IN THE UNITED STATES DISTRICT COURT
FOR THE DISTRICT OF DELAWARE

UNIVERSITY OF TENNESSEE RESEARCH FOUNDATION, ***Plaintiff,*** v. **ANTHROPIC PBC,** ***Defendant.***	**Civil Action No. _________** **JURY TRIAL DEMANDED**

COMPLAINT FOR PATENT INFRINGEMENT

University of Tennessee Research Foundation ("UTRF" or "Plaintiff") brings this action and makes the following allegations of patent infringement relating to U.S. Patent No(s). 10,019,470 (the "'470 Patent") and 10,095,718 (the "'718 Patent") (collectively, the "Patents-in-Suit"). Defendant Anthropic PBC ("Anthropic" or "Defendant") infringes the Patents-in-Suit in violation of the patent laws of the United States of America, 35 U.S.C. §§ 1 *et seq*.

INTRODUCTION

1. This is an action for infringement of patents developed through groundbreaking research conducted at The University of Tennessee. The Patents-in-Suit are owned by UTRF, a non-profit organization responsible for protecting, licensing, and commercializing intellectual property developed by The University of Tennessee's faculty and researchers.

2. Established in 1794, The University of Tennessee is Tennessee's flagship public research institution. Through its faculty, researchers, and students, The University of Tennessee has developed numerous innovations across a wide range of scientific and technological disciplines.

3. To maximize the public benefit generated by university research, inventions developed by University of Tennessee researchers are often assigned to UTRF, which is responsible for protecting, managing, licensing, and enforcing those intellectual property rights.

4. This case arises from Defendant's unauthorized use of technologies claimed in the Patents-in-Suit. As described below, Defendant has made, used, sold, offered for sale, imported, and/or provided products and services that practice the inventions claimed in the Patents-in-Suit without authorization from UTRF.

PLAINTIFF

5. Plaintiff University of Tennessee Research Foundation is a Tennessee non-profit organization existing under Section 501(c)(3) of the Internal Revenue Code.

6. Originally established in 1935 as the University of Tennessee Research Corporation, UTRF is among the oldest university research foundations in the United States. UTRF's mission is to promote, support, and carry out the research mission of The University of Tennessee. UTRF facilitates the commercialization of university-developed technologies and encourages innovation, entrepreneurship, and economic development through technology transfer and licensing.

7. UTRF owns by assignment all right, title, and interest in the Patents-in-Suit, including the right to recover damages for past infringement.

DEFENDANT

8. Defendant Anthropic PBC ("Anthropic") is a Delaware corporation with its principal place of business at 500 Howard Street, San Francisco, California 94105. Anthropic can be served via its registered agent for service of process, Corporation Service Company, 251 Little Falls Drive, Wilmington, Delaware 19808.

9. Anthropic is organized as a Public Benefit Corporation; however, it is a for-profit corporation that, in its short history, has been no friend to the developers and owners of intellectual property rights. In *Bartz, et al. v. Anthropic PBC*, No. 24-cv-05417-WHA (N.D. Cal. June 23, 2025), Northern District of California Judge William Alsup explained Anthropic's rampant copyright infringement in connection with the development of its large language model products. In connection with the training of these products, Judge Alsup explained, "[b]efore buying books for its central library, Anthropic downloaded over seven million pirated copies of books, paid nothing, and kept these pirated copies in its library even after deciding it would not use them to train its AI (at all or ever again). Authors argue Anthropic should have paid for these pirated library copies. This order agrees." (D.I. 231 at 18.) Anthropic has agreed to pay more than $1.5 billion to resolve the copyright claims at issue in that case. (D.I. 362 (Unopposed Motion for Preliminary Approval of Settlement) at 20-21.)

10. As alleged herein, Anthropic's cavalier approach to others' intellectual property rights in the development of its products extends beyond the use of copyrighted material. Anthropic infringed UTRF's patent rights, including infringing each of the Patents-in-Suit without the permission of or compensation to UTRF.

JURISDICTION AND VENUE

11. This action arises under the patent laws of the United States, Title 35 of the United States Code. Accordingly, this Court has exclusive subject matter jurisdiction over this action under 28 U.S.C. §§ 1331 and 1338(a).

12. This Court has personal jurisdiction over Anthropic in this action because Anthropic has committed acts within the State of Delaware giving rise to this action and has established minimum contacts with this forum such that the exercise of jurisdiction over Defendant would not offend traditional notions of fair play and substantial justice. Anthropic, directly and/or

through subsidiaries or intermediaries, has committed and continues to commit acts of infringement in this District by, among other things, offering to sell and selling products and/or services that infringe the Patents-in-Suit. Moreover, Anthropic actively directs its activities to customers located in the State of Delaware.

13. The Court further has personal jurisdiction over Anthropic because it is organized and existing under the laws of the State of Delaware and maintains a registered agent in Delaware.

14. Venue is proper in this District under 28 U.S.C. §§ 1391(b)-(d) and 1400(b). Defendant is organized and existing under the laws of the State of Delaware.

THE UNIVERSITY OF TENNESSEE'S PIONEERING RESEARCH IN ARTIFICIAL INTELLIGENCE AND NEUROSCIENCE-INSPIRED COMPUTING

15. Artificial intelligence ("AI") systems have become increasingly important across a wide range of industries and applications. Modern AI systems are capable of performing tasks involving classification, prediction, pattern recognition, anomaly detection, decision making, language processing, image analysis, and other forms of machine intelligence.

16. Many AI systems utilize artificial neural networks. Artificial neural networks are computational systems that process information using interconnected computational elements commonly referred to as neurons and synapses. These systems are often inspired by the structure and operation of biological neural networks found in the human brain.

17. As artificial neural networks became more sophisticated, researchers encountered significant technical challenges relating to network construction, architecture design, learning, adaptation, optimization, monitoring, interpretability, and scalability. Conventional neural network systems often relied upon fixed architectures, manually designed network structures, and limited mechanisms for understanding, monitoring, or visualizing how neural networks arrived at particular outputs or decisions.

18. Researchers also faced challenges in designing artificial neural networks capable of adapting to changing conditions, efficiently identifying useful neural substructures, reusing learned functionality, and improving performance without excessive computational cost or manual intervention.

19. Long before the recent surge in commercial artificial intelligence products and services, researchers at The University of Tennessee were conducting pioneering research into neuroscience-inspired computing systems and advanced artificial neural network architectures. This research explored ways to construct, optimize, visualize, monitor, and improve artificial neural networks by drawing inspiration from the structure, behavior, and adaptive characteristics of biological neural systems.

20. The inventions claimed in the Patents-in-Suit were developed by researchers affiliated with The University of Tennessee, including Dr. J. Douglas Birdwell, a longtime faculty member and researcher whose work focused on artificial intelligence, machine learning, neural networks, and advanced computing systems.

21. Dr. Birdwell is an Emeritus Professor who holds a Ph.D. from the Massachusetts Institute of Technology and spent more than four decades at The University of Tennessee. He is a Fellow of both the IEEE and the National Academy of Inventors. He has authored over 100 publications, holds 36 U.S. patents, and has directed over $10 million in sponsored research. Dr. Birdwell has worked extensively on control systems, artificial intelligence, secure information technologies, and neuromorphic computing. He has served in senior leadership roles within the IEEE Control Systems Society, including serving as its President in 2004. His extensive experience in computer hardware and software applications, signal processing, and inductive

machine learning formed the foundation for the neuroscience-inspired artificial neural network technologies claimed in the Patents-in-Suit.

22. The research benefited from the contributions of Dr. Mark Dean, a pioneering computer engineer, inventor, and technology leader whose achievements shaped modern computing. Dr. Dean holds a Ph.D. from Stanford University and spent nearly 40 years at International Business Machines (IBM), where he rose to become one of IBM's most lauded technologists. In 1995, he became the first African American named an IBM Fellow, the company's highest technical honor. He later served as IBM's Vice President of Systems and Chief Technology Officer for the Middle East and Africa. Dr. Dean directed IBM's Almaden Research Center while overseeing technical strategy for IBM's global operations.

23. During his career, Dr. Dean received numerous honors and made groundbreaking contributions to the field of computing. Dr. Dean is a member of the National Academy of Engineering and a Fellow of the Institute of Electrical and Electronics Engineers (IEEE). Dr. Dean co-invented the Industry Standard Architecture (ISA) bus, a fundamental technology that allowed peripheral devices such as disk drives, printers, and monitors to connect to the motherboard of the original IBM personal computer. Dr. Dean's co-invention of the ISA bus expanded the capabilities of personal computing and led to his induction into the National Inventors Hall of Fame. Dr. Dean's contributions outside of the personal computer included leading the team that developed the Blue Gene supercomputer and overseeing the creation of the first gigahertz microprocessor (a chip capable of performing one billion calculations per second). After retiring from IBM, Dr. Dean joined The University of Tennessee as a Distinguished Professor, where he continues his research into advanced computer architecture, data-centric computing, and computational sciences.

24. Through their research, Drs. Dean, Birdwell, and their co-inventors developed innovative solutions to technical challenges associated with artificial neural networks and machine-learning systems. The inventions disclosed in the Patents-in-Suit are directed to improvements in artificial intelligence systems, neuroscience-inspired computing systems, artificial neural networks, dynamic neural architectures, neural network optimization, neural pathway visualization, adaptive learning, and related technologies.

25. The inventions disclosed in the Patents-in-Suit describe, among other things, techniques for constructing neuroscience-inspired artificial neural networks, dynamically adapting neural network architectures, identifying and reusing useful neural substructures, visualizing neural pathways and network activity, monitoring neural network operation, implementing biologically inspired learning mechanisms, and improving the performance and operation of artificial intelligence systems.

26. The patented technologies addressed technical problems associated with the construction, operation, training, monitoring, explainability, and improvement of artificial neural networks years before artificial intelligence systems became widely deployed in commercial products and services.

27. The patented inventions represent significant contributions to the fields of artificial intelligence, machine learning, neuromorphic computing, and neuroscience-inspired computing and reflect years of research conducted at The University of Tennessee concerning biologically inspired approaches to machine intelligence.

<u>THE ASSERTED PATENTS</u>

<u>U.S. PATENT NO. 10,019,470</u>

28. U.S. Patent No. 10,019,470 (the "'470 Patent") entitled, *Method and Apparatus for Constructing, Using and Reusing Components and Structures of an Artificial Neural Network*, was

filed on October 14, 2014. The ‘470 Patent claims priority to U.S. Provisional Patent Application Nos. 61/891,621 (filed on October 16, 2013); 61/934,052 (filed on January 31, 2014); 61/946,179 (filed on February 28, 2014); 61/951,690 (filed on March 12, 2014); 62/001,951 (filed on May 22, 2014); and 62/024,081 (filed on July 14, 2014). The ‘470 Patent is subject to a 35 U.S.C. § 154(b) term extension of 194 days. A true and correct copy of the ‘470 Patent is attached hereto as Exhibit 1.

29. The ‘470 Patent has been in full force and effect since its issuance. UTRF owns by assignment the entire right, title, and interest in and to the ‘470 Patent.

30. The ‘470 Patent family has been cited by several hundred United States and international patents and patent applications as relevant prior art. The following companies and research institutions have cited the ‘470 Patent family as relevant prior art:

- Apple Inc.
- Google LLC
- Microsoft Technology Licensing, LLC
- Amazon Technologies, Inc.
- Samsung Electronics Co., Ltd.
- International Business Machines Corporation
- Facebook, Inc.
- Sony Group Corporation
- Intel Corporation
- Princeton University
- California Institute of Technology
- Tsinghua University
- University of California
- University of Toronto
- Purdue University
- University of Illinois
- University of Texas
- LG Electronics
- General Electric Company
- Siemens Aktiengesellschaft
- AT&T Intellectual Property I, L.P.
- Cisco Technology, Inc.
- Qualcomm Incorporated

- Micron Technology, Inc.
- Hewlett Packard Enterprise Development LP
- SoftBank Corp.
- Motorola Solutions, Inc.

U.S. PATENT NO. 10,095,718

31. U.S. Patent No. 10,095,718 (the "'718 Patent") entitled, *Method and Apparatus for Constructing a Dynamic Adaptive Neural Network Array (DANNA)*, was filed on October 14, 2014. The '718 Patent claims priority to U.S. Provisional Patent Application Nos. 61/891,621 (filed on October 16, 2013); 61/934,052 (filed on January 31, 2014); 61/946,179 (filed on February 28, 2014); 61/951,690 (filed on March 12, 2014); 62/001,951 (filed on May 22, 2014); and 62/024,081 (filed on July 14, 2014). The '718 Patent is subject to a 35 U.S.C. § 154(b) term extension of 581 days. A true and correct copy of the '718 Patent is attached hereto as Exhibit 2.

32. The '718 Patent has been in full force and effect since its issuance. UTRF owns by assignment the entire right, title, and interest in and to the '718 Patent.

33. The '718 Patent family has been cited by several hundred United States and international patents and patent applications as relevant prior art. The following companies and research institutions have cited the '718 Patent family as relevant prior art:

- Apple Inc.
- Google LLC
- Microsoft Technology Licensing, LLC
- Amazon Technologies, Inc.
- Samsung Electronics Co., Ltd.
- International Business Machines Corporation
- Facebook, Inc.
- Sony Group Corporation
- Intel Corporation
- Princeton University
- California Institute of Technology
- Tsinghua University
- University of California
- University of Toronto

- Purdue University
- University of Illinois
- University of Texas
- LG Electronics
- General Electric Company
- Siemens Aktiengesellschaft
- AT&T Intellectual Property I, L.P.
- Cisco Technology, Inc.
- Qualcomm Incorporated
- Micron Technology, Inc.
- Hewlett Packard Enterprise Development LP
- SoftBank Corp.
- Motorola Solutions, Inc.

COUNT I
INFRINGEMENT OF U.S. PATENT NO. 10,019,470

34. Plaintiff references and incorporates by reference the preceding paragraphs of this Complaint as if fully set forth herein.

35. Anthropic designs, makes, uses, sells, and/or offers for sale in the United States products and/or services for constructing a central pattern generator of a neuromorphic network for use in a process control application of an external automated process.

36. Anthropic designs, makes, sells, offers to sell, imports, and/or uses Anthropic's Claude Code and its underlying agentic software architecture (collectively, the "Anthropic '470 Product(s)").

37. One or more Anthropic subsidiaries and/or affiliates use the Anthropic '470 Product(s) in regular business operations.

38. The Anthropic '470 Products perform a method comprising a "central pattern generator" (CPG) of the "neuromorphic network" that is implemented as the background execution scheduling system (cronScheduler.ts) and the background memory consolidation engine (autoDream.ts), which automatically manages, compresses, and restructures memory states in a

recurring, cyclic process to control local terminal environments or remote container sandboxes (acting as the external automated system or external device).

39. The Anthropic '470 Products control background memory consolidation processes using, in part, the config.ts file within the autoDream functionality of the Anthropic '470 Products. The Anthropic '470 Products integrate the main execution loop to run the consolidation agent using autoDream.ts.

40. The Anthropic '470 Products perform memory consolidation asynchronously on a recurring basis to optimize memory states. "Dreaming is a process that looks for patterns and mistakes across your recent agent sessions and their transcripts and automatically produces organized and up-to-date memory content. … It's a batch asynchronous process that runs separately from the work that you're doing … on kind of a cron basis." Mahesh Murag, *Memory and dreaming for self-learning agents,* YOUTUBE.COM at 11:29-11:38, 12:14-12:30 (May 8, 2026), available at: https://www.youtube.com/watch?v=RtywqDFBYnQ.

41. The Anthropic '470 Products perform the step of controlling the external automated system or external device upon prediction of a component failure. Specifically, the Anthropic '470 Products monitor session and workspace health to predict and prevent system failures (*e.g.*, context window overflow, runtime crashes, memory saturation, or syntactic/security command errors). To prevent syntax or security exceptions from causing an application crash (*i.e.*, a predicted "component failure"), the Anthropic '470 Products run pre-execution Abstract Syntax Tree (AST) parsing on shell inputs to catch syntax errors or security bypasses before they execute on the external automated system (the shell actuator). Additionally, the memory consolidation engines in the Anthropic '470 Products (autoDream.ts) execute when session logs reach a defined volume threshold, predicting and preventing context window saturation and memory failures.

42. The Anthropic '470 Products perform static analysis on bash commands using a fail-closed parser to catch dangerous expansions or syntax anomalies before they reach the shell actuator, preventing system failure. Specifically, the Anthropic '470 Products contain ast.ts code that performs this process.

43. The Anthropic '470 Products contain a memory consolidation system that is triggered when session counts exceed a defined threshold to prevent context window saturation (predicted component failure due to memory overflow).

44. The Anthropic '470 Products evaluate tool parameters and catch execution errors inside checkPermissionsAndCallTool, formatting error strings back to the model as a tool_result loop to auto-correct, ensuring continuous process control.

45. The Anthropic '470 Products use an Abstract Syntax Tree (AST) validation to parse and control bash tools before execution. *See* Thariq Shihipar, *Claude Agent SDK [Full Workshop]*, YOUTUBE.COM at 14:03-14:11 (Jan. 5, 2026), available at: https://www.youtube.com/watch?v=TqC1qOfiVcQ ("AST pass parser on the bash tool, for example, so we know um, fairly reliably like what the bash tool is actually doing").

46. The Anthropic '470 Products use a multi-stage permission classifier (permissions.ts). The permission classifier in the Anthropic '470 Products uses a fast single-token filter to predict a "safety blast radius" and prevent mistakes and/or malicious commands from causing component failure. "[O]n entering auto mode, we drop permission rules that are known to grant arbitrary code execution, including blanket shell access, wildcarded script interpreters (python, node, ruby, and similar), and package manager run commands… Stage 1 is a fast single-token (yes/no) filter instructed to err toward blocking." Anthropic, *How we built Claude Code*

auto mode: a safer way to skip permissions, ANTHROPIC WEBSITE at 3 (Mar. 15, 2026), available at: https://www.anthropic.com/news/claude-code-auto-mode.

47. The Anthropic '470 Products comprise an addressably configurable circuit element for providing inputs or outputs to components of the external automated system or external device comprising one of a display, a camera, a radio, and a scanner device external to the neuromorphic network, the neuromorphic network receiving input data from and transmitting output data to the external automated system or external device. Specifically, the Anthropic '470 Products contain addressable Model Context Protocol (MCP) servers, tool wrappers, and host adapters (hostAdapter.ts). These modules provide addressable inputs and outputs to external hardware and OS-level devices. For example, the computerUse host adapter (hostAdapter.ts) configures OS-level permissions to control display and input devices (*e.g.*, Accessibility and Screen Recording APIs for display outputs and inputs). At the same time, the MCP server (mcpServer.ts) uses system tools, such as Spotlight, to scan the host environment for installed applications (serving as a scanning device).

48. The Anthropic '470 Products contain source code (hostAdapter.ts) that configures OS-level permissions to interact with display and input devices (*i.e.*, providing inputs or outputs to external components). In addition, the Anthropic '470 Products contain source code (mcpServer.ts) that uses a spotlight process scanner to scan the host for installed applications, feeding these details as input to the tool descriptor.

49. The Anthropic '470 Products utilize different task autonomy levels with modular permission configurations.

> *Safety vs. autonomy.* The permission modes (five always present, plus `auto` when the classifier feature flag is active, and the internal `bubble` mode) create a gradient from `plan` (user approves all plans) through `default`, `acceptEdits`, `auto` (ML classifier), `dontAsk`, to `bypassPermissions` (skips most prompts but safety-critical checks remain). The progression represents a monotonically decreasing safety gradient with increasing autonomy. Not restoring permissions on resume reflects a deliberate choice to err toward safety: security state does not persist implicitly across session boundaries.

Jiacheng Liu et al., *Dive into Claude Code: The Design Space of Today's and Future AI Agent Systems,* ARXIV at 29 (2026) (emphasis added).

50. The Anthropic '470 Products utilize a "dangerously-skip-permissions" flag that disables all permission prompts and enables the Anthropic '470 Products to act "freely."

> `--dangerously-skip-permissions` flag that disables all permission prompts and lets Claude act freely, which is unsafe in most situations. Figure 1 lays out the tradeoff space. Sandboxing is safe but high-maintenance: each new capability needs configuring, and anything requiring network or host access breaks isolation. Bypassing permissions is zero-maintenance but offers no protection. Manual prompts sit in the middle, and in practice users accept 93% of them anyway.

Anthropic, *How we built Claude Code auto mode: a safer way to skip permissions,* ANTHROPIC ENGINEERING WEBSITE (Mar. 15, 2026), available at: https://www.anthropic.com/news/claude-code-auto-mode.

51. The Anthropic '470 Products perform the step of addressably configuring at least one addressably configurable input neuron circuit element and at least one addressably configurable output neuron circuit, the addressably configurable input and output circuit elements located at an edge of the two-dimensional addressably configured array of neuron and synapse circuit elements. Specifically, the Anthropic '470 Products map input and output boundary nodes (neurons) to layout node adapters located at the edges of the two-dimensional flexbox rendering grid. The layout system in the Anthropic '470 Products explicitly maps coordinates to the physical edges (left, right, top, bottom) of the visual display matrix, serving as the input and output neuron circuit elements along the array's edges. The node.ts code within the Anthropic '470 Products

defines the edge coordinates of structural nodes (acting as neurons located at the edges) in the layout array.

52. The Anthropic '470 Products contain source code, including the index.ts file, which enables the Anthropic '470 Products to resolve style edges into absolute physical edge coordinates at the boundary of each bounding box array.

53. The Anthropic '470 Products perform one or more layout reconciliation loops inside createRenderer, computing layout coordinate mappings dynamically on the terminal grid using code including renderer.ts.

54. The Anthropic '470 Products neural network weights represent addressably configured synthetic neurons organized into circuits. The Anthropic '470 Products further contain feedforward loops and refusal circuits.

> **Convergent paths and shortcuts.** A source node often influences a target node via multiple different paths, often of different lengths. For instance, in § 3 Multi-step Reasoning, we observe that "Texas" and "say a capital" features upweight the "Austin" response via direct connections to the output and indirectly via "say Austin" features. Similarly, although we focused on the two-step path from Dallas → Texas → Austin, there also exist direct positive connections from "Dallas" features to "Austin" features! In the taxonomy of Alon [55], this corresponds to a "coherent feedforward loop," a commonly observed circuit motif in biological systems.

Jack Lindsey, et. al., *On the Biology of a Large Language Model,* TRANSFORMER CIRCUITS THREAD at § 13 (2025), available at: https://transformer-circuits.pub/2025/attribution-graphs/biology.html (emphasis added).

55. The Anthropic '470 Products perform the step of addressably configuring one synapse circuit element to connect one addressably configured neuron circuit element to another addressably configured neuron circuit element to form an initial addressably configured array comprising an initial two-dimensional neuromorphic network. Specifically, in the Anthropic '470 Products, synapses (connections between neuron nodes) are implemented as at least the following: parent-child tree node references, JSON-RPC channels, and structural parent-child connections.

In the Anthropic '470 Products, layout nodes are connected and bound via insertChild and removeChild, which form an initial, addressable two-dimensional network grid of nodes. The layout engine (engine.ts) in the Anthropic '470 Products sets up and forms these connection nodes. "This is our implementation of swarms. It's very exciting because uh it just lets Claude do more work for longer, more autonomously. You have a bunch of different uh uncorrelated context windows and you have this kind of communication between agents. They can just do more." Boris Cherny & Gergely Orosz, *Building Claude Code with Boris Cherny,* YOUTUBE.COM at 01:00:46-01:00:59 (Mar. 4, 2026), available at: https://www.youtube.com/watch?v=julbw1JuAz0.

56. The Anthropic '470 Products perform the step of addressably reconfiguring the neuromorphic network comprising a different two-dimensional array of neuron and synapse circuit elements. Specifically, the Anthropic '470 Products perform addressable reconfiguration when a two-dimensional layout tree is recomputed. For example, the Yoga Flexbox layout engine in the Anthropic '470 Products traverses the tree structure, calculates updated layout boundaries, and mutates component coordinates (*e.g.*, when nodes are added, removed, or resized). This produces a newly structured, different two-dimensional array of coordinates. The Anthropic '470 Products contain source code, including yoga.ts, that enables them to recompute a tree coordinate array, thereby updating absolute bounds to reconfigure the network.

57. The Anthropic '470 Products contain source code (renderer.ts) that performs the re-rendering of node structures so that they map onto the physical terminal grid.

58. The Anthropic '470 Products comprise the central pattern generator producing cyclic or regenerative behavior through at least a neuron and synapse pair of addressably configurable circuit elements. Specifically, the central pattern generator's cyclic or regenerative behavior is implemented through two recurring processes working in tandem: (1) the background

task scheduler, which utilizes a timer thread to poll and execute cyclical operations at defined intervals, and (2) the memory consolidation background loop (autoDream.ts), which continuously evaluates temporal and session gate conditions to periodically summary-prune active states, looping back recursively. The cronScheduler.ts within the Anthropic '470 Products triggers cyclic execution of defined commands based on a recurring polling interval.

59. The Anthropic '470 Products contain a memory consolidation background module (autoDream.ts) that executes a cyclical background agent loop when temporal and session volume gates open. Security audits and system design analyses confirm that this background CPG runs periodically to perform memory consolidation and optimization within the Anthropic '470 Products.

> DD.md), and an optional DREAMS.md for dreaming sweep summaries. When an embedding provider is configured, memory search uses hybrid retrieval combining vector similarity with keyword matching. An experimental dreaming system performs background consolidation, scoring candidates and promoting only qualified items from short-term recall into long-term memory. Before compaction, OpenClaw automatically reminds the agent to save important notes to memory files, preventing context loss. Both systems share the design commitment to user-visible, editable memory. OpenClaw invests more heavily in structured long-term memory promotion (dreaming, daily notes, memory search), while Claude Code invests more in graduated context compression (five layers with cache awareness). OpenClaw also supports pluggable compaction providers and session pruning, but its compaction pipeline is less graduated than Claude Code's five-layer system.

Jiacheng Liu et al., *Dive into Claude Code: The Design Space of Today's and Future AI Agent Systems,* ARXIV at 26 (April 14, 2026) (emphasis added).

60. The Anthropic '470 Products comprise the addressably configurable neuron generating a sequence of discrete events. Specifically, the platform's command parsers, lexers, and stream controllers act as neuron elements that decompose unstructured terminal inputs or execution logs into an ordered sequence of discrete, structured command events. This includes parsed subshell actions, variable assignments, standard input/output redirects, and operator pipelines. The Anthropic '470 Products contain source code (treeSitterAnalysis.ts) that walks the input command stream, decomposing it into a sequence of discrete structural segments and

operator events. Further, the AST parser in ast.ts in the Anthropic '470 Products generates a sequence of discrete, structured SimpleCommand types from the raw input.

61. The Anthropic '470 Products perform the step of outputting a process control signal to the external automated system or external device via the application of the central pattern generator. Specifically, the Anthropic '470 Products emit programmatic process control signals (e.g., virtual key presses, mouse clicks, shell commands, and terminal execution signals) to actuate control over external host systems, container processes, and remote environments. In addition, the Anthropic '470 Products contain a shell execution tool (BashTool.tsx) that handles command execution via a generator-based executor (runShellCommand), and outputting process control signals into the active host shell.

62. The Anthropic '470 Products comprise a structure of components of the constructed central pattern generator. Specifically, the structure of the central pattern generator is defined by the programmatic orchestration of the task scheduler, background daemon configurations, and concurrent locking mechanisms working in tandem to maintain safety and consistency in background processing. This is confirmed by the structure of the source code of the Anthropic '470 Products, which shows components such as autoDream, cronScheduler, and consolidationLock within a single directory.

63. The Anthropic '470 Products comprise a neuron of the neuron and synapse pair of addressably configurable circuit elements having a programmable threshold and a programmable refractory period. Specifically, the Anthropic '470 Products implement "programmable thresholds" and "programmable refractory periods" inside their memory consolidation and polling components. These parameters act as temporal and volume gating rules. The memory consolidation trigger (autoDream.ts) applies programmable thresholds (*e.g.*, requiring 24 elapsed

hours and 5 accumulated sessions) before permitting task execution. Furthermore, a refractory period (cooldown scan interval and file lock expiry) is enforced to throttle and prevent rapid overlapping executions, safeguarding system execution against feedback loops. Further, the Anthropic '470 Products contain a file lock utility (consolidationLock.ts) that implements a PID-backed refractory period to prevent concurrent execution conflicts, enforcing a strict timeout parameter before the lock can be recycled.

64. The Anthropic '470 Products comprise a synapse of the neuron and synapse pair of addressably configurable circuit elements having a programmable delay or distance parameter value. Specifically, programmable delay parameter values are implemented as jittered backoff intervals, network connection retries, and tool timeouts. In command scope analysis, a programmable "distance" parameter maps to scope-level tracking constraints. This resolves shell variable propagation across sequential commands (varScope) but isolates and copies scope (strictly limiting the propagation distance) across nested pipelines or subshells. Further, the network connection interface in the Anthropic '470 Products (withRetry.ts) implements connection retries as an asynchronous generator that uses custom retry logic. *See* Alex Albert & Cat Noone*, Building and prototyping with Claude Code,* YOUTUBE.COM at 09:05-09:11 (Aug. 21, 2025), available at: https://www.youtube.com/watch?v=DAQJvGjlgVM ("And we also handle interacting with the underlying API. So we make sure that we have backoff if there's any API errors. We very aggressively prompt cache to make sure that your requests are token-efficient.").

65. Responsive to a neuron that is addressably configurable, the Anthropic '470 Products perform the step of generating a sequence of discrete events, the addressably configurable synapse permitting at least two discrete events to propagate together through the synapse. Specifically, the Anthropic '470 Products' synapse (implemented as the communication, stream,

or parsing channel) permits multiple discrete events (such as stdout/stderr streams, parallel tool uses, or piped subcommands) to propagate and resolve together in parallel or as concatenated payloads. Further, the syntax utility in the Anthropic '470 Products (ast.ts) processes concatenation nodes to merge and pass multiple distinct arguments together.

66. The Anthropic '470 Products perform the step of applying a process control signal to an actuator for controlling the external automated system or external device in response to the prediction of the component failure and a reconfiguration and optimization of the neuromorphic network. Specifically, when the Anthropic '470 Products predict context window saturation or memory limit saturation (predicted "component failure"), the Anthropic '470 Products automatically execute the background memory consolidation process. The Anthropic '470 Products thus act as a central pattern generator that outputs process control signals (by invoking a restricted tool-use loop via runForkedAgent with read-only bash limits, restricting to specific AST commands to avoid security exceptions) to reconfigure, summarize, and optimize the neuromorphic memory structure (MEMORY.md) so that the active memory network fits within a strict 200-line or 25KB limit, thereby optimizing network performance.

67. Anthropic has directly infringed and continues to directly infringe the '470 Patent by, among other things, making, using, offering for sale, and/or selling technology for constructing a neuromorphic network comprising at least one artificial neural network, including but not limited to the Anthropic '470 Product(s).

68. The Anthropic '470 Product(s) are available to businesses and individuals throughout the United States.

69. The Anthropic '470 Product(s) are provided to businesses and individuals located in the District of Delaware.

70. By making, using, testing, and/or performing a method of constructing a neuromorphic network comprising at least one artificial neural network, including but not limited to the Anthropic ‘470 Product(s), Anthropic has injured The University of Tennessee Research Foundation (“UTRF”) and is liable to UTRF for directly infringing one or more claims of the ‘470 Patent, including at least claim 10 pursuant to 35 U.S.C. § 271(a).

71. Anthropic also indirectly infringes the ‘470 Patent by actively inducing infringement under 35 U.S.C. § 271(b).

72. Anthropic has had knowledge of the ‘470 Patent since at least service of this Complaint or shortly thereafter, and Anthropic knew of the ‘470 Patent and knew of its infringement, including by way of this lawsuit.

73. Anthropic intended to induce patent infringement by third-party customers and users of the Anthropic ‘470 Product(s) and had knowledge that the inducing acts would cause infringement or was willfully blind to the possibility that its inducing acts would cause infringement. Anthropic specifically intended and was aware that the normal and customary use of the accused products would infringe the ‘470 Patent. Anthropic performed the acts that constitute induced infringement, and would induce actual infringement, with knowledge of the ‘470 Patent and with the knowledge that the induced acts would constitute infringement. For example, Anthropic provides the Anthropic ‘470 Product(s) that have the capability of operating in a manner that infringe one or more of the claims of the ‘470 Patent, including at least claim 10, and Anthropic further provides documentation and training materials that cause customers and end users of the Anthropic ‘470 Product(s) to utilize the products in a manner that directly infringe one or more claims of the ‘470 Patent. By providing instruction and training to customers and end-users on how to use the Anthropic ‘470 Product(s) in a manner that directly infringes one or more

claims of the ‘470 Patent, including at least claim 10, Anthropic specifically intended to induce infringement of the ‘470 Patent. Anthropic engaged in such inducement to promote the sales of the Anthropic ‘470 Product(s), *e.g.*, through Anthropic user manuals, product support, marketing materials, and training materials to actively induce the users of the accused products to infringe the ‘470 Patent. Accordingly, Anthropic has induced and continues to induce users of the accused products to use the accused products in their ordinary and customary way to infringe the ‘470 Patent, knowing that such use constitutes infringement of the ‘470 Patent.

74. To the extent applicable, the requirements of 35 U.S.C. § 287(a) have been met with respect to the ‘470 Patent.

75. As a result of Anthropic’s infringement of the ‘470 Patent, UTRF has suffered monetary damages, and seeks recovery in an amount adequate to compensate for Anthropic’s infringement, but in no event less than a reasonable royalty for the use made of the invention by Anthropic together with interest and costs as fixed by the Court.

COUNT II
INFRINGEMENT OF U.S. PATENT NO. 10,095,718

76. Plaintiff references and incorporates by reference the preceding paragraphs of this Complaint as if fully set forth herein.

77. Anthropic designs, makes, sells, offers to sell, imports, and/or uses the following products: Anthropic’s Claude Code and its underlying agentic software architecture (collectively, the “Anthropic ‘718 Product(s)”).

78. One or more Anthropic subsidiaries and/or affiliates use the Anthropic ‘718 Product(s) in regular business operations.

79. The Anthropic ‘718 Products construct a multi-agent system representing a software-based neuromorphic network.

80. The Anthropic '718 Products perform a computer-implemented method for constructing a neuromorphic network comprising at least one artificial neural network.

81. The Anthropic '718 Products construct a multi-agent system representing a software-based neuromorphic network. The artificial neural network is the Claude LLM, which represents an addressably reconfigurable multi-dimensional array of weights (interconnected circuit elements) in a high-dimensional vector/token embedding space.

82. The Anthropic '718 Products contain a "coordinator system" prompt that explicitly configures the agent workers as discrete nodes (neurons) that perform research, implementation, and verification tasks. *See* Boris Cherny, *Head of Claude Code: What Happens After Coding Is Solved*, YOUTUBE.COM at 57:29-57:47 (February 19, 2026), available at: https://www.youtube.com/watch?v=We7BZVKbCVw. ("What is your brain? Like what are what is it? It's like it's like a bunch of neurons that are connected. And so what you can do is like in a human brain or an animal brain, you can study it at this kind of mechanistic level to understand what the neurons are doing. It turns out surprisingly a lot of this does translate to models also. So model neurons are not the same as animal neurons, but they behave similarly in a lot of ways.").

83. Anthropic has described the Anthropic '718 Products as comprising a neural network that contains neurons (features), weights, and circuits (pathways connecting features). *See* Chris Olah, *Chris Olah - Looking Inside Neural Networks with Mechanistic Interpretability*, YOUTUBE.COM at 06:52-07:20 (Sept. 1, 2023), available at: https://www.youtube.com/watch?v=2Rdp9GvcYOE ("I like to think about this is in terms of three kinds of objects... features... which are... say the easiest way to think about this would be a neuron and it might be something like a car detector or a curve detector or an edge detector ... weights

which connect together features and then circuits which are the combinations of weights and features together.”).

84. The Anthropic ‘718 Products represent concepts via linear combinations of the network’s underlying biological/artificial neurons.

> [E]ach concept <u>is represented across many neurons</u>, and each neuron is involved in representing many concepts. . . any internal state of the model can be represented in terms of a few active features instead of many active neurons. Just as every English word in a dictionary is made by combining letters, and every sentence is made by combining words, every feature in an AI model is made by combining neurons, and every internal state is made by combining features.

Anthropic, *Mapping the Mind of a Large Language Model*, ANTHROPIC WEBSITE (May 21, 2024), available at: https://www.anthropic.com/research/mapping-mind-language-model (emphasis added).

85. The Anthropic ‘718 Products perform the method using the special purpose computer processor comprising an artificial neural network configuration structure for addressably configuring the multi-dimensional array of addressably configurable circuit elements and an interface and control structure for connecting the addressably configured and connected multi-dimensional array of circuit elements to the external process; the special purpose computer being responsive to input data from the external process for addressably configuring the multi-dimensional array.

86. The Anthropic ‘718 Products comprise an artificial neural network configuration structure for addressably configuring the multi-dimensional array of addressably configurable circuit elements and an interface and control structure for connecting the addressably configured and connected multi-dimensional array of circuit elements to the external process; the special purpose computer being responsive to input data from the external process for addressably configuring the multi-dimensional array.

87. The Anthropic ‘718 Products contain QueryEngine message submission logic, and the main loop in query.ts acts as the artificial neural network configuration structure, adjusting

prompt contexts, compacting history, and configuring token budgets to reconfigure the LLM's attention weights. The interface and control structure are implemented in toolExecution.ts and query.ts in the Anthropic '718 Products, which hook the model's outputs (such as JSON-formatted tool calls) to external OS processes (executing bash, files, etc.) and stream results back.

88. The Anthropic '718 Products comprise an "auto mode" configuration and control loops. Boris Cherny & Alex Kantrowitz, *Claude Code Head Boris Cherny: Insane Growth, Tokenmaxxing, AI Agents' Next Frontie*r, YOUTUBE.COM at 38:36-38:49 (May 20, 2026), available at: https://www.youtube.com/watch?v=Z6IT4gjrcPE ("And so auto mode is the answer. And this is a new way of routing these tool calls. And the way that it works is whenever Claude wants to use a tool, it asks another Claude, is it safe to use this tool.").

89. The Anthropic '718 Products contain control features like retry logic, routing, planning, and verification loops that are integrated into the Anthropic '718 Products' runtime. *See* Lucas Gonzalez, *The Expanding Toolkit,* YOUTUBE.COM at 20:00-20:11 (May 8, 2026), available at: https://www.youtube.com/watch?v=KLCuxMDZSDg ("So, things like retry logic, routers, planners, verification loops, all of these are going to get absorbed in the model and they have been getting absorbed in the model as I've shown you in the prior slides in this talk.").

90. Anthropic's guide to building effective agents describes the routing and orchestrator architecture patterns that govern the Anthropic '718 Products' control flow.

> Routing classifies an input and directs it to a specialized followup task. This workflow allows for separation of concerns, and building more specialized prompts. Without this workflow, optimizing for one kind of input can hurt performance on other inputs. . . . In the orchestrator-workers workflow, a central LLM dynamically breaks down tasks, delegates them to worker LLMs, and synthesizes their results.

Anthropic, *Building effective agents*, Anthropic Website (Dec. 19, 2024), available at: https://www.anthropic.com/engineering/building-effective-agents.

91. The Anthropic '718 Products perform at least one circuit element of the addressably configured multi-dimensional array addressably configured as an input neuron.

92. The Anthropic '718 Products comprise a user input prompt and text attachments that are formatted and parsed into structured message arrays (mutableMessages). These message arrays represent input neurons that initialize attention weights inside the transformer network.

93. The Anthropic '718 Products contain input neuron elements that are endpoints of the network's weights. *See* Chris Olah & Interviewer, *Chris Olah - Looking Inside Neural Networks with Mechanistic Interpretability*, YOUTUBE.COM at 08:07-08:20 (Sept. 1, 2023), available at: https://www.youtube.com/watch?v=2Rdp9GvcYOE ("[B]ut if we just look at the weights in isolation they're not that interesting right we have the input neuron and the output neuron but yeah the weights by themselves you know they don't say that much but if we then go and connect them to features that we understand um suddenly that becomes very different.").

94. The Anthropic '718 Products perform the step of at least one circuit element of the addressably configured multi-dimensional array addressably configured as an output neuron. Specifically, the output generated by the Anthropic '718 Products (tokens, text block completions, and tool use JSON schemas) represents output neurons. The streamed output is captured by the QueryEngine streaming generator and normalized into assistant message structures.

95. The Anthropic '718 Products contain an output neuron element as the terminal point of the network's weight array. Chris Olah, *Looking Inside Neural Networks with Mechanistic Interpretability*, YOUTUBE.COM at 08:07-08:20 (Sept. 1, 2023), available at: https://www.youtube.com/watch?v=2Rdp9GvcYOE ("[B]ut if we just look at the weights in isolation they're not that interesting right we have the input neuron and the output neuron but yeah

the weights by themselves you know they don't say that much but if we then go and connect them to features that we understand um suddenly that becomes very different.").

96. The Anthropic '718 Products contain an input neuron layer that uses overcomplete features that represent concepts within linear combinations of underlying neurons. *See* Trenton Bricken, et al., *Towards Monosemanticity: Decomposing Language Models With Dictionary Learning*, TRANSFORMER CIRCUITS THREAD, at 3 (2023), available at: https://transformer-circuits.pub/2023/monosemantic-features/index.html ("[W]here a neural network represents more independent "features" of the data than it has neurons by assigning each feature its own linear combination of neurons. If we view each feature as a vector over the neurons, then the set of features form an overcomplete linear basis for the activations of the network neurons.").

97. The Anthropic '718 Products perform, under a special purpose program control, one to multiple circuit elements of the addressably configurable multi-dimensional array addressably configured as a neuron or a synapse forming a pathway between the input neuron receiving input data from the external process and the output neuron outputting data to the external process. Specifically, in coordinator mode, the Anthropic '718 Products configure pathways using multiple agents and tool execution loops. The Anthropic '718 Products contain a coordinator that spawns workers via AgentTool (neurons) that communicate via XML <task-notification> (synapses). Furthermore, toolExecution.ts in the Anthropic '718 Products implement self-correcting feedback loops (if a tool execution fails or returns an error, it is caught in a catch block and fed back as a new input to reconfigure the LLM's attention pathway).

98. In the Anthropic '718 Products, background memory consolidation in autoDream.ts runs a sub-agent to prune and update MEMORY.md, reinforcing valuable pathways (analogous to

Long-Term Potentiation). Further, the Anthropic '718 Products tool execution functionality (toolExecution.ts) catches execution errors in the feedback loop input packages.

99. The Anthropic '718 Products contain a background memory consolidation engine that uses the runForkedAgent helper to consolidate facts into MEMORY.md.

100. The Anthropic '718 Products utilize multi-agent swarm pathways, self-correcting feedback mechanisms, background memory consolidation (dreaming), and proactive execution pathways. "[R]unning more quads in parallel... very few people nowadays run one quad [Claude] code at a time. Most people run many many quad [Claude] codes, you know, ranging from, you know, a few to thousands." Boris Cherny & Alex Kantrowitz, *Claude Code Head Boris Cherny: Insane Growth, Tokenmaxxing, AI Agents' Next Frontier,* YOUTUBE.COM at 39:37-39:50 (May 20, 2026), available at: https://www.youtube.com/watch?v=Z6IT4gjrcPE.

101. The Anthropic '718 Products use asynchronous background "dreaming" that consolidates agent execution transcripts, identifies errors, and updates the shared memory state.

> Dreaming is a process [in the Anthropic '718 Products] that looks for patterns and mistakes across your recent agent sessions and their transcripts and automatically produces organized and up-to-date memory content . . . . It's a batch asynchronous process that runs separately from the work that you're doing uh within a specific session that's working on a specific task . . . . And dreaming comprehensively looks through recent transcripts, looks for common mistakes, things that a bunch of agents are doing like a failed tool call or strategies that are working out for them, and finds opportunities to update the memory state that will improve it in the future . . . . And again, because dreaming is an out-of-band process, it's in the background, it does this without adding any latency to the hot path of an agent's existing task.

Mahesh Murag, *Memory and Dreaming for Self-Learning Agents*, YOUTUBE.COM at 11:29-11:38, 12:14-12:23, 12:44-12:58, 14:45-14:53 (May 8, 2026), available at: https://www.youtube.com/watch?v=RtywqDFBYnQ.

102. The Anthropic '718 Products incorporate a dreaming consolidation engine that identifies temporal patterns and retry logic latency spikes (specifically, 60-second delay spikes) across separate agent runs. Mahesh Murag, *Memory and Dreaming for Self-Learning Agents,* YOUTUBE.COM at 21:52-22:13 (May 8, 2026), available at:

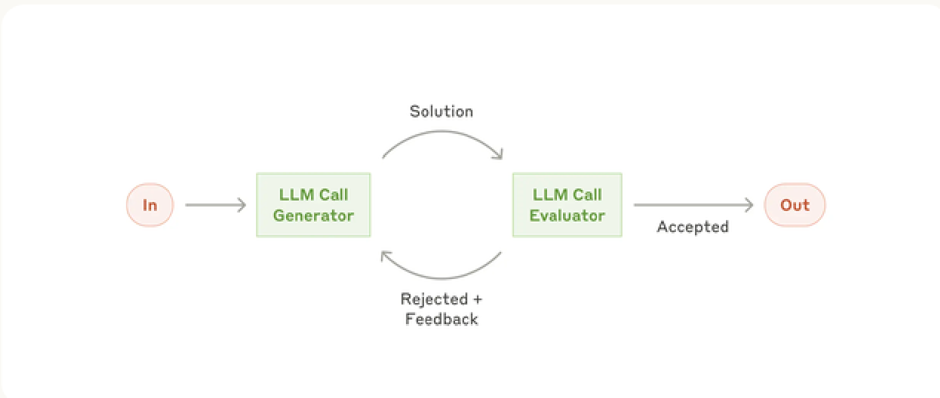
https://www.youtube.com/watch?v=RtywqDFBYnQ ("It said, 'Hey, a bunch of these agents were triggered exactly 60 seconds after an upstream spike in CPU utilization.' And it kind of figures out based on that pattern that there might be some retry logic that's getting triggered that's really inefficient and leading to a lot of wasted time when we're actually triaging this stuff. So, it identifies that because each of the individual agents aren't really noticing that pattern.").

103. The Anthropic '718 Products utilize proactive routines and scheduling structures to automate execution pathways. This functionality causes the Anthropic '718 Products to utilize autonomous workflows. *See* Maya Nielan, *Build a Proactive Agent Workflow with Claude Code*, YOUTUBE.COM at 21:10-21:52 (May 20, 2026), available at: https://www.youtube.com/watch?v=eSP7PLTXNy8 ("Proactive agents beat reactive agents... You can move from an agent that is waiting for you to actually press enter and create a PR to an agent that reacts to problems and opens a PR itself... You're a single schedule commands inside of Claude code away from creating your very first routine.").

104. The Anthropic '718 Products use multi-hop activation pathways and self-correcting feedback loops. The activation patterns in the Anthropic '718 Products propagate across multi-hop computational pathways between inputs, intermediate clusters, and output features. *See* Jack Lindsey, et al., *On the Biology of a Large Language Model*, TRANSFORMER CIRCUITS THREAD, at 14 (2025) ("The Texas features and the say a capital features jointly upweight the probability of the model saying Austin. They do so via two pathways: directly impacting the Austin output, and indirectly, by activating a cluster of say Austin output features. There also exists a "shortcut" edge directly from Dallas to say Austin. <u>The graph indicates that the replacement model does in fact perform 'multi-hop reasoning'</u> — that is, its decision to say Austin hinges on a chain of several intermediate computational steps.") (emphasis added).

105. The Anthropic '718 Products' implementation of iterative correction and optimizer loops functions as self-correcting feedback pathways.

In the evaluator-optimizer workflow, one LLM call generates a response while another provides evaluation and feedback in a loop.



The evaluator-optimizer workflow

Anthropic, *Building Effective Agents*, ANTHROPIC WEBSITE (Dec. 19, 2024), available at: https://www.anthropic.com/engineering/building-effective-agents.

106. Anthropic has directly infringed and continues to directly infringe the '718 Patent by, among other things, making, using, offering for sale, and/or selling technology for constructing a neuromorphic network comprising at least one artificial neural network, including but not limited to the Anthropic '718 Product(s).

107. The Anthropic '718 Product(s) are available to businesses and individuals throughout the United States.

108. The Anthropic '718 Product(s) are provided to businesses and individuals located in the District of Delaware.

109. By making, using, testing, and/or performing a method of constructing a neuromorphic network comprising at least one artificial neural network, including but not limited

to the Anthropic ‘718 Product(s), Anthropic has injured UTRF and is liable to UTRF for directly infringing one or more claims of the ‘718 Patent, including at least claim 35 pursuant to 35 U.S.C. § 271(a).

110. Anthropic also indirectly infringes the ‘718 Patent by actively inducing infringement under 35 U.S.C. § 271(b).

111. Anthropic has had knowledge of the ‘718 Patent since at least service of this Complaint or shortly thereafter, and Anthropic knew of the ‘718 Patent and knew of its infringement, including by way of this lawsuit.

112. Anthropic intended to induce patent infringement by third-party customers and users of the Anthropic ‘718 Product(s) and had knowledge that the inducing acts would cause infringement or was willfully blind to the possibility that its inducing acts would cause infringement. Anthropic specifically intended and was aware that the normal and customary use of the accused products would infringe the ‘718 Patent. Anthropic performed the acts that constitute induced infringement, and would induce actual infringement, with knowledge of the ‘718 Patent and with the knowledge that the induced acts would constitute infringement. For example, Anthropic provides the Anthropic ‘718 Product(s) that have the capability of operating in a manner that infringe one or more of the claims of the ‘718 Patent, including at least claim 35, and Anthropic further provides documentation and training materials that cause customers and end users of the Anthropic ‘718 Product(s) to utilize the products in a manner that directly infringe one or more claims of the ‘718 Patent. By providing instruction and training to customers and end-users on how to use the Anthropic ‘718 Product(s) in a manner that directly infringes one or more claims of the ‘718 Patent, including at least claim 35, Anthropic specifically intended to induce infringement of the ‘718 Patent. Anthropic engaged in such inducement to promote the sales of

the Anthropic '718 Product(s), *e.g.*, through Anthropic user manuals, product support, marketing materials, and training materials to actively induce the users of the accused products to infringe the '718 Patent. Accordingly, Anthropic has induced and continues to induce users of the accused products to use the accused products in their ordinary and customary way to infringe the '718 Patent, knowing that such use constitutes infringement of the '718 Patent.

113. To the extent applicable, the requirements of 35 U.S.C. § 287(a) have been met with respect to the '718 Patent.

114. As a result of Anthropic's infringement of the '718 Patent, UTRF has suffered monetary damages and seeks recovery in an amount adequate to compensate for Anthropic's infringement, but in no event less than a reasonable royalty for the use made of the invention by Anthropic, together with interest and costs as fixed by the Court.

PRAYER FOR RELIEF

WHEREFORE, Plaintiff University of Tennessee Research Foundation respectfully requests that this Court enter:

A. A judgment in favor of Plaintiff that Anthropic has infringed, either literally and/or under the doctrine of equivalents, the '470 and '718 Patents;

B. An Order permanently enjoining Anthropic, and all those acting in concert with it, from further infringement of the '470 and '718 Patents in accordance with 35 U.S.C. § 283;

C. An award of damages resulting from Anthropic's acts of infringement in accordance with 35 U.S.C. § 284, as well as costs and prejudgment and post-judgment interest;

D. A judgment and order finding that this is an exceptional case within the meaning of 35 U.S.C. § 285 and awarding reasonable attorneys' fees to Plaintiff against Anthropic.

E. Any and all other relief to which Plaintiff may show itself to be entitled.

JURY TRIAL DEMANDED

Pursuant to Rule 38 of the Federal Rules of Civil Procedure, Plaintiff University of Tennessee Research Foundation requests a trial by jury of any issues so triable by right.

Dated: July 20, 2026

Respectfully submitted,

FARNAN LLP

/s/ Brian E. Farnan

Brian E. Farnan (Bar No. 4089)
Michael J. Farnan (Bar No. 5165)
919 N. Market St., 12th Floor
Wilmington, DE 19801
(302) 777-0300
bfarnan@farnanlaw.com
mfarnan@farnanlaw.com

Attorneys for Plaintiff University of Tennessee Research Foundation

Of Counsel:

Dorian S. Berger
Daniel P. Hipskind
Erin E. McCracken
BERGER & HIPSKIND LLP
9538 Brighton Way, Ste. 320
Beverly Hills, CA 90210
Telephone: 323-886-3430
Facsimile: 323-978-5508
E-mail: dsb@bergerhipskind.com
E-mail: dph@bergerhipskind.com
E-Mail: eem@bergerhipskind.com