Expected Value Intelligence Engine (EVIE)

Claire Longo[1]

[1]Statistician in Stilettos

July 8, 2025

A Probabilistic Framework for Poker Tournament Value Estimation with Conversational Feedback via LLMs

Abstract

This paper presents the design and architecture of EVIE (Expected Value Intelligence Engine), a personalized poker coaching system that combines probabilistic modeling of tournament outcomes with a Retrieval-Augmented Generation (RAG) interface powered by large language models (LLMs). We introduce a probabilistic framework that models live multi-table tournament performance using individualized skill distributions and payout structures: player rankings are normalized across tournament sizes using a continuous beta distribution to estimate finish percentiles, while a power-law function captures the payout curve. This enables personalized expected value (EV) estimation across diverse tournament formats. The RAG-based chatbot delivers conversational, context-aware coaching by integrating insights from a user's historical tournament results, prior coaching chat patterns, and model-driven EV projections. The result is an interpretable, scalable system that empowers amateur and semi-professional players to improve strategic decision-making through natural language feedback grounded in real placement data. We detail the system's architecture, mathematical foundations, and implementation strategies for real-world use.

1 Introduction

Poker is a game of incomplete information, strategic reasoning, and long-term statistical convergence. While much of the existing AI research in poker has focused on building agents that play optimally, relatively little work addresses the complementary goal of supporting human players with personalized coaching tools. This gap is particularly evident for amateur and semi-professional players participating in live multi-table tournaments, who often seek to improve their outcomes, select better tournaments, and track trends in their performance over time.

Advanced players commonly rely on solvers, advanced data tools, or human coaching, but these resources tend to be either too complex, too costly, or insufficiently tailored for everyday players. Tools like PioSolver and GTO+ are powerful for studying Game Theory Optimal (GTO) strategies, but they are not designed to help humans improve through personalized feedback. They lack interpretability, adaptability, and support for reasoning across diverse tournament structures, payout models, and field sizes. Moreover, they do not offer intuitive, conversational interfaces that allow users to explore insights in natural language. Tools like PokerGPT offer an AI chat coaching interface, bjut rely heavily on hand-by-hand analysis and GTO play. Our system rejects GTO as the universal mathematical framework for poker and takes a different approach rooted in long-term probabilistic modeling that reflect real-world tournament dynamics and draws inference from the model through practical AI coaching conversations. Our system is also focused tournament play not just hand by hand analysis. By prioritizing strategic tournament selection, adaptive feedback, and conversational coaching over perfect-play simulation, this application fills a critical gap: it supports the human player's personal journey across tournaments, rather than just evaluating isolated hands.

This work presents a lightweight, user-centric AI coaching tool, EVIE, that analyzes historical tournament data and draws from coaching chat histories to provide individualized advice through a natural language interface. Our goals are twofold: (1) to democratize personalized poker coaching through an accessible dialogue interface powered by AI, and (2) to enable data-driven tournament selection and strategy refinement using meaningful expected value estimations derived from a player's skill and performance history.

To achieve this, we introduce a probabilistic framework that models a player's EV across tournaments using a normalized, continuous representation of historical placements. The model accounts for tournament-specific payout structures and player skill distributions, enabling generalization across a wide variety of tournament formats. This statistical model is embedded within a Retrieval-Augmented Generation (RAG) system that leverages LLMs to deliver conversational coaching. The AI chatbot integrates insights from tournament results and prior coaching exchanges to deliver context aware, personalized feedback.

The result is a scalable and interpretable coaching system that evolves with the user's skill development and supports better decision-making. By combining statistical modeling with the conversational strengths of LLMs, this tool offers a novel and practical approach to personalized poker coaching.

2 System Design Architecture

The EVIE system is built as a modular, end-to-end pipeline that supports scalable ingestion of tournament data, probabilistic skill modeling, and interactive explanation via the LLM-based RAG interface.

2.1 Architecture Overview

The architecture design consists of three main modular subsystems:

1. Data Layer
2. Probabilistic Modeling Module
3. LLM RAG Chat Interface

Each module is designed to be loosely coupled and independently testable, allowing flexibility for future improvements, such as incorporating new model types, live data streams, or fine-tuned models.

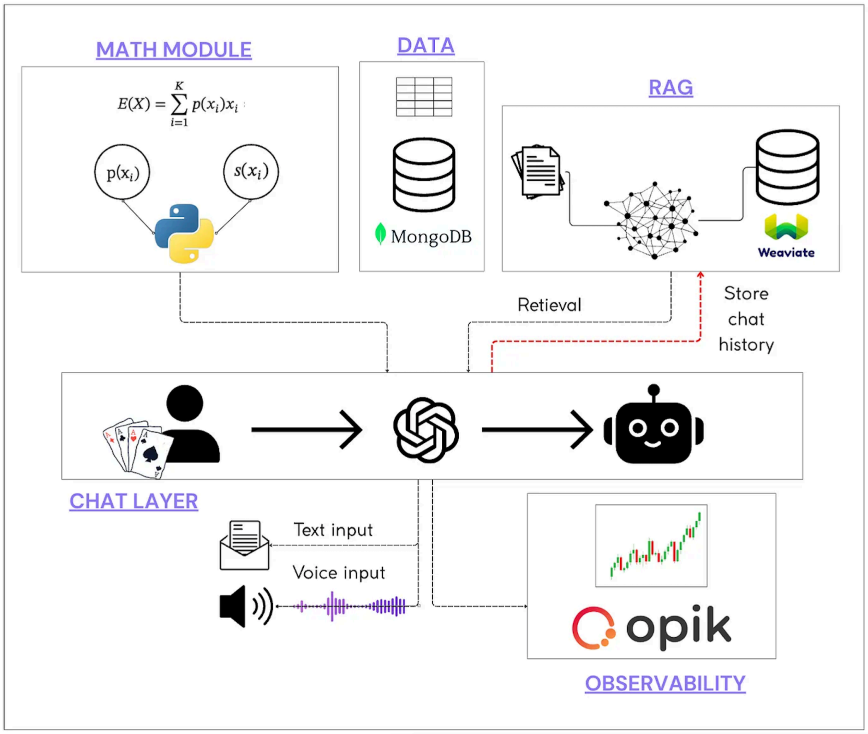

Figure 1: Modular Architecture Design for the EVIE system

2.2 1. Data Layer

Raw tournament data, including player placement, number of entrants, re-entries, and payout structures, is collected manually through user data entry and stored in a central database.

Each tournament entry is converted into a standardized schema including: Player ID, Casino, Placement rank, Payout, Number of Entrants, Additional tournament metadata (if available), Free Text user notes (if available)

This allows the EVIE database to serve as a system of record for a player's tournament performance, collecting a wide range of variables, including both tabular and free-text notes, which can be used to correlate performance in a mathematical model.

2.3 2. Probabilistic Modeling Module

EVIE's modeling engine estimates a player's underlying skill distribution using a Beta distribution over the normalized finish percentile r. Payout structures are modeled using a power-law function to approximate real-world tournament dynamics.

These two components are combined to compute the player's EV in any future tournament setting. Parameters of these distributions such as α, β, and γ are estimated via the Method of Moments and stored for each player profile.

2.4 3. LLM RAG Chat Interface

RAG architectures have shown effectiveness in domain-specific knowledge applications [2]. We apply this paradigm to build an adaptive poker coaching interface into EVIE that is powered by patters detected in past chat histories with the user, allowing natural language exploration of personal performance data. This RAG system is what powers the unique personalization and memory capabilities in the the interactive coaching chatbot interface.

- **Embeddings:** A simple embedding-based vector store contains user chat histories.
- **Retriever:** a retrieval step searches and injects relevant patterns from past chats into the prompts for current conversations, allowing the AI poker coach to identify player patterns and coach based on relevant user context in the current discussion with the player. Output from the user's results data ad probabilistic models and patterns are also considered.
- **Generator:** An LLM generates explanations, recommendations, and answer rationales based on the retrieved context and relevant data.

The RAG chat Agent in EVIE is optimized for providing coaching in the poker context, prioritizing mathematical reasoning and personalized output over stylistic generation. This is accomplished through model selection and prompt engineering.

As shown by Shojaee et al. [3] in the popular paper "The Illusion of Thinking: Understanding the Strengths and Limitations of Reasoning Models via the Lens of Problem Complexity", LLM models can collapse on complex tasks. LLMs often simulate reasoning rather than truly perform it, especially in domains requiring step-by-step logic, probability, or mathematics. When applied to these tasks, LLMs can provide fluent, convincing, but wrong answers, especially in high-stakes or numerically sensitive contexts. The authors suggest that LLMs should be paired with symbolic systems, retrieval modules, or probabilistic programs to better handle reasoning tasks.

Therefore, we have carefully designed EVIE to separate the mathematical calculations and LLM-powered chat component into different modules. EVIE's AI chatbot component is constricted to not perform any of the advanced mathematical calculations presented in this paper. Instead, the LLM will review the output of the equation to make optimal recommendations to the player, and deliver these recommendations in a chat interface. This allows the LLM chat to serve as a as a kind of interface between the user to the probabilistic models and calculations.

3 Methods

3.1 Mathematical Modeling

Expected Value Let $x \in \{1, \ldots, K\}$ be a discrete variable denoting a player's finishing place in a tournament with K entrants. Let $p_p(x)$ be the player's skill distribution for player p, and $s_t(x)$ the tournament payout structure for tournament t. Then the expected value from the tournament session is:

$$\mathbb{E}_{p,t}(X) = \sum_{i=1}^{K} p_p(x_i)\, s_t(x_i) \tag{1}$$

Player Ranking Model A players chances of placing in rank x can be modeled with the assuming a Binomial distribution:

$$P(X = x) = \binom{K}{x} p^x (1-p)^{K-x}, \quad x = 1, 2, \ldots, K \tag{2}$$

Raw placements like "2nd place" or "10th place" don't mean the same thing in a 10-player vs. 100-player tournament. For example, a 2nd-place finish in a 10-player tournament isn't equivalent to 2nd in a 50-player tournament. We need a common scale so we normalized placement as a continuous variable r This gives us a percentile representation of the finishing rank independent of the number of players K so we can easily across tournaments of different sizes and still use data from various tournaments to inform the model and estimate the parameters. To normalize placement across tournaments of different sizes, define:

$$r = \frac{x-1}{K-1} \in [0, 1] \tag{3}$$

Now we can model the shape of this player's finishing tendency over many tournament a smooth distribution over the range [0,1]. When the number of tournaments T played is large enough we can approximate the binomial distribution with a continuous Beta distribution.

$$\begin{aligned} P(X) = \binom{K}{x} p^x (1-p)^{K-x} \\ \sim Beta(\alpha, \beta) \end{aligned} \tag{4}$$

The probability density function (PDF) of the Beta distribution is given by:

$$p(r; \alpha, \beta) = \frac{r^{\alpha-1}(1-r)^{\beta-1}}{B(\alpha, \beta)} \tag{5}$$

Where $B(\alpha, \beta) = \int_0^1 t^{\alpha-1}(1-t)^{\beta-1}dt$. The mean of this distribution is:

$$\mu = \frac{\alpha}{\alpha + \beta}$$

and

α The first shape parameter of the distribution; controls the skewness.

β The second shape parameter of the distribution; also controls the skewness.

The Beta distribution is flexible and changes shape dramatically depending on α and β. This models individual player skill levels well.

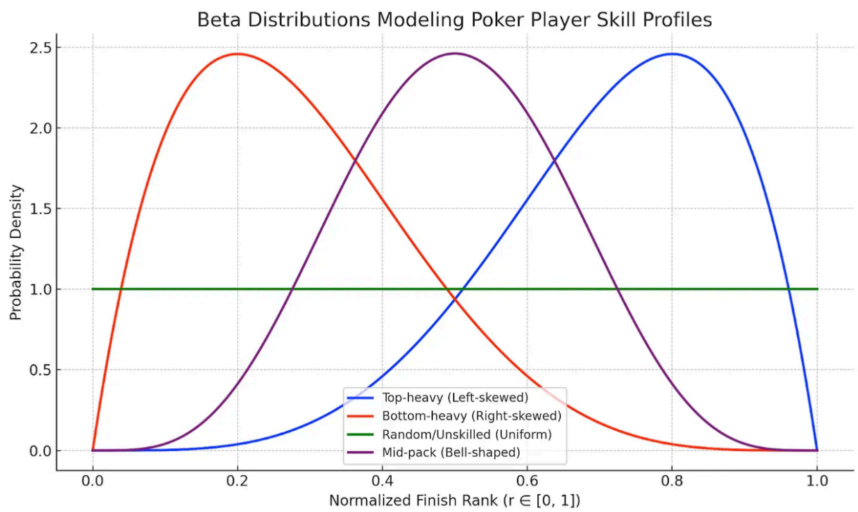

Figure 2: Example of beta distribution modeling different player skill profiles.

Payout Structure Model Most tournaments only pay the top $m = 10\% - 15\%$ of players. When a player places in a paid rank, this is called "ITM" or "In the Money". This payout structure depends on the casino and the casinos typically use a complex algorithm to determine the payout structure as it varies by number of entrants and other tournament features. Our goal, however, is to model a player's abilities and estimate their EV across a variety of tournaments, so we derive a function that closely models common payout structures.

To accurately approximate payout structure for the purpose of modeling EV across many tournaments, we will use a power law function. The power law function captures the shape of the payout structure where top places are favored with heavier payouts.

$$s(r) = \begin{cases} \frac{1}{(r+\epsilon)^{\gamma}} \cdot [W - (bC + a)] & \text{if } r \leq m \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

Where:

- W: prize pool, $W = K(C - c)$
- C: buy-in cost
- a: add-on cost
- b: number of rebuys
- γ: payout steepness
- m: payout percentile cutoff
- ϵ: small constant to prevent divide-by-zero

Note that any player who places below m, is not "ITM", and will receive a payout of $0.

The parameter γ controls how steep the payout structure is, with high values favoring top players model. γ and m can be adjusted to closely approximate the exact payout structure a specific tournament event t, allowing us to still model the dynamic structure of the tournament based on the number of entrants K. Players are are typically allowed to rebuy - allowing K to vary through the early phase of the tournament.

Final EV Expression We now have flexible models for tournament payout structure $s(r)$, and for a players ability $p(r)$ Putting it all together, the expected value of $s(r)$ under the Beta distribution is: Combining the player skill model and payout model to define a generalized expected value function:

$$
\begin{aligned}
\mathbb{E}_{p,t}(X) &= \sum_{i=1}^{K} p_p(x_i) s_t(x_i) \\
&\sim \mathbb{E}[s(r)] \\
&= \int_0^1 \frac{1}{B(\alpha,\beta)} r^{\alpha-1}(1-r)^{\beta-1} \cdot \frac{1}{(r+\varepsilon)^{\gamma}}[W - (bC + a)]dr
\end{aligned} \tag{7}
$$

Using the data collected from the user on past tournament results, we can use the Method of Moments (MOM) to estimate the parameters for the player's skill distribution, and make accurate EV estimations using equation 7 above.

Assume T tournaments are played. Let $r_1, \ldots, r_T$ be the normalized finish positions. Use method of moments to estimate Beta parameters:

$$\hat{\mu} = \frac{1}{T} \sum_{j=1}^{T} r_j \tag{8}$$

$$\hat{v} = \frac{1}{T} \sum_{j=1}^{T} (r_j - \hat{\mu})^2 \tag{9}$$

Then:

$$\hat{\alpha} = \hat{\mu}\left(\frac{\hat{\mu}(1-\hat{\mu})}{\hat{v}} - 1\right) \tag{10}$$

$$\hat{\beta} = (1-\hat{\mu})\left(\frac{\hat{\mu}(1-\hat{\mu})}{\hat{v}} - 1\right) \tag{11}$$

4 Conclusion

We introduced a probabilistic framework for modeling tournament performance using beta distributions and power-law payout functions. This enables personalized expected value analysis and underpins a coaching system delivered through a LLM-based chat interface.

This research introduces a novel architecture for EVIE that emphasizes accessibility, interpretability, and personalization. By grounding statistical modeling in natural dialogue, the system delivers tailored coaching for live tournament players who are undeserved by current software tools.

Unlike traditional poker engines that focus on solving hands, this system focuses on helping users analyze outcomes and trends across tournaments. The combination of a simple probabilistic model and a responsive RAG chatbot bridges a key usability gap in poker analytics. Future work will focus on refining the player clustering model, improving personalized recommendations, and integrating richer metadata (e.g., time of day, fatigue indicators, or chip stack size at bust).

This project demonstrates how modern AI tools can be used not only for high-performance reasoning in poker, but for human-aligned interpretation of the game dynamics. In doing so, it aims to help a broader population of players get better at poker — one tournament at a time.

5 Related Work

While most existing poker analysis tools rely on game-theoretic solvers (e.g., PioSolver, GTO+), these systems are often difficult to use and provide limited interpretability for non-technical players. Our work differs by focusing on modeling individual player performance across tournaments using statistical techniques, and providing insights via a conversational LLM interface.

Prior research has explored probabilistic modeling in poker using Bayesian methods [1], as well as expected value analysis based on tournament payout structures. However, few if any approaches cast the player finish distribution as a normalized continuous variable, nor attempt to generalize across tournaments of varying structure.

To our knowledge, this is the first approach to combine normalized percentile modeling with RAG-based interaction in a poker coaching application.

References

[1] Jonathan Schaeffer Duane Szafron Darse Billings, Denis Papp. Opponent modeling in poker. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*. AAAI Press, 1998.

[2] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Devang Kulkarni, Panupong Pasupat, Douwe Kiela, Dhruv Batra, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474, 2020.

[3] Parshin Shojaee, Iman Mirzadeh, Keivan Alizadeh, Maxwell Horton, Samy Bengio, and Mehrdad Farajtabar. The illusion of thinking: Understanding the strengths and limitations of reasoning models via the lens of problem complexity. June 2025.