

The State of Computational Chess: Architectures, Interfaces, and Ecosystem Dynamics (2020–2025)

Executive Summary

The half-decade spanning 2020 to 2025 represents a pivotal epoch in the history of computer chess, characterized by the total obsolescence of classical handcrafted evaluation functions, the consolidation of commercial engine development under platform monopolies, and a radical democratization of high-level analysis tools through open-source innovation. This report provides a comprehensive survey of the current landscape, analyzing the technical specifications, market positioning, and user demographics of the dominant engines and graphical user interfaces (GUIs).

The primary finding of this analysis is the bifurcation of the engine market into two distinct hegemonies: the open-source supremacy of **Stockfish** and the proprietary, cloud-centric dominance of **Torch**. Stockfish, utilizing the Efficiently Updatable Neural Network (NNUE) architecture, effectively "solved" the problem of CPU-based evaluation, achieving ELO ratings exceeding 3600 and rendering traditional engines obsolete. Conversely, Torch, developed by Chess.com, signifies a shift toward "Software as a Service" (SaaS) in chess engines, where top-tier analysis is gated behind platform subscriptions rather than sold as standalone binaries. Simultaneously, the GUI sector has undergone a renaissance driven by the "Free and Open Source Software" (FOSS) community. While **ChessBase 17** retains its status as the industry standard for professional preparation—anchored by its massive proprietary database ecosystem and "Beauty" annotation algorithms—it faces unprecedented competition from **En Croissant** and **BanksiaGUI**. These modern interfaces, built on frameworks like Tauri and React, offer cross-platform compatibility and streamlined user experiences that challenge the legacy Windows-centric dominance of ChessBase. Furthermore, the restriction of binary execution in Android 14 has forced the mobile sector to adapt via the Open Exchange (OEX) format, reshaping how millions of mobile users interact with engines.

The emergence of Large Language Models (LLMs) and "Agentic" interfaces in 2025 marks the frontier of this domain. Projects like **ChessAgine** and **Maia Chess** act as semantic bridges, translating the raw numerical output of engines into human-intelligible coaching. While LLMs currently lack the tactical precision to compete with Stockfish, their integration via protocols like the Model Context Protocol (MCP) suggests a future where the value proposition of chess software shifts from "finding the best move" to "explaining the move."

This report concludes that while the "engine war" for ELO supremacy is largely stagnant—reduced to marginal gains in draw-heavy super-tournaments—the "interface war" for user attention and comprehension is just beginning.

Comparison Matrix: High-Performance Chess Engines & GUIs (2025)

Software Name	Type	License	Strength / Status	Typical User	Integration	Strengths & Key Features	Notable Repos/Projects
Stockfish 17.1	Engine	GPLv3	~3645 CCRL (Supremacy)	Beginner to World Champ	UCI	Unrivaled CPU strength via NNUE; Lazy SMP scaling; flawless endgame tablebase use.	official-stockfish/Stockfish
Torch	Engine	Proprietary	~3637 CCRL (Top 2)	Platform Subscriber	API (Cloud)	Exceptional speed in blitz/bullet; PyTorch-based; exclusive to Chess.com ecosystem.	N/A (Closed Source)
Leela Chess Zero (Lc0)	Engine	GPLv3	~3706 (Blitz, GPU dependent)	Analyst / Researcher	UCI	"Alien" intuition; GPU-accelerated (CUDA/OpenCL); distinct handling of long-term compensation.	LeelaChessZero/lc0
Komodo Dragon 3	Engine	Proprietary	~3626 CCRL (Retired)	Club / Correspondence	UCI	MCTS-based "human" style; "Optimism" settings; development team now inactive.	komodochess.com
Maia Chess	Engine	GPLv3	1100–1900 Elo (Human-aligned)	Learner / Coach	UCI	Predicts human errors rather than optimal moves;	CSSLab/macia-chess

Software Name	Type	License	Strength / Status	Typical User	Integration	Strengths & Key Features	Notable Repos/Projects
						trained on millions of amateur games.	
Nalwald	Engine	GPLv3	~3400+ (Superhuman)	Developer / Hobbyist	UCI	Written in Nim; utilizes unique "piece-relative" PSTs and 3x3 pawn structure tables.	tsoj/Nalwald
ChessBase 17	GUI	Commercial	N/A (Standard)	Grandmaster / Pro Coach	Proprietary	Massive DB management; cloud engine access; Repertoire reports; "Beauty" annotations.	N/A
En Croissant	GUI	GPLv3	N/A (Modern FOSS)	Analyst / Club Player	UCI / Lichess API	Modern UI (React/Tauri); direct download of online games; cross-platform (Win/Mac/Linux).	franciscoB Salgueiro/en-croissant
BanksiaGUI	GUI	Freeware	N/A (Tournament)	Engine Tester / Developer	UCI / WinBoard	Advanced tournament orchestration; remote engine play; massive customization.	nguyenpham/Banksia

Software Name	Type	License	Strength / Status	Typical User	Integration	Strengths & Key Features	Notable Repos/Projects
Nibbler	GUI	GPLv3	N/A (Visualization)	Lc0 Analyst	UCI	Real-time winrate graphing; visualization of neural net probabilities and "best" vs "human" moves.	rooklift/nibbler
ChessAgame	Tool	MIT	N/A (Agentic)	Developer / AI Researcher	MCP / API	Bridges LLMs (Claude/GPT) with Stockfish via Model Context Protocol for natural language analysis.	jalapp/chessagine-mcp

1. The Engine Hegemony: From Calculation to Neural Intuition

The trajectory of chess engines from 2020 to 2025 is defined by the absolute victory of neural network architectures over classical, handcrafted evaluation. This transition was not gradual; it was a precipitous paradigm shift initiated by the release of Stockfish 12 in late 2020, which integrated NNUE (Efficiently Updatable Neural Network). This technology allowed engines to evaluate positions with the nuanced "intuition" of deep learning without the prohibitive hardware costs associated with the AlphaZero approach.

1.1 Stockfish: The Apex of CPU Computing

License: GPLv3 (Open Source) **Repository:** official-stockfish/Stockfish **Integration:** Standard UCI (Universal Chess Interface) **Typical User:** Universal (Novice to Super-GM)
Stockfish remains the indisputable hegemon of the chess engine world. As of December 2025, Stockfish 17.1 holds the highest rating on the CCRL 40/15 list at 3645 Elo, maintaining a significant margin over its closest competitors.

1.1.1 The NNUE Architecture

The core of Stockfish's dominance lies in its adoption and refinement of NNUE. Originally ported from Shogi programming, this architecture embeds a shallow neural network directly into the engine's evaluation function. Unlike deep convolutional networks (like Leela Chess Zero) that require powerful GPUs, NNUE is optimized for modern CPU instruction sets (AVX2, AVX-512). This allows Stockfish to evaluate tens of millions of positions per second (NPS) while maintaining a high quality of evaluation.

- **Evolution (2020–2025):** Early NNUE iterations in Stockfish 12/13 used relatively simple network topologies. By Stockfish 16 and 17, the architecture evolved to support larger, dual-network setups ("Big" and "Small" nets) where the engine dynamically switches networks based on search depth and complexity. This hybrid approach ensures maximum speed in simple tactical sequences and maximum understanding in complex positional battles.
- **Lazy SMP:** Stockfish's scaling on multicore systems utilizes "Lazy Symmetric Multi-Processing." Instead of complex thread locking, threads largely search independently and share information via a transposition table. This allows the engine to scale efficiently on high-end consumer hardware (e.g., Threadripper CPUs) and massive cloud clusters.

1.1.2 Weaknesses and Criticism Despite its strength, Stockfish faces criticism for its "inhuman" perfection. In analysis, it often provides evaluation swings that are incomprehensible to humans (e.g., changing from 0.00 to +4.00 based on a 20-move tactical sequence that no human could see). Furthermore, its "sterile" playstyle in balanced positions—prioritizing absolute safety over practical complexity—can lead to draws in situations where a riskier engine might win against a human. However, forks and configuration options (like "Contempt") attempt to mitigate this.

1.2 Torch: The Corporate Challenger

License: Proprietary (Closed Source) **Integration:** Chess.com API (Cloud-based) **Typical**

User: Chess.com Premium Subscriber, Online Analyst

Torch represents a significant deviation from the open-source ethos that dominated the 2010s. Developed by a team hired by Chess.com (including developers from Ethereal and Koivisto), Torch appeared initially as a "Mystery Engine" in the Computer Chess Championship (CCC) before its official branding.

1.2.1 Architecture and Performance

Torch utilizes a PyTorch-based neural network backend, distinguishing it from the C++ NNUE implementation of Stockfish. It has established itself as the solid #2 engine in the world, rated approximately 3637 in CCRL 40/15. Its primary strength lies in shorter time controls (Bullet and Blitz), where its lighter neural networks allow it to out-calculate competitors. In CCC events, it has notably traded blows with Stockfish, securing wins in bullet championships.

1.2.2 The "Walled Garden" Controversy

Unlike Stockfish, Torch is not available as a downloadable binary. It exists solely as a backend service for Chess.com's analysis board. This has profound implications for the ecosystem: professional players cannot use Torch for offline preparation in air-gapped environments, and

researchers cannot independently verify its behavior without using the platform's API. Community efforts to "wrap" the API into a pseudo-UCI engine (e.g., torch-v2 scripts on GitHub) highlight the demand for local access, though these solutions are unofficial and often unstable.

1.3 Leela Chess Zero (Lc0): The Creative Visionary

License: GPLv3 (Open Source) **Repository:** LeelaChessZero/Lc0 **Integration:** UCI (Requires GPU Backend) **Typical User:** Correspondence Player, Opening Theoretician

Lc0 is the community-driven implementation of DeepMind's AlphaZero papers. It differs fundamentally from Stockfish and Torch in that it relies on Monte Carlo Tree Search (MCTS) guided by a deep policy and value network.

1.3.1 Hardware Dependency

Lc0's performance is strictly bound to GPU power. On a standard laptop, it is mediocre; on an NVIDIA RTX 4090, it is a monster. This reliance makes it less accessible to the average user but indispensable for high-end analysis. Its rating fluctuates heavily based on the hardware used in testing pools (TCEC vs CCRL), often leading to debates about its "true" strength.

1.3.2 Strategic Profile

Lc0 is celebrated for its understanding of "long-term compensation." It is famous for playing distinct opening lines that Stockfish initially dislikes but eventually comes to accept as sound. This "alien" intuition makes it the engine of choice for finding novelties in the Ruy Lopez or Sicilian, where positional factors outweigh immediate tactical calculation.

1.4 Komodo Dragon 3: The Sunset of a Titan

License: Proprietary (Commercial) **Strength:** ~3626 CCRL **Typical User:** Club Player, Human-vs-Engine sparring

Once the primary rival to Stockfish, Komodo transitioned to the "Dragon" architecture (NNUE + MCTS) in 2020. This hybrid approach aimed to combine tactical precision with the "human-like" probability assessment of MCTS.

1.4.1 The "Human" Playing Style

Dragon is renowned for its "MCTS Optimism" parameter. Unlike Alpha-Beta search, which assumes the opponent will always find the best defense, MCTS evaluates the *probability* of the opponent finding the best move. This allows Dragon to play "swindles"—moves that are objectively equal but practically difficult for a human to defend. This makes it an excellent training partner.

1.4.2 End of Development

Following the acquisition of the Komodo team by Chess.com, the lead developers (Mark Lefler and GM Larry Kaufman) announced their semi-retirement. As of 2025, Dragon 3.3 is likely the

final major release, with the team's expertise largely redirected toward Torch or internal projects.

1.5 The Open Source Vanguard (Tier 2 Engines)

Beyond the "Big Four," a vibrant ecosystem of open-source engines exists, pushing the boundaries of programming languages and novel heuristics.

- **Nalwald:** An engine written in the **Nim** programming language, notable for its use of piece-relative Piece-Square Tables (PSTs) and 3x3 pawn structure tables. It demonstrates that competitive engines (~3400+ Elo) need not be written in C++ or Rust.
- **Obsidian & PlentyChess:** Both have breached the 3600 Elo barrier, utilizing their own NNUE implementations. They serve as critical benchmarks in rating lists, ensuring that Stockfish's dominance is constantly stress-tested against varied coding approaches.
- **Berserk:** Known for its aggressive pruning and high tactical volatility, making it a fan favorite in engine tournaments where draws are common.

2. The Analyst's Workbench: Graphical User Interfaces (GUIs)

While engines provide the intelligence, GUIs provide the workflow. The 2020–2025 period has seen a dramatic improvement in the quality of free interfaces, challenging the long-standing dominance of commercial software.

2.1 ChessBase 17: The Industry Standard

License: Commercial (Expensive) **Integration:** Proprietary / Cloud **Typical User:** Professional Grandmaster, Serious Coach

ChessBase remains the unparalleled tool for data management. Its proprietary .cbh and newer .2cbh formats allow for the efficient storage and searching of databases containing 10+ million games.

2.1.1 Strengths

- **Repertoire Management:** The "Opening Report" feature aggregates statistics from millions of games to show the most critical lines, scoring percentages, and historical trends for any given move.
- **Search Booster:** Its ability to search for specific positional patterns (e.g., "White knight on f5, Black king on g8, no queens") across millions of games in seconds is unrivaled by FOSS alternatives.
- **Cloud Integration:** The "Engine Cloud" allows users to rent private, high-performance hardware for analysis, directly integrated into the GUI.

2.1.2 Weaknesses

- **UX/UI:** The interface remains rooted in Windows design paradigms from the early 2010s. It is "busy," often unintuitive for new users, and strictly Windows-only.

- **Stability:** Users frequently report crashes and database corruption issues with the new 2CBH format.

2.2 En Croissant: The Modern FOSS Challenger

License: GPLv3 **Repository:** franciscoBSalgueiro/en-croissant **Typical User:** Analyst, Developer, Cross-Platform User

En Croissant represents the most significant disruption in the GUI market in years. Built on a modern tech stack (Tauri, React, Rust), it offers a user experience comparable to sleek web apps like Lichess but with the power of a desktop application.

2.2.1 Integration and Features

- **Seamless Import:** It connects directly to Lichess and Chess.com APIs, allowing users to auto-download their games into a local database with a single click.
- **Engine Management:** It simplifies the complex process of installing UCI engines, managing binaries and weights automatically for users who find command-line interfaces daunting.
- **Cross-Platform:** Unlike ChessBase, En Croissant runs natively on macOS (Apple Silicon optimized), Linux, and Windows, making it the de-facto standard for non-Windows chess players.

2.3 BanksiaGUI: The Tournament Orchestrator

License: Freeware (Closed GUI / Open Source Core) **Website:** banksiogui.com **Typical User:** Engine Developer, Rating List Maintainer

BanksiaGUI fills the niche of "scientific testing." It is designed not just for playing chess, but for running experiments.

2.3.1 Key Capabilities

- **Concurrency:** Banksia can run dozens of engine matches simultaneously on a multicore processor, allowing developers to quickly test if a new version of an engine is statistically stronger than the previous one.
- **Remote Play:** It supports connecting to engines over a local network, allowing a weak laptop to run the GUI while a powerful server in the other room performs the calculations.
- **Customization:** It offers extreme granular control over time controls, opening books (PolyGlot), and adjudication rules, making it the preferred tool for organizations like CCRL.

2.4 Nibbler: The Neural Visualizer

License: GPLv3 **Repository:** rooklift/nibbler **Typical User:** Leela Chess Zero Enthusiast

Nibbler is a specialized GUI designed specifically for Leela Chess Zero. While standard GUIs display a single evaluation number (e.g., "+0.55"), Nibbler visualizes the internal "mind" of the neural network.

2.4.1 Visualization Features

- **Winrate Graphs:** It displays the live win/draw/loss probability distribution for every move, not just the best line.
- **Saliency:** In conjunction with specific Lc0 backends, it can highlight which squares on the board the network is "paying attention to," offering a glimpse into the AI's intuition.

2.5 Scid vs PC: The Reliable Legacy

License: GPLv2 **Typical User:** Linux Power User, Database Archivist

A fork of the original SCID, this GUI remains relevant due to its sheer speed and reliability. While its Tk/Tk interface is archaic compared to En Croissant, it handles massive databases (5M+ games) with lower RAM overhead than any Electron/web-based app. It remains the tool of choice for users on older hardware or those who prioritize function strictly over form.

3. The Mobile Ecosystem: Android Restrictions and Adaptation

The years 2024 and 2025 introduced a significant technical hurdle for mobile chess analysis: the security hardening of the Android operating system.

3.1 The Android 14 Binary Ban

Historically, Android chess apps (like *Chess for Android* or *DroidFish*) allowed users to download a compiled engine binary (e.g., stockfish-android-armv8) and import it from the file system. Android 14 introduced strict "W^X" (Write XOR Execute) policies, preventing apps from executing code that was downloaded to storage by the user. This effectively broke the "Import Engine" feature for millions of users.

3.2 The Open Exchange (OEX) Standard

The industry solution has been a migration to the **OEX** format. In this model, the chess engine is packaged as a standalone Android application (APK) that exposes a service to other apps.

- **Implementation:** Apps like **ChessIs** and **Analyze Your Chess Pro** now rely on the user installing a "Stockfish OEX" app. The GUI app then binds to the OEX service to communicate via UCI.
- **Impact:** This has fragmented the ecosystem slightly, as users must now manage multiple apps to get one working analysis board, but it has standardized engine distribution on mobile.

3.3 Leading Mobile Analysis Tools

- **ChessIs:** A Freemium app that focuses on "Game Reports." It mimics the Chess.com "Game Review" feature, generating accuracy scores and classifying moves (Blunder,

Mistake, Great Move) locally on the device using OEX engines. It is popular for its visual feedback and low latency.

- **Analyze Your Chess Pro:** A robust PGN viewer/editor that focuses on compatibility. It supports the latest OEX engines and allows for deep manipulation of PGN annotations, catering to users who study on tablets.

4. The Agentic Frontier: LLMs and Human Alignment

The most cutting-edge development in 2025 is the intersection of chess engines and Generative AI. While Stockfish provides the "truth," it often fails to provide the "why." New tools are emerging to bridge this semantic gap.

4.1 ChessAgine and the Model Context Protocol (MCP)

Repository: jalpp/chessagine-mcp **Function:** Semantic Analysis Bridge

ChessAgine represents the first generation of "Agentic" chess tools. It does not try to replace the engine; instead, it wraps the engine in a semantic layer using the Model Context Protocol.

4.1.1 Mechanism of Action

When a user interacts with ChessAgine (via an LLM interface like Claude or GPT-4), the system operates as follows:

1. **User Query:** "Why is moving my knight to f3 a mistake?"
2. **MCP Call:** ChessAgine translates this natural language into a UCI command (position fen... moves..., go depth 20).
3. **Engine Analysis:** Stockfish returns the evaluation and principal variation (PV).
4. **Semantic Translation:** ChessAgine parses the PV and feeds it back to the LLM, which then generates a human-readable explanation: "Moving the knight blocks your bishop's control of the diagonal and allows Black to push the e-pawn...".

This architecture solves the "hallucination" problem of LLMs (where they invent illegal moves) by grounding their output in verifiable engine data.

4.2 Maia Chess: Predicting Human Error

Repository: CSSLab/maia-chess **License:** GPLv3

While Stockfish strives for divine perfection, **Maia** strives for human imperfection. Trained on millions of games from Lichess, Maia learns to predict the move a human *would* play at a specific rating level (e.g., 1100, 1500, 1900).

4.2.1 Application in Training

- **Maia 2 (2025):** The latest iteration uses a unified model to predict moves across the rating spectrum. It is used in "Bot or Not" tests and for identifying "natural" blunders.
- **Use Case:** If a student makes a blunder that Maia predicts with high confidence, it

indicates a gap in fundamental understanding common to that skill level. If the blunder is something Maia *didn't* predict, it might be a simple calculation error or a mouse slip. This distinction allows for targeted coaching.

4.3 Benchmarking LLMs in Chess

The **LLM CHESS** benchmark has been established to measure the reasoning capabilities of general-purpose models. Findings in 2025 indicate:

- **Reasoning Gap:** Models like OpenAI's o3 and o4-mini show drastically different performance. o3 demonstrates strategic planning capabilities roughly equivalent to a 1200 Elo player, while standard models struggle to beat random movers.
- **Implication:** Chess has become a "canary in the coal mine" for testing the multi-step reasoning and instruction-following capabilities of AGI systems.

5. Conclusion and Future Outlook

The landscape of 2025 is defined by a shift from **capability to usability**. The raw strength of chess engines has plateaued at a level so far beyond human comprehension that further gains in ELO are mathematically interesting but practically irrelevant for 99% of users.

Strategic Implications:

1. **The Commoditization of Strength:** With open-source engines like Stockfish and PlentyChess freely available, the commercial value of selling a "strong engine" has evaporated. The only viable commercial models remaining are **Platform** ([Chess.com/Torch](#)) and **Data** (ChessBase).
2. **The Rise of the "Wrapper":** The most vibrant development is happening in the layer between the user and the engine. Tools like **En Croissant** (UX wrapper) and **ChessAgine** (Semantic wrapper) are where innovation is concentrated. They do not make the engine smarter; they make the user smarter.
3. **The Mobile-Desktop Convergence:** With the power of mobile hardware and OEX integration, the gap between desktop analysis and mobile analysis is closing. However, the OS-level restrictions on Android signal a future where "power user" freedom is increasingly constrained by mobile OS security models, potentially driving serious analysts back to Linux/Desktop environments.

For the typical user in 2025:

- **The Beginner** should use **Maia** and **ChessIs** to understand their mistakes in human terms.
- **The Club Player** is best served by **En Croissant** running **Stockfish 17** for a free, professional-grade analysis suite.
- **The Professional** remains tethered to **ChessBase 17** for its deep data mining, but likely supplements this with **Leela/Nibbler** for creative inspiration.

The "Engine War" is over; Stockfish won. The "Interface War" is just beginning.

Works cited

1. Complete rating list - CCRL, https://computerchess.org.uk/ccrl/4040/rating_list_all.html
2. Stockfish (chess) - Wikipedia, [https://en.wikipedia.org/wiki/Stockfish_\(chess\)](https://en.wikipedia.org/wiki/Stockfish_(chess))
3. Strongest Chess

Engines (2025) - Reddit,
https://www.reddit.com/r/chess/comments/1naldoq/strongest_chess_engines_2025/ 4. Torch - Chessprogramming wiki, <https://www.chessprogramming.org/Torch> 5. Announcing Torch: New #2 Chess Engine, <https://www.chess.com/news/view/torch-chess-engine> 6. Torch Engine Now Available On Chess.com's Analysis Page, <https://www.chess.com/news/view/torch-engine-available-on-chesscom> 7. Run Torch V2 locally : r/chess - Reddit, https://www.reddit.com/r/chess/comments/1ivbn58/run_torch_v2_locally/ 8. Dragon Chess Engine - Official Site - Komodo chess engine, <https://komodochess.com/dragon-3.htm> 9. BREAKING news : chess.com buys Komodo/Dragon, <https://www.chess.com/blog/thechesscorner64/significant-news-chesscom-buys-komodo-dragon> 10. tsoj/Nalwald - Chess engine written in Nim - GitHub, <https://github.com/tsoj/Nalwald> 11. ChessBase 17 - Review, <https://www.chessengenia.eu/post/chessbase-17-review> 12. franciscoBSalgueiro/en-croissant: The Ultimate Chess Toolkit - GitHub, <https://github.com/franciscoBSalgueiro/en-croissant> 13. En Croissant - Chess Forums, <https://www.chess.com/forum/view/chess-equipment/en-croissant> 14. BanksiaGUI – A Free Chess Graphical User Interface, <https://banksiogui.com/> 15. Banksia GUI released - Page 115 - TalkChess.com, <https://talkchess.com/viewtopic.php?t=72350&start=1140> 16. Jac-Zac/Nibbler_MacOS: Version of Nibbler for MacOS - GitHub, https://github.com/Jac-Zac/Nibbler_MacOS 17. Scid vs. PC Reviews - 2025 - SourceForge, <https://sourceforge.net/projects/scidvspc/reviews/> 18. UCI and XBoard Protocols for Android - Aart Bik's, https://www.aartbik.com/android_eng.php 19. What the heck happened to chess on Android?, <https://www.chess.com/forum/view/chess-equipment/what-the-heck-happened-to-chess-on-android> 20. Chessis: Chess Analysis - Apps on Google Play, <https://play.google.com/store/apps/details?id=com.chessimprovement.chesis> 21. Analyze your Chess Pro - Apps on Google Play, <https://play.google.com/store/apps/details?id=com.lucian.musca.chess.analyzeyourchess.pro> 22. ChessAgine: Bridging AI and Grandmaster Strategy with MCP - Skywork.ai, <https://skywork.ai/skypage/en/chess-ai-grandmaster-strategy/1981573338766610432> 23. jalpp/chessagineweb: A modern FOSS chess interface that combines LLMs and chess engines into one unified platform. Built for the chess community - GitHub, <https://github.com/jalpp/chessagineweb> 24. Analysis - Maia Chess, <https://www.maiachess.com/analysis> 25. Maia Chess, <https://www.maiachess.com/> 26. LLM CHESS: Benchmarking Reasoning and Instruction-Following in LLMs through Chess, <https://openreview.net/forum?id=65R1Dbfwzk> 27. The 2025 AI Chess Tournament: A Battle of Minds - BCC Research Blog, <https://blog.bccresearch.com/the-2025-ai-chess-tournament-a-battle-of-minds>