r (T1)	r (T2)
Pooled	1561	0.164	0.513
Annotator 1	1257	0.250	0.527
Annotator 2	305	0.013	0.541

As shown in Table 1, T1's correlation with human ground truth is near zero across all splits, indicating that the predicted opening-end move carries minimal predictive value regarding expert placement. Conversely, T2 shows moderate and consistent correlation (r $\approx$ 0.51–0.54).

Table 2 evaluates tolerance-based accuracy Acc@K on the pooled set. T1 accuracy climbs rapidly as tolerance widens (reaching 68.8% within $\pm$15 moves), whereas T2 improves slowly, requiring a broader window to achieve high confidence due to its higher standard deviation (SD 37–54 for T2 vs. 17–24 for T1). Table 2 also compares the heuristics with direct switching on the same per-move probabilities: T1 is the first move whose argmax is not the opening, and T2 is the first move after T1 whose argmax is the endgame. Backward confirmation improves T2 at every tolerance. For T1, the drop threshold is itself a first-switch rule, and the two give similar results.

Table 2. Tolerance-based accuracy Acc@K (%), pooled set. Baseline: direct switching. $\pm$ is the 95% half-width.

Tolerance ($\pm$)	T1 Acc@K	T2 Acc@K	T1 baseline	T2 baseline
1	14.0% $\pm$ 1.7%	18.4% $\pm$ 1.9%	19.5% $\pm$ 2.0%	13.3% $\pm$ 1.7%
5	35.4% $\pm$ 2.4%	21.8% $\pm$ 2.0%	34.7% $\pm$ 2.4%	17.7% $\pm$ 1.9%
10	53.0% $\pm$ 2.5%	26.9% $\pm$ 2.2%	54.0% $\pm$ 2.5%	21.6% $\pm$ 2.0%
15	68.8% $\pm$ 2.3%	32.9% $\pm$ 2.3%	66.2% $\pm$ 2.3%	26.0% $\pm$ 2.2%
20	79.8% $\pm$ 2.0%	38.5% $\pm$ 2.4%	75.0% $\pm$ 2.1%	30.2% $\pm$ 2.3%
25	85.9% $\pm$ 1.7%	43.3% $\pm$ 2.5%	81.4% $\pm$ 1.9%	34.5% $\pm$ 2.4%

5.2 Aggregate vs. Per-Game Discrepancy

After filtering untriggered or short-game records (reducing the dataset from 62,669 to 32,380 games), Table 3 compares aggregate corpus-level phase gaps against per-game Mean Absolute Error (MAE).

Table 3. Aggregate phase-length gap vs. per-game MAE (moves). $\pm$ is the 95% half-width.

Phase	Aggregate gap	Per-game MAE	Acc@10
Opening	0.3 $\pm$ 2.3 (1.4%)	5.5 $\pm$ 0.1	85.6% $\pm$ 0.4%
Middlegame	13.3 $\pm$ 7.2 (5.8%)	19.1 $\pm$ 0.2	30.4% $\pm$ 0.5%
Endgame	20.3 $\pm$ 8.5	39.3 $\pm$ 0.3	13.5% $\pm$ 0.4%
Overall	—	34.6 $\pm$ 0.3 (15.2%)	15.3% $\pm$ 0.4%

While aggregate gaps appear small, per-game MAE highlights substantial individual misalignment, particularly in the middle game and endgame. Figure 1 illustrates a representative trajectory where probability oscillations persist near phase boundaries before stabilizing.

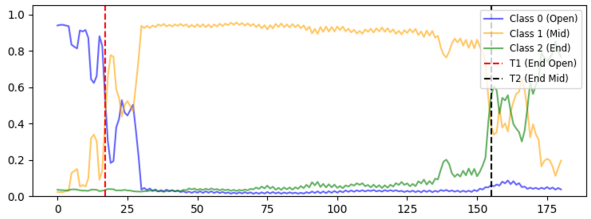


Fig. 1. Per-move class probability trajectory, one game, with T1 (red) / T2 (black).

All evaluations rest on a single backbone (the 2023 AI CUP champion model, repurposed for this task); whether the conservative, low-variance behavior is a property of this architecture or would recur with a different sequence model remains open. The two annotation experts also labeled subsets of the same underlying game pool used to build the training and application sets, rather than an entirely independent corpus — our correlations therefore test agreement on familiar games, not yet generalization to an unfamiliar collection. The planned AI CUP release is a natural opportunity to address this directly.

6. Conclusion

We built a data-preprocessing pipeline and an N+1 state classification model that segments Go games into opening, middle-game, and endgame phases with 87–89% held-out move-level accuracy, together with two heuristics that convert its per-move probabilities into two transition points per game. Two independent lines of evidence — pooled-versus-per-annotator cross-validation (5.1) and an aggregate-versus-per-game comparison of corpus-level phase-length statistics (5.2) — both show T1/T2 are considerably less reliable at the individual-game level than move-level accuracy implies, a bias we attribute to the resampling-based data augmentation and the fixed-threshold deployment heuristics.

References

[1] G. Erickson and M. Buro, "Global state evaluation in StarCraft," in Proc. AAAI Conf. Artif. Intell. Interact. Digit. Entertain., vol. 10, no. 1, 2014, pp. 73–79.

[2] N. Justesen and S. Risi, "Learning macromanagement in StarCraft from replays using deep learning," in Proc. IEEE Conf. Comput. Intell. Games (CIG), 2017, pp. 162–169.

[3] S. Roohi, M. Takano, and T. Nakajima, "Predicting game difficulty and player performance using deep learning," arXiv:2107.12061, 2021.

[4] H. Zhou, H. Tang, M. Li, et al., "Explaining how a neural network plays the Go game and let people learn," arXiv:2310.09838, 2023.

[5] T.-Y. Chen, C.-L. Yu, C.-W. Su, K.-H. Liao, and H.-Y. Hsiao, "Cognitive abilities across the opening, middle-game, and endgame phases of Go," Chinese J. Psychology, vol. 61, no. 3, pp. 173–196, 2019.

[6] J. Moudrík, P. Baudiš, and R. Neruda, "Evaluating Go game records for prediction of player attributes," in Proc. IEEE Conf. Comput. Intell. Games (CIG), 2015, pp. 162–168.

© 2026 Information Processing Society of Japan