Proceedings of the 20th Conference on Computer Science and Intelligence Systems (FedCSIS) pp. 819–824
DOI: 10.15439/2025F1698
ISSN 2300-5963 ACSIS, Vol. 43

A Stacking-Based Ensemble Approach for Predicting Chess Puzzle Difficulty

Alan Liang
University of Southern California
aqliang@usc.edu

Cenzhi Liu
Fuhua Singapore
cenzhi128@gmail.com

Kai Wang
University of Pennsylvania
kai6@sas.upenn.edu

Ethan Liu
Rose-Hulman Institute of Technology
liuj15@rose-hulman.edu

***Abstract*—FedCSIS 2025 competition is to predict the difficulty of chess puzzles, we present a structured multi-stage regression pipeline developed for the FedCSIS 2025 Challenge. The approach consists of three stages: (i) four Elo-banded base models trained on separate rating ranges to capture localized difficulty semantics and mitigate bias in imbalanced datasets; (ii) a feature-level stacking ensemble combining base predictions with structural attributes, such as success probabilities, failure distributions, and solution length, to enhance cross-band generalization; and (iii) a lightweight post-hoc residual correction to reduce systematic prediction biases. Additionally, an uncertainty-aware mask-based evaluation is introduced to identify the 10% most challenging puzzles for extended scoring.**

Our method achieved competitive results, ranking 7th in the final leaderboard, while maintaining low computational cost. These findings demonstrate that lightweight, interpretable models, when combined with structural reasoning and uncertainty estimation, can rival more complex deep-learning approaches. This study highlights the potential of structured machine learning pipelines for scalable, human-centric chess puzzle analytics.

***Index Terms*—Chess puzzle difficulty prediction, Elo-banded modeling, Stacking ensemble, Meta-learning, Structural features, Residual correction**

I. Introduction

CHESS puzzle difficulty prediction involves assessing not only the tactical correctness of moves but also their perceived complexity for human players. Human performance depends on multiple factors, such as move sequence length, tactical motifs, time pressure, and psychological biases, which are often poorly correlated with engine evaluations. The increasing availability of large-scale puzzle-solving data from online platforms has fueled interest in data-driven approaches to this problem.

A. Related Work

With the rise of deep learning, end-to-end approaches became dominant, as mentioned in the previous IEEE BigData 2024 Cup: chess puzzle competition report [1]. Woodruff et al. [2] proposed neural models and won the IEEE BigData 2024 Cup. Miłosz and Kapusta [3] proposed GlickFormer, a spatio-temporal transformer jointly modeling board states and move sequences, significantly outperforming earlier transformer-based models and ranking among the top entries in the IEEE BigData 2024 Cup, while Ruta *et al.* [4] introduced a convolutional neural network (CNN) that mapped board configurations to difficulty ratings, achieving strong correlation with human ratings.

B. Our Contributions

Our work proposes a structured and computationally efficient pipeline for chess puzzle difficulty prediction. The key contributions are:

- Training four Elo-banded base models on separate rating ranges to capture localized difficulty semantics and mitigate bias in imbalanced data.
- Combining base predictions with structural puzzle features in a heterogeneous stacking ensemble, improving generalization across diverse puzzle types.
- Applying a post-hoc residual correction to reduce systematic biases and introducing an uncertainty-aware mask to identify the 10% most challenging puzzles for extended evaluation.
- Achieving competitive performance, i.e. 6th in the preliminary and 7th in the final leaderboard, while maintaining low computational cost, demonstrating that structured lightweight models can rival more complex deep-learning approaches.
- For computational efficiency, we intentionally avoid excessively long training schedules. Instead of relying on prolonged base model training, which is time-consuming for millions of samples, later stages—stacking and residual correction—are designed to refine predictions using cross-band interactions and structural reasoning. This strategy provides a better balance between accuracy and runtime in practical competition settings.

The remainder of this paper is organized as follows: Section II introduces the competition task, dataset characteristics, and evaluation protocol. Section III describes the proposed methodology, including the band-specific base models, the stacking ensemble, and the post-hoc residual correction with uncertainty estimation. Section V presents the experimental setup, ablation studies, and official leaderboard results. Finally, Section VI concludes the paper and discusses potential future work.

II. Competition Description

The FedCSIS 2025 Challenge [5] organized on the KnowledgePit platform[1] is the continuation of the highly successful first edition organized as part of the IEEE BigData Cup 2024 [1] . This second edition further extends the benchmark by

[1] https://knowledgepit.ai/

providing an updated large-scale dataset and refined evaluation protocol, aimed at advancing algorithms that estimate human-perceived puzzle difficulty. Unlike chess engines optimized for best-move accuracy, the task focuses on modeling human solving performance, a key requirement for adaptive training systems, personalized recommendation engines, and educational applications. The difficulty level is measured as the rating on the lichess platform[2].

A. Task Definition

Participants are required to predict a continuous difficulty rating for each chess puzzle, expressed as a Glicko-2 rating[3] equivalent. The official evaluation metric is the Mean Squared Error (MSE)[4]:

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2, \tag{1}$$

where y_i is the ground-truth human-derived rating and $\hat{y}_i$ is the predicted rating. The competition adopts a two-stage leaderboard system:

- Preliminary Stage: A public leaderboard based on a subset of the test set for iterative submissions.
- Final Stage: A private leaderboard evaluated on the full hidden test set, determining the official ranking.

B. Dataset and Features

The official dataset comprises a large labeled training set and an unlabeled test set:

- Training Set: 4,557,000 puzzles annotated with human-derived Glicko-2 ratings and engine-computed statistics.
- Test Set: 2,235 puzzles sharing the same feature structure but without difficulty ratings. Predictions for this set are used for final evaluation.

Each puzzle is described by 32 structured features in the training set and 25 features in the test set:

1) Core Information: PuzzleId, Forsyth–Edwards Notation (FEN)[5] for board state, and Portable Game Notation (PGN)[6] for solution moves.
2) Human-Performance Annotations (training only): Rating (Glicko-2 difficulty), RatingDeviation, Popularity, and NbPlays.
3) Contextual Metadata: Themes, GameUrl, and OpeningTags.
4) Engine-Derived Success Probabilities: 10 rapid-mode columns (success_prob_rapid_1050–2050) and 10 blitz-mode columns (success_prob_blitz_1050–2050), representing estimated human success rates at different skill levels.

By decoding the FEN, the chessboard can be illustrated into image. For example, the below Figure 1 shows the chessboard initial state decoded by FEN and the rating is 1300.

[2]https://lichess.org/
[3]https://en.wikipedia.org/wiki/Glicko_rating_system
[4]https://en.wikipedia.org/wiki/Mean_squared_error
[5]https://en.wikipedia.org/wiki/ForsythEdwards_Notation
[6]https://en.wikipedia.org/wiki/Portable_Game_Notation

Figure 1. FEN: "8/4R3/1p2P3/p4r2/P6p/1P3Pk1/4K3/8 w - - 1 64".

C. Key Challenges

The challenge introduces several unique difficulties:

- Human-Centric Bias: Ratings are derived from solver statistics rather than engine evaluations, so engine-trivial tactics may still be difficult for humans.
- Imbalanced Difficulty Distribution: Sparse high-Elo samples are prone to underestimation by global models trained on mid-range-dominated data.
- High-Dimensional Structured Features: Success probabilities across multiple skill bands must be effectively combined without overfitting.
- Scalability and Interpretability: Models must efficiently process millions of samples while maintaining transparency for educational use cases.

D. Extended Mask-Based Evaluation

An additional subtask evaluates uncertainty estimation. Participants submit a binary mask identifying the 10% most error-prone test puzzles. Scores are recomputed by replacing masked predictions with ground-truth ratings, and rankings are determined by the ratio between the adjusted score and the theoretical "perfect mask." This extension highlights the importance of reliable uncertainty estimation.

III. Methodology

This section presents the detailed methodology of our solution for the FedCSIS 2025 Challenge main task. The proposed approach follows a structured multi-stage pipeline designed to balance accuracy, interpretability, and computational efficiency. We first provide an overview of the entire pipeline, then highlight its key methodological contributions, followed by a detailed description of each stage.

A. Overall Pipeline

The pipeline consists of three major stages:

1) Elo-Banded Base Models: Four band-specific models are trained on separate Elo ranges using structured puzzle features (engine-derived statistics and positional indicators) to capture localized difficulty semantics and

reduce prediction bias caused by highly imbalanced rating distributions.

2) Feature-Level Stacking Ensemble: Outputs from base models are combined with structural puzzle features in a heterogeneous meta-learning framework, improving cross-band generalization.
3) Post-Processing Rating Prediction: A lightweight residual correction adjusts systematic biases, producing the final predicted ratings submitted to the competition.

This design allows predictions to be progressively refined: base models specialize in local rating regions, stacking integrates global patterns, and post-processing corrects residual systematic errors.

B. Major Contributions in Method Design

The key methodological contributions include:

- Band-Specific Specialization: Instead of a single global model, band-wise training explicitly targets the diverse difficulty distributions across Elo ranges.
- Structured Feature Utilization: Engine-derived success probabilities and handcrafted positional indicators are explicitly exploited both in base and stacking models, improving interpretability and generalization.
- Hybrid Meta-Learning: By combining a linear model (Ridge), a tree-based model (XGBoost), and a neural network (MLP), the stacking ensemble exploits complementary strengths.
- Lightweight but Scalable: All components are computationally efficient and scalable to millions of samples, unlike many deep-learning-based solutions.
- Bias Mitigation via Post-Processing: A simple, interpretable residual correction effectively addresses underestimation in high-Elo regions.

C. Step 1: Elo-Banded Base Models

Puzzle ratings span from approximately 400 Elo (basic tactics) to over 3000 Elo (master-level combinations), leading to a potential imbalanced difficulty distribution, as summarized in Fig. 2. Although high-Elo puzzles (≥ 1700) constitute over one-third of the data, their solving patterns differ substantially from lower bands, and a single global model trained on such mixed distributions often overfits mid-range samples while underestimating high-Elo puzzle difficulty.

To address this, we adopt a band-wise modeling strategy, training four independent models, each specialized for a designated Elo range [2]. This specialization allows each model to focus on localized difficulty patterns, improving prediction accuracy across heterogeneous rating bands.

1) Training Data Selection: All training samples are assigned to exactly one band-specific model according to their difficulty ratings. Unlike the default global-training setup of the baseline regression code, we manually configured the training pipeline to filter puzzles into four disjoint Elo bands:

- **Small model:** Rating $<$ 1000 (beginner-level puzzles with simple tactical motifs),

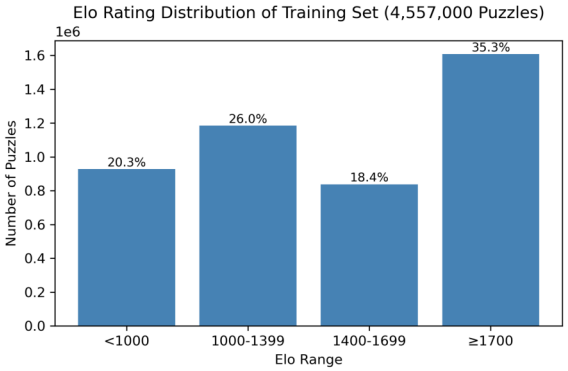

Figure 2. Elo rating distribution of the training set (4,557,000 puzzles), showing heterogeneous distributions across rating bands.

- **1300 model:** $1000 \leq$ Rating $<$ 1400 (intermediate-level puzzles),
- **1500 model:** $1400 \leq$ Rating $<$ 1700 (club-level puzzles with mixed tactical and strategic depth),
- **1700+ model:** Rating $\geq$ 1700 (advanced puzzles requiring deeper tactical reasoning).

This standardized partitioning ensures full data utilization and enables each model to learn the statistical patterns and solving dynamics specific to its designated Elo range, which is particularly beneficial for the sparse high-Elo band.

2) Model Architecture: All four band-specific models are trained as Multi-Layer Perceptron regressors (MLP), following a standard supervised learning framework:

- Input Representation: Each puzzle is represented entirely by structured features extracted from the official dataset:
  - Engine-derived statistics such as per-move success probabilities, mean, standard deviation, max, and min probabilities,
  - failure probability and distribution skewness,
  - material balance,
  - solution length and other positional descriptors.

  All continuous features are normalized to the range [0,1].
- Network Structure: Each model consists of several fully connected layers with ReLU activations, followed by a single linear regression head that outputs a scalar difficulty rating. This MLP architecture is well-suited to tabular structured data and provides efficient training on millions of samples.
- Training Objective: All models are optimized with the standard Mean Squared Error (MSE):

$$\mathcal{L}_{\text{base}} = \frac{1}{N}\sum_{i=1}^{N}(y_i - f_\theta(x_i))^2, \tag{2}$$

  where f_θ is the band-specific MLP and y_i the ground-truth difficulty rating.
- Epochs and Fine-Tuning: The number of training epochs is determined empirically based on validation performance, typically ranging from 10 to 20 epochs depending

on band size, model convergence speed, and computational budget. This relatively small epoch range reflects a trade-off between convergence and time efficiency, as prolonged training provides diminishing returns and is computationally expensive for large-scale data.
An internal validation set, sampled as 10% of the band-specific training data, is used exclusively for hyperparameter tuning and to save the best-performing checkpoints based on validation MSE.

3) Inference and Outputs: After training, each model is applied independently to the official test set using a standardized inference pipeline. The four models generate separate predictions corresponding to the small, 1300, 1500, and 1700 Elo bands. These predictions form the base inputs for the stacking ensemble in Step 2.

D. Step 2: Stacking Ensemble

1) Motivation: While band-specific models effectively capture localized difficulty patterns, they lack global consistency and fail to fully exploit cross-band correlations. A meta-learning strategy can integrate predictions from multiple bands and leverage structural attributes to correct residual inconsistencies. In particular, features such as failure probability and move counts provide complementary information about puzzle-solving dynamics that is not fully encoded in the band-wise models.

2) Meta-Feature Construction: A comprehensive meta-feature vector is constructed by combining base-model predictions and structural puzzle attributes:

$$\begin{aligned}\mathbf{z} = [&p_1,\ p_2,\ p_3,\ p_4,\ \tilde{p},\ \sigma_p,\\ &\text{avg\_success},\ \text{fail\_prob},\ \text{inflection\_rating},\\ &\text{fail\_skew},\ \text{num\_move}]\end{aligned} \tag{3}$$

where:

$$\tilde{p} = \text{median}(p_i), \qquad \sigma_p = \sqrt{\frac{1}{4}\sum_{i=1}^{4}(p_i - \tilde{p})^2}. \tag{4}$$

Here, $\tilde{p}$ represents the robust central tendency of the base predictions, while σ_p serves as an implicit confidence measure, indicating inter-model disagreement.

3) Meta-Learners and Complementarity: The stacking ensemble integrates three heterogeneous meta-learners chosen for their complementary modeling capacities:

- Ridge Regression: A linear regression model with L_2 regularization, which stabilizes coefficient estimation by penalizing large weights. Ridge captures global linear trends between structural attributes (e.g., average success probability, number of moves) and target ratings. Its interpretability provides valuable insight into the relative importance of meta-features.
- XGBoost: An ensemble of gradient-boosted regression trees that sequentially fits residual errors. XGBoost is well-suited for modeling nonlinear feature interactions and conditional relationships, such as detecting puzzles that are deceptively difficult despite short move sequences.
- MLP Regressor: A lightweight feedforward neural network configured with two hidden layers (128 and 64 neurons, ReLU activation). This configuration provides a balance between model capacity and overfitting risk for the low-dimensional meta-feature vector, while effectively capturing high-order nonlinear dependencies not easily approximated by tree-based methods.

4) Prediction Aggregation: The outputs of the three meta-learners are aggregated via an adaptive weighted average:

$$\hat{y}_{stack} = w_{\text{ridge}}\hat{y}_{\text{ridge}} \ + \ w_{\text{xgb}}\hat{y}_{\text{xgb}} \ + \ w_{\text{mlp}}\hat{y}_{\text{mlp}}, \tag{5}$$

where the weights are inversely proportional to the validation residual variance:

$$w_j = \frac{1/\sigma_j^2}{\sum_k (1/\sigma_k^2)}, \qquad j \in \{\text{ridge}, \text{xgb}, \text{mlp}\}. \tag{6}$$

This dynamic weighting emphasizes models with more stable validation performance. Ridge contributes stability and interpretability, XGBoost captures local conditional interactions, and the MLP learns complex nonlinear relationships, resulting in improved generalization across heterogeneous puzzle types.

E. Step 3: Post-Processing Rating Prediction

1) Motivation: Although the stacking ensemble improves overall prediction accuracy, residual analysis reveals systematic biases: high-Elo puzzles tend to be underestimated, while some low-Elo puzzles are slightly overestimated. These biases arise because the meta-learners only indirectly exploit engine-derived structural signals, which often contain strong priors about puzzle difficulty.

2) Structure-Aware Residual Correction: To explicitly incorporate these priors, we introduce a lightweight residual correction that adjusts the stacking predictions toward a refined structural difficulty estimate.

First, a baseline structural estimate s is computed as a failure-probability-weighted average of rating buckets:

$$s = \frac{\sum_j (1 - \bar{p}_j)\, r_j}{\sum_j (1 - \bar{p}_j)}, \tag{7}$$

where $(1 - \bar{p}_j)$ represents the failure probability in bucket r_j. This estimate is then refined by incorporating two additional structural signals:

- The inflection point r_{inf}, indicating the rating at which the success probability changes most sharply.
- The failure-probability skewness γ, which captures asymmetric difficulty distributions where certain skill groups systematically misjudge a puzzle. highlighting difficulty transitions perceived by human solvers;

The final refined structural estimate is defined as:

$$s^* = 0.6\, s \ + \ 0.2\, r_{\text{inf}} \ + \ 0.2\,(1700 + 300\,\gamma) \tag{8}$$

Finally, the corrected rating prediction is obtained through a residual fusion with the stacking output:

$$\hat{y}_{\text{final}} = \hat{y}_{\text{stack}} \; + \; \lambda \left(s^* - \hat{y}_{\text{stack}} \right), \qquad \lambda = 0.3. \tag{9}$$

This correction introduces no additional learnable parameters and is computationally efficient. By explicitly leveraging interpretable structural priors, it consistently improves accuracy, particularly in high-Elo regions where data sparsity makes residual biases more severe.

IV. Mask-Based Uncertainty Evaluation

In addition to the main rating prediction task, the FedCSIS 2025 Challenge introduced an extended evaluation explicitly designed to assess uncertainty estimation. Participants were required to submit a binary mask marking the 10% of test puzzles most likely to be mispredicted by their models. The final score was recalculated by replacing predictions for these masked samples with ground-truth ratings, and the ratio between the adjusted score and the theoretically optimal score determined the subtask ranking.

We designed a structure-aware uncertainty estimation method that integrates model disagreement, prediction instability, and structural difficulty indicators into a unified scoring framework. This approach balances model-based and feature-based uncertainty cues, improving the alignment of selected samples with actual model errors.

A. Scoring Function Design

The composite uncertainty score for each puzzle is defined as:

$$u_i = \alpha \cdot \sigma_{p,i} \; + \; \beta \cdot \delta_i \; + \; \gamma \cdot \psi_i \tag{10}$$

where:

- $\bar{p}_i$: average success probability across rating buckets for puzzle i,
- z_i: Z-score of the material balance in the training data; for the test set, this term defaults to zero as no material information is provided,
- $\mathbb{I}(|z_i| > \tau)$: indicator of extreme material imbalance, with $\tau = 3.0$ as the outlier threshold.

The weighting coefficients were empirically set to $\alpha = 1.0$, $\beta = 1.0$, and $\gamma = 0.4$ to optimize the uncertainty ranking quality.

B. Mask Generation

Puzzles are ranked according to u_i, and the top 10% are assigned a mask value of 1:

$$\text{mask}_i = \begin{cases} 1, & i \text{ in top } 10\% \text{ highest scores}, \\ 0, & \text{otherwise}. \end{cases} \tag{11}$$

This binary mask was submitted separately from the rating predictions, providing an explicit measure of the model's uncertainty estimation capability.

C. Discussion

The proposed uncertainty estimation method combines model-driven and feature-driven indicators in a single interpretable framework. Its main advantages include:

- improved consistency between selected samples and actual prediction errors compared to variance-only methods;
- and explicit consideration of structural puzzle difficulty, beyond pure statistical disagreement.

However, the method still relies on the alignment of raw and post-processed predictions and may overlook systematic biases shared across all models. Despite these limitations, the approach provides a strong baseline for uncertainty-aware chess puzzle difficulty estimation.

Our uncertainty mask ratio is 1.696, placing us 8th out of the 9 teams that chose to participate in this additional task. Using our submitted mask, our final score is approximately 57,931, while using a "perfect mask" would yield a score of about 34,162. The full results for all teams will be published in the competition report [5].

V. Experimental Setup

A. Dataset and Preprocessing

The dataset provided by the organizers consists of two disjoint parts:

- Training set: Approximately 4,557,000 puzzles, each labeled with a human-perceived difficulty rating derived from aggregated player performance in Lichess (Glicko-2 system). This set is fully annotated and serves as the only source of labeled data for model development.
- Test set: An unlabeled set of 2,235 puzzles used exclusively for the final evaluation. Participants are required to submit predicted ratings for this set, while the ground-truth ratings remain hidden.

To ensure balanced representation across different rating bands, the official training set is split using stratified sampling within each Elo band:

- Base-model training subset: 90% of puzzles per Elo band, used for training the four band-specific base models.
- Validation subset: 10% of puzzles per Elo band, reserved exclusively for generating unbiased out-of-sample predictions for stacking and for residual bias calibration.

The stacking meta-learners are trained solely on this held-out validation subset to prevent data leakage. To improve generalization within the stacking stage, a 10-fold cross-validation is performed on the validation subset: in each fold, the meta-learners are trained on 90% of the fold and validated on the remaining 10%, and the out-of-fold predictions are averaged for final submission.

All continuous features, including success probabilities, failure distributions, and structural statistics, are normalized to the [0,1] range. PGN sequences are parsed to compute derived attributes such as the number of solution moves (*num_move*), and FEN strings are used to extract positional features such as castling rights and piece material balance.

B. Training Details

Each Elo-banded base model is implemented as a lightweight multi-layer perceptron (MLP) following the official baseline recommendations. The models are trained for 20–30 epochs with early stopping based on validation loss. The best-performing checkpoints are selected, and no cross-validation is applied at the base-model stage to keep training computationally efficient.

The meta-learner integrates Ridge regression (L2 regularization), XGBoost (max depth = 8, learning rate = 0.01-0.05, 1000-2000 boosting rounds) and a lightweight MLP (two hidden layers: 128 and 64 neurons, ReLU activation, dropout = 0.1). The meta-features are constructed exclusively from base model predictions on the validation subset (out-of-sample), and a 10-fold cross-validation within this subset is used to improve the robustness of the meta-learners.

A structure-aware residual correction is fitted to the validation residuals, explicitly adjusting the stacking predictions toward structural difficulty estimates, especially mitigating underestimation in sparse high-Elo regions.

C. Overall Performance

Table I shows the progressive improvements across pipeline stages.

Table I
PROGRESSIVE IMPROVEMENTS ACROSS PIPELINE STAGES (INTERNAL HOLD-OUT TEST SET).

Pipeline Stage	MSE	Relative Gain
Base Models (avg/median)	78,000–80,000	–
Stacking Ensemble (10-Fold)	71,000–73,000	+9–11%
Post-Processing + Best Avg	66,600–68,000	+6–8%

The pipeline shows clear incremental gains: stacking substantially improves cross-band consistency, while residual correction yields additional improvements in high-Elo regions.

D. Ablation Study

Table II highlights the contribution of each major component.

Table II
ABLATION STUDY: CONTRIBUTION OF EACH COMPONENT (VALIDATION SET).

Configuration	MSE	Change vs. Full
Full Pipeline	66,600–68,000	Baseline
w/o Structural Features	∼72,000	+6–8%
w/o Stacking Ensemble	∼78,000	+15–18%
Single Global Model	>85,000	+25%

Removing structural features increases variance, confirming their role as complementary difficulty indicators. Stacking yields the largest single improvement by integrating base-model predictions with structural cues, while replacing band-wise models with a single global model leads to severe bias, particularly for high-Elo puzzles.

E. Discussion

These results confirm the effectiveness of a structured and interpretable pipeline:

- Elo-banded modeling alleviates data imbalance and reduces extreme rating bias;
- Stacking with structural reasoning provides the largest performance gain by capturing cross-band interactions;
- Residual calibration yields additional improvements in sparse high-Elo regions with negligible computational cost.

The final MSE of 62,685 ranks among the top solutions, demonstrating that a lightweight and transparent ensemble can rival more computationally expensive deep-learning models, making it suitable for educational or large-scale online puzzle recommendation systems.

VI. CONCLUSION

This study presented a structured and interpretable pipeline for chess puzzle difficulty prediction in the FedCSIS 2025 Challenge. The approach combines three complementary stages: (1) Elo-banded base models specialized for different rating ranges to reduce distributional bias, (2) a feature-level stacking ensemble that integrates base predictions with structural puzzle attributes to improve cross-band generalization, and (3) a lightweight structure-aware residual correction to mitigate systematic errors, particularly in sparse high-Elo regions.

Extensive experiments demonstrated clear incremental improvements at each stage, achieving a final MSE of 62,685 and ranking 7$^{\text{th}}$ in the final leaderboard among resource-intensive deep-learning solutions. The results confirm that carefully designed ensembles, when combined with domain-specific structural reasoning, can achieve competitive accuracy with far lower computational cost and higher interpretability.

REFERENCES

[1] J. Zyśko, M. Świechowski, S. Stawicki, K. Jagieła, A. Janusz and D. Ślęzak, "IEEE Big Data Cup 2024 Report: Predicting Chess Puzzle Difficulty at KnowledgePit.ai," 2024 IEEE International Conference on Big Data (BigData), Washington, DC, USA, 2024, pp. 8423-8429, doi: 10.1109/BigData62323.2024.10825289.

[2] T. Woodruff, O. Filatov and M. Cognetta, "The bread emoji Team's Submission to the IEEE BigData 2024 Cup: Predicting Chess Puzzle Difficulty Challenge," 2024 IEEE International Conference on Big Data (BigData), Washington, DC, USA, 2024, pp. 8415-8422, doi: 10.1109/BigData62323.2024.10826037.

[3] K. Miłosz and A. Kapusta, "GlickFormer: A Spatio-Temporal Transformer for Chess Puzzle Difficulty Prediction," in *Proc. IEEE BigData Conf.*, 2024.

[4] D. Ruta, M. Liu and L. Cen, "Moves Based Prediction of Chess Puzzle Difficulty with Convolutional Neural Networks," 2024 IEEE International Conference on Big Data (BigData), Washington, DC, USA, 2024, pp. 8390-8395, doi: 10.1109/BigData62323.2024.10825595.

[5] J. Zyśko, M. Ślęzak, D. Ślęzak, and M. Świechowski, "FedCSIS 2025 knowledgepit.ai Competition: Predicting Chess Puzzle Difficulty Part 2 & A Step Toward Uncertainty Contests," in *Proc. 20th Conf. Comput. Sci. Intell. Syst. (FedCSIS)*, vol. 43, Polish Inf. Process. Soc., 2025. doi: http://dx.doi.org/10.15439/2025F5937.