



(51) International Patent Classification:

G06N 3/092 (2023.01) E21B 33/14 (2006.01)
G06N 3/04 (2006.01)

(21) International Application Number:

PCT/US2024/039385

(22) International Filing Date:

24 July 2024 (24.07.2024)

(25) Filing Language:

English

(26) Publication Language:

English

(30) Priority Data:

18/544,772 19 December 2023 (19.12.2023) US

(71) Applicant: **HALLIBURTON ENERGY SERVICES, INC.** [US/US]; 3000 N. Sam Houston Parkway E., Houston, Texas 77032-3219 (US).

(72) Inventors: **YERUBANDI, K V V N Krishna Babu**; 3000 N. Sam Houston Parkway E., Houston, Texas 77032-3219

(US). **JIMENEZ, Walmy Cuello**; 3000 N. Sam Houston Parkway E., Houston, Texas 77032-3219 (US). **JOGDAND, Anoop Sheshrao**; Sai Radhe Building, 100+101 Kennedy Road, Pune 411001 (IN).

(74) Agent: **TUMEY, Corey S.**; P.O. Box 890226, Houston, Texas 77062-9998 (US).

(81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CV, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IQ, IR, IS, IT, JM, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, MG, MK, MN, MU, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, WS, ZA, ZM, ZW.

(54) Title: LARGE LANGUAGE MODEL INTERFACE FOR WELLBORE CEMENT JOB DESIGN

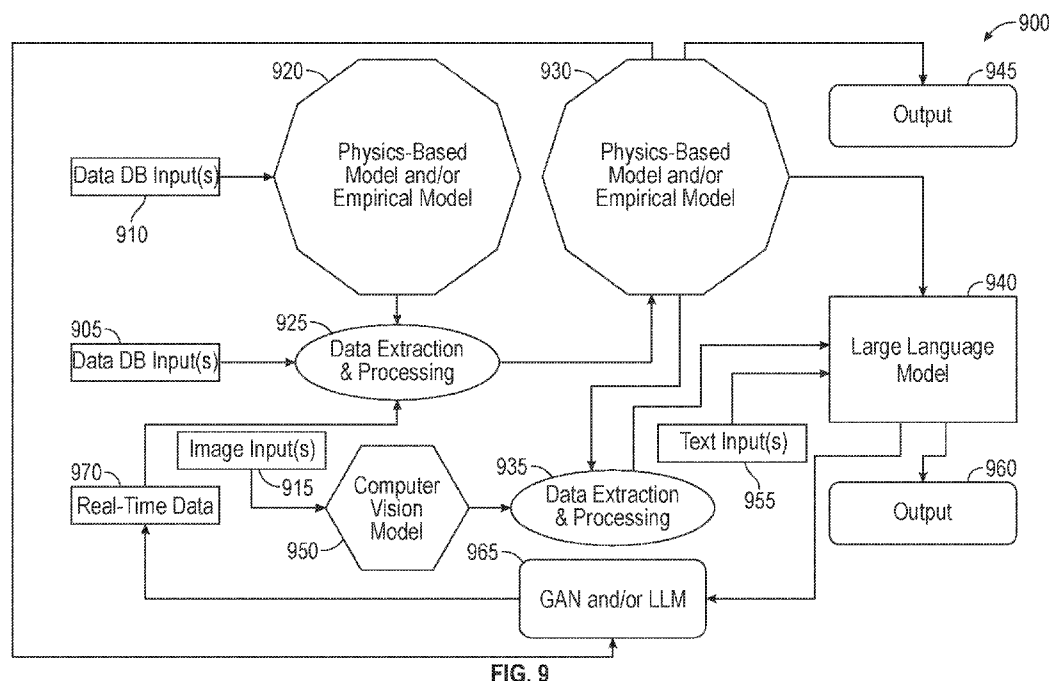


FIG. 9

(57) Abstract: A method may include: providing one or more inputs to a hybrid data generator, wherein one of the one or more inputs is based at least in part on a wellsite location, wherein the hybrid data generator comprises a large language model, and wherein the large language model is based at least in part on a machine learning algorithm; utilizing an information handling system to generate a cement job design based at least in part on the one or more inputs and the hybrid data generator; performing at least a portion of a cementing operation based at least in part on the cement job design; and collecting at least one measurement from at least one sensor during the cementing operation.

(84) **Designated States** (*unless otherwise indicated, for every kind of regional protection available*): ARIPO (BW, CV, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SC, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, ME, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report (Art. 21(3))*

Large Language Model Interface for Wellbore Cement Job Design

BACKGROUND

[0001] The oil and gas industry may use wellbores as fluid conduits to access subterranean deposits of various fluids and minerals which may include hydrocarbons. A drilling operation may be utilized to construct the fluid conduits which are capable of producing hydrocarbons disposed in subterranean formations. Wellbores may be incrementally constructed as tapered sections, which sequentially extend into a subterranean formation. Cement compositions are used in a variety of subterranean operations. For example, in subterranean well construction, a pipe string (e.g., casing, liners, expandable tubulars, etc.) can be run into a borehole and cemented in place. The process of cementing the pipe string in place is commonly referred to as “primary cementing.” In a typical primary cementing method, a cement composition can be pumped into an annulus between the walls of the wellbore and the exterior surface of the pipe string disposed therein. The cement composition may set in the annular space, thereby forming an annular sheath of hardened, substantially impermeable cement (i.e., a cement sheath) that may support and position the pipe string in the wellbore and may bond the exterior surface of the pipe string to the subterranean formation. Among other things, the cement sheath surrounding the pipe string functions to prevent the migration of fluids in the annulus, as well as protecting the pipe string from corrosion. Cement compositions also may be used in remedial cementing methods, for example, to seal cracks or holes in pipe strings or cement sheaths, to seal highly permeable formation zones or fractures, to place a cement plug, and the like.

[0002] Designing cement compositions suitable for wellbore use is a complicated task where many process variables much be simultaneously satisfied including physical properties of the cement slurry as well as the physical properties of the set cement. The cement design process is typically performed in an iterative manner whereby a cement design is changed repeatedly until a cement composition which converges to satisfy all process variables is reached.

BRIEF DESCRIPTION OF THE DRAWINGS

[0003] These drawings illustrate certain aspects of some examples of the present disclosure and should not be used to limit or define the disclosure.

[0004] FIG. 1a illustrates surface equipment used in placement of a cement composition.

[0005] FIG. 1b illustrates a cement composition placement into a subterranean formation.

[0006] FIG. 2 illustrates a schematic view of an information handling system.

[0007] FIG. 3 illustrates another schematic view of an information handling system.

[0008] FIG. 4 illustrates a schematic view of a network.

[0009] FIG. 5 illustrates a schematic view of transformer architecture.

[0010] FIG. 6 illustrates a schematic view of an encoder.

[0011] FIG. 7 illustrates a schematic view of a decoder.

[0012] FIG. 8 illustrates a schematic of a machine learning algorithm which may be used for deep learning.

[0013] FIG. 9 illustrates a hybrid data generator which incorporates deep learning and a physical-model.

[0014] FIG. 10 illustrates a method of using a hybrid data generator to generate a cement job design.

DETAILED DESCRIPTION

[0015] This disclosure details methods and systems which utilize hybrid data generators which may include at least a Large Language Model (“LLM”) to aid in the creation of a wellbore cement job designs which may be used to construct subterranean wellbores. Cement job designs can include representative data about the wellbore, service fluids for use in the cementing job, centralizer location, and pump schedule including rates and volumes of each service fluid. A large language model takes natural language as input to generate a structured query in response to the input. The structured query is utilized as input to one or more job design models which in turn executes the query generated by the LLM to calculate one or more components of the wellbore cement job design. The LLM receives the wellbore cement job design and displays the output of the cement job design to the user.

[0016] As discussed above, wellbore cement job design is an iterative process where a cement design is changed repeatedly until a cement composition which converges to satisfy all process variables is reached. In addition to the physical properties including for example, 24-hour and ultimate compressive strength, rheology, and density, of the cement which must be satisfied, a cement job design must also satisfy industry best practices, company work methods, and industry safety standards, for example. However, the documentation which contains the best practices, work methods, safety standards, and other important information required when designing a wellbore cement job design are scattered across multiple systems. The lack of a centralized and organized system makes it difficult to keep tabs on the latest revisions, updates, or modifications to engineering calculations and work methods. Users may inadvertently refer to outdated documents. In dynamic engineering environments where new technologies and methodologies emerge frequently, it becomes crucial to have an efficient method of dissemination and tracking to ensure the most up-to-date practices are followed. Inefficiencies and reduced productivity are also potential outcomes when engineering calculations and work methods are scattered across

various sources. Users may spend excessive time searching for the correct information or duplicating efforts to develop similar solutions.

[0017] A large language model interface is able to synthesize all the industry best practices, company work methods, and industry safety standards as well as the relevant engineering calculations into a common interface. In this way, the user can be assured that the wellbore cement job design meets all the physical requirements as well as the non-physical requirements. In an embodiment, a large language model is trained using a dataset which is representative of the industry best practices, company work methods, and industry safety standards.

[0018] In some examples, historical multi-disciplinary datasets may be used to at least partially inform future wellbore construction and cementing operations by being at least partially incorporated into a cement job design which may be used to construct a subterranean formation. For example, multi-disciplinary datasets may at least partially inform components of a cement job design including borehole design (e.g., alternatively wellbore design), cement designs, operational plans, equipment utilized for the wellbore construction, fluid parameters, and/or hydraulic calculations. The multi-disciplinary datasets may further include information from technical fields including engineering design, geology, geophysics, and operational execution. For example, multi-disciplinary datasets may include at least one or more of engineering data, geological data, geo-mechanical data, geo-physical data, data from lab-based tests, data modelled from simulations, data modelled from empirical models, data modelled from physics-based models, data from physics-informed neural networks, operational data from current cementing operations, operational data from previous cementing operations, measurements collected from current cementing operations, measurements collected from previous cementing operations, information collected from previous cementing reports, previously created cement job design, logging data, available equipment in a given region, and combinations thereof. In addition to the foregoing datasets, information from public datasets including the National Oceanic and Atmospheric Administration (“NOAA”), public geological databases, weather databases and models, traffic and road restriction information, and combinations thereof may be included in the multi-disciplinary datasets.

[0019] In some examples, the borehole design may include a starting and ending depth of each borehole section, a diameter of each borehole section, a casing setting depth of each section, and/or any information related to one or more trajectories in which the borehole section extends (e.g., a directional plan including azimuth, inclination, dogleg severity, build rate, and/or walk rate. In some non-limiting examples, the borehole design may at least partially account for the one or more types of rock that are encountered in the subterranean formation, the one or more pore pressures

of the various rock layers in the subterranean formation, the stresses and forces of the subterranean formation (e.g., basin stresses), and the stability of the rocks in the subterranean formation. In some non-limiting examples, the operational plans may include operational or engineering parameters including casing setting depths as a function of engineering parameters, depth, etc. In some examples, calculations such as mechanical specific energy or specific mechanical energy (“SME”) may be at least partially used to guide the cement job design.

[0020] In some non-limiting examples, the utilized equipment may include rigs, drill pipe, spiraled drill pipe, drilling collars, reamers, casing and liners, isolation plugs and/or packers, casing patches, cementing equipment, wellbore logging equipment, drill bits, mills and milling assemblies, cutting tools and cutting assemblies, centrifuges, degassers, desanders, and various measurement devices and sensors which may be disposed on any portion of the drilling and/or wellbore equipment. In further examples, measurement devices and sensors may be included in or on a bottom hole assembly (“BHA”), which may be disposed at the distal end of the drill string, towards the drill bit of cutting assemblies. Additional equipment utilized during drilling and wellbore construction operations will be described further below. In some non-limiting examples, the fluid parameters may include components for drilling mud and other drilling treatment fluids such as base fluids (e.g., water-based fluids, invert emulsions, and direct emulsions), clay (e.g., bentonite), weighting agents (e.g., barite), chemical additives (e.g., shale inhibitors, scale inhibitors, flocculants, foaming agents, stabilizers, surfactants, emulsifiers, and/or friction reducers), lost circulation materials, completions fluids, and other circulating fluids. In further examples, the fluid parameters may include loadings for the planning drilling fluid components and additives.

[0021] As previously mentioned, cement job performance may be assessed in a multitude of ways. In some examples, a range of cement job designs may be created from which a cement job design may be selected for execution. In some non-limiting examples, operational features which may be utilized to assess cement job performance and optimize a cement job design may include maximizing fluid displacement, optimizing centralizer location and centralizer type, selecting fluids suitable for the particular cementing operation, optimizing fluid volumes and rates, satisfying physical property requirements of the cement slurry including rheology and density, ensuring that the cement slurry is mixable and pumpable, ensuring that the cement slurry has correct transient properties such as thickening time, satisfying physical property requirements of the set cement, satisfying chemical requirements of the cement slurry and other fluids such as displacement fluids, ensuring compatibility between subsequently pumped fluids, maximizing hole stability, minimizing total cementing cost, minimizing cost per wellbore section, minimizing

time spent on each wellbore section, and combinations thereof. In some examples, cement job designs generated by hybrid data generators may include constraints which optimize a cement job design for a single operational feature or a weighted set of operational features. As such, the one or more cement job designs, which may at least partially be generated by hybrid data generators, may be optimized with respect to one or more target objectives or operational features.

[0022] In some examples, cementing operations may be time sensitive operations. In further examples, it may be time consuming for a human to identify, analyze, and incorporate all relevant data into an initial cement job design in a timely manner. Additionally, it may be desirable to update cement job designs during the cementing operations, however time constraints may restrict the breadth and depth of analysis that may be performed using current methods to generate the updated cement job design. The foregoing may be true despite the use of currently available automatic and/or semi-automatic tools which may be implemented on an information handling system to analyze, compile, organize, and/or structure some of the datasets. Additionally, analysis of the available data may require subject matter expertise for adequate analysis and development of the cement job design. In some examples, impactful relationships between independent variables in a multi-disciplinary dataset and the cement job performance may not yet have been identified.

[0023] Given the time constraints and analytical constraints imposed by the current methods, it may be beneficial to utilize a hybrid data generator to generate cement job designs. The cement job designs may at least partially guide the construction of a subterranean wellbore. In some examples, hybrid data generators which utilize Large Language Models, may be able to generate initial and updated cement job designs at a faster pace than the traditional processes. As previously mentioned, traditional processes utilized to develop cement job designs may include manual processes, partially automated processes, fully automated processes, and combinations thereof. However, traditional processes may not have previously used hybrid data generators which utilize Large Language Models. In further examples, utilizing hybrid data generators may allow for humans, including subject matter experts, to develop and update cement job designs in a more efficient manner. For example, with the benefit of cement job designs constructed from hybrid data generators, personnel may spend less time generating cement job designs and more time reviewing and optimizing cement job designs while reducing the overall time required to generate an adequate cement job design. As such, the use of hybrid data generators which may be at least partially supported by Large Language Models, may be beneficial to the process of creating cement job designs.

[0024] An example technique for placing a cement composition into a subterranean formation will now be described with reference to FIG. 1a and FIG. 1b. FIG. 1a illustrates surface equipment 100 that may be used in placement of a cement composition in accordance with certain embodiments. It should be noted that while FIG. 1a generally depicts a land-based operation, however, the principles described herein are equally applicable to subsea operations that employ floating or sea-based platforms and rigs, without departing from the scope of the disclosure. As illustrated by FIG. 1a, surface equipment 100 may include a cementing unit 102, which may include one or more cement trucks. Cementing unit 102 may include mixing equipment and pumping equipment. The cementing unit 102 may pump a cement composition 108 through a feed pipe 104 and to a cementing head 106 which conveys the cement composition 108 downhole.

[0025] Turning now to FIG. 1b, the cement composition 108 may be placed into a subterranean formation 130 in accordance with example embodiments. As illustrated, wellbore 110 may be drilled into the subterranean formation 130. While wellbore 110 is shown extending generally vertically into the subterranean formation 130, the principles described herein are also applicable to wellbores that extend at an angle through the subterranean formation 130, such as horizontal and slanted wellbores. As illustrated, wellbore 110 comprises walls 112. In the illustrated embodiment, surface casing 114 has been inserted into wellbore 110. The surface casing 114 may be cemented to the walls 112 of the wellbore 110 by cement sheath 116. In the illustrated embodiment, one or more additional conduits (e.g., intermediate casing, production casing, liners, etc.), shown here as casing 118 may also be disposed in wellbore 110. As illustrated, there is a wellbore annulus 120 formed between the casing 118 and the walls 112 of the wellbore 110 and/or the surface casing 114. One or more centralizers 122 may be attached to casing 118, for example, to centralize the casing 118 in the wellbore 110 prior to and during the cementing operation.

[0026] With continued reference to FIG. 1b, cement composition 108 may be pumped down the interior of casing 118. Cement composition 108 may be allowed to flow down the interior of the casing 118 through the casing shoe 124 at the bottom of the casing 118 and up around the casing 118 into the wellbore annulus 120. Cement composition 108 may be allowed to set in the wellbore annulus 120, for example, to form a cement sheath that supports and positions the casing 118 in the wellbore 110. While not illustrated, other techniques may also be utilized for introduction of the cement composition 108. By way of example, reverse circulation techniques may be used that include introducing the cement composition 108 into the subterranean formation 130 by way of the wellbore annulus 120 instead of through the casing 118.

[0027] As it is introduced, the cement composition 108 may displace other fluids 126, such as drilling fluids and/or spacer fluids that may be present in the interior of the casing 118 and/or the

wellbore annulus 120. At least a portion of the displaced fluids 218 may exit the wellbore annulus 214 via a flow line and be deposited, for example, in one or more retention pits (e.g., a mud pit). Referring again to FIG. 1b, a bottom plug 128 may be introduced into the wellbore 110 ahead of the cement composition 108, for example, to separate the cement composition 108 from the fluids 126 that may be inside the casing 118 prior to cementing. After the bottom plug 128 reaches the landing collar 132, a diaphragm or other suitable device should rupture to allow the cement composition 108 through the bottom plug 128. In FIG. 1b, the bottom plug 128 is shown on the landing collar 132. In the illustrated embodiment, a top plug 134 may be introduced into wellbore 110 behind the cement composition 108. The top plug 134 may separate the cement composition 108 from a displacement fluid 136 and push the cement composition 108 through the bottom plug 128.

[0028] The operations of surface equipment 100 may be guided by a cement job design. In some examples, an initial cement job design may be generated prior to moving any cementing equipment to a wellsite location. In further examples, the cement job design may be generated from a hybrid data generator which may further utilize a Large Language Model, physical models, empirical models, cost models, material supply models, and/or combinations thereof.

[0029] Without limitation, surface equipment 100 may be connected to and/or controlled by information handling system 131. Without limitation, information handling system 131 may be disposed down hole in a bottom hole assembly. Information handling system 131 may be connected to sensors disposed on or operatively connected to any piece of equipment used in surface equipment 100 and sensors disposed within the wellbore 110. Processing of information recorded may occur down hole and/or on at the surface. Processing occurring downhole may be transmitted to the surface to be recorded, observed, and/or further analyzed. Additionally, information recorded on information handling system 131 that may be disposed down hole may be stored until a bottom hole assembly may be brought to the surface. In examples, information handling system 131 may communicate with surface equipment 100 through a communication line. In examples, wireless communication may be used to transmit information back and forth between information handling system 131 and surface equipment 100. Information handling system 131 may transmit information to surface equipment 100 and may receive as well as process information recorded by bottom surface equipment 100. In examples, a downhole information handling system may include, without limitation, a microprocessor or other suitable circuitry, for estimating, receiving, and processing signals. Downhole information handling system may further include additional components, such as memory, input/output devices, interfaces, and the like. In examples, a bottom hole assembly may include one or more additional components, such as

analog-to-digital converter, filter, and amplifier, among others, that may be used to process the measurements of a bottom hole assembly before they may be transmitted to the surface. Alternatively, raw measurements from a bottom hole assembly may be transmitted to the surface. [0030] Any suitable technique may be used for transmitting signals from information handling system 131 to surface equipment 100, including, but not limited to, wired methods, acoustic methods, and electromagnetic methods. Surface equipment 100 may include a telemetry subassembly that may transmit telemetry data to information handling system 131. The information handling system 131 may communicate with surface equipment 100 via a communication link 140, which may be a wired or wireless link. The telemetry data may be analyzed and processed by information handling system 131. In some examples, information handling system 131 may be configured to update a hybrid data generator to generate an updated cement job design based on the measurements gathered from the various sensors disposed on the surface equipment. In some examples, threshold values set for various cementing parameters, engineering parameters, operational parameters, and/or fluid parameters, which may be measured by any one or more of the sensors disposed within the cementing operation, may trigger the hybrid data generator to generate an updated cement job design. In further examples, the information handling system may be configured to update the hybrid data generator such that the cement job design is updated continuously, at set intervals, at random intervals, by manual execution as determined by personnel, when a threshold is met for any one or more parameters as described above, or combinations thereof. In some examples, manual input may be provided which may be utilized to update the hybrid data generator. In further examples the updated cement job design may be automatically implemented or may require review and approval by personnel prior to implementation.

[0031] As illustrated, communication link 140 (which may be wired or wireless, for example) may be provided that may transmit data from surface equipment 100 to an information handling system 131. Information handling system 131 may include a personal computer 141, a video display 142, a keyboard 144 (i.e., other input devices.), and/or non-transitory computer-readable media 146 (e.g., optical disks, magnetic disks) that can store code representative of the methods described herein. In addition to, or in place of processing at the surface, processing may occur downhole. As will be discussed below, the hybrid data generator may be executed on information handling system 131, both before cementing operations commence, while cementing operations are occurring, or during periods where cementing operations are stalled, to generate an initial and/or an updated cementing program.

[0032] Information handling system 131 may include any instrumentality or aggregate of instrumentalities operable to compute, estimate, classify, process, transmit, receive, retrieve, originate, switch, store, display, manifest, detect, record, reproduce, handle, or utilize any form of information, intelligence, or data for business, scientific, control, or other purposes. For example, an information handling system 131 may be a personal computer, a network storage device, or any other suitable device and may vary in size, shape, performance, functionality, and price. Information handling system 131 may include random access memory (RAM), one or more processing resources such as a central processing unit (CPU) or hardware or software control logic, ROM, and/or other types of nonvolatile memory. Additional components of the information handling system 131 may include one or more disk drives 146, output devices 142, such as a video display, and one or more network ports for communication with external devices as well as an input device 144 (e.g., keyboard, mouse, etc.). Information handling system 131 may also include one or more buses operable to transmit communications between the various hardware components.

[0033] Alternatively, systems and methods of the present disclosure may be implemented, at least in part, with non-transitory computer-readable media. Non-transitory computer-readable media may include any instrumentality or aggregation of instrumentalities that may retain data and/or instructions for a period of time. Non-transitory computer-readable media may include, for example, storage media such as a direct access storage device (e.g., a hard disk drive or floppy disk drive), a sequential access storage device (e.g., a tape disk drive), compact disk, CD-ROM, DVD, RAM, ROM, electrically erasable programmable read-only memory (EEPROM), and/or flash memory; as well as communications media such wires, optical fibers, microwaves, radio waves, and other electromagnetic and/or optical carriers; and/or any combination of the foregoing.

[0034] FIG. 2 illustrates an example information handling system 131 which may be employed to perform various steps, methods, and techniques disclosed herein. Persons of ordinary skill in the art will readily appreciate that other system examples are possible. As illustrated, information handling system 131 includes a processing unit (CPU or processor) 202 and a system bus 204 that couples various system components including system memory 206 such as read only memory (ROM) 208 and random-access memory (RAM) 210 to processor 202. Processors disclosed herein may all be forms of this processor 202. Information handling system 131 may include a cache 212 of high-speed memory connected directly with, in close proximity to, or integrated as part of processor 202. Information handling system 131 copies data from memory 206 and/or storage device 214 to cache 212 for quick access by processor 202. In this way, cache 212 provides a performance boost that avoids processor 202 delays while waiting for data. These and other

modules may control or be configured to control processor 202 to perform various operations or actions. Other system memory 206 may be available for use as well. Memory 206 may include multiple different types of memory with different performance characteristics. It may be appreciated that the disclosure may operate on information handling system 131 with more than one processor 202 or on a group or cluster of computing devices networked together to provide greater processing capability. Processor 202 may include any general-purpose processor and a hardware module or software module, such as first module 216, second module 218, and third module 220 stored in storage device 214, configured to control processor 202 as well as a special-purpose processor where software instructions are incorporated into processor 202. Processor 202 may be a self-contained computing system, containing multiple cores or processors, a bus, memory controller, cache, etc. A multi-core processor may be symmetric or asymmetric. Processor 202 may include multiple processors, such as a system having multiple, physically separate processors in different sockets, or a system having multiple processor cores on a single physical chip. Similarly, processor 202 may include multiple distributed processors located in multiple separate computing devices but working together such as via a communications network. Multiple processors or processor cores may share resources such as memory 206 or cache 212 or may operate using independent resources. Processor 202 may include one or more state machines, an application specific integrated circuit (ASIC), or a programmable gate array (PGA) including a field PGA (FPGA).

[0035] Each individual component discussed above may be coupled to system bus 204, which may connect each and every individual component to each other. System bus 204 may be any of several types of bus structures including a memory bus or memory controller, a peripheral bus, and a local bus using any of a variety of bus architectures. A basic input/output (BIOS) stored in ROM 208 or the like, may provide the basic routine that helps to transfer information between elements within information handling system 131, such as during start-up. Information handling system 131 further includes storage devices 214 or computer-readable storage media such as a hard disk drive, a magnetic disk drive, an optical disk drive, tape drive, solid-state drive, RAM drive, removable storage devices, a redundant array of inexpensive disks (RAID), hybrid storage device, or the like. Storage device 214 may include software modules 216, 218, and 220 for controlling processor 202. Information handling system 131 may include other hardware or software modules. Storage device 214 is connected to the system bus 204 by a drive interface. The drives and the associated computer-readable storage devices provide nonvolatile storage of computer-readable instructions, data structures, program modules and other data for information handling system 131. In one aspect, a hardware module that performs a particular function includes the software component

stored in a tangible computer-readable storage device in connection with the necessary hardware components, such as processor 202, system bus 204, and so forth, to carry out a particular function. In another aspect, the system may use a processor and computer-readable storage device to store instructions which, when executed by the processor, cause the processor to perform operations, a method or other specific actions. For example, the hybrid data generator, which may include a Large Language Model or other models derived from machine learning and deep learning algorithms, may include computational instructions which may be executed on a processor to generate an initial and/or an updated cement job design. In some examples, the deep learning algorithms may include convolutional neural networks, long short term memory networks, recurrent neural networks, generative adversarial networks, attention neural networks, zero-shot models, fine-tuned models, domain-specific models, multi-modal models, transformer architectures, radial basis function networks, multilayer perceptrons, self-organizing maps, deep belief networks, and combinations thereof. The basic components and appropriate variations may be modified depending on the type of device, such as whether information handling system 131 is a small, handheld computing device, a desktop computer, or a computer server. When processor 202 executes instructions to perform “operations”, processor 202 may perform the operations directly and/or facilitate, direct, or cooperate with another device or component to perform the operations.

[0036] As illustrated, information handling system 131 employs storage device 214, which may be a hard disk or other types of computer-readable storage devices which may store data that are accessible by a computer, such as magnetic cassettes, flash memory cards, digital versatile disks (DVDs), cartridges, random access memories (RAMs) 210, read only memory (ROM) 208, a cable containing a bit stream and the like, may also be used in the exemplary operating environment. Tangible computer-readable storage media, computer-readable storage devices, or computer-readable memory devices, expressly exclude media such as transitory waves, energy, carrier signals, electromagnetic waves, and signals per se.

[0037] To enable user interaction with information handling system 131, an input device 222 represents any number of input mechanisms, such as a microphone for speech, a touch-sensitive screen for gesture or graphical input, keyboard, mouse, motion input, speech and so forth. An output device 224 may also be one or more of a number of output mechanisms known to those of skill in the art. In some instances, multimodal systems enable a user to provide multiple types of input to communicate with information handling system 131. Communications interface 226 generally governs and manages the user input and system output. There is no restriction on

operating on any particular hardware arrangement and therefore the basic hardware depicted may easily be substituted for improved hardware or firmware arrangements as they are developed.

[0038] As illustrated, each individual component describe above is depicted and disclosed as individual functional blocks. The functions these blocks represent may be provided through the use of either shared or dedicated hardware, including, but not limited to, hardware capable of executing software and hardware, such as a processor 202, that is purpose-built to operate as an equivalent to software executing on a general-purpose processor. For example, the functions of one or more processors presented in FIG. 2 may be provided by a single shared processor or multiple processors. (Use of the term “processor” should not be construed to refer exclusively to hardware capable of executing software.) Illustrative examples may include microprocessor and/or digital signal processor (DSP) hardware, read-only memory (ROM) 208 for storing software performing the operations described below, and random-access memory (RAM) 210 for storing results. Very large-scale integration (VLSI) hardware examples, as well as custom VLSI circuitry in combination with a general-purpose DSP circuit, may also be provided.

[0039] Cement slurries and compositions described herein may generally include a hydraulic cement and water. A variety of hydraulic cements may be utilized in accordance with the present disclosure, including, but not limited to, those comprising calcium, aluminum, silicon, oxygen, iron, and/or sulfur, which set and harden by reaction with water. Suitable hydraulic cements may include, but are not limited to, Portland cements, pozzolana cements, gypsum cements, high alumina content cements, silica cements, and any combination thereof. In certain examples, the hydraulic cement may include a Portland cement. In some examples, the Portland cements may include Portland cements that are classified as Classes A, C, H, and G cements according to American Petroleum Institute, *API Specification for Materials and Testing for Well Cements*, API Specification 10, Fifth Ed., July 1, 1990. In addition, hydraulic cements may include cements classified by American Society for Testing and Materials (ASTM) in C150 (Standard Specification for Portland Cement), C595 (Standard Specification for Blended Hydraulic Cement) or C1157 (Performance Specification for Hydraulic Cements) such as those cements classified as ASTM Type I, II, or III. The hydraulic cement may be included in the cement slurry in any amount suitable for a particular composition. Without limitation, the hydraulic cement may be included in the cement slurries in an amount in the range of from about 10% to about 80% by weight of dry blend in the cement slurry. For example, the hydraulic cement may be present in an amount ranging between any of and/or including any of about 10%, about 15%, about 20%, about 25%, about 30%, about 35%, about 40%, about 45%, about 50%, about 55%, about 60%, about 65%, about 70%, about 75%, or about 80% by weight of the cement slurries.

[0040] The water may be from any source provided that it does not contain an excess of compounds that may undesirably affect other components in the cement slurries. For example, a cement slurry may include fresh water or saltwater. Saltwater generally may include one or more dissolved salts therein and may be saturated or unsaturated as desired for a particular application. Seawater or brines may be suitable for use in some examples. Further, the water may be present in an amount sufficient to form a pumpable slurry. In certain examples, the water may be present in the cement slurry in an amount in the range of from about 33% to about 200% by weight of the cementitious materials. For example, the water cement may be present in an amount ranging between any of and/or including any of about 33%, about 50%, about 75%, about 100%, about 125%, about 150%, about 175%, or about 200% by weight of the cementitious materials. The cementitious materials referenced may include all components which contribute to the compressive strength of the cement slurry such as the hydraulic cement and supplementary cementitious materials, for example.

[0041] As mentioned above, the cement slurry may include supplementary cementitious materials. The supplementary cementitious material may be any material that contributes to the desired properties of the cement slurry. Some supplementary cementitious materials may include, without limitation, fly ash, blast furnace slag, silica fume, pozzolans, kiln dust, and clays, for example.

[0042] The cement slurry may include kiln dust as a supplementary cementitious material. “Kiln dust,” as that term is used herein, refers to a solid material generated as a by-product of the heating of certain materials in kilns. The term “kiln dust” as used herein is intended to include kiln dust made as described herein and equivalent forms of kiln dust. Depending on its source, kiln dust may exhibit cementitious properties in that it can set and harden in the presence of water. Examples of suitable kiln dusts include cement kiln dust, lime kiln dust, and combinations thereof. Cement kiln dust may be generated as a by-product of cement production that is removed from the gas stream and collected, for example, in a dust collector. Usually, large quantities of cement kiln dust are collected in the production of cement that are commonly disposed of as waste. The chemical analysis of the cement kiln dust from various cement manufactures varies depending on a number of factors, including the particular kiln feed, the efficiencies of the cement production operation, and the associated dust collection systems. Cement kiln dust generally may include a variety of oxides, such as SiO_2 , Al_2O_3 , Fe_2O_3 , CaO , MgO , SO_3 , Na_2O , and K_2O . The chemical analysis of lime kiln dust from various lime manufacturers varies depending on several factors, including the particular limestone or dolomitic limestone feed, the type of kiln, the mode of operation of the kiln, the efficiencies of the lime production operation, and the associated dust collection systems. Lime kiln dust generally may include varying amounts of free lime and free magnesium, limestone,

and/or dolomitic limestone and a variety of oxides, such as SiO_2 , Al_2O_3 , Fe_2O_3 , CaO , MgO , SO_3 , Na_2O , and K_2O , and other components, such as chlorides. A cement kiln dust may be added to the cement slurry prior to, concurrently with, or after activation. Cement kiln dust may include a partially calcined kiln feed which is removed from the gas stream and collected in a dust collector during the manufacture of cement. The chemical analysis of CKD from various cement manufactures varies depending on several factors, including the particular kiln feed, the efficiencies of the cement production operation, and the associated dust collection systems. CKD generally may comprise a variety of oxides, such as SiO_2 , Al_2O_3 , Fe_2O_3 , CaO , MgO , SO_3 , Na_2O , and K_2O . The CKD and/or lime kiln dust may be included in examples of the cement slurry in an amount suitable for a particular application.

[0043] In some examples, the cement slurry may further include one or more of slag, natural glass, shale, amorphous silica, or metakaolin as a supplementary cementitious material. Slag is generally a granulated, blast furnace by-product from the production of cast iron including the oxidized impurities found in iron ore. The cement may further include shale. A variety of shales may be suitable, including those including silicon, aluminum, calcium, and/or magnesium. Examples of suitable shales include vitrified shale and/or calcined shale. In some examples, the cement slurry may further include amorphous silica as a supplementary cementitious material. Amorphous silica is a powder that may be included in embodiments to increase cement compressive strength. Amorphous silica is generally a byproduct of a ferrosilicon production process, wherein the amorphous silica may be formed by oxidation and condensation of gaseous silicon suboxide, SiO , which is formed as an intermediate during the process.

[0044] In some examples, the cement slurry may further include a variety of fly ashes as a supplementary cementitious material which may include fly ash classified as Class C, Class F, or Class N fly ash according to American Petroleum Institute, API Specification for Materials and Testing for Well Cements, API Specification 10, Fifth Ed., July 1, 1990. In some examples, the cement slurry may further include zeolites as supplementary cementitious materials. Zeolites are generally porous alumino-silicate minerals that may be either natural or synthetic. Synthetic zeolites are based on the same type of structural cell as natural zeolites and may comprise aluminosilicate hydrates. As used herein, the term “zeolite” refers to all natural and synthetic forms of zeolite.

[0045] Where used, one or more of the aforementioned supplementary cementitious materials may be present in the cement slurry. For example, without limitation, one or more supplementary cementitious materials may be present in an amount of about 0.1% to about 80% by weight of the cement slurry. For example, the supplementary cementitious materials may be present in an

amount ranging between any of and/or including any of about 0.1%, about 10%, about 20%, about 30%, about 40%, about 50%, about 60%, about 70%, or about 80% by weight of the cement.

[0046] In some examples, the cement slurry may further include hydrated lime. As used herein, the term “hydrated lime” will be understood to mean calcium hydroxide. In some embodiments, the hydrated lime may be provided as quicklime (calcium oxide) which hydrates when mixed with water to form the hydrated lime. The hydrated lime may be included in examples of the cement slurry, for example, to form a hydraulic composition with the supplementary cementitious components. For example, the hydrated lime may be included in a supplementary cementitious material-to-hydrated-lime weight ratio of about 10:1 to about 1:1 or 3:1 to about 5:1. Where present, the hydrated lime may be included in the set cement slurry in an amount in the range of from about 10% to about 100% by weight of the cement slurry, for example. In some examples, the hydrated lime may be present in an amount ranging between any of and/or including any of about 10%, about 20%, about 40%, about 60%, about 80%, or about 100% by weight of the cement slurry. In some examples, the cementitious components present in the cement slurry may consist essentially of one or more supplementary cementitious materials and the hydrated lime. For example, the cementitious components may primarily comprise the supplementary cementitious materials and the hydrated lime without any additional components (e.g., Portland cement, fly ash, slag cement) that hydraulically set in the presence of water.

[0047] Lime may be present in the cement slurry in several forms, including as calcium oxide and or calcium hydroxide or as a reaction product such as when Portland cement reacts with water. Alternatively, lime may be included in the cement slurry by amount of silica in the cement slurry. A cement slurry may be designed to have a target lime to silica weight ratio. The target lime to silica ratio may be a molar ratio, molal ratio, or any other equivalent way of expressing a relative amount of silica to lime. Any suitable target lime to silica weight ratio may be selected including from about 10/90 lime to silica by weight to about 40/60 lime to silica by weight. Alternatively, about 10/90 lime to silica by weight to about 20/80 lime to silica by weight, about 20/80 lime to silica by weight to about 30/70 lime to silica by weight, or about 30/70 lime to silica by weight to about 40/63 lime to silica by weight.

[0048] Other additives suitable for use in subterranean cementing operations also may be included in embodiments of the cement slurry. Examples of such additives include, but are not limited to: weighting agents, lightweight additives, gas-generating additives, mechanical-property-enhancing additives, lost-circulation materials, filtration-control additives, fluid-loss-control additives, defoaming agents, foaming agents, thixotropic additives, and combinations thereof. In embodiments, one or more of these additives may be added to the cement slurry after storing but

prior to the placement of a cement slurry into a subterranean formation. In some examples, the cement slurry may further include a dispersant. Examples of suitable dispersants include, without limitation, sulfonated-formaldehyde-based dispersants (e.g., sulfonated acetone formaldehyde condensate) or polycarboxylated ether dispersants. In some examples, the dispersant may be included in the cement slurry in an amount in the range of from about 0.01% to about 5% by weight of the cementitious materials. In specific examples, the dispersant may be present in an amount ranging between any of and/or including any of about 0.01%, about 0.1%, about 0.5%, about 1%, about 2%, about 3%, about 4%, or about 5% by weight of the cementitious materials.

[0049] In some examples, the cement slurry may further include a set retarder. A broad variety of set retarders may be suitable for use in the cement slurries. For example, the set retarder may comprise phosphonic acids, such as ethylenediamine tetra(methylene phosphonic acid), diethylenetriamine penta(methylene phosphonic acid), etc.; lignosulfonates, such as sodium lignosulfonate, calcium lignosulfonate, etc.; salts such as stannous sulfate, lead acetate, monobasic calcium phosphate, organic acids, such as citric acid, tartaric acid, etc.; cellulose derivatives such as hydroxyl ethyl cellulose (HEC) and carboxymethyl hydroxyethyl cellulose (CMHEC); synthetic co- or ter-polymers comprising sulfonate and carboxylic acid groups such as sulfonate-functionalized acrylamide-acrylic acid co-polymers; borate compounds such as alkali borates, sodium metaborate, sodium tetraborate, potassium pentaborate; derivatives thereof, or mixtures thereof. Examples of suitable set retarders include, among others, phosphonic acid derivatives. Generally, the set retarder may be present in the cement slurry in an amount sufficient to delay the setting for a desired time. In some examples, the set retarder may be present in the cement slurry in an amount in the range of from about 0.01% to about 10% by weight of the cementitious materials. In specific examples, the set retarder may be present in an amount ranging between any of and/or including any of about 0.01%, about 0.1%, about 1%, about 2%, about 4%, about 6%, about 8%, or about 10% by weight of the cementitious materials.

[0050] In some examples, the cement slurry may further include an accelerator. A broad variety of accelerators may be suitable for use in the cement slurries. For example, the accelerator may include, but are not limited to, aluminum sulfate, alums, calcium chloride, calcium nitrate, calcium nitrite, calcium formate, calcium sulfoaluminate, calcium sulfate, gypsum-hemihydrate, sodium aluminate, sodium carbonate, sodium chloride, sodium silicate, sodium sulfate, ferric chloride, or a combination thereof. In some examples, the accelerators may be present in the cement slurry in an amount in the range of from about 0.01% to about 10% by weight of the cementitious materials. In specific examples, the accelerators may be present in an amount ranging between any of and/or

including any of about 0.01%, about 0.1%, about 1%, about 2%, about 4%, about 6%, about 8%, or about 10% by weight of the cementitious materials.

[0051] Cement slurries generally should have a density suitable for a particular application. By way of example, the cement slurry may have a density in the range of from about 8 pounds per gallon (“ppg”) (959 kg/m³) to about 20 ppg (2397 kg/m³), or about 8 ppg to about 12 ppg (1437 kg/m³), or about 12 ppg to about 16 ppg (1917.22 kg/m³), or about 16 ppg to about 20 ppg, or any ranges therebetween. Examples of the cement slurry may be foamed or unfoamed or may comprise other means to reduce their densities, such as hollow microspheres, low-density elastic beads, or other density-reducing additives known in the art.

[0052] The cement slurries disclosed herein may be used in a variety of subterranean applications, including primary and remedial cementing. The cement slurries may be introduced into a subterranean formation and allowed to set. In primary cementing applications, for example, the cement slurries may be introduced into the annular space between a conduit located in a wellbore and the walls of the wellbore (and/or a larger conduit in the wellbore), wherein the wellbore penetrates the subterranean formation. The cement slurry may be allowed to set in the annular space to form an annular sheath of hardened cement. The cement slurry may form a barrier that prevents the migration of fluids in the wellbore. The cement slurry may also, for example, support the conduit in the wellbore. In remedial cementing applications, the cement slurry may be used, for example, in squeeze cementing operations or in the placement of cement plugs. By way of example, the cement slurry may be placed in a wellbore to plug an opening (e.g., a void or crack) in the formation, in a gravel pack, in the conduit, in the cement sheath, and/or between the cement sheath and the conduit (e.g., a micro annulus).

[0053] FIG. 3 illustrates an example information handling system 131 having a chipset architecture that may be used in executing the described method and generating and displaying a graphical user interface (GUI). Information handling system 131 is an example of computer hardware, software, and firmware that may be used to implement the disclosed technology. Information handling system 131 may include a processor 202, representative of any number of physically and/or logically distinct resources capable of executing software, firmware, and hardware configured to perform identified computations. Processor 202 may communicate with a chipset 300 that may control input to and output from processor 202. In this example, chipset 300 outputs information to output device 224, such as a display, and may read and write information to storage device 214, which may include, for example, magnetic media, and solid-state media. Chipset 300 may also read data from and write data to RAM 210. A bridge 302 for interfacing with a variety of user interface components 304 may be provided for interfacing with chipset 300.

Such user interface components 304 may include a keyboard, a microphone, touch detection and processing circuitry, a pointing device, such as a mouse, and so on. In general, inputs to information handling system 131 may come from any of a variety of sources including machine generated and/or human generated.

[0054] Chipset 300 may also interface with one or more communication interfaces 226 that may have different physical interfaces. Such communication interfaces may include interfaces for wired and wireless local area networks, for broadband wireless networks, as well as personal area networks. Some applications of the methods for generating, displaying, and using the GUI disclosed herein may include receiving ordered datasets over the physical interface or be generated by the machine itself by processor 202 analyzing data stored in storage device 214 or RAM 210. Further, information handling system 131 may receive one or more inputs from a user via user interface components 304 and execute appropriate functions, such as browsing functions by interpreting these inputs using processor 202.

[0055] In examples, information handling system 131 may also include tangible and/or non-transitory computer-readable storage devices for carrying or having computer-executable instructions or data structures stored thereon. Such tangible computer-readable storage devices may be any available device that may be accessed by a general purpose or special purpose computer, including the functional design of any special purpose processor as described above. By way of example, and not limitation, such tangible computer-readable devices may include RAM, ROM, EEPROM, CD-ROM or other optical disk storage, magnetic disk storage or other magnetic storage devices, or any other device which may be used to carry or store desired program code in the form of computer-executable instructions, data structures, or processor chip design. When information or instructions are provided via a network, or another communications connection (either hardwired, wireless, or combination thereof), to a computer, the computer properly views the connection as a computer-readable medium. Thus, any such connection is properly termed a computer-readable medium. Combinations of the above should also be included within the scope of the computer-readable storage devices.

[0056] Computer-executable instructions include, for example, instructions and data which cause a general-purpose computer, special purpose computer, or special purpose processing device to perform a certain function or group of functions. Computer-executable instructions also include program modules that are executed by computers in stand-alone or network environments. Generally, program modules include routines, programs, components, data structures, objects, and the functions inherent in the design of special-purpose processors, etc. that perform particular tasks or implement particular abstract data types. Computer-executable instructions, associated data

structures, and program modules represent examples of the program code means for executing steps of the methods disclosed herein. The particular sequence of such executable instructions or associated data structures represents examples of corresponding acts for implementing the functions described in such steps.

[0057] In additional examples, methods may be practiced in network computing environments with many types of computer system configurations, including personal computers, hand-held devices, multi-processor systems, microprocessor-based or programmable consumer electronics, network PCs, minicomputers, mainframe computers, and the like. Examples may also be practiced in distributed computing environments where tasks are performed by local and remote processing devices that are linked (either by hardwired links, wireless links, or by a combination thereof) through a communications network. In a distributed computing environment, program modules may be located in both local and remote memory storage devices.

[0058] During cementing operations, information handling system 131 may process different types of the real time data originated from varied sampling rates and various sources, such as diagnostics data, sensor measurements, operations data, and/or the like. These one or more measurements from surface equipment and/or bottom hole assembly may allow for information handling system 131 to perform real-time health assessment of the cementing operation. In some examples, the foregoing one or more measurements may be utilized to generate an updated cement job design when the one or more measurements are supplied to the hybrid data generator. Cementing tools and equipment may further comprise a variety of sensors which may be able to provide one or more real-time measurements and data relevant to pumping the cement into the wellbore in adherence to a well plan such as rate and volume of the fluids pumped. In some examples this cementing equipment may include cementing trucks, cementing pumps, recirculating cementing mixers, liquid additive pumps, rigs, gyroscopes, accelerometers, magnetometers, and other wellbore cementing tools and equipment. In the context of cementing operations, “real-time,” may be construed as monitoring, gathering, assessing, and/or utilizing data contemporaneously with the execution of the cementing operation. Real-time operations may further comprise modifying the initial design or execution of the planned operation in order to modify a well plan of a cementing operation. In some examples, the modifications to the cementing operation may occur through automated or semi-automated processes. An example of an automated cementing process may include relaying or downlinking a set of operational commands (control commands) to an RSS in order to modify a cementing operation to achieve a certain objective. In other examples, operational commands (control commands), which may be derived from an initial or an updated cementing program may be automatically relayed surface equipment.

In other examples, the operational commands (control commands) may be relayed to the rig personnel for review prior to implementation. In some examples, one or more cementing objectives and operational features may be incorporated into the cementing operation through the utilization of a cost function. In further examples, the cost function may be optimized for one or more operational features including but not limited to fluid displacement, satisfying physical property requirements of the cement slurry including rheology and density, ensuring that the cement slurry is mixable and pumpable, ensuring that the cement slurry has correct transient properties such as thickening time, satisfying physical property requirements of the set cement, satisfying chemical requirements of the cement slurry and other fluids such as displacement fluids, ensuring compatibility between subsequently pumped fluids, maximizing hole stability, minimizing total cementing cost, minimizing cost per wellbore section, minimizing time spent on each wellbore section, and combinations thereof.

[0059] FIG. 4 illustrates an example of one arrangement of resources in a computing network 400 that may employ the processes and techniques described herein, although many others are of course possible. As noted above, an information handling system 131, as part of their function, may utilize data, which includes files, directories, metadata (e.g., access control list (ACLS) creation/edit dates associated with the data, etc.), and other data objects. The data on the information handling system 131 is typically a primary copy (e.g., a production copy). During a copy, backup, archive or other storage operation, information handling system 131 may send a copy of some data objects (or some components thereof) to a secondary storage computing device 404 by utilizing one or more data agents 402.

[0060] A data agent 402 may be a desktop application, website application, or any software-based application that is run on information handling system 131. As illustrated, information handling system 131 may be disposed at any rig site (e.g., referring to FIG. 1a) or repair and manufacturing center. The data agent may communicate with a secondary storage computing device 404 using communication protocol 408 in a wired or wireless system. The communication protocol 408 may function and operate as an input to a website application. In the website application, field data related to pre- and post-operations, generated DTCs, notes, and the like may be uploaded. Additionally, information handling system 131 may utilize communication protocol 408 to access processed measurements, operations with similar DTCs, troubleshooting findings, historical run data, and/or the like. This information is accessed from secondary storage computing device 404 by data agent 402, which is loaded on information handling system 131.

[0061] Secondary storage computing device 404 may operate and function to create secondary copies of primary data objects (or some components thereof) in various cloud storage sites 406A-

N. Additionally, secondary storage computing device 404 may run determinative algorithms on data uploaded from one or more information handling systems 131, discussed further below. Communications between the secondary storage computing devices 404 and cloud storage sites 406A-N may utilize REST protocols (Representational state transfer interfaces) that satisfy basic C/R/U/D semantics (Create/Read/Update/Delete semantics), or other hypertext transfer protocol (“HTTP”)-based or file-transfer protocol (“FTP”)-based protocols (e.g., Simple Object Access Protocol).

[0062] In conjunction with creating secondary copies in cloud storage sites 406A-N, the secondary storage computing device 404 may also perform local content indexing and/or local object-level, sub-object-level or block-level deduplication when performing storage operations involving various cloud storage sites 406A-N. Cloud storage sites 406A-N may further record and maintain DTC code logs for each downhole operation or run, map DTC codes, store repair and maintenance data, store operational data, and/or provide outputs from determinative algorithms and or models that are located in cloud storage sites 406A-N. In a non-limiting example, this type of network may be utilized as a platform to store, backup, analyze, import, and perform extract, transform and load (“ETL”) processes to the data gathered during a cementing operation. In further examples, this type of network may be utilized to execute a hybrid data generator to generate an initial and/or an updated cement job design.

[0063] As previously mentioned, the hybrid data generator may include a stack of models which are run in series, in parallel, or combinations thereof to produce a cement job design. The development of cement job designs, whether executed using a hybrid data generator or using traditional methods, may require the analysis of text-based data. Additionally, the cement job designs (e.g., an output from a hybrid data generator) themselves may include text-based data. In some examples, Large Language Models may be proficient in analyzing input provided in the form of text, while providing an output in the form of text. As such, a Large Language Model may be included in the stack of models which form the hybrid data generator. In some examples, Large Language Models may be trained on large amounts of text data including but not limited to books, technical papers, articles, previous cement job designs, previous cement job reports, documentation regarding industry best practices, documents regarding company specific work methods, documents regarding company specific trade secrets, documentation regarding industry safety standards, web-based content, emails, technical presentations, and various other forms of text-based data. In some examples, a Large Language Model algorithm may include a deep learning architecture which may be referred to as a transformer architecture. The transformer architecture may allow for a language model to perform natural language processing tasks in a

fashion that mimics human-like responses. In some examples, tasks performed by natural language processing may include text-based content creation and generation, next-word predictions in sentence construction, summarization, machine translation, application (e.g., computer-based “apps”) generation, and/or answering text-based questions with text-based responses. In further examples, large language models supported by transformer architecture may be able to learn the patterns and structures of language.

[0064] FIG. 5 illustrates one example of a transformer architecture 500 which may be executed on an information handling system (e.g., information handling system 131 in FIG. 1) which may further be able to generate a model capable of performing natural language processing tasks. Transformer architecture 500 may include block 505 where one or more inputs are provided to transformer architecture 500. In some examples, the one or more inputs provided in block 505 may be provided in text-form, numerical-form, or combinations thereof. The one or more inputs provided in block 505 may be tokenized into distinct elements which may be referred to as tokens. In some examples, a tokenized sentence is a fixed-length sentence. In further examples, the tokenized sentence may further include words and/or sub-words. However, information handling systems may not be able to understand text-based tokens and therefore, may convert or transform the text-based tokens into a numerical format which may be referred to as input embeddings. For example, the tokens may be converted to integer indices associated with a vocabulary dataset. In block 510, the one or more inputs from block 505 may be translated into input embeddings. In some examples, input embeddings may represent words in a numerical format which may be processed by a machine learning algorithm and/or a machine learning model. In further examples, the input embeddings of block 510 may place the tokenized inputs of block 505 in a mathematical space such that words are placed in proximity to each other relative to their similarity. For example, the input embeddings may create vectors to associate words of similar meaning, which may further determine the location of the tokenized inputs from block 505 in the mathematic space. As such, the inputs may be tokenized, encoded in a numerical format, and converted to input embeddings where the tokens are placed in a vector-space representation to preserve their meaning.

[0065] In block 515, positional encoding may be applied to the inputs of block 505 to construct a sequence of embedding vectors. In some examples, each vector may represent the semantics and position of each token. As such, positional encoding may include encoding the sequential location or position of each word from the one or more inputs of block 505 as a set of numbers. In some examples, providing the sequential location of the words from the input as a positional encoding may allow the transformer architecture to more effectively understand how humans construct and order sentences. Additionally, the positional encoding may benefit the transformer architecture’s

ability to generate grammatically correct sentences with semantically meaningful responses. The positional encoding, which identifies the location of each word in the inputs may further be provided to an encoder in block 520 along with the input embeddings of block 510. The encoder of block 520 may be part of a neural network that processes the input embeddings and positional encodings of block 510 and block 515. In some examples, the encoder of block 520 may generate a series of hidden states that may capture the meaning and context of the input provided in block 505. In some examples, the encoder of block 520 may generate a series of hidden states that represent the input text at multiple levels of abstraction. In some examples, multiple layers of the encoder may be utilized in a transformer architecture. Additional information about the encoder structure will be detailed below.

[0066] The outputs from the encoder of block 520, which may include an encoded output sequence, may be provided to a decoder of block 525. A decoder may be part of a neural network that processes an encoded output sequence to generate a decoded output sequence. In some examples, the decoder of 525 may be trained to learn how to guess the next word in an output sequence based on the words that preceded the word to be guessed. In further examples, multiple layers of the decoder may be utilized in a transformer architecture. In addition to the encoded output sequence provided by the encoder of block 520, the decoder of block 525 may also receive output embeddings of block 530 and positional encoding of block 535 as inputs. In some examples, the positional encoding of block 535 may include an output sequence which is shifted to the right by one position. Since the information handling systems may not be able to understand text directly, the output sequence provided by the positional encoding of block 535 may be formatted as output embeddings. In some examples, a loss function may be utilized to adjust the output embeddings and positional encoding of block 530 and block 535. During the model training process, output embeddings may compute the loss function and update the model parameters to improve the difference between the model's predictions and the actual target values (e.g., model performance). During inference, the output embeddings may generate the output text by mapping the model's predicted probabilities of each token to one or more corresponding tokens in the vocabulary. Additional information about the decoder structure will be detailed below.

[0067] The decoded output sequence determined in block 525 may be provided to a linear layer in block 540. In some examples, a linear layer may map the decoded output sequence to a higher-dimensional space which may transform the decoded output sequence to the original input space. The output created from block 540 may be provided to a softmax function in block 545 which may generate a probability distribution for each output token in a vocabulary. The softmax function of

block 545 may additionally generate output tokens with probabilities, such as the output probabilities of block 550.

[0068] As described above, the transformer architecture of FIG. 5 may include an encoder. An example schematic of the structure for an encoder may be illustrated in FIG. 6 with encoder architecture 600. The input embeddings and positional encodings in block 605 (e.g., from block 510 and block 515 of FIG. 5) may be provided to a multi-head attention in block 610. In some examples the multi-head attention may include a self-attention mechanism to enrich the embedding vectors with contextual information from the whole sentence (e.g., from the inputs). For example, depending on the proximity of the words in a sentence, the words may have more than one semantic and/or functional purpose. By utilizing the self-attention mechanism, the model may be able to assess multiple embedding subspaces. In some examples, the multi-head attention may utilize eight or more parallel attention calculations.

[0069] The outputs from the multi-head attention in block 610 may be provided as inputs to the position-wise feed-forward network of block 615. The feed-forward network of block 615 may include one or more linear layers which may further include a rectified linear unit (“ReLU”) activation function. In some examples, ReLU may introduce non-linearity to the feed-forward network to resolve a vanishing gradient issue. The position-wise feed-forward network may process each embedding vector independently with identical weights to provide further transformation of the embedding vectors. In some examples, the linear transformations may be equivalent across different positions, however they may use different parameters from a preceding linear layer to a subsequent linear layer. Encoder architecture 600 may include multiple layers of multi-attention heads and feed-forward networks, where the encoder uses residual connections and layer normalization 620. In some examples, residual connections between each layer may perform element-wise addition to carry over the previous embeddings to subsequent layers. In further examples, this may allow the encoder to enrich the embeddings vectors with additional information obtained from the outputs of the multi-head attention of block 610 and the feed-forward network of block 615. Layer normalization may also be applied after each layer, in conjunction with the application of residual connections. In some examples, layer normalization may reduce the effect of covariant shift. In some examples, reducing covariant shift may prevent migration of the mean and standard deviation of embedding vector elements. Encoder architecture 600 may only depict a single multi-head attention layer and a single position-wise feed forward layer, however, any number of multi-head attention layers and/or position-wise feed forward layers may be utilized in an encoder architecture. In such examples, the outputs from a preceding layer become the inputs to a subsequent layer. As such, multiple layers of residual connection and layer normalization may

be utilized in association with the multi-head attention layers and position-wise feed forward layers.

[0070] As described above, the transformer architecture of FIG. 5 may include a decoder. An example schematic of the structure for a decoder may be illustrated in FIG. 7 with decoder architecture 700. In some examples, the input to decoder architecture 700 may be an output from the encoder (e.g., encoder architecture 600 of FIG. 6) with a positional encoding that is shifted to the right. In some examples, decoders may be similar to encoders in that they generate enriched embeddings. In block 705, inputs may be provided to a masked multi-head attention in block 710. In some examples, the inputs of block 705 may be the outputs from the encoder with positional encoding adjusted to the right. The mask-aspect of the masked multi-head attention may hide or mask certain information provided in the inputs of block 705. In some examples, when a mask is applied to a set of inputs, the masked inputs may not be usable to the multi-head attention of masked multi-head attention in block 710. The information to be masked may be determined by position. The masked multi-attention head in block 710 may include self-attention mechanisms to enrich the non-masked embedding vectors with contextual information from the unmasked portion of the sentence. As with the decoder architecture, the outputs from the masked multi-head attention in block 710 may pass through residual connection and layer normalization in block 715. As previously described, layer normalization may reduce the effect of covariant shift. In some examples, reducing covariant shift may prevent migration of the mean and standard deviation of embedding vector elements. Additionally, residual connections between each layer may perform element-wise addition to carry over the previous embeddings to subsequent layers. The outputs from block 715 may be provided as inputs to the multi-head attention layer in block 720 which may function substantially similar to the multi-head attention layer from the encoder architecture (e.g., encoder architecture 600 in FIG. 6). The outputs from block 720 may pass through residual connection and layer normalization in block 715 to provide inputs for position-wise feed forward in block 725. The position-wise feed forward layer in block 725 may function substantially similar to the position-wise feed forward layer from the encoder architecture (e.g., encoder architecture 600 in FIG. 6) before going being passed back through residual connection and layer normalization in block 715.

[0071] The encoder of FIG. 6 and the decoder of FIG. 7 may utilize machine learning algorithms and deep learning algorithms to create a machine learning model which may be utilized in a transformer architecture. In some examples, the deep learning algorithms may include convolutional neural networks, long short term memory networks, recurrent neural networks, generative adversarial networks, attention neural networks, zero-shot models, fine-tuned models,

domain-specific models, multi-modal models, transformer architectures, radial basis function networks, multilayer perceptrons, self-organizing maps, deep belief networks, and combinations thereof. Additionally, one or more of the empirical models incorporated in the stack of models which form the hybrid data generator may be developed from a machine learning algorithm. As such, the hybrid data generator may include additional machine learning-based models aside from the Large Language Model. A machine learning model may be an empirically derived model which may result from a machine learning algorithm identifying one or more underlying relationships within a dataset. In comparison to a physics-based model, which may be derived from first principals and define the mathematical relationship of a system, a pure machine learning model may not be derived from first principals. Once a machine learning model is developed, it may be queried in order to predict one or more outcomes for a given set of inputs. The type of input data used to query the model to create the prediction may correlate both in category and type to the dataset from which the model was developed. In the case of Large Language Models, the inputs utilized to query the model may not be an exact match to the dataset on which the model is trained.

[0072] The structure of, and the data contained within a dataset provided to a machine learning algorithm may vary depending on the intended function of the resulting machine learning model. In some examples, the data provided in a dataset may contain one or more independent values. The independent values of a dataset may be referred to as “features,” and a collection of features may be referred to as a “feature space.” Additionally, datasets may contain corresponding dependent values. The dependent values may be the result or outcome associated with a set of independent values. In some examples, the dependent values may be referred to as “target values.” Although dependent values may be a necessary component of a dataset for certain algorithms, not all algorithms require a dataset with dependent values. Furthermore, both the independent and dependent values of the dataset may comprise either numerical, categorical, or text-based data.

[0073] While it may be true that machine learning model development is more successful with a larger dataset, it may also be the case that the whole dataset is not used to train the model. A test dataset may be a portion of the original dataset which is not presented to the algorithm for model training purposes. Instead, the test dataset may be used for what may be known as “model validation,” which may be a mathematical evaluation of how successfully a machine learning algorithm has learned and incorporated the underlying relationships within the original dataset into a machine learning model. This may comprise evaluating model performance according to whether the model is over-fit or under-fit. As it may be assumed that all datasets contain some level of error, it may be important to evaluate and optimize the model performance and associated model fit by means of model validation. In general, the variability in model fit (e.g.: whether a model is

over-fit or under-fit) may be described by the “bias-variance trade-off.” As an example, a model with high bias may be an under-fit model, where the developed model is over-simplified, and has either not fully learned the relationships within the dataset or has over-generalized the underlying relationships. A model with high variance may be an over-fit model which has overlearned about non-generalizable relationships within training dataset which may not be present in the test dataset. In a non-limiting example, these non-generalizable relationships may be driven by factors such as intrinsic error, data heterogeneity, and the presence of outliers within the dataset. The selected ratio of training data to test data may vary based on multiple factors, including, in a non-limiting example, the homogeneity of the dataset, the size of the dataset, the type of algorithm used, and the objective of the model. The ratio of training data to test data may also be determined by the validation method used, wherein some non-limiting examples of validation methods comprise k-fold cross-validation, stratified k-fold cross-validation, bootstrapping, leave-one-out cross-validation, resubstitution, random subsampling, and percentage hold-out.

[0074] In some examples, training Large Language Models may include a training component referred to as reinforcement learning. In further examples, reinforcement learning may utilize human input, computer input (e.g., artificial intelligence), and combinations thereof. In some examples, reinforcement learning may include querying a trained model to receive one or more responses and ranking and quality and correctness of the one or more responses provided by the model. In further examples, the responses may be approved or rejected. In some examples, the assessment of the model-developed responses may be provided to the model in order to further train the model and improve the subsequent responses.

[0075] In addition to the parameters that exist within the dataset, such as the independent and dependent variables, machine learning algorithms may also utilize parameters referred to as “hyperparameters.” Each algorithm may have an intrinsic set of hyperparameters which guide what and how an algorithm learns about the training dataset by providing limitations or operational boundaries to the underlying mathematical workflows on which the algorithm functions. Furthermore, hyperparameters may be classified as either model hyperparameters or algorithm parameters.

[0076] Model hyperparameters may guide the level of nuance with which an algorithm learns about a training dataset, and as such model hyperparameters may also impact the performance or accuracy of the model that is ultimately generated. Modifying or tuning the model hyperparameters of an algorithm may result in the generation of substantially different models for a given training dataset. In some cases, the model hyperparameters selected for the algorithm may result in the development of an over-fit or under-fit model. As such, the level to which an algorithm may learn

the underlying relationships within a dataset, including the intrinsic error, may be controlled to an extent by tuning the model hyperparameters.

[0077] Model hyperparameter selection may be optimized by identifying a set of hyperparameters which minimize a predefined loss function. An example of a loss function for a supervised regression algorithm may include the model error, wherein a selected set of hyperparameters correlates to a model which produces the lowest difference between the predictions developed by the produced model and the dependent values in the dataset. In addition to model hyperparameters, algorithm hyperparameters may also control the learning process of an algorithm, however algorithm hyperparameters may not influence the model performance. Algorithm hyperparameters may be used to control the speed and quality of the machine learning process. As such, algorithm hyperparameters may affect the computational intensity associated with developing a model from a specific dataset.

[0078] Machine learning algorithms, which may be capable of capturing the underlying relationships within a dataset, may be broken into different categories. One such category may comprise whether the machine learning algorithm functions using supervised, unsupervised, semi-supervised, or reinforcement learning. The objective of a supervised learning algorithm may be to determine one or more dependent variables based on their relationship to one or more independent variables. Supervised learning algorithms are named as such because the dataset comprises both independent and corresponding dependent values where the dependent value may be thought of as “the answer,” that the model is seeking to predict from the underlying relationships in the dataset. As such, the objective of a model developed from a supervised learning algorithm may be to predict the outcome of one or more scenarios which do not yet have a known outcome. Supervised learning algorithms may be further divided according to their function as classification and regression algorithms. When the dependent variable is a label or a categorical value, the algorithm may be referred to as a classification algorithm. When the dependent variable is a continuous numerical value, the algorithm may be a regression algorithm. In a non-limiting example, algorithms utilized for supervised learning may comprise Neural Networks, K-Nearest Neighbors, Naïve Bayes, Decision Trees, Classification Trees, Regression Trees, Random Forests, Linear Regression, Support Vector Machines (SVM), Gradient Boosting Regression, Genetic Algorithm, and Perception Back-Propagation.

[0079] The objective of unsupervised machine learning may be to identify similarities and/or differences between the data points within the dataset which may allow the dataset to be divided into groups or clusters without the benefit of knowing which group or cluster the data may belong to. Datasets utilized in unsupervised learning may not comprise a dependent variable as the

intended function of this type of algorithm is to identify one or more groupings or clusters within a dataset. In a non-limiting example, algorithms which may be utilized for unsupervised machine learning may comprise K-means clustering, K-means classification, Fuzzy C-Means, Gaussian Mixture, Hidden Markov Model, Neural Networks, and Hierarchical algorithms.

[0080] The machine learning algorithms utilized in a Large Language Model which may utilize transformer architecture as described in FIGs. 5-7 may include one or more neural network algorithms as illustrated in Figure 8. In addition to the Large Language Model, the hybrid data generator may additionally include other machine learning-based models which may work in conjunction with the Large Lange Model. For example, the hybrid data generator may include models based on Gaussian Mixture Models, Hidden Markov Models, Support Vector Machines, Principal Component Analysis (“PCA”) models built on a variety of neural networks. Examples of machine learning algorithms that fall into the category of neural networks may comprise Perceptron, Multi-Layer Perceptron, Feed Forward, Radial Basis Network, Deep Feed Forward, Recurrent Neural Network, Long Term Memory, Short Term Memory, Deep Neural Network, Gated Recurrent Unit, Auto Encoder, Variational AE, Denoising AE, Sparse AE, Markov Chain, Hopfield Network, Boltzmann Machine, Restricted Boltzmann Machine, Deep Belief Network, Deep Convolutional Network, Deconvolutional Network, Deep Convolutional Inverse Graphics Network, Generative Adversarial Network, Liquid State Machine, Extreme Learning Machine, Echo State Network, Deep Residual Network, Kohonen Network, Support Vector Machine, and Neural Turing Machine. Neural network 800 of Figure 8 may be utilized to draw a relationship between independent and dependent variables, or to identify relationships within a set of exclusively independent variables as described herein. Neural network 800 may be an artificial neural network with one or more hidden layers 802 between input layer 804 and output layer 806. As illustrated, input layer 804 may include multi-disciplinary datasets as described in the foregoing, whereas output layers 806 may include data which may further feed the model stack, or may provide outputs used to populate a cement job design. As such, the outputs from neural network 800 may provide results which are directly included in the cement job design, or may function as inputs to a subsequent model or series of models which may then provide results which may be included in the cement job design. Input data is taken by neurons 812 in first layer which then provide an output to the neurons 812 within next layer and so on which provides a final output in output layer 806. Each layer may have one or more neurons 812. The connection between two neurons 812 of successive layers may have an associated weight. The weight defines the influence of the input to the output for the next neuron 812 and eventually for the overall final output. The process of training the neural network may entail determining the suitable weights that produce a

model capable of being utilized in a hybrid data generator to generate one or more cement job design. Furthermore, building the machine learning model may be an iterative process which comprises a validation component and/or reinforcement learning, as previously mentioned. Once a model which meets one or more criterion for deployment, which in a non-limiting example may comprise achieving a certain level of accuracy, it may be incorporated into a hybrid data generator to generate one or more cement job designs. In some examples, the level of accuracy which meets the deployment criterion may range from about 50% to about 100%. Alternatively, the level of accuracy may range from about 50% to about 60%, about 60% to about 70%, about 70% to about 80%, about 80% to about 90%, or about 90% to about 100%. Finally, if the historical dataset, which may further comprise multi-disciplinary datasets, increases in size due to the acquisition of additional data, the model may be retrained to incorporate the learnings of the additional data. In some examples, data generated by a hybrid data generator may additionally be incorporated into the dataset to augment the training dataset.

[0081] The machine learning models and Large Language Models as described in the foregoing may be incorporated into the stack of models which forms a hybrid data generator which may be further described in FIG.9. A hybrid data generator 900 of FIG. 9 may include a stack of models which may be executed in series, in parallel, and combinations thereof. The stack of models in hybrid data generator 900, which may further include machine learning models and deep learning models, may receive inputs from block 905, block 910, and block 915. In some examples, the inputs supplied in block 905 and block 910 may include a dataset which may further include historical data from multi-disciplinary datasets. For example, data gathered from previously performed cementing operations may be included in a historical multi-disciplinary dataset. In some examples, the datasets included in the inputs supplied in block 905 and block 910 may include public datasets or public databases as previously described (e.g., weather and/or traffic related data). In some examples, the inputs may additionally include data from previously generated cement job designs. These cement job designs may include cement job designs generated by personnel, cement job designs generated by a hybrid data generator, and combinations thereof. In some examples, the inputs supplied in block 915 may include image data which may further include path referencing image storage, wellbore logging images, formation images from wellbore image logs, process implementation schematics, wellsite configuration schematics, seismic images, microseismic images, pdfs or text-based images from previous wellbore construction reports (e.g., daily reports or operational reports), and images of previous cement job designs.

[0082] The dataset included in the inputs of block 910 may be provided to one or more models in block 920. The models of block 920 may include physics-based models, empirically derived

models, and combinations thereof. For example, the models of block 920 may include any combination of physics-based models, physics-informed neural networks, deep learning models, machine learning models, Gaussian Mixture Models, Hidden Markov Models, Support Vector Machine Models, Principal Component Analysis Models, and combinations thereof. The outputs of block 920 may provide for inputs to block 925 where data extraction and data processing may occur. In some examples, the outputs of block 920 may undergo an Extract, Transform, and Load (“ETL”) process which may identify relevant data, process it, and perform any transformations prior to providing the data to block 930. In some examples, the ETL process may identify a subset of relevant data from a larger dataset and create a data subset for utilization in the hybrid data generator. In some examples, ETL processes may be considered a component of “data cleaning,” which may organize, partition, structure, and standardize an unorganized dataset into a cohesive dataset. Generating a cohesive dataset may be beneficial if the dataset is to be utilized for machine learning and/or deep learning processes. The ETL process in block 925 may additionally utilize logic to determine how to handle null values in the dataset provided by block 920. In some examples, the ETL processes may time-align or depth-align data contained in two distinct databases which may further be processed prior to utilization in block 930.

[0083] In addition to receiving the output dataset from block 920, block 925 may also receive the inputs supplied in block 905. The same or substantially similar ETL processes which were applied to the dataset of block 920 may likewise be applied to the dataset from block 905. In some examples, the dataset provided by block 920 and the dataset provided by block 905 may be joined into a cohesive dataset prior to being supplied to block 930. In other examples, the dataset from block 905 undergo separate ETL processes from the dataset of block 920 in order to maintain two distinct datasets which may be handled separately by block 930.

[0084] In block 930, the one or more datasets received from block 925 may be provided as inputs to one or more models. As previously mentioned, block 925 may receive a cohesive dataset when the outputs from block 920 and inputs from block 905 are joined into a single dataset. However, block 930 may additionally receive separate datasets from block 925, where the ETL processes on block 920 and block 905 are performed separately. The models of block 930 may include physics-based models, empirically derived models, and combinations thereof. For example, the models of block 920 may include any combination of physics-based models, physics-informed neural networks, deep learning models, machine learning models, Gaussian Mixture Models, Hidden Markov Models, Support Vector Machine Models, Principal Component Analysis Models, and combinations thereof.

[0085] In some examples, the outputs of block 930 may provide for inputs to block 945 where results from the models in block 930 may be displayed in a multitude of formats including numerical, graphical, and image. The outputs from block 945 may further include reports, graphs, images, further data analytics, and data extraction. For example, the outputs from block 930 may provide results from one or more models including physics-based models and empirical models. In further examples, the results of these models may be analyzed, reviewed, and validated in block 945 to ensure that the model outputs track with an expected output.

[0086] In some examples, the outputs of block 930 may additionally provide for inputs to block 935 where the dataset may undergo additional data extraction and data processing. In some examples, the outputs of block 920 may undergo an Extract, Transform, and Load (“ETL”) process which may identify relevant data, process it, and perform any transformations prior to providing the data to a Large Language Model in block 940. In some examples, the ETL process may identify a subset of relevant data from a larger dataset and create a data subset for utilization in the hybrid data generator. In some examples, ETL processes may be considered a component of “data cleaning,” which may organize, partition, structure, and standardize an unorganized dataset into a cohesive dataset. Generating a cohesive dataset may be beneficial if the dataset is to be utilized for machine learning and/or deep learning processes. The ETL process in block 935 may additionally utilize logic to determine how to handle any null values in the dataset provided by block 930.

[0087] As previously mentioned, inputs in image-format or video-format may be received by hybrid data generator 900 in block 915. The one or more images of block 915 may be inputs to a computer vision model in block 950. In some examples, the image inputs may include path referencing image storage, wellbore logging images, formation images from wellbore image logs, process implementation schematics, wellsite configuration schematics, seismic images, microseismic images, portable document format files (“PDF”) or text-based images from previous wellbore construction reports (e.g., daily reports or operational reports), and images of previous cement job designs. In some examples, a computer vision model may be able to translate visual data based on features and contextual information contained in the visual data. For example, computer vision models may be able to identify or detect objects, visual elements, or visual features within an image. In further examples, computer vision model algorithms may rely on convolutional neural networks and a multi-layered architecture along with a training dataset to construct the computer vision model. The output from block 950 may be provided as an input to block 935 where it may undergo data extraction and data processing. The output from the data extraction and data processing of block 935 may then be provided as an input to the Large Language Model of block 940.

[0088] As mentioned in the foregoing, the outputs from block 935 which may include be numerical outputs created by the one or more models of block 930, and the image related outputs created in block 950, may be supplied as inputs to the Large Language Model of block 940. In addition to these inputs, one or more text inputs of block 955 may be provided to the Large Language Model of block 940. These text inputs may include questions which may further include prompt engineering. In some examples, prompt engineering may include crafting a textual input for a Large Language Model with consideration for appropriate words, phrases, sentence structures, and problem formulation. The text inputs may also be crafted to benefit problem formulation which may further defining a problem for the Large Language Model to solve with consideration for delineating the problem focus, scope, and boundaries. The inputs provided to the Large Language Model of block 940 may be any combination or multiplicity of outputs from block 935 and block 955. The outputs from block 940 may feed into block 960 to form a final result which may include reports, graphs, images, data analytics, data extraction, and combinations thereof. In some examples, the reports of block 960 may include an initial cement job design or an updated cement job design.

[0089] One example of where an updated cement job design may be generated is when new data is acquired which was not present in the training dataset (e.g., the historical multi-disciplinary dataset). In further examples, new data may be obtained during cementing operations on a given well or on an offset well. In some examples, the outputs of block 930, which may include numerical outputs from physics-based and machine learning-based models may be further provided to a Generative Adversarial Network and/or a Large Language Model in block 965. In further examples, the models of block 965 may be trained and utilized for making or updating decisions during real-time operations. The models developed in block 965 may be applied to one or more real-time datapoints in block 970. Similar to the other numerical inputs provided to hybrid data generator 900, the real-time data collected in block 970 may undergo an ETL process in block 925 before being utilized by the stacked models of hybrid data generator 900.

[0090] The proposed methods and systems may make use of multi-disciplinary datasets in conjunction with a hybrid generator to create one or more cement job designs in a faster and more efficient manner than previous processes have allowed for. In further examples, changes experienced during a cementing operation could be incorporated into an updated cement job design with a decreased turn-around time. For example, the overall capability to access multitudes of information 'on the fly' or in real-time by simply querying a hybrid data generator which incorporates some level of predictive analytics allows for better planning, increased vigilance and may result in better efficiency in cementing operations. In some examples, data which may not

have previously been incorporated into a cement job design could be included. In some examples, the addition of new data sources may result in an improved cement job design. In some examples, the utilization of hybrid data generators may improve knowledge sharing among various teams by connecting and/or incorporating previously disconnected data sources. For example, incorporating data gathered from previous wells and/or a variety of teams, which may not have previously been in communication, may allow for rapid knowledge sharing which may further allow for utilization of larger datasets when developing new and revised cement job designs. In some examples, hybrid data generators may allow for the system to access databases and pick input well data based on the location (distance from upcoming location), provide weather predictions, provide logistic limitations, and incorporate local laws that might significantly impact delivery of equipment/products.

[0091] FIG. 10 is a flowchart of method 1000 for using a Large Language Model to develop a cement job design. Method 1000 begins with block 1002 where a user inputs, using natural language, into a Large Language Model prompt parameters for a cement job. For example, a prompt may include the geographical location where the cement job is to be pumped, a geological formation where the cement job is to be pumped, a coordinate location where the cement job is to be pumped, and/or the identity of the borehole the cement job is to be pumped. The prompt may further specify the type of cement job to be pumped such as cementing an intermediate casing at a particular depth. The prompt may further include parameters such as required 24-hour compressive strength or any other relevant parameter a user selects. From block 1002 the user input is sent to Large Language Model 1004, such as those Large Language Models discussed herein. During the user input step, Large Language Model 1004 may suggest further user input based on the user input. For example, an input of “design an offshore intermediate liner job” may result in the Large Language Model 1004 requesting additional details such as the geographical location of the job as well as the client for who the cement job design will be presented which would inform the Large Language Model about applicable local requirements for wellbore construction as well as client requirements for wellbore construction. Once the prompting is complete the Large Language Model 1004 generates a structured output based on the users input and passes the structured output to a cement design simulator 1006. The structured output includes a cement job design which includes representative data about the wellbore, composition of the fluid service fluid train e.g. spacer, cement, and drilling mud, centralizer location, and pump schedule including rates and volumes of each service fluid.

[0092] The structured output is in a form that cement design simulator 1006 can take as input and generate a structured output comprising the simulated element of the cement design. Simulated

elements include, for example, compressive strength, set time, rheology, density, fluid loss, fluid compatibility, and temperature resistance. Additional simulated elements include fluid flow modeling by computational fluid mechanics which may have as input fluid properties such as viscosity and yield stress of the drilling mud, spacer fluid, and cement slurry, flow rates, pressures, and the geometry of the wellbore. Additional simulated elements may include fluid train modeling to optimize the design of the fluid train, including the selection of spacer fluids and displacement volumes to determine the most effective sequence and volume of fluids to achieve desired displacement and minimize fluid channeling or contamination. Additional simulated elements include fluid interface tracking simulations to track the movement and interaction of fluid interfaces during displacement to evaluate and identify any areas of potential fluid entrapment or inadequate coverage. Another simulated element includes evaluation of displacement efficiency using a computational fluid dynamic simulation to provide quantitative data on the displacement efficiency, such as the percentage of drilling mud or spacer fluid removed from the wellbore. Additional simulated elements include sensitivity analysis simulations to determine the impact of different parameters on the displacement process such as varying flow rates, fluid properties, wellbore geometry, or operational conditions to identify optimal design parameters and mitigate potential issues. Another simulation is fluid hydraulics to predict and analyze fluid flow dynamics during the cementing process. Hydraulics modeling may consider factors such as flow rates, pressures, and fluid properties to assess the hydraulic behavior of the fluids in the wellbore to identify potential issues such as excessive pressure drops, fluid losses, and inadequate fluid distribution. Another simulation may include cement slurry rheology modeling for assessing the flow behavior and viscosity of the cement slurry under various conditions. Rheological modeling helps optimize the cement slurry formulation, considering factors such as cement additives, temperature, pressure, and shear rates. Additional simulations may include heat transfer modeling simulations to assess the temperature distribution and heat transfer within the wellbore during the cementing operation. Heat transfer simulations consider factors such as wellbore geometry, fluid properties, and heat generation rates to evaluate the thermal behavior of the system. Another simulation includes cement setting and strength development modeling which assess the chemical reactions and hydration processes of the cement slurry over time and typically consider factors such as cement composition, temperature, and pressure, to predict the setting time and strength development of the cement. Another simulated element includes centralizer selection and placement optimization. Centralization simulations are performed to evaluate the effectiveness of different centralizer designs in achieving desired casing centralization. For example, the simulations consider factors such as wellbore geometry, casing properties, centralizer

specifications, and fluid flow conditions. Centralizers may be selected based on several factors, including wellbore conditions, casing specifications, and cementing objectives as input by the user or determined by the Large Language Model 1004 where the centralizer selection process may include modeling parameters such as centralizer material, design type (e.g., bow spring, rigid, or composite), centralizer spacing, centralizer size, and placement optimization. Cement design simulator 1006 may include any suitable model such as empirical models, physics-based models, physics-informed neural networks, as well as computational methods such as numerical analysis (e.g. computational fluid dynamics (CFD)).

[0093] After the simulation is completed, cement design simulator 1006 passes the structured output back to Large Language Model 1004 where the structured output from cement design simulator 1006 is evaluated to ensure that the cement job design meets all requirements. Large Language Model 1004 can generate new cement job designs and pass the cement job designs to cement design simulator 1006 which then performs additional simulations with the new cement job designs. Finally, in block 1008, Large Language Model 1004 displays to the user the cement job design in a natural language.

[0094] The systems and methods may include any of the various features disclosed herein, including one or more of the following statements. The systems and methods may include any of the various features disclosed herein, including one or more of the following statements.

[0095] Statement 1. A method comprising: providing one or more inputs to a hybrid data generator, wherein one of the one or more inputs is based at least in part on a wellsite location, wherein the hybrid data generator comprises a large language model, and wherein the large language model is based at least in part on a machine learning algorithm; utilizing an information handling system to generate a cement job design based at least in part on the one or more inputs and the hybrid data generator; performing at least a portion of a cementing operation based at least in part on the cement job design; and collecting at least one measurement from at least one sensor during the cementing operation.

[0096] Statement 2. The method of statement 1, wherein the large language model is trained using a dataset comprising at least one type of data selected from the group consisting of engineering data, geological data, geo-mechanical data, geo-physical data, data from lab-based tests, data modelled from simulations, data modelled from empirical models, data modelled from physics-based models, data from physics-informed neural networks operational data from current cementing operations, operational data from previous cementing operations, measurements collected from current cementing operations, measurements collected from previous cementing

operations, information collected from previous cementing reports, previously created cement job design, logging data, available equipment in a given region, and combinations thereof.

[0097] Statement 3. The method of any of statements 1-2, wherein the one or more inputs further comprises at least one input selected from the group consisting of engineering data, geological data, geo-mechanical data, geo-physical data, data from lab-based tests, data modelled from simulations, data modelled from empirical models, data modelled from physics-based models, data from physics-informed neural networks, operational data from current cementing operations, operational data from previous cementing operations, measurements collected from current cementing operations, measurements collected from previous cementing operations, information collected from previous cementing reports, previously created cement job design, logging data, available equipment in a given region, and combinations thereof.

[0098] Statement 4. The method of any of statements 1-3, wherein training the large language model further comprises reinforcement learning.

[0099] Statement 5. The method of any of statements 1-4, wherein the machine learning algorithm is utilized in a transformer architecture.

[0100] Statement 6. The method of statement 5, wherein the transformer architecture includes at least one architecture component selected from the group consisting of an encoder, a decoder, and combinations thereof.

[0101] Statement 7. The method of any of statements 1-6, wherein the machine learning algorithm comprises a deep learning algorithm further comprising at least one type of algorithm selected from the group consisting of convolutional neural networks, long short term memory networks, recurrent neural networks, generative adversarial networks, attention neural networks, zero-shot models, fine-tuned models, domain-specific models, multi-modal models, transformer architectures, radial basis function networks, multilayer perceptrons, self-organizing maps, deep belief networks, and combinations thereof.

[0102] Statement 8. The method of any of statements 1-7, further comprising updating the cement job design using the at least one measurement collected from the at least one sensor during the cementing operation, wherein the at least one measurement is added to the inputs provided to the hybrid data generator.

[0103] Statement 9. The method of statement 8, wherein updating the cement job design comprises updating the cement job design using at least one method selected from the group consisting of continuously updating the cement job design, updating the cement job design at set intervals of time, updating the cement job design when manually executed, updating the cement job design when a threshold is met, or combinations thereof.

[0104] Statement 10. The method of any of statements 1-9, wherein the large language model is optimized for at least one operational feature, wherein the at least one operational feature is at least one feature selected from the group consisting of fluid displacement, centralizer location, centralizer type, fluid composition, fluid volume, fluid rate, rheology, density, mixability, pumpability, thickening time, compressive strength, set time, fluid loss, fluid compatibility, temperature resistance, compatibility between subsequently pumped fluids, maximizing hole stability, minimizing total cementing cost, minimizing cost per wellbore section, minimizing time spent on each wellbore section, operational safety, and combinations thereof.

[0105] Statement 11. The method of claim 1 further comprising collecting at least one measurement from at least one sensor during the cementing operation.

[0106] Statement 12. A system comprising: a hybrid data generator comprising a large language model, wherein the large language model is based at least in part on a machine learning algorithm; an information handling system configured to execute the hybrid data generator to generate a cement job design, wherein the generated cement job design is based at least in part on one or more inputs and wherein at least one of the one or more inputs is based at least in part on a wellsite location; and a sensor in communication with the information handling system, wherein the sensor measures at least one measurement during a cementing operation.

[0107] Statement 13. The system of statement 12, wherein the large language model is trained using a dataset comprising at least one type of data selected from the group consisting of engineering data, geological data, geo-mechanical data, geo-physical data, data from lab-based tests, data modelled from simulations, data modelled from empirical models, data modelled from physics-based models, operational data from current cementing operations, operational data from previous cementing operations, measurements collected from current cementing operations, measurements collected from previous cementing operations, information collected from previous cementing reports, previously created cement job design, logging data, available equipment in a given region, and combinations thereof.

[0108] Statement 14. The system of any of statements 12-13, wherein the one or more inputs further comprises at least one input selected from the group consisting of engineering data, geological data, geo-mechanical data, geo-physical data, data from lab-based tests, data modelled from simulations, data modelled from empirical models, data modelled from physics-based models, operational data from current cementing operations, operational data from previous cementing operations, measurements collected from current cementing operations, measurements collected from previous cementing operations, information collected from previous cementing

reports, previously created cement job design, logging data, available equipment in a given region, and combinations thereof.

[0109] 15. The system of any of statements 12-14, wherein training the large language model further comprises reinforcement learning.

[0110] 16. The system of any of statements 12-15, wherein the machine learning algorithm is utilized in a transformer architecture and wherein the transformer architecture includes at least one architecture component selected from the group consisting of an encoder, a decoder, and combinations thereof.

[0111] 17. The system of any of statements 12-16, wherein the machine learning algorithm comprises a deep learning algorithm further comprising at least one type of algorithm selected from the group consisting of convolutional neural networks, long short term memory networks, recurrent neural networks, generative adversarial networks, attention neural networks, zero-shot models, fine-tuned models, domain-specific models, multi-modal models, transformer architectures, radial basis function networks, multilayer perceptrons, self-organizing maps, deep belief networks, and combinations thereof.

[0112] 18. The system of any of statements 12-17, wherein the information handling system is configured to update the cement job design based at least in part on the one measurement collected by the sensor during the cementing operation.

[0113] 19. The system of any of statements 12-18, wherein the information handling system is configured to update the cement job design using at least one method selected from the group consisting of continuously updating the cement job design, updating the cement job design at set intervals, updating the cement job design when manually executed, updating the cement job design when a threshold is met, or combinations thereof.

[0114] 20. The system of any of statements 12-19, wherein the large language model is optimized for at least one operational feature, wherein the at least one operational feature is at least one feature selected from the group consisting of fluid displacement, centralizer location, centralizer type, fluid composition, fluid volume, fluid rate, rheology, density, mixability, pumpability, thickening time, compressive strength, set time, fluid loss, fluid compatibility, temperature resistance, compatibility between subsequently pumped fluids, maximizing hole stability, minimizing total cementing cost, minimizing cost per wellbore section, minimizing time spent on each wellbore section, operational safety, and combinations thereof.

[0115] Although the present disclosure and its advantages have been described in detail, it should be understood that various changes, substitutions and alterations may be made herein without departing from the spirit and scope of the disclosure as defined by the appended claims. The

preceding description provides various examples of the systems and methods of use disclosed herein which may contain different method steps and alternative combinations of components. It should be understood that, although individual examples may be discussed herein, the present disclosure covers all combinations of the disclosed examples, including, without limitation, the different component combinations, method step combinations, and properties of the system. It should be understood that the compositions and methods are described in terms of “comprising,” “containing,” or “including” various components or steps, the compositions and methods can also “consist essentially of” or “consist of” the various components and steps. Moreover, the indefinite articles “a” or “an,” as used in the claims, are defined herein to mean one or more than one of the element that it introduces.

[0116] For the sake of brevity, only certain ranges are explicitly disclosed herein. However, ranges from any lower limit may be combined with any upper limit to recite a range not explicitly recited, as well as, ranges from any lower limit may be combined with any other lower limit to recite a range not explicitly recited, in the same way, ranges from any upper limit may be combined with any other upper limit to recite a range not explicitly recited. Additionally, whenever a numerical range with a lower limit and an upper limit is disclosed, any number and any included range falling within the range are specifically disclosed. In particular, every range of values (of the form, “from about a to about b,” or, equivalently, “from approximately a to b,” or, equivalently, “from approximately a-b”) disclosed herein is to be understood to set forth every number and range encompassed within the broader range of values even if not explicitly recited. Thus, every point or individual value may serve as its own lower or upper limit combined with any other point or individual value or any other lower or upper limit, to recite a range not explicitly recited.

[0117] Therefore, the present examples are well adapted to attain the ends and advantages mentioned as well as those that are inherent therein. The particular examples disclosed above are illustrative only and may be modified and practiced in different but equivalent manners apparent to those skilled in the art having the benefit of the teachings herein. Although individual examples are discussed, the disclosure covers all combinations of all of the examples. Furthermore, no limitations are intended to the details of construction or design herein shown, other than as described in the claims below. Also, the terms in the claims have their plain, ordinary meaning unless otherwise explicitly and clearly defined by the patentee. It is therefore evident that the particular illustrative examples disclosed above may be altered or modified and all such variations are considered within the scope and spirit of those examples. If there is any conflict in the usages of a word or term in this specification and one or more patent(s) or other documents that may be

incorporated herein by reference, the definitions that are consistent with this specification should be adopted.

Claims

What is claimed is:

1. A method comprising:

providing one or more inputs to a hybrid data generator, wherein one of the one or more inputs is based at least in part on a wellsite location, wherein the hybrid data generator comprises a large language model, and wherein the large language model is based at least in part on a machine learning algorithm;

utilizing an information handling system to generate a cement job design based at least in part on the one or more inputs and the hybrid data generator; and

performing at least a portion of a cementing operation based at least in part on the cement job design.

2. The method of claim 1, wherein the large language model is trained using a dataset comprising at least one type of data selected from the group consisting of engineering data, geological data, geo-mechanical data, geo-physical data, data from lab-based tests, data modelled from simulations, data modelled from empirical models, data modelled from physics-based models, data from physics-informed neural networks, operational data from current cementing operations, operational data from previous cementing operations, measurements collected from current cementing operations, measurements collected from previous cementing operations, information collected from previous cementing reports, previously created cement job design, logging data, available equipment in a given region, and combinations thereof.

3. The method of claim 1, wherein the one or more inputs further comprises at least one input selected from the group consisting of engineering data, geological data, geo-mechanical data, geo-physical data, data from lab-based tests, data modelled from simulations, data modelled from empirical models, data modelled from physics-based models, data from physics-informed neural networks, operational data from current cementing operations, operational data from previous cementing operations, measurements collected from current cementing operations, measurements collected from previous cementing operations, information collected from previous cementing reports, previously created cement job design, logging data, available equipment in a given region, and combinations thereof.

4. The method of claim 1, wherein training the large language model further comprises reinforcement learning.

5. The method of claim 1, wherein the machine learning algorithm is utilized in a transformer architecture.
6. The method of claim 5, wherein the transformer architecture includes at least one architecture component selected from the group consisting of an encoder, a decoder, and combinations thereof.
7. The method of claim 1, wherein the machine learning algorithm comprises a deep learning algorithm further comprising at least one type of algorithm selected from the group consisting of convolutional neural networks, long short term memory networks, recurrent neural networks, generative adversarial networks, attention neural networks, zero-shot models, fine-tuned models, domain-specific models, multi-modal models, transformer architectures, radial basis function networks, multilayer perceptrons, self-organizing maps, deep belief networks, and combinations thereof.
8. The method of claim 1, further comprising updating the cement job design using the at least one measurement collected from the at least one sensor during the cementing operation, wherein the at least one measurement is added to the inputs provided to the hybrid data generator.
9. The method of claim 8, wherein updating the cement job design comprises updating the cement job design using at least one method selected from the group consisting of continuously updating the cement job design, updating the cement job design at set intervals of time, updating the cement job design when manually executed, updating the cement job design when a threshold is met, or combinations thereof.
10. The method of claim 1, wherein the large language model is optimized for at least one operational feature, wherein the at least one operational feature is at least one feature selected from the group consisting of fluid displacement, centralizer location, centralizer type, fluid composition, fluid volume, fluid rate, rheology, density, mixability, pumpability, thickening time, compressive strength, set time, fluid loss, fluid compatibility, temperature resistance, compatibility between subsequently pumped fluids, maximizing hole stability, minimizing total cementing cost, minimizing cost per wellbore section, minimizing time spent on each wellbore section, operational safety, and combinations thereof.

11. The method of claim 1 further comprising collecting at least one measurement from at least one sensor during the cementing operation.

12. A system comprising:

a hybrid data generator comprising a large language model, wherein the large language model is based at least in part on a machine learning algorithm;

an information handling system configured to execute the hybrid data generator to generate a cement job design, wherein the generated cement job design is based at least in part on one or more inputs and wherein at least one of the one or more inputs is based at least in part on a wellsite location; and

a sensor in communication with the information handling system, wherein the sensor measures at least one measurement during a cementing operation.

13. The system of claim 12, wherein the large language model is trained using a dataset comprising at least one type of data selected from the group consisting of engineering data, geological data, geo-mechanical data, geo-physical data, data from lab-based tests, data modelled from simulations, data modelled from empirical models, data modelled from physics-based models, operational data from current cementing operations, operational data from previous cementing operations, measurements collected from current cementing operations, measurements collected from previous cementing operations, information collected from previous cementing reports, previously created cement job design, logging data, available equipment in a given region, and combinations thereof.

14. The system of claim 12, wherein the one or more inputs further comprises at least one input selected from the group consisting of engineering data, geological data, geo-mechanical data, geo-physical data, data from lab-based tests, data modelled from simulations, data modelled from empirical models, data modelled from physics-based models, operational data from current cementing operations, operational data from previous cementing operations, measurements collected from current cementing operations, measurements collected from previous cementing operations, information collected from previous cementing reports, previously created cement job design, logging data, available equipment in a given region, and combinations thereof.

15. The system of claim 12, wherein training the large language model further comprises reinforcement learning.

16. The system of claim 15, wherein the machine learning algorithm is utilized in a transformer architecture and wherein the transformer architecture includes at least one architecture component selected from the group consisting of an encoder, a decoder, and combinations thereof.

17. The system of claim 12, wherein the machine learning algorithm comprises a deep learning algorithm further comprising at least one type of algorithm selected from the group consisting of convolutional neural networks, long short term memory networks, recurrent neural networks, generative adversarial networks, attention neural networks, zero-shot models, fine-tuned models, domain-specific models, multi-modal models, transformer architectures, radial basis function networks, multilayer perceptrons, self-organizing maps, deep belief networks, and combinations thereof.

18. The system of claim 12, wherein the information handling system is configured to update the cement job design based at least in part on the one measurement collected by the sensor during the cementing operation.

19. The system of claim 18, wherein the information handling system is configured to update the cement job design using at least one method selected from the group consisting of continuously updating the cement job design, updating the cement job design at set intervals, updating the cement job design when manually executed, updating the cement job design when a threshold is met, or combinations thereof.

20. The system of claim 12, wherein the large language model is optimized for at least one operational feature, wherein the at least one operational feature is at least one feature selected from the group consisting of fluid displacement, centralizer location, centralizer type, fluid composition, fluid volume, fluid rate, rheology, density, mixability, pumpability, thickening time, compressive strength, set time, fluid loss, fluid compatibility, temperature resistance, compatibility between subsequently pumped fluids, maximizing hole stability, minimizing total cementing cost, minimizing cost per wellbore section, minimizing time spent on each wellbore section, operational safety, and combinations thereof.

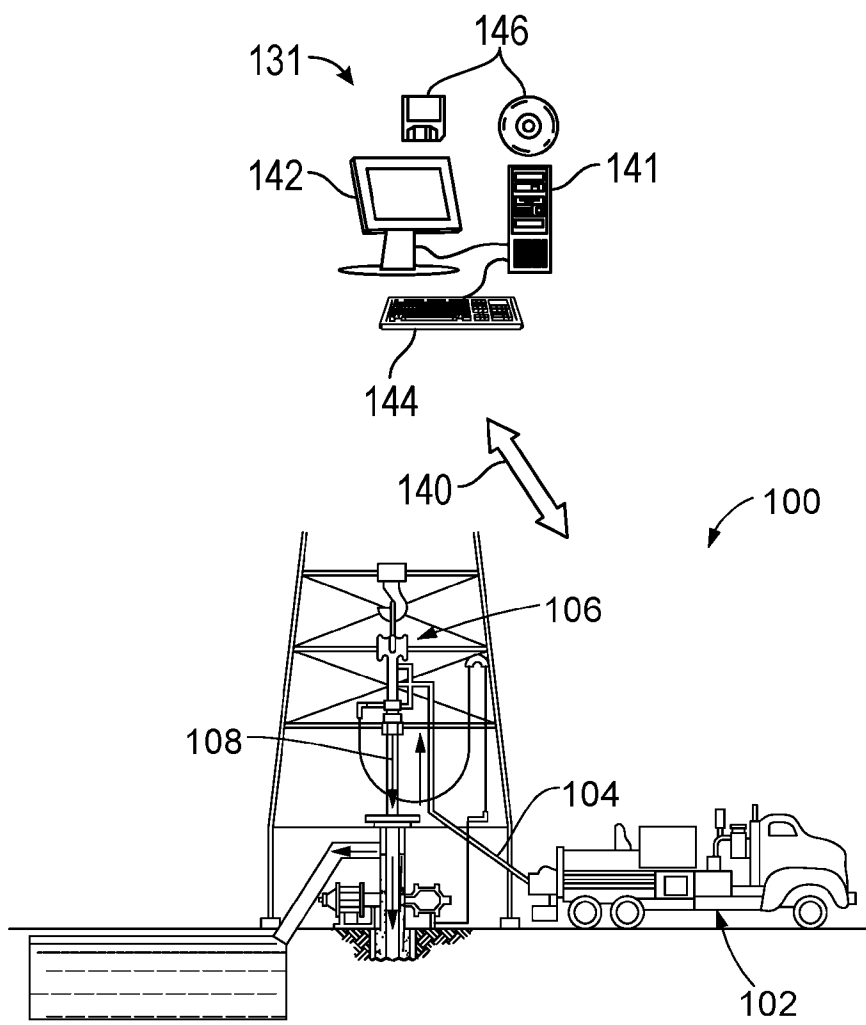


FIG. 1a

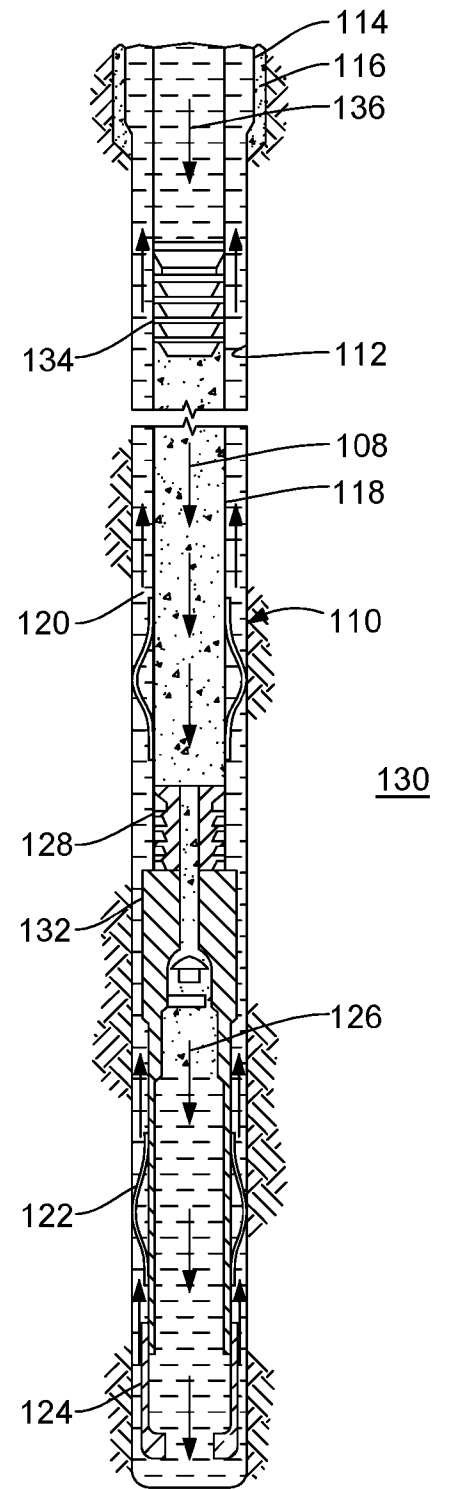


FIG. 1b

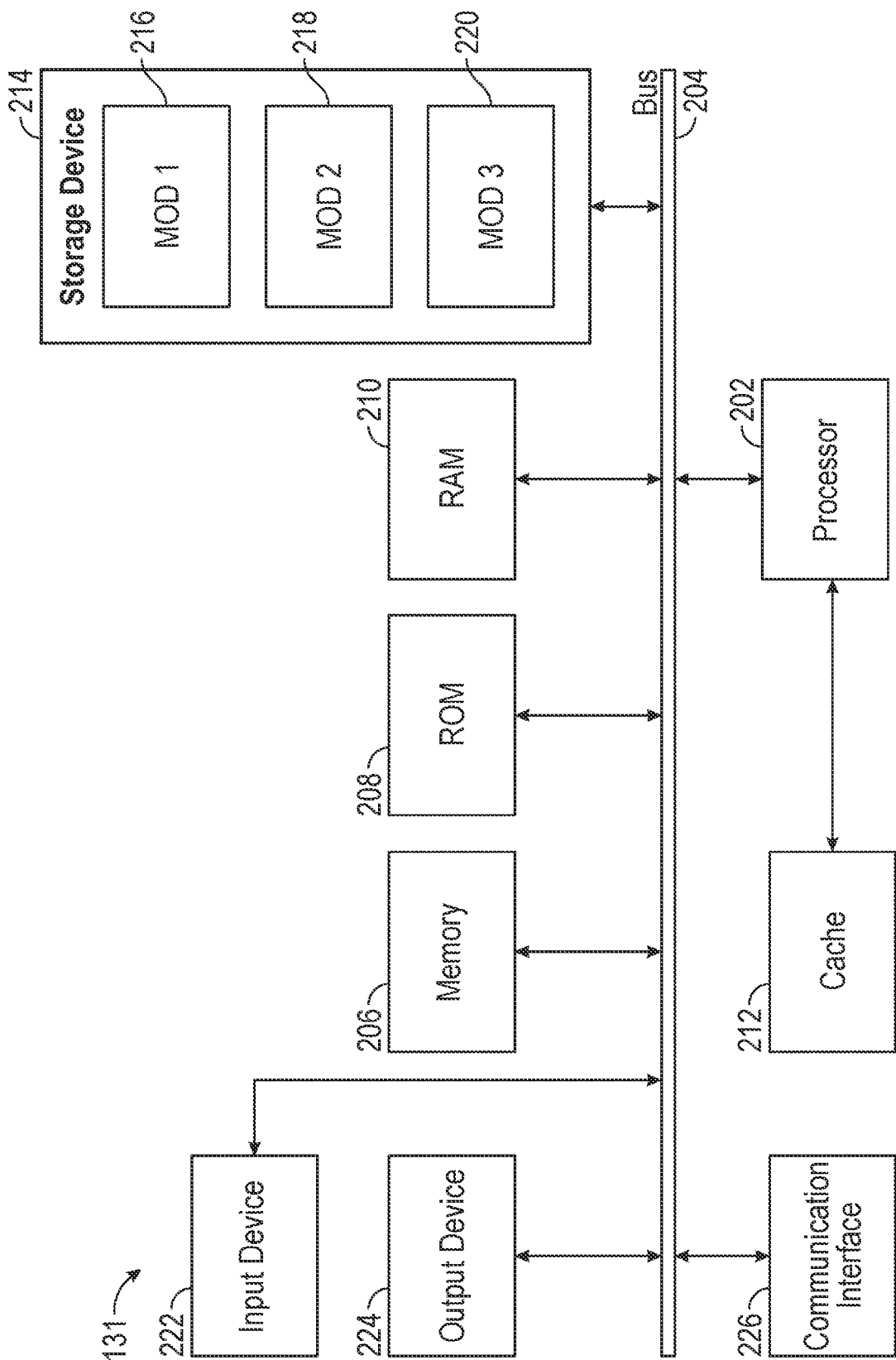


FIG. 2

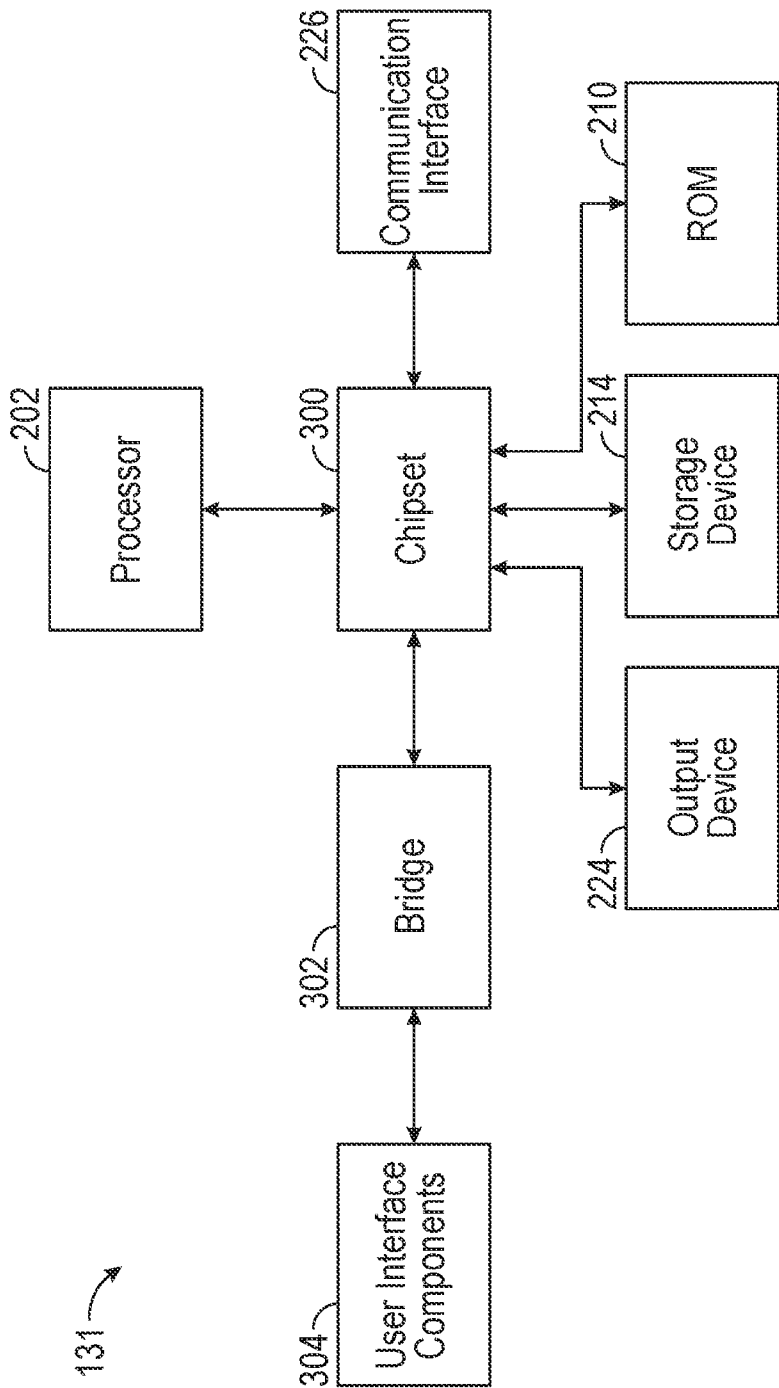


FIG. 3

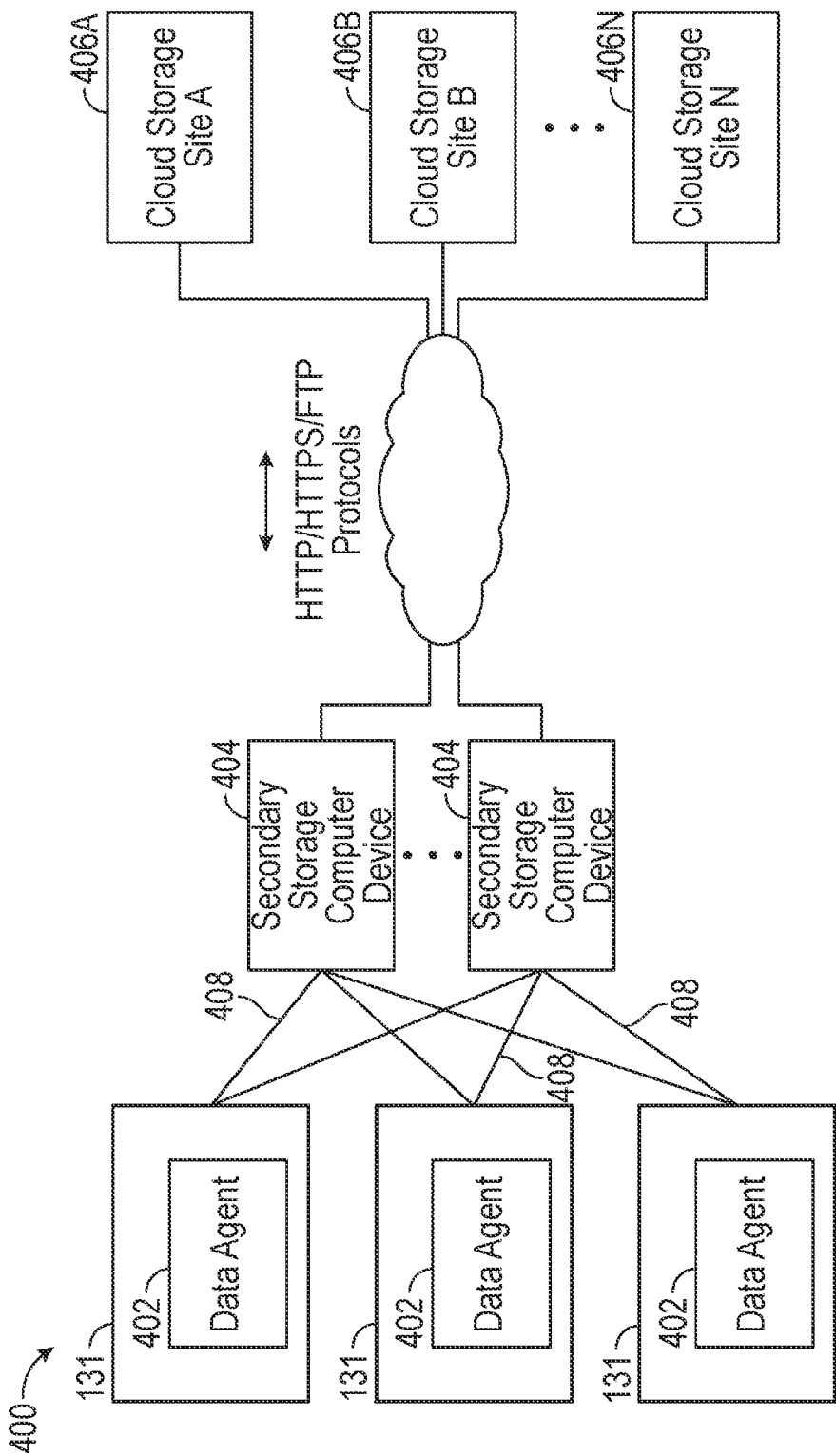


FIG. 4

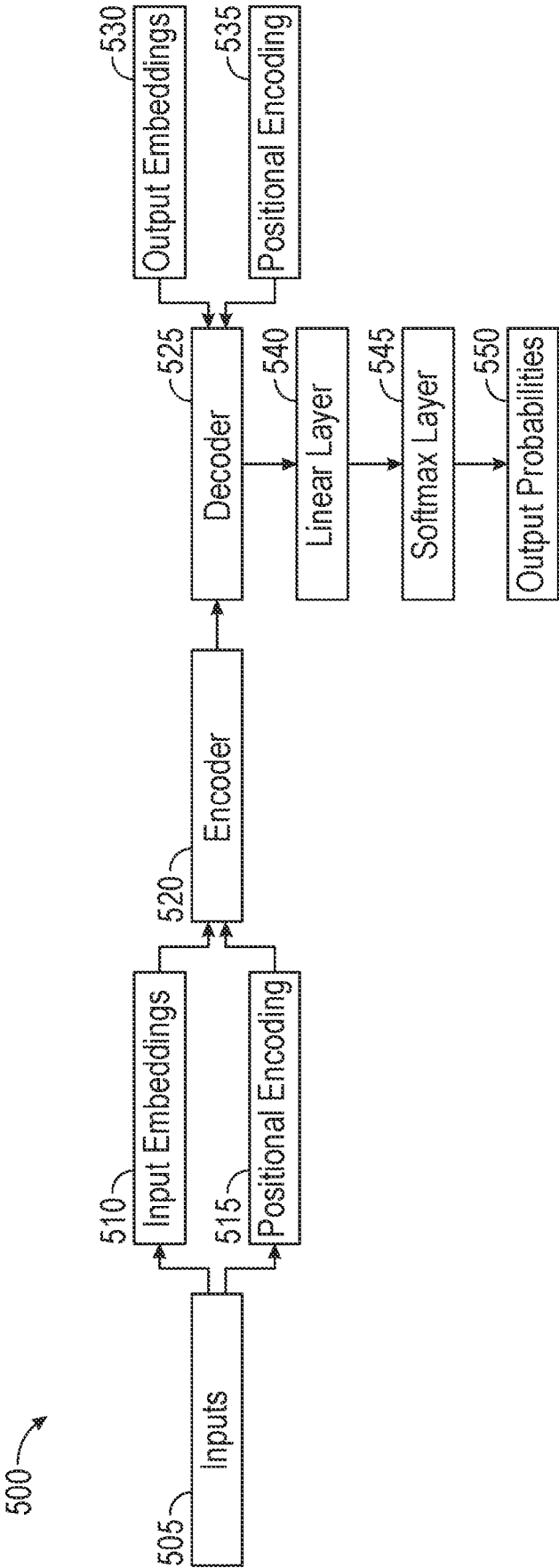


FIG. 5

6/10

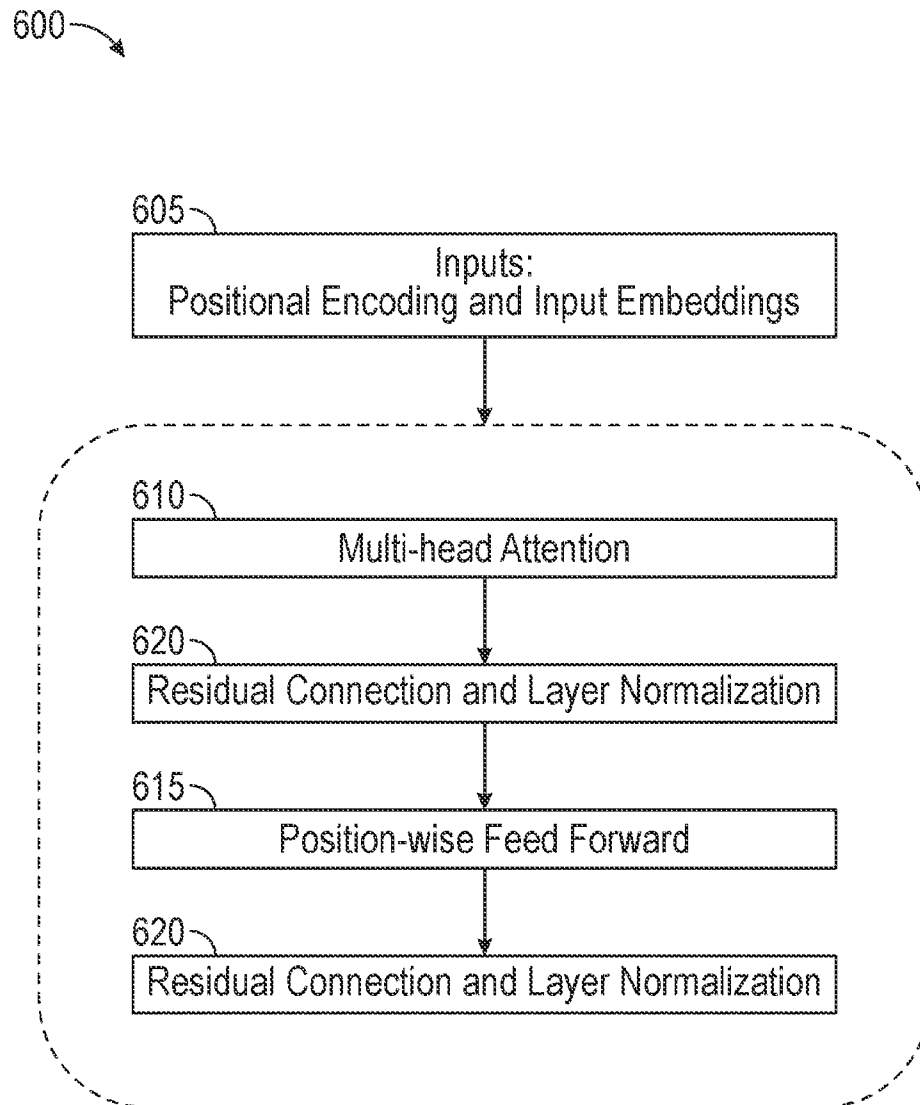


FIG. 6

7/10

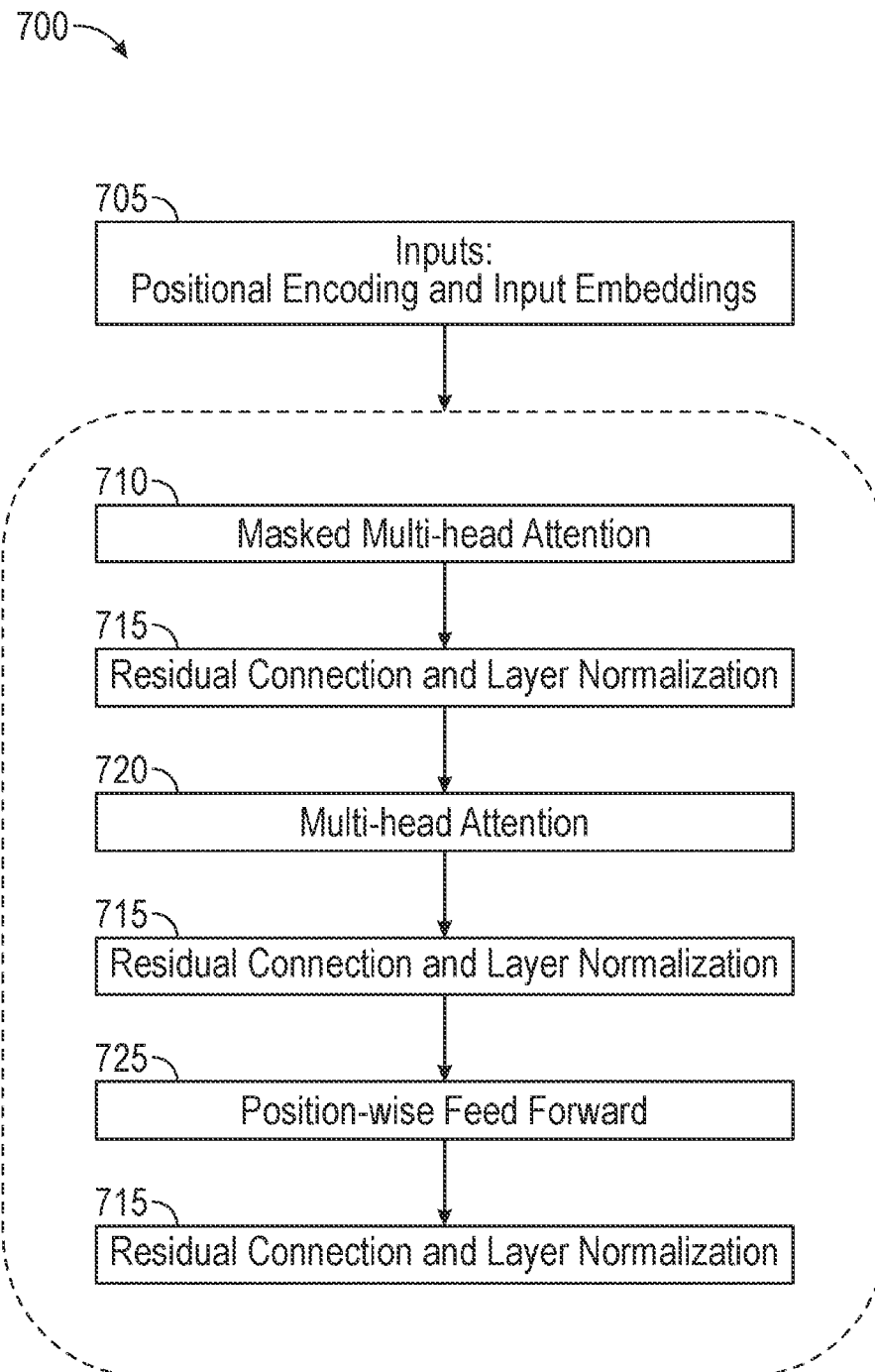
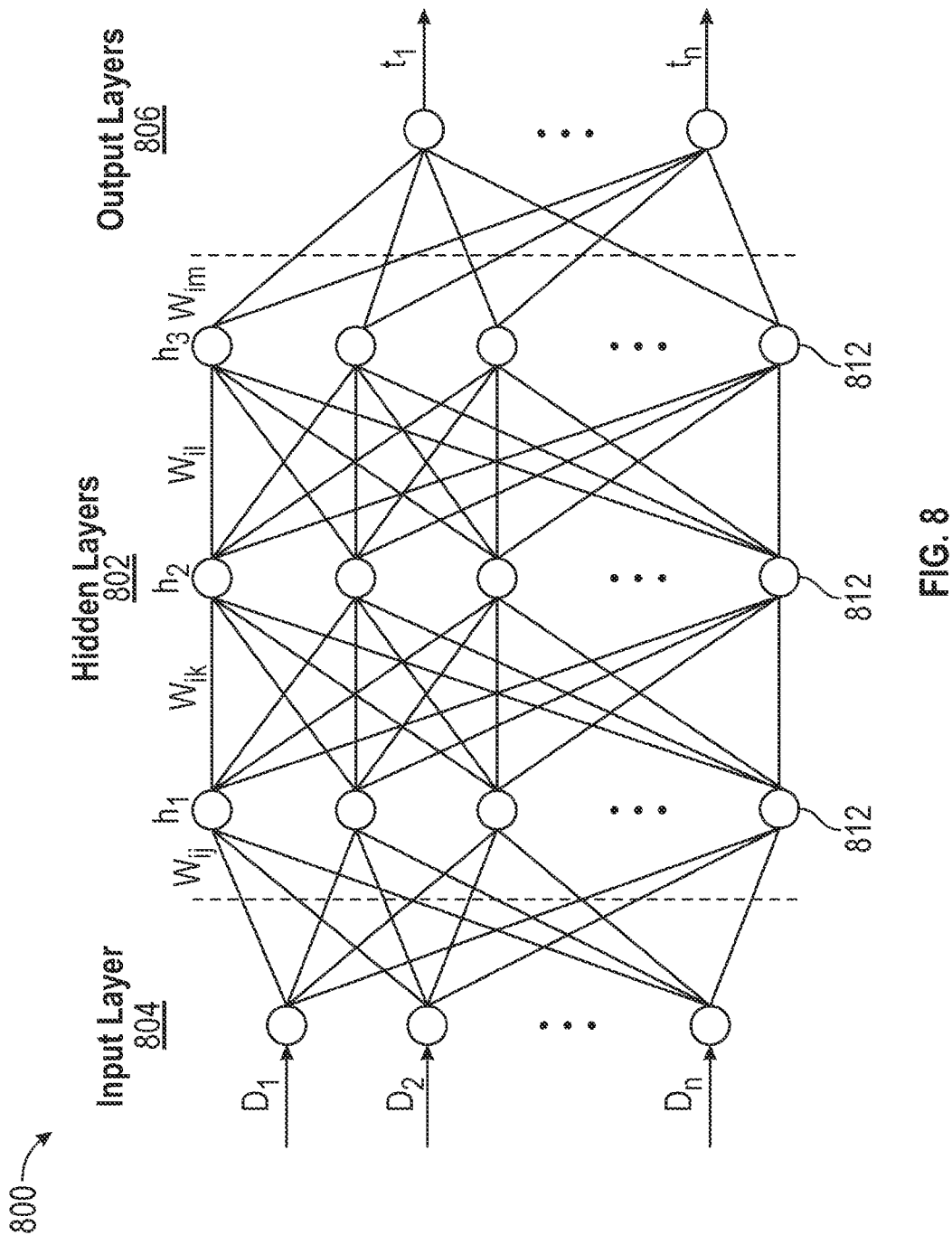


FIG. 7



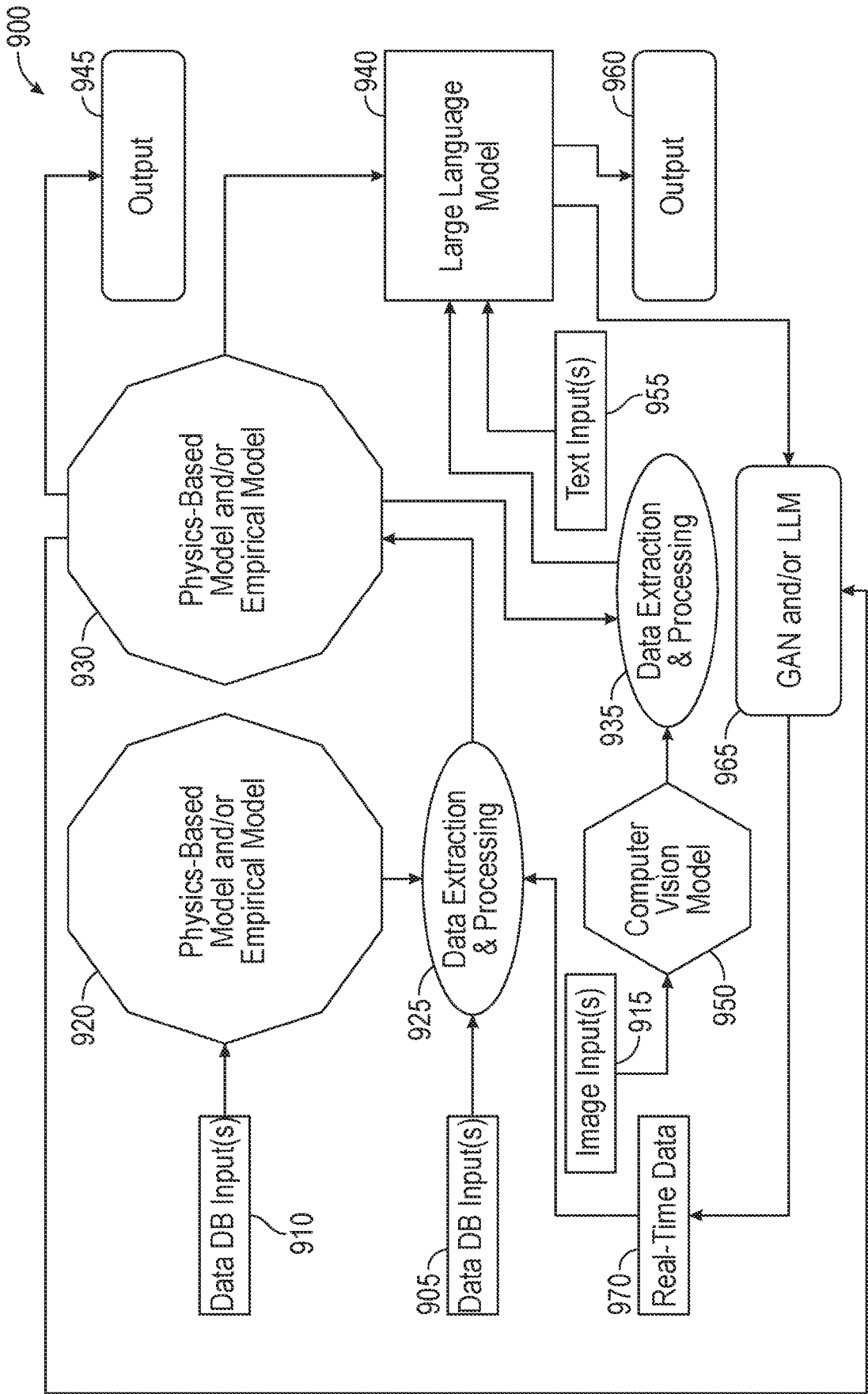


FIG. 9

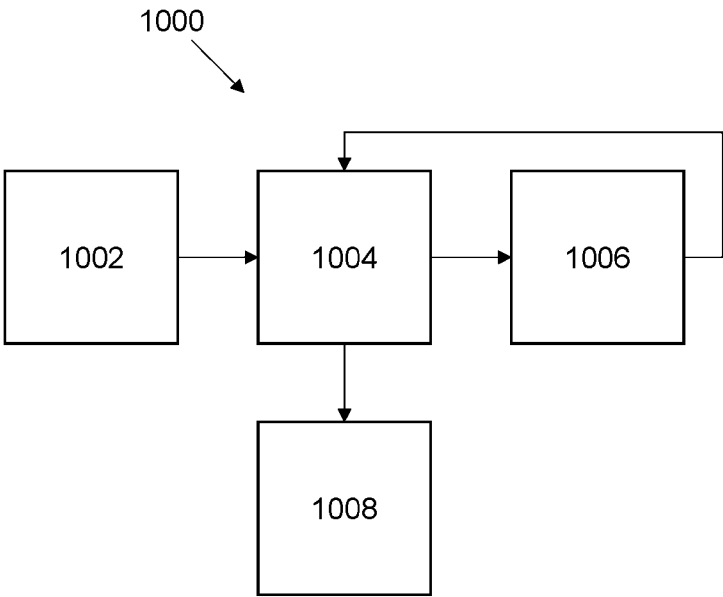


FIG. 10

INTERNATIONAL SEARCH REPORT

International application No.

PCT/US2024/039385

A. CLASSIFICATION OF SUBJECT MATTER G06N 3/092(2023.01)i; G06N 3/04(2006.01)i; E21B 33/14(2006.01)i According to International Patent Classification (IPC) or to both national classification and IPC		
B. FIELDS SEARCHED Minimum documentation searched (classification system followed by classification symbols) G06N 3/092(2023.01); E21B 44/00(2006.01); E21B 47/12(2006.01); E21B 49/00(2006.01); G01V 11/00(2006.01); G01V 99/00(2009.01); G06F 30/27(2020.01); G06N 20/20(2019.01) Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched Korean utility models and applications for utility models Japanese utility models and applications for utility models Electronic data base consulted during the international search (name of data base and, where practicable, search terms used) eKOMPASS(KIPO internal) & Keywords: large language model, wellbore, drilling operation, cement job design, hybrid data generator		
C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 2023-0055082 A1 (HALLIBURTON ENERGY SERVICES, INC.) 23 February 2023 (2023-02-23) paragraphs [0049], [0055], [0057], [0059], [0064]; and claim 1	1-20
Y	MING JIN et al., 'Large Models for Time Series and Spatio-Temporal Data: A Survey and Outlook', arXiv:2310.10196v2, October 2023 [retrieved on 2024.10.21]. Retrieved from <https://arxiv.org/pdf/2310.10196v2> pages 1, 4, 10; and figure 1	1-20
A	US 2023-0063424 A1 (SAUDI ARABIAN OIL COMPANY) 02 March 2023 (2023-03-02) paragraph [0045]; claim 1; and figure 4	1-20
A	WO 2023-164048 A1 (SCHLUMBERGER TECHNOLOGY CORPORATION et al.) 31 August 2023 (2023-08-31) claims 1-3	1-20
A	US 2021-0355805 A1 (LANDMARK GRAPHICS CORPORATION) 18 November 2021 (2021-11-18) paragraphs [0017]-[0018]	1-20
☐ Further documents are listed in the continuation of Box C. ☒ See patent family annex.		
* Special categories of cited documents: “A” document defining the general state of the art which is not considered to be of particular relevance “D” document cited by the applicant in the international application “E” earlier application or patent but published on or after the international filing date “L” document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) “O” document referring to an oral disclosure, use, exhibition or other means “P” document published prior to the international filing date but later than the priority date claimed “T” later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention “X” document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone “Y” document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art “&” document member of the same patent family		
Date of the actual completion of the international search 31 October 2024		Date of mailing of the international search report 31 October 2024
Name and mailing address of the ISA/KR Korean Intellectual Property Office 189 Cheongsa-ro, Seo-gu, Daejeon 35208, Republic of Korea Facsimile No. +82-42-481-8578		Authorized officer YANG, Jeong Rok Telephone No. +82-42-481-5709

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.

PCT/US2024/039385

Patent document cited in search report			Publication date (day/month/year)	Patent family member(s)			Publication date (day/month/year)
US	2023-0055082	A1	23 February 2023	AU	2021-461284	A1	07 December 2023
				NO	20231406	A1	29 December 2023
				WO	2023-027737	A1	02 March 2023
US	2023-0063424	A1	02 March 2023	US	11988795	B2	21 May 2024
WO	2023-164048	A1	31 August 2023	US	2023-0267594	A1	24 August 2023
US	2021-0355805	A1	18 November 2021	US	2021-0056447	A1	25 February 2021
				US	2021-0058235	A1	25 February 2021
				WO	2021-040764	A1	04 March 2021
				WO	2021-041251	A1	04 March 2021
				WO	2021-041252	A1	04 March 2021
				WO	2021-041254	A1	04 March 2021