Human-AI Common Ground for Training and Operations

Spencer K. Lynn, Susan S. Latiff, William Norsworthy, Jr.
Charles River Analytics, Inc.
Cambridge, MA
slynn@cra.com, slatiff@cra.com, wnorsworthy@cra.com

Mark Turner
Case Western Reserve University
Cleveland, OH
mark.turner@case.edu

Peter Weyhrauch
Charles River Analytics, Inc.
Cambridge, MA
pweyhrauch@cra.cm

ABSTRACT

How do we create artificially intelligent agents capable of meaningful and trusted teaming with humans for training and operations? “Common ground” refers to congruent knowledge, beliefs, and assumptions among a team about their objectives, context, and capabilities. It has been a guiding principle in cognitive systems engineering for human-AI interaction, where research has focused on improving communication between human and machines. Coordination (e.g., directability) and transparency (e.g., observability and predictability) are important for establishing, maintaining, and repairing both human-AI and human-human common ground. Nonetheless, human-AI common ground remains relatively impoverished, and AI remains a tool rather than a teammate Communication between humans and machines plays a crucial role in establishing the machine's state. Conversely, when machines communicate with humans, it provides transparency by revealing the machine's state to the human. Among humans, common ground occurs at the level of concept structure; however, human concepts are not merely variables to be parameterized, but are constructed during discourse. For example, an instructor uses communication to activate and shape concepts (through dialog) in the student’s mind, contextualizing and refining concepts until shared perceptions are categorized (understood) in a common way. To increase autonomy and human-AI teaming, the challenge is to provide the AI with human-like conceptual structure. An architecture to enable human-AI common ground must provide the AI with representational capacity and algorithms that mimic features of human conceptual structure and flexibility. Here, we identify critical features of human conceptual structure, including Conceptual Blending, Situated Categorization, and Concept Degeneracy. We evaluate challenges of implementing these features in AI and we outline technical approaches for hybrid symbolic/subsymbolic AI to meet those challenges. As contemporary human-factors approaches to human-AI common ground continue to mature, common ground issues will move from interface transparency to concept congruency.

ABOUT THE AUTHORS

Spencer Lynn, Ph.D., is a Senior Scientist at Charles River Analytics. He applies principles of behavioral ecology and neuroscience to create biologically inspired cognitive architectures and autonomous agents. His work focuses on development of technology that applies evidence-based, computational modeling of cognitive and behavioral processes to human-machine teaming, situation awareness, and operational readiness.

Susan Latiff, Ph.D., is a Scientist at Charles River Analytics. She uses cognitive systems engineering and ecological interface design to create human-centered intelligence systems. Her research focuses on understanding work systems and technology design with respect to decision making in complex, multimodal, dynamic environments.

William Norsworthy, Jr., is a Software Engineer at Charles River Analytics. His interests include implementation of agent architectures and advanced typed ontology systems, interface design, and game development.

Mark Turner, Ph.D., is Institute Professor and Professor of Cognitive Science, Case Western Reserve University. He is a distinguished researcher in the field of cognitive linguistics. He has written 12 books, published widely in peer reviewed journals, and lectured globally. He co-directs the International Distributed Little Red Hen Lab.

Peter Weyhrauch, Ph.D., is Principal Scientist and Vice President of the Human-Centered Artificial Intelligence Division at Charles River Analytics. Dr. Weyhrauch’s research interests include artificial intelligence, simulation-based training, models of human performance, as well as scenario generation and computational narrative.

Human-AI Common Ground for Training and Operations

Spencer K. Lynn, Susan S. Latiff, William Norsworthy, Jr.
Charles River Analytics, Inc.
Cambridge, MA
slynn@cra.com, slatiff@cra.com, wnorsworthy@cra.com

Mark Turner
Case Western Reserve University
Cleveland, OH
mark.turner@case.edu

Peter Weyhrauch
Charles River Analytics, Inc.
Cambridge, MA
pweyhrauch@cra.cm

CHALLENGES TO ACHIEVING HUMAN-AI COMMON GROUND

How do we create artificially intelligent teammates capable of meaningful and trusted teaming with humans? A critical barrier to achieving this vision is equipping an Artificial Intelligence (AI) teammate with the means to develop *common ground* (Clark & Wilkes-Gibbs, 1986; Klein et al., 2004; Dafoe et al., 2021; Lynn et al., 2023) with its human counterpart by sharing congruent knowledge, beliefs, and assumptions about the team's objectives, context, and capabilities. Across the Services, strategic vision over the coming decades calls for increasing operational agility via human-autonomy teaming (e.g., Endsley, 2015; Zacharias, 2019). Yet, computers and humans have a limited ability to communicate in ways that are intuitive and expressive to humans, resulting in a meager common ground on which AI is perceived as a tool rather than a teammate. Thus, a navigation computer can effortlessly detect its precise position in space, receive information about the coordinates of a specific Point of Interest (POI), and calculate the distance and duration required to reach the POI. Alternatively, a computer vision classifier can undergo training to recognize formations of sensed contacts (such as aircraft formations like wall, echelon, or finger-four formations). However, while these computers can more precisely and quickly calculate statistics about the spatial information than its human teammate, and accurately execute actions based on them, rich communication about spatial relations in human terms is limited by the relatively impoverished and brittle nature of machine concepts (Brennan, 1998).

Human language and co-speech gesture, by contrast, are exceptionally powerful for conveying the overall spatial organization and dynamics of a scene. A human might say, "the pattern is {clearing, enlarging, exploding, contracting, swooping left}" with accompanying gestures; the other humans in the conversation know exactly what to look for and easily comprehend the details. A computer's ability to instead give the human all the spatial coordinate details for the elements in the pattern is almost useless. So, while humans have intuitive-seeming concepts invoked via natural-language to create common ground, humans have not invented tools sufficient for interacting with computational devices to *develop* the ground as the communication progresses. Here, our goal is to provide the industry designs toward a generalizable computational framework that captures key aspects of human conceptual flexibility to support developing common ground within human-AI teams.

Research and development on human-AI common ground has occurred around several themes. Some approaches focus on the AI's ability to parse human input, which is important when the same intent can be expressed in variable ways. For example, getting directions to Mom's house might be said as: "Go to Mom's", or "Take me to my mother's home", or "Please show me how to drive to Mom". Some approaches focus on reassuring the human that the AI has a correct understanding the human's instructions, e.g., by supplementing the AI's response to the human's prompt with specific contextualization. For example, a human using a device geolocated to Cambridge, Massachusetts, asks: "What's the weather today?" and the AI responds: "The weather *in Cambridge today* is mostly sunny." Contributions brought by the AI also add new facts to the common ground that the human would like to know. Likewise, providing a usable, efficient, non-overwhelming interface for the human is critical to enabling the human to understand the state of the AI. In addition, however, among humans common ground is also about congruent understanding of the external world as a domain of common input. Of the three often-cited pillars of common ground (knowledge, beliefs, and assumptions), only knowledge has been the concretely operationalized in human-AI common ground. Theory developed around the role of knowledge in common ground (Klein et al., 2005; McDermott et al., 2018) has largely been around facts or procedural knowledge that teammates need to know about the joint activity or the teammates. It stops short of establishing and refining shared mental models that put the facts to use. Nonetheless, the flexibility of

such models, e.g., how they are communicated and shaped by building on existing knowledge and other models, is key to how humans achieve common ground with other humans.

Among humans, the speaker uses communication to activate and construct concepts in the hearer's mind. Humans have a rich, dynamic network of concepts, of course. Therefore, the multitude of ways to refer to these concepts in communication with an AI (such as differences in grammar and labels) represents just one aspect of the shared challenge. Another aspect of the challenge is to provide the AI with human-like conceptual structures and abilities to reason over them. A system to enable the establishment, maintenance, and repair of human-AI common ground must provide the AI with representational capacity and algorithms that reflect important features of human conceptual structure and flexibility.

Human utterances consist of *blends* of grammatical constructions (including words, clausal and phrasal patterns, but also prosody and gesture). The speaker has a conceptual network that they want to share with the hearers. Grammatical form-meaning pairs (i.e., communicative constructions) encoding the meanings and relations in the conceptual network are blended by the speaker into a coherent *form*, a performance (i.e., a communicative act). The hearer's brain infers candidate form-meaning pairs that could have blended to create this performance and uses that inference to construct the conceptual network—developing common ground with the speaker. This is a flexible, adaptive, and creative process. However, the process works only with other human beings, not with computers.

Current AI possesses impoverished conceptual structures relative to a human teammate and this is one reason human-AI common ground has been limited. Three characteristics of human concepts contribute to the development of common ground among humans: concepts are blended, situated, and degenerate. These characteristics arise from the nature of human perception and categorization–the on-the-fly functional *construction* (Barrett et al., 2015) of conceptual instances (that is, categorization of incoming sensory data) and recognition of situations in which familiar percepts are present in perhaps novel combinations. An approach to modeling these features and processes for an AI is needed to enable the establishment, maintenance, and repair of a richer common ground within human-AI teams (Lynn et al., 2023). Table 1 contrasts these properties for human and AI concepts.

Table 1. Differences Between Human Concepts and AI Concepts

Human Concepts	AI Concepts
Function: Concepts provide understanding of perceptions given perceiver's goals	Function: Concepts are used to classify perceptions (including commands) as if the classes are objectively valid
Blended: Real-time construction of concepts (categories) is constrained by current goals	Static: A Class is pre-defined by a collection of features that is exclusive to that class
Situated: Context informs construction and can be a feature of the concept	Situated: Context used to condition classification
Degenerate: Multiple concepts may validly "explain" a given perception	Non-degenerate: One-to-one mapping between concept and perception; exceptions are "noise" rather than functional variability

AI Concepts Must Blend

Human concepts can be *blended* (Turner, 1996, 2014). Concepts have structure: an aircraft has parts, like wings, and attributes, like mass. These elements of structure can be shared with other concepts. Blending mixes elements from different concepts to categorize familiar elements they are encountered in novel configurations or situations. Conceptual blending enables the human teammate to reason about novel configurations of objects, or parts of objects, by applying the knowledge and entailments inherited by the blended concept from its contributing source concepts. For example, when you read the sentence "The greeble is on the table" you can infer features of greebles by blending the concept [on][1] with [the table] and use those inferences to create a novel concept of [greeble], with uncertain, yet experientially constrained, attributes. Specifically, you might infer that greebles (Gauthier & Tarr, 1997) are probably physical objects, and their size and mass is small enough to fit on and be supported by your prototypical concept of a table. In a human-AI teaming context, blending provides the AI an ability to mix and match perceptual primitives to construct a concept that "explains" an observed scene.

[1] Square brackets denote human or machine concepts, as distinct from words or labels used to refer to the concept.

AI Categories Must Be Situated

Human categorization, the act of judging a perception to be an example of a concept, is *situated* in a context (Barsalou, 2015). Situated categorization means that context influences which concepts are used to understand a situation. For example, when you read the sentence "The greeble is near the cup on the table," you can infer some constraint on greeble size and mass, because the meaning of the spatial relationship "near" is situated; this meaning, and how you can reason about it, depends on the context in which it was used. The distance "near" is dependent on relative sizes of the objects concerned; here, the greeble, the cup, and the table. To qualify as being "near" a cup on a table, given prototypical notions of those objects, one can reason that a greeble is probably not vastly larger than the cup. Thus, situated categorization enables context, including already-active concepts, to influence forthcoming categorization. In a human-AI teaming context, situated categorization means that the concepts a human teammate invokes to make sense of their perceptions are influenced by the goals of the team and aspects of context, such as the state of the mission, progress toward the goals, and the condition of the team or resources. For an AI-piloted autonomous Uninhabited Aerial System (UAS) to converse with a human teammate about being spatially "near" a target in a way that reflects common ground depends on whether its payload is a long-range sensor or short-range sensor: that context influences the meaning of the concept [near] for both teammates.

AI Concepts Must Exhibit Degeneracy

Degeneracy refers to the partial overlap of function by multi-functional components (Edelman & Gally, 2001). In biology, a physiological function can be successfully achieved by more than one pathway. Concept degeneracy refers to the idea different concepts can sometimes be applied equally well to categorize the same perception. Colloquially, there may be more than one way to understand the world, all of which might make sense; perceptions can be suitably categorized by more than concept. In a human-AI teaming context, the AI may encounter situations in which an observation may be suitably categorized as instances of more than one concept, all of which may be good fits to the observation. Also, situations will arise in which the human and AI may characterize observations of the same scene differently from one another. As in human teams, when such differences in understanding occur, teammates must have the means to negotiate which concept should be operative.

DESIGN PRINCIPLES FOR AN EMBODIED COMMON CONCEPT ONTOLOGY

We describe design principles for a system that can exhibit conceptual blending, situated categorization, and concept degeneracy. We call this design an Embodied Common Concept Ontology (ECCO). Embodiment (Lakoff, 1987; Varela et al., 2017) refers to the idea that meaning is relative to the agent (e.g., the agent's sensory capabilities, motor capabilities, and goals: what the agent needs and can do in the world). On this view, the idea that an entity (such as objects that computer vision classifiers recognize) has an objective, intrinsic definition is a fallacy. Instead, what an object *is* is defined entirely by what it affords the agent with respect to the agent's goals. This notion explains the brittleness of both classical symbolic AI and neural networks: definitions are only "true" to the extent that those concerned have common ground with respect to their goals and perceptions. Thus, when a human's goal becomes incongruent with an AI's training, the AI's output becomes untrustable. In the open world, agents (i.e., humans, and eventually AI teammates) have many and dynamic goals, so static definitions of objects in the world are insufficient to support flexible behavior.

For scoping, we limit ECCO's example domain of operation to spatial conceptualization and reasoning. In our example, ECCO is a component of a UAS teamed with a human-piloted aircraft to engaging in tactical spatial configurations with respect to targets and one another. The domain of spatial conceptualization and reasoning is a useful example for developing human-AI common ground because UAS tactics (in, e.g., search and rescue or combat operations), can be defined as establishing specific spatial relationships with respect to objects. Examples include following teammates, observing adversaries, surrounding POIs, and avoiding collisions (Lynn et al., 2020). Physical space is also an interesting domain in which to begin because it is a root conceptual domain within the Theory of Embodied Cognition (TEC). TEC posits that more abstract human conceptualization uses the same representational and reasoning mechanisms as those of directly embodied domains, such as space (Lakoff & Johnson, 1980).

The principles we describe differ from prior approaches to common in notable ways. ECCO communicates about contextually relative categorizations rather than objective facts. ECCO's locus of variability is at the mapping of observations to degenerate concepts rather than at the mapping of variable grammar to unequivocal parameters. ECCO uses human-like conceptual structures to construct a model of the world that is congruent with that of the human teammate rather than representing facts in the human's mind. Figure 1 illustrates the general architecture for an ECCO-backed agent teaming with a human.

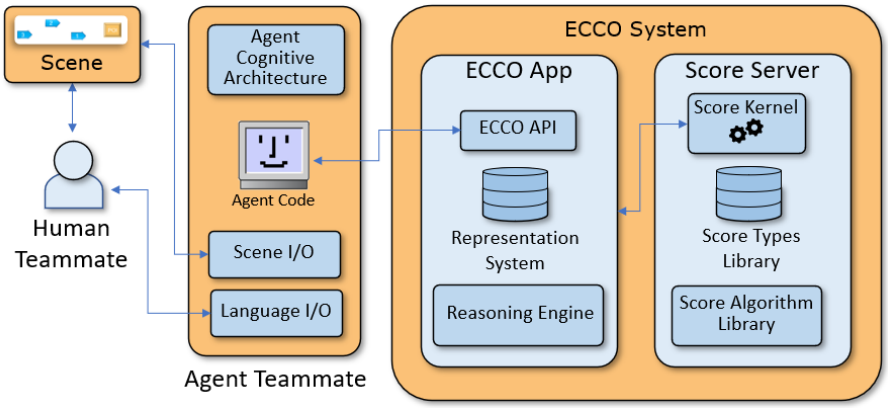

Figure 1. ECCO Architecture

On the left of Figure 1, our teammates observe scenes depicting spatial configurations of objects such as UASs tracking objects of interest on the ground or aircraft in flying formations. To complete a mission or other shared task, human and software agent must develop common ground understanding of the spatial relationships among the objects in the scenes (for example, recognizing tactics as they unfold) and take action in the scenes (for example, correctly joining a formation of aircraft).

The agent interfaces to ECCO via an Application Programming Interface (API). By providing an API and a controlled language, the ECCO system is not limited to working with a particular agent architecture. ECCO is agnostic to the cognitive architecture with which agent teammates are implemented; it provides any agent with concepts that structure a domain of human experience. The agent interacts with the *scene* (sensing it and acting within it) via a *Scene I/O* component that interfaces with the world or a simulation environment responsible for generating the scene. The agent interacts with the *human* via the *Language I/O* component. There are, of course, many approaches to implementing an agent's Scene and Language I/O components and a wealth of prior work in these domains, including subsymbolic computer vision and natural language processing. Our focus here is instead on the ECCO components rather than the agent components.

The agent's understanding of the scene and ability to reason about it in human terms are provided by ECCO. ECCO itself has two main parts, a Representation System and a Reasoning Engine. The Representation System is a types-based, probabilistic, ontological knowledgebase containing all the concepts the system is capable of recognizing, associated with one another in attribute- and part- relationships to provide conceptual structure. The Reasoning Engine has a kernel, programming language, and libraries of type-specific algorithms that operate on entries in the Representation System. Depicted in Figure 1, the ECCO features described here are built on a prototype ontology kernel, called Score.

To establish a common understanding of spatial relationships and reasoning using ECCO, the agent converts scene descriptions into ECCO's vector-based notation. The agent then interacts with the ECCO system, which categorizes the spatial relationships and returns concept names. Additional API calls using the concept names allow the agent to reason and carry out common ground processes. Some API calls return vectors back to the agent, which it can transform into a suitable representation, such as a waypoint, for taking actions in the environment.

The Representation System

To provide a computational representation of objects and spatial relations that can support blending, situated categorization, and degeneracy, ECCO has a representation system. Blending means that humans combine parts of different concepts, therefore the schema must support conceptual *structure* that has elements which can be shared among concepts. Situated categorization means that context influences interpretation, which requires that the instancing of a concept can be influenced by other active concepts. Degeneracy means that more than one concept can be a good candidate for categorizing a perception, therefore the schema must accommodate a notion of goodness-of-fit between perceptions and conceptual structures. Table 2 describes key features of the ECCO representation system. These features combine to support real-time construction of concepts to categorize incoming sensor data blending to produce situated categorization and degeneracy.

Table 2. Key Features of the ECCO Representation System

Feature	Description
Types	Ontology entries are computational *types,* to support processes such as creating new knowledgebase entries that can inherit features of their parent entries.
Entries with internal structure	Annotations, parts, and attributes of a type can have different functions with respect to inheritance during processes such as blending.
Contexts	[context]s are types that have a *simulation structure* with a typed ontology, a *language* with mappings from symbols in the language to types in the simulation structure, and *facts* expressed in the language and assigned to instances of types.
Runtime creation of new ontology entries	New types (e.g., novel combinations of ontology parts and attributes) can be created programmatically by the system, a requirement for blending.
Multiple inheritance	The system supports unions of types rather than requiring a strict hierarchy of types, for example, a blend is the child of its source concepts but is not necessarily a strict subtype or example of any of them.
Depth	Parts and attributes are themselves ontological entries (with their own parts and attributes), to support situated categorization.

To support blending, entries in the ontology are structured by parts and attributes that can be shared among concepts, so a new blended concept entry can be constructed by mixing parts and/or attributes from existing contributing entries. To support situated categorization, all parts and attributes are themselves ontological entries in the knowledgebase, enabling entries to influence one another. To support degeneracy, the values that parts and attributes can take on are defined as probabilistic distributions, providing numerical values for goodness-of-fit measures between perceptions and the candidate conceptual structures for which the perceptions are potential exemplars. Context data types are central to all of these because they can be used to represent complex concepts and exchange information with other contexts (Weyhrauch, 1978; Talcott & Weyhrauch, 1990; Weyrauch & Talcott, 1997).

As an example of representation system features, the knowledgebase entry for the spatial relationship [between] has parts that include an object of focus (the object that is in the [between] relationship to other objects) and two or more objects of reference (the objects flanking the focal object). An object of reference is itself an ontology entry, with two attributes that define the distributions (e.g., means and variances) of the idealized angle and distance to the focal object. The knowledgebase entry for the spatial relationship [beside] also has focal and reference objects but has different idealized distributions than [between]. Figure 2 illustrates this approach with an example of degeneracy in the application of [between] versus [beside]. In Figure 2, is UAS 2 between or beside UASs 1 and 3? It is ambiguous; UAS 2 is positioned in a region of equal [between] versus [beside] density, therefore either term is a suitable categorization of UAS 2's spatial relationship with UASs 1 and 3. The human might use one and ECCO the other, in which case they can invoke a decision rule to choose which term to use (for example, ECCO could adopt the human's preferred term).

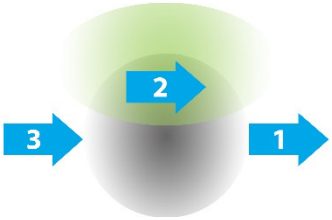

Figure 2. Concept Degeneracy of Between Versus Beside. Looking down on three UAS traveling in the same direction at the same altitude in a staggered formation. The grey (lower, circular) blurred region represents the density function of [between] spatial relationships. The green (upper, elongated) blurred area represents the density function of [beside] spatial relationships. Darker shade corresponds to higher prior expectation.

The ECCO ontology captures key features of human concepts about objects and spatial relationships present in the scene. In ECCO, concepts describing spatial relationships among objects, such as [between] and [beside], can be defined by a type that specifies spatial relationships as *distributions* of vectors between objects or points of interest (expected angle and distance; Figure 3). A vector distribution notation supports situated cognition by representing distances (vector magnitudes) scaled relative to the size of the objects so that, for example, formations of small UAS are proportionally distanced with respect to formations of piloted aircraft. Distributional attributes also support degeneracy. Degeneracy is modeled as similarity of goodness-of-fit measures between an observed spatial relationship and ontology entries that are candidates for categorizing the observation.

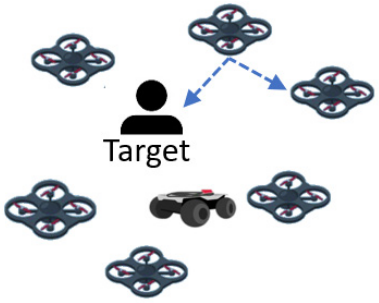

Figure 3. [surround] Spatial Relationship. Defined by probability distributions of vectors (denoted by dashed arrows) among swarm vehicles and the target.

The Reasoning Engine

To reason over concepts and categorized observations, ECCO has a Reasoning Engine. This component implements algorithms, which operate over ECCO knowledgebase content, to perform processes of establishing, maintaining, and repairing human-AI common ground based on principles of embodied human conceptualization. Every ontology entry in the ECCO representation system is a computational type constructed from more primitive computational types. For example, a reference object's vector attribute is comprised of a direction distribution and a distance distribution, and the distributions may be characterized by a mean and variance, which are themselves floating point numbers (i.e., floats). Floats (and integers and characters) are familiar computational types, and in a computer, each has type-specific algorithms that operate on it. Similarly, the reasoning engine defines algorithms that operate over the representation system's ontology entries as constructed types. Thus, when the reasoning engine operates on, e.g., a mean or variance, those types afford additional reasoning (algorithms) over and above their status as floats. The ontological relationships among types and the parts and attributes from which they are composed provide a basis for machine-reasoning across the knowledgebase as a semantic network. The reasoning engine has several key and interrelated functions, including categorization of observations, conceptual blending, situated categorization, and support for action.

Categorization of Observations

For our purposes, categorization is the act of applying a concept (for ECCO, a type definition) to an observation, and we take the position that this act is the mechanism by which a human or AI comes to "understand" the observation. One way ECCO categorizes observations (e.g., spatial relations among objects) is by comparing vector representations of an observed scene to vector definitions of types to calculate the goodness-of-fit with which the type explains the observation (Figure 4). Goodness-of-fit is implemented as a joint likelihood function. Distributional attributes provide variability over which to calculate goodness-of-fit measurements between the observation and candidate concepts. ECCO selects the concept with which to categorize an observation based on goodness-of-fit measurements.

	Direction (deg)		Magnitude (normed)	
Pair	Mu	Var	Mu	Var
1 to 2	45	2	1	2
1 to 3	350	2	2	2
1 to 4	315	2	3	2

Frame of reference: from nose, 0 deg=>up x-axis, increasing counter-clockwise

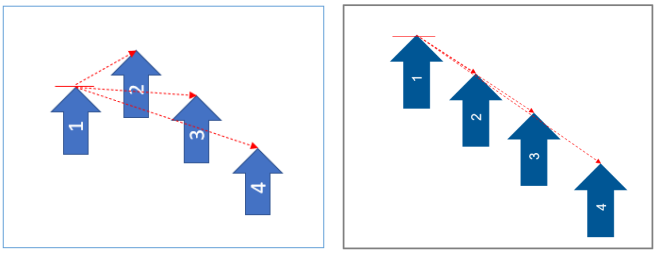

Figure 4. Spatial relations defined by distributional attributes (left, definition of a finger-four formation) afford goodness of fit calculations between the definition and a given observation (middle, a good fit to a finger-four formation; right, a poor fit to finger four but a good fit to a ladder formation)

Conceptual Blending

ECCO is designed to blend by combining recognized parts and attributes from different known types to create instances of known types as well as novel types. For example, blending can include the commonplace application of a known spatial relation concept to known objects that have none-the-less not been seen before in that relation. Blending can also encompass understanding novelty by constructing a new concept (i.e., run-time creation of a new type in the ontology) that has attributes and parts from known concepts but replaces the parent distributions with observed values and the parent parts with observed instances of those parts.

As illustrated with the greeble example, blending can support inferences about the attributes of unknown objects. In addition, ECCO is designed to use blending to learn new concepts, such as spatial configurations that are outside of ECCO's current knowledgebase, illustrated in Figure 5. One intriguing possibility this application of blending offers is the construction of concepts collaboratively with ECCO over a series of conversations, starting from building-block concepts. If the human were also learning the terms at the same time, this method could be an interesting model for human-machine co-training (van den Bosch et al., 2019).

1. The agent teammate has existing concept definitions for several spatial relationships (e.g., [between], [wall formation], [above]), some of which have parts for multiple UASs (e.g., [wall formation]), some of which are expressed relative to a POI (e.g., [above])
2. Agent observes a spatial configuration: several UASs are positioned around a point of interest (e.g., a building)
3. This observation has a poor goodness-of-fit with all concepts that the agent currently knows, e.g., the observed number of UAS does not match some concepts, and the observed vectors are unlikely to have been produce the others
4. Nonetheless, the observation has commonalities with existing concepts: e.g., [wall formation] accommodates >1 UAS, [above] accommodates a point of interest.
5. ECCO creates a new spatial relation type that uses these commonalities it recognizes with the observed vectors as estimates of the vectors that characterize the new relation
6. Then, agent asks the human teammate for a label for the new type
7. Human says, "That type of spatial relationship is called [surround]"

Figure 5. Novel Concept Generation

Situated Categorization

In situated categorization, previously instanced concepts modify the attribute and part values of types that are candidates for use in categorizing the scene. Adjusting the prior distributions of the attributes of candidate concepts is a mechanism by which context (i.e., other concepts involved in the shared task and the scene) can influence the understanding of the scene. This influence of context of categorization is exemplified by an evaluation of an ECCO software prototype, described below. In that example, the meaning of [near] is situated by an agent's objective (e.g., to deploy long-range vs. short-range sensors). The objective constrains the goodness-of-fit with which [near] characterizes this scene by adjusting the distribution of the vector magnitude that defines [near] in the ontology.

Action

The reasoning engine also includes algorithms to support agent actions in the shared task. It may solve for relative positions to enable the agent to attain a required spatial relationship with respect to some target, or to otherwise take actions that change the scene such that it exemplifies a specific spatial relationship. For example, if the shared task

requires the agent, as an autonomous UAS, to adopt and maintain an echelon formation with the human commander's aircraft, then ECCO determines the correct spatial relationship with the commander's aircraft, using the agent as a part in the echelon formation type, with instances of distributional attributes, and returns the vector notation for that relative position back to the agent. The agent then uses its own internal algorithms to translate the vector notation into appropriate flight maneuvers.

Human-AI Communication

The human and agent teammates *need a language with which to communicate* to establish, maintain, and repair common ground. Directives, questions, answers, and reports between the teammates must be encoded in a mutually understood form. As a starting point, the grammar is based on the ontology kernel's syntax. The lexicon of the controlled language consists of the names of all the entries in the ECCO knowledgebase. In our example domain of spatial relations relevant to human teaming with autonomous UAS, for example, the lexicon includes not just the relevant common spatial relations ("between", "beside", "on") but also technical terms for spatial relations that support the scenarios and shared tasks ("echelon formation", "wall formation") and supporting objects ("AeroVironment Wasp III", "MQ-1B Predator", "Point of Interest", "High Value Target", and others). In addition, the types associated with these lexical entries are defined by specific attributes and parts, which are themselves entries in the ontology and part of the lexicon, down to primitive types (e.g., [float], [integer]). The lexicon also includes all the names or symbols of all the type-specific algorithms in the ECCO reasoning engine, both those constructed for ECCO and primitive type-specific algorithms such as mathematical operations. The lexicon is thus potentially large and spans a broad hierarchy of specificity, from general types (e.g., [aircraft]) through increasingly refined types ([F-16 Fighting Falcon] and [Human Teammate's F-16 Fighting Falcon]).

Feasibility Assessment

We developed prototype ECCO software that implements a context data structures, ontological types, and type-specific algorithms. In a simulator, we created an operational vignette illustrating situated categorization (Figure 6). We asked subject matter experts in computer science, cognitive linguistics, human factors, and Government stakeholders to evaluate ECCO human-AI common ground. In the vignette, mission objectives influence the meaning of the concept [near]. Specifically, ECCO's current goal (the current mission objective) situates how ECCO understands its observation of a spatial relationship in response to a human teammate's query (e.g., Is the agent "near" the target?). Mission objectives were implemented as a [context] data structure. One objective was to identify Target 1 using a camera that required the agent to be within 0.3 miles of the target. A second objective was to use locate Target 2 using a camera that required a distance < 0.9 miles. A third objective was to listen to Target 3 using a radio that required a distance of < 1.8 miles. The concept [near] was represented as a function requiring three inputs: two entities and a critical distance. To ask ECCO if the agent is "near" a target, the human teammate executed a query at the ECCO command line: ASK (NEAR ECCO1 TARGET). ECCO extracts the entities (ECCO1 and TARGET) from the query, looking to the currently active [context] for assignment of the lexical entries "ECCO1" and "TARGET" to knowledge present in the [context]'s ontology. The query also prompts ECCO to categorize the observed relationship between ECCO1 and the target as instance of [near] and assess the observation's goodness-of-fit to the meaning of [near] as situated by the current mission objective. Because [near] requires a critical distance, ECCO must search the [context] for knowledge that expresses a distance relationship between the entities. That association is provided in details of the objective: observe a target using a particular method. The method itself (use of a particular sensor) contains information about the distance that the method requires.

ECCO correctly responded to queries about being near the three possible targets contingent on the current mission objective. Evaluation confirmed ECCO technical merits, demonstrating development of common ground processes at the level of conceptual structure. Specifically, evaluation showed establishment of common ground understanding of [near] as situated by a mission objective and maintenance of common ground as the objective changed. These common ground processes were enabled by technical successes, including: integration of ontology, agent, and simulator technologies enabling the system to direct agents to take action; distributional attributes and their use in the goodness-of-fit algorithm to assess likelihood that an observation is an example of alternative spatial configurations; an algorithm for goal-directed blending of known types into a novel type; and encoding of scenes in the ontology.

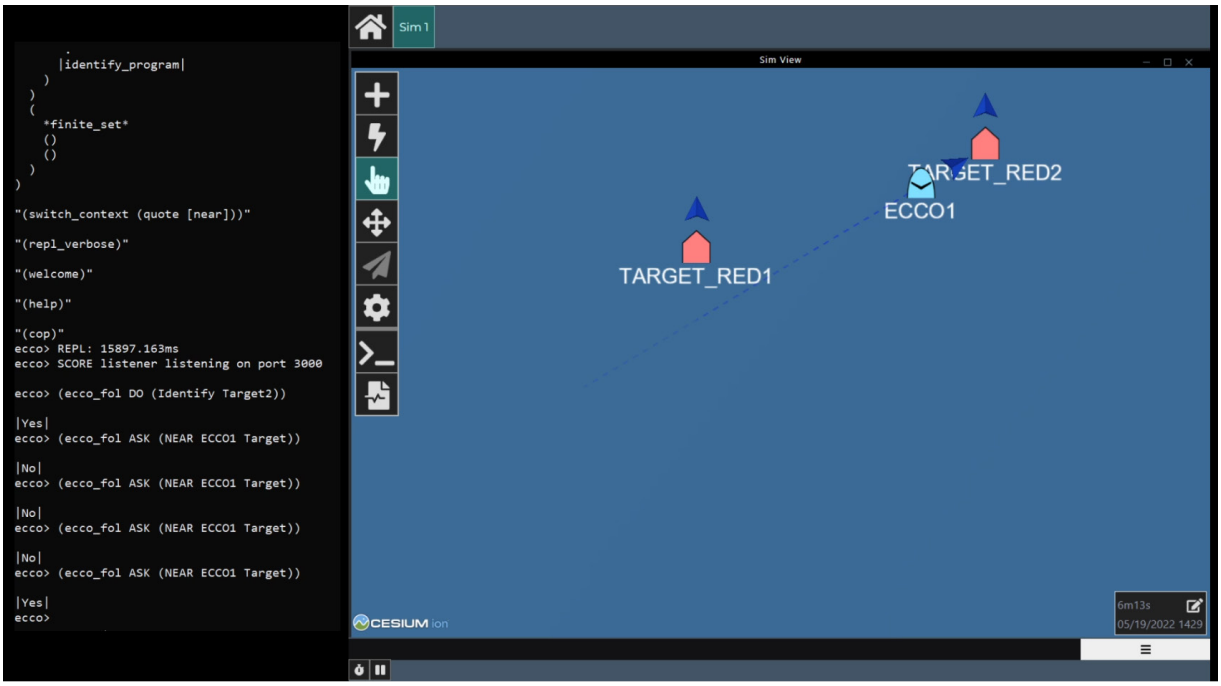

Figure 6: Screenshot from an evaluation of ECCO situated categorization of "near".

THE WAY FORWARD

The sharing of conceptual structures with the capability for blending, situated categorization, and degeneracy presents a significant opportunity to advance the state of the art in human-AI common ground. Augmenting existing AI tools with ECCO capabilities will enable humans to communicate at a more tactical level with their AI teammates on shared tasks. For example, applied to the domain of spatial concepts, these features combine to enable common ground understanding about relative, ambiguous, and novel relationships among objects in the environment. The conceptual, embodied, nature of the human-AI common ground made possible by these features affords shared understanding at a functional level (e.g., "get near the target"), rather than at a conventional computational tool's more mechanistic level of the actions needed to perform a tactic (e.g., "move to a waypoint').

These features support the use of a more natural-language style of terminology that is intuitive to the human, with the trade-off of some additional ambiguity of meaning that must be negotiated among the teammates. Instilling an AI with these features, and a mechanism for negotiating resulting ambiguity, provides a basis for the human to trust that the AI teammate understands its perceptions in a way that is functionally similar to how the human performs. Technical challenges to implementing the design principles described here include development of a sufficiently expressive ontology kernel, scalability of the system in terms of both deployment and domains of experience, and integration of subsymbolic and symbolic approaches to perception, representation, and reasoning. Successful implementation of an embodied, constructivist approach to conceptual structure and categorization such as that adopted here will facilitate additional opportunities for the advancement of AI, such as extensions to common sense reasoning and abstract conceptualization via conceptual metaphor.

It is worth contrasting the approach described here with those of transformer-based Large Language Models (LLMs, see Manning, 2022), such as OpenAI's GPT and Google's LaMDA, including image transformers, such as OpenAI's DALL-E and Stability AI's Stable Diffusion. Current endeavors to make LLMs serviceable in new domains (Bornstein & Radovanovic, 2023), such as internet-search, show that humans and LLMs often have different goals. For example, LLMs were not initially designed (i.e., imbued with a goal) to return factual responses to search queries, yet humans use LLMs with that goal in mind. This misalignment of "human-AI team goals" is a

lack of common ground: The human and the LLM both "perceive" the same human-supplied search query, but the LLM has a different goal than the human, which shapes its "understanding" of the query and guides its response.

As well illustrated in the popular press, LLMs produce compelling output, which is rich with blends. LLMs produce blends because human language contains phrases that are generated, in part, by concept blending (Turner, 1996, 2014). LLMs are trained to produce naturalistic sequences of words by ingesting vast amounts of natural language to learn what can come next, phrase-by-phrase. Therefore, LLMs blend by dint of having implicit knowledge of what can come next during sentence construction, having been trained on content that contains blends. However, language merely *expresses* blends, which reflect interactions among human concepts (in fact, language is a primary source of evidence for the theory of blending). LLMs are not blending parts of models (representations of concepts) nor using blends to understand their perceptions, reason about their perceptions, or take actions (Chomsky et al., 2023). Elsewhere, we have argued that generative transformer architectures are part of the solution to creating artificial organic-like cognitive architectures, but where LLMs generate surface behavior (language), the generativity of organic intelligence starts at perception and extends through action (Lynn et al., 2023). ECCO's design is intended to be an approach for increasing the utility of LLM-like architectures for human-AI teaming and common ground.

As contemporary human-factors approaches to human-AI common ground continue to mature, common ground issues will move from interface transparency to concept congruency. The representation and reasoning systems described here provide flexibility to adapt to changing circumstances, for example enabling the meaning of "near" to change as mission goals change. Furthermore, the typed ontology is designed to enable abstraction. For example, given a [context] that contained a refinement of physical distance, as an ontological type, to a new type, e.g., distance in a multidimensional mathematical space, ECCO would still be able to answer queries about nearness of two entities in that abstract space. As a generalizable framework, the ECCO approach is intended to model the structure and transformations of human concepts to provide an AI with knowledge, beliefs, and assumptions similar to those of a human teammate, facilitating trust in the AI. Beyond our UAS example, achieving common ground at the level of flexible conceptual structure would support training and operations in other areas where human and AI teammates must support each other to make best use of their distinctive strengths.

ACKNOWLEDGEMENTS

We are grateful to Richard Weyhrauch for his contributions to the design of the underlying ontology system, Score. We are grateful to Chris Muller for his engineering expertise in the prototyping of ECCO. We thank Rishabh Kaushik for his helpful manuscript review. This material is based upon work supported by the United States Air Force under Contract No. FA8650-22-P-6413 with the Air Force Research Laboratory, RHW/RHWM Cognitive Models Branch, 711th Human Performance Wing, Wright-Patterson Air Force Base. The authors thank Dr. Sarah Bibyk for her support and direction on that project. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the United States Air Force.

REFERENCES

Barrett, L. F., Wilson-Mendenhall, C. D., & Barsalou, L. W. (2015). The conceptual act theory: A roadmap. In *The psychological construction of emotion* (pp. 83–110). The Guilford Press.

Barsalou, L. W. (2015). Situated conceptualization. In *Perceptual and emotional embodiment: Foundations of embodied cognition* (pp. 1–11). Routledge.

Bornstein, M., & Radovanovic, R. (2023). Emerging architectures for LLM applications. *Andreessen Horowitz Enterprise Newsletter*. https://a16z.com/2023/06/20/emerging-architectures-for-llm-applications/.

Brennan, S. E. (1998). The grounding problem in conversations with and through computers. In S. R. Fussell & R. J. Kreuz (Eds.), *Social and Cognitive Psychological Approaches to Interpersonal Communication* (pp. 201–225). Lawrence Erlbaum.

Chomsky, N., Roberts, I., & Watumull, J. (2023, March 8). The false promise of ChatGPT. *The New York Times*.

Clark, H. H., & Wilkes-Gibbs, D. (1986). Referring as a collaborative process. *Cognition*, *22*(1), 1–39.

Dafoe, A., Bachrach, Y., Hadfield, G., Horvitz, E., Larson, K., & Graepel, T. (2021). Cooperative AI: machines must learn to find common ground. *Nature*, *593*(7857), 33–36.

Edelman, G. M., & Gally, J. A. (2001). Degeneracy and complexity in biological systems. *Proc Natl Acad Sci USA*, *98*(24), 13763–13768. https://doi.org/10.1073/pnas.231499798.

Endsley, M. R. (2015). *Autonomous Horizons: System Autonomy in the Air Force—A Path to the Future* (AF/ST TR#15-01). Office of the Chief Scientist, United States Air Force.

Gauthier, I., & Tarr, M. J. (1997). Becoming a "Greeble" expert: Exploring mechanisms for face recognition. *Vision Research*, *37*(12), 1673–1682. https://doi.org/10.1016/s0042-6989(96)00286-6. PMID 9231232.

Klein, G., Feltovich, P. J., Bradshaw, J. M., & Woods, D. D. (2005). Common Ground and Coordination in Joint Activity. In W. B. Rouse & K. R. Boff (Eds.), *Organizational Simulation* (pp. 139–184). John Wiley & Sons, Inc. https://doi.org/10.1002/0471739448.ch6.

Klein, G., Woods, D. D., Bradshaw, J. M., Hoffman, R. R., & Feltovich, P. J. (2004). Ten Challenges for Making Automation a "Team Player" in Joint Human-Agent Activity. *IEEE Intelligent Systems*, *19*(6), 91–95. https://doi.org/10.1109/MIS.2004.74.

Lakoff, G. (1987). *Women, Fire, and Dangerous Things: What Categories Reveal About the Mind*. University of Chicago Press.

Lakoff, G., & Johnson, M. (1980). *Metaphors We Live By*. University of Chicago Press.

Lynn, S. K., Loyall, B., & Niehaus, J. (2023). Growing An Embodied Generative Cognitive Agent. *Proceedings of the AAAI Symposium Series*, *2*(1), 315–319. https://ojs.aaai.org/index.php/AAAI-SS/article/view/27694.

Lynn, S., Koelle, D., & Wronski, R. (2020). Drone swarms: A transformational technology. *Aerospace & Defense Technology*, *May*, 14–17.

Manning, C. D. (2022). Human language understanding & reasoning. *Daedalus*, *151*(2), 127–138.

McDermott, P., Dominguez, C., Kasdaglis, N., Ryan, M., Trahan, I., & Nelson, A. (2018). *Human-Machine Teaming Systems Engineering Guide*. The MITRE Corporation. chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://www.mitre.org/sites/default/files/2021-11/prs-17-4208-human-machine-teaming-systems-engineering-guide.pdf.

Talcott, C. L., & Weyhrauch, R. W. (1990). Towards a Theory of Mechanizable Theories: I, FOL Contexts: The Extensional View. *European Conference on Artificial Intelligence*, 634–639.

Turner, M. (1996). *The Literary Mind: The Origin of Thought and Language*. Oxford University Press.

Turner, M. (2014). *The Origin of Ideas: Blending, Creativity, and the Human Spark*. Oxford University Press.

van den Bosch, K., Schoonderwoerd, T. A. J., Blankendaal, R., & Neerincx, M. (2019). Six challenges for human-AI co-learning. *International Conference on Human-Computer Interaction*, 572–589. https://doi.org/10.1007/978-3-030-22341-0_45.

Varela, F. J., Thompson, E., & Rosch, E. (2017). *The Embodied Mind, Revised Edition: Cognitive Science and Human Experience*. MIT press.

Weyhrauch, R. W. (1978). *Prolegomena to a Theory of Formal Reasoning* (STAN-CS-78-687). Stanford Artificial Intelligence Laboratory Memo AIM-3 16, Stanford University Computer Science Department. http://infolab.stanford.edu/pub/cstr/reports/cs/tr/78/687/CS-TR-78-687.pdf.

Weyrauch, R. W., & Talcott, C. (1997). *WristWatch—An FOL theory of time*. Unpublished manuscript. https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=5dd2524f17b8ac1763cba75a00c8c9b756b52258.

Zacharias, G. (2019). *Autonomous Horizons: The Way Forward*. Air University Press.