Agent-Oriented Modeling for the Age of AI – Nine Pivots towards a Reconceptualization of Requirements Engineering

Eric Yu
University of Toronto, Canada

Abstract Modeling techniques have been instrumental for conceptualizing and architecting complex systems. Since the early days of Requirements Engineering (RE), concepts such as *process* and *object* have served as core abstractions undergirding many modeling languages, and have seen extensive application in all kinds of domains. Today, software systems are engaging with humans in ever deeper, more intricate, sometimes even intimate ways. They leverage pervasive digitalization to learn from data, acquiring increasingly human-like characteristics. An abstract concept of *agent*, proposed much earlier in AI for characterizing artificial agents, suitably extended and generalized, could potentially serve as a core abstraction for requirements modeling for today's complex sociotechnical environment. In this paper, we outline a modeling ontology by elaborating on the sociotechnical actor concept originally proposed in i*. The proposal also suggests shifting the focus of RE from the specification of a target technology system to one of assisting stakeholders to explore how they might augment their "selves". A personal wellness app is used as a motivating example.

Keywords requirements engineering, agent-oriented modeling, intentional stance, goal modeling, iStar, aaIstar

1 Introduction

Digital technologies have vastly transformed everyday life in recent years. Today, further advances in machine learning (ML) and artificial intelligence (AI) offer promises of dramatic benefits, yet could pose serious threats to lives and livelihoods.

For requirements engineering (RE), the challenge is to provide methods and techniques to help stakeholders explore the space of potential technology solutions so as to identify those that offer desired benefits while minimizing potential negative consequences.

Most RE techniques in use today were conceived in earlier eras, and are ill-equipped to respond to today's much richer and faster moving environments. In the past, most software and information systems offered solutions that are task-oriented and narrowly confined, often in work or business-related settings where users were trained to use the system. Requirements techniques aimed to define intended functions and interactions precisely and in great detail.

Today's "smart" machines are increasingly general purpose, and aim to respond flexibly to a wide range of user needs. With advanced machine learning algorithms and models, machines are able to recognize human behavioral characteristics, and adjust their own behavior accordingly. As systems today incorporate more advanced ML and AI components, we can expect them to engage with humans in increasingly human-like ways.

From a modeling perspective, RE techniques need to be able to adequately describe and analyze the kinds of services and capabilities that today's systems offer, and how problems, issues, and concerns that arise can be addressed. They need to map out the space of potential technology solutions, and to analyze their implications and consequences, so that stakeholders can make informed decisions.

The agent concept had been used as an abstraction for characterizing machine behavior since the early days of AI [44]. The i* framework [67] adapted the concept for the purpose of requirements modeling and analysis. In this paper, we argue that an agent-orient modeling approach is well suited to address the RE challenges in the age of AI. We put forward a vision for extending the agent-oriented modeling approach as initially proposed in i* [66, 64]. The proposal is presented in terms of nine "pivots", together constituting a transition from the prevailing way of thinking about RE to a new conceptualization.

2 A Motivating Example

Consider a health and wellness app that can be personalized to help you stay healthy and fit [12, 9]. By emphasizing prevention, it has the potential to reduce healthcare costs, overcome shortages of health professional, and improve outcome for an entire population. It draws on personal health records and population health data, as well as ongoing advances in medical knowledge and expertise. It taps into live data on your wearable and mobile devices, and gives you timely advice on your dietary choices and suggestions on how to stay active. To make effective suggestions, it adapts to your predilections and inclinations, and can sense and help you manage your emotions. Your healthcare team is kept up-to-date on your conditions and will take proactive action when necessary. Your health insurer will reward you for avoiding expensive procedures.

What features of such an app would you want or not want? What aspects of your daily life would you be willing to expose to the app in return for what benefits? If you are the family physician, how would you want the app to participate in the care of your patient, and how would your experience and expertise contribute? If you are the regulator, how would you ensure that the service would not be harmful to individuals and would serve the public interest? If you are the developer or service provider, how would you balance the various interests to achieve a viable and sustainable business model? How would you ensure that the system is functioning properly as intended and whether it is living up to various stakeholders' expectations? [26]

There are numerous opportunities to incorporate AI into the app and supporting platform, from statistical ML to generative AI using foundation models. During requirements analysis, stakeholders would want to assess the potential benefits and pitfalls of various technology options.

3 From Process to Objects to Agents

Models are necessarily simplified representations of reality. They simplify by omitting unnecessary details while capturing essential aspects. Requirements modeling languages typically offer a small number of concepts (comprising the modeling ontology [40]) based on abstractions that are considered sufficient for the purpose of requirements analysis, while "abstracting away" the rest.

In the early days of computing, the concept of process served as a powerful core abstraction for requirements modeling [52, 17]. In the era of interactive and distributed computing, object orientation became the dominant modeling paradigm [15]. Today, as software systems interact with the world in increasingly human-like ways, the concept of agent holds promise for modeling how machines relate to the world and to each other.

The agent concept was first proposed as an abstraction for AI by Allen Newell [44]. It offered a higher level characterization of the behavior of a machine, in terms of the goals that it can achieve. It served as a specification for building AI agents, and for reasoning about how agents behave at a high level independently of software implementation mechanisms.

The i* framework adapted the agent concept for the purpose of requirements modeling and analysis, for reasoning about how machines are situated in the world in a social setting. i* actors relate to each other at an intentional level. They depend on each other for goals to be achieved, tasks to be performed, resources to be furnished, and quality goals (softgoals) to be satisfied. When an actor is depended upon to achieve a goal, it has the freedom to choose among alternative ways for achieving the goal.

Intentional relationships allow for a high-level way of describing how actors relate to each other. This is well suited to the age of AI, where machines increasingly have the ability to do what is expected, without prior specification of every interaction in detail.

While the i* framework could serve as a foundation for requirements analysis, it abstracted away aspects that have become important today, particularly concerning rich human-like behavior such as learning. In the paper, we outline how we might extend the agent-oriented modeling approach to meet today's challenges.

The transition from process to object to agent orientation can be viewed as a natural progression of requirements modeling paradigms [64]. Each paradigm relies on abstractions that make simplifying assumptions about the world. Each step in the progression can be seen as an incremental removal or relaxation of earlier assumptions so as to be able to cope with the increasingly complex ways that today's powerful technology systems engage with the world.

In the process modeling paradigm, the analyst/designer has a full view of the entire end-to-end process, has control over how any parts of it is to be redesigned, and can globally optimize. In the object-oriented paradigm, behavior is distributed and localized. Local information is encapsulated and hidden. The prior assumptions of the omniscient and omnipotent analyst/designer is replaced by localized decision making. The criteria for decision making, however, are not captured in process models or object models. Intentionality resides in the analysts/designers. In goal-oriented requirements modeling [41, 60], intentionality is made explicit in the model, and becomes available for analysis. In agent-oriented modeling, such as in i*, intentionality is distributed and localized to each actor. Each actor is autonomous, although their freedoms are constrained by dependencies among them.

4 Nine Pivots towards a Reconceptualization of RE

The proposed vision is premised on an ontology[1] of the world revolving around strategic actors and their relationships, rather than processes, objects, or user stories or scenarios that are the mainstay of current RE practices [49, 42]. The first two pivots recall the central ideas behind i*. Each of the subsequent pivots elaborates and extends the agent-oriented modeling paradigm to encompass a richer ontology and a broader vision of RE. The proposal is meant to stimulate further developments in requirements modeling concepts and languages, agent-oriented or otherwise.

Pivot 1 – From unidirectional stipulations to multilateral relationships

Requirements are often thought of as “shall” statements about what users and stakeholders want from a target system and how the system should behave, to serve as guidance and objectives for developers to meet [30]. To arrive at such statements, the various parties would want to explore different options to gain understanding about the implications of the choices they make. As technology systems offer new and unfamiliar functionalities and venture into new application areas, this exploratory stage of requirements analysis becomes more important [66], as there are greater uncertainties about potential benefits as well as vulnerabilities and pitfalls.

i* thus shifts focus from the usual understanding of requirements as prescriptive or optative statements directed from the client to developers, to the analysis of the multilateral relationships among stakeholders and machines, and the changes to those relationships arising from new technology solutions. Stakeholders aim to advance or protect their strategic interests as technology proposals are considered. The relationships are expressed at the intentional level, leaving detailed interactions to be specified during the later stages of requirements definition. With highly flexible and general-purpose AI applications, specification of detailed behavior may not be possible or necessary.

---

[1] The term ontology is used in this paper to refer to the set of concepts underlying a modeling framework [40]. The paper outlines a skeletal ontology, leaving issues of language design open for future work.

In i*, stakeholders are modeled in terms of Actors[2]. Actors achieve goals by performing tasks with available resources. Alternative ways for achieving goals are differentiated by how they contribute towards quality goals (softgoals). By depending on others, an actor can take advantage of opportunities beyond its own capabilities. At the same time, it becomes vulnerable to potential failures beyond its control.

Actors are intentional – their actions are motivated by goals and softgoals. They are autonomous in that they have freedom to choose among alternative ways for achieving goals. They are social in that their freedoms are constrained by dependencies with other actors. They reason about alternatives as best they can, and seek to advance their strategic self-interest by leveraging opportunities while mitigating vulnerabilities [66].

In the wellness app, each of the stakeholders – the wellness seeker, the physician, etc., would aim to benefit from the technology, while avoiding negative consequences. They would have dependencies on the machine, but the machine also depend on the people around it if it is to function as intended. During requirements exploration and negotiation, each stakeholder would assess whether a proposed configuration of relationships is to its advantage, and will seek to advance or at least protect its position. They would assess the viability of goal achievement by analyzing how success or failure would propagate across the network of dependencies and rationales.

Pivot 2 – From AI agents to a generalized actor abstraction

In AI, the term "agent" refers to a computational entity – a machine [53, 62]. In humanistic disciplines, agent often refers to a human individual. For RE, we seek abstractions that allow for decisions about the automation boundary to be made during RE, not as prior givens. In process modeling as well as object-oriented modeling, processes and objects can be analyzed at an abstract level[3] independent of implementation considerations, so that allocation of responsibilities to humans or machines can be determined as a result of the RE process. The same principle is also recognized in the goal-oriented RE approach [22].

In i*, the Actor is an abstraction that is independent of the medium or mechanisms of its eventual realization[4]. The abstraction adopts an external view and at-

---

[2] In this paper, terms such as Actor, Role, and Agent are capitalized when used as defined in i*. When uncapitalized, their meanings are as in common usage.

[3] In the Structured Analysis method based on process modeling [17], the abstract level is referred to as the Logical Model, in contract to the Physical Model. These are analogous to abstraction levels in data modeling.

[4] Although the i* Actor abstraction was partly inspired by the concept of agent in multi-agent systems and agent-oriented software engineering [62, 35], and appear to share some of the same properties, e.g., autonomy, sociality, etc., these terms have different connotations in their respective contexts [65]. An autonomous agent in AI is a machine that has the ability to act intelligently on its own

tributes intentional properties to an entity regardless of its internal makeup or mechanisms [18]. It is therefore equally applicable to humans and machines, or any combinations thereof, of any size or complexity. Machines are attributed intentionality and agency because they embody design intents and rationales. By explicitly representing intents and rationales in requirements modeling, the impact of design options on stakeholders can be explored and analyzed.

While analysis at an abstract level allows design options to be explored with the greatest flexibility, the abstract entities ultimately need to be related to their potential realizational counterparts. i* modeling provides for abstract Actors[5] (called Roles) as well as realizational[6] Actors (called Agents) to be included in the same model, so that dependencies among them can be recognized and analyzed. In the age of AI, where many automated systems operate based on probabilistic principles, one can expect that Agents may not fully meet the expectations defined in Roles. In traditional programmed software, even where functional requirements are fully met (sometimes supported by proofs of correctness), the non-functional or quality requirements may be met only up to a certain degree [14]. In the case of humans, roles often serve as aspirational guidance for actual behavior.

In i* modeling, the realization relationship from Agent to Role is denoted by the PLAYS association link. An Agent can depend on another Agent for achieving its goals, but those dependencies can be mediated by a Role, so that different candidate Agents for PLAYing that Role can be considered. Note that since Roles and Agents are autonomous Actors, they can have mutual dependencies, unlike in conventional software architectures where intentionality flows one-way along the realization relationship[7]. By using IS-A hierarchies to distinguish among different types of Agents, classes of machines and humans with various capabilities and characteristics can be defined, e.g., those with and without learning, different types of learning, etc.

---

without human intervention. Sociality refers to machines cooperating and acting collectively. In i*, these terms refer to properties of abstract Actors regardless of their realizational mechanisms.

[5] The terms agent and actor are used with a variety of connotations in various disciplines, with different degrees of abstraction. In the human and social disciplines, an agent or actor often refers to a human individual, but can sometimes refer to a larger social entity, such as an organization or a nation state. In AI, an agent is an artifact to be implemented in software. In i*, Actor, Role, and Agent are concepts for use in modeling so are therefore all abstractions. In i* models, they generally refer to classes, but can also appear as instances of classes. The Actor is the generic concept, with Role and Agent being specializations. They can be further specialized into subclasses.

[6] We refer to an Agent as being realizational rather than realized, since it is still an abstraction. When a Role is PLAYED by an Agent, the relationship takes the Role one step closer towards realization.

[7] Top-down in a layered architecture. Lower layer components are used to meet the objectives of higher layer components.

Pivot 3 – From unitary actor to composite of multiple autonomous actors

It is commonplace to think of a person as having one or more roles at work – project lead, software developer, etc., in addition to multiple roles in personal and social life – parent, citizen, fitness club member, etc. We learn to manage the competing demands from these roles, each representing different sets of interests and values. Different individuals may live up to the expectations of these roles to varying degrees. In simpler times, when software applications were narrowly focused, each stakeholder could be treated as a unitary actor with a single set of more or less coherent goals and strategic interests. In the age of AI, we are increasingly dealing with broad-based open-ended applications, possibly trained on all-encompassing data sources, as in the case of large language models (LLMs). Social psychologists have studied in great depth how humans manage multiple identities and conceptions of selves [59, 54]. In any given context, one or a small number of one's selves may be salient, i.e., dominate over others.

For RE, we propose to model multiple identities by treating any Actor as potentially composite, with each constituent Actor being a full-fledged Actor like any other (i.e., has all the properties of intentionality, autonomy, sociality, etc.). RE analysis could then be as fine-grained as desired. The wellness seeker, in addition to a StayingHealthy Role, may have other Roles such as BecomingWealthy. The wellness app may well be discerning among these multiple identities (the healthy versus the wealthy person), and targeting them selectively to effect behavioral change. It may offer an AI agent that nudges the StayingHealthy role to adhere to an exercise or diet regime.

We use the PART association link to denote the part-whole relationship between the composite Actor and its constituents. An Actor could be a Role, or an Agent capable of realizing a Role. Goal analysis is conducted by propagating goal satisfaction values through explicitly defined intentional dependencies between any kind of Actor, including between Agent and Role, between a part and the whole, as well as among the parts. By including a Role and the Agent PLAYing that Role as constituent Actors within a composite Actor, we are able to analyze the mismatches between Role and Agent, which appear as unsatisfied dependencies.

Beyond the human individual, we apply the composite autonomous Actor concept to sociotechnical structures of any size and complexity, comprising of Agents and Roles that may or may not yet exist, and which in turn may be composite[8]. The composite Actor concept can therefore be used to model an organization normally thought of as being made up of human and technology elements.

[8] By conceptualizing the Actor as a generalized abstraction, it can be recursively refined or composed, as in the generalized abstractions of process and object. Note however, that compositionality of the generalized intentional Actor is a more complex concept, as each Actor is autonomous and has agency.

By considering the Roles that the human and machine Agents are PLAYing, one could explore whether those Roles, or some of its subparts, can potentially be realized through different combinations of humans and technology. In the wellness app, the LoggingPhysicalActivities Role, previously played by the human, can now be played by the wellness app equipped with sensors. Whether the Nudging Role would be better played by the app or by another human may depend on the context and could be assessed empirically.

Pivot 4 – From separate to intertwined development and use

In the past, software took years to develop, and once delivered, would be updated infrequently. Development processes were modeled and analyzed separately from the processes in which the systems are used, such as business processes, and typically using different modeling languages and analysis techniques, e.g., SPEM [45] vs. BPMN [46]. In the digital age where usage is constantly monitored to inform developers, there is a much tighter coupling between development and use. Users' expectations on functionality and quality evolve continuously. There are numerous decisions and choices during development that can result in success or failure during usage. When systems do not behave as expected, users often adapt the system or devise workarounds to fit their own processes.

RE modeling can be used to expose and analyze how choices made throughout development and use affect each other. Following the agent-oriented approach, we model development and usage in terms of strategic Actors, with dependencies that can crisscross among development and use. With a generalized Actor concept, we aim to cover a notion of development that is applicable to human development as well as the development of machines.

We introduce an association link A DEVELOPS-THRU B to mean that Actor A develops through the efforts of Actor B. The direction of the link suggests that, even though A is the beneficiary of the development efforts of B, it is A that wants to be "developed", to acquire new or enhanced capabilities. This is consistent with the notion of a person going through development, e.g., through professional training programs. Both A and B are autonomous actors, and would assess the relationship strategically in terms of opportunities and vulnerabilities. This is in contrast to the conventional view in systems development, where A, normally thought of as the user, is a passive recipient of the development efforts.

We refer to A as the Developient and B as the Developier (words coined to dissociate from the conventional concepts of user and developer, but to rhyme with recipient and developer for mnemonic aid). The terms refer to the two ends of the association relationship, rather than special kinds of Actors. This leaves the specifics of the "development" process in B and the "usage" process in A entirely open, so as to accommodate any kinds of development methods. Similarly, the usage process is unconstrained, and can include unintended use cases and even misuse. RE analysis would aim to uncover issues and alternative solutions from the strategic viewpoints of stakeholders. A and B can be refined into sub-Actors for detailed analysis.

In the digital age, we as users of technology frequently evolve our own practices in our various Roles, as we integrate various systems and services into our lives, which themselves are evolving all the time. By modeling development and use within the same framework, RE analysis can examine and reconsider the allocation of responsibilities and decision rights between development and use. For example, in considering alternative design for personalization, a wellness app that automatically detects and infers user preferences may be simpler to use, but renders the user feeling less in control.

Pivot 5 – From separate to intertwined learning and doing

Recent dramatic advances in AI are achieved by applying machine learning algorithms on massive amounts of data. It is essential for RE to offer suitable abstractions that will enable stakeholders to assess technology systems today as they incorporate AI. As these systems become deeply embedded in human lives, it is equally important to include human learning in RE analysis.

We take the same approach for modeling learning as for development (Pivot 4), in order to support analysis of strategic relationships among autonomous Actors capable of learning.

We introduce LEARNS-THRU as an association link from A to B to mean that B contributes to the learning of A. We refer to A as the Learnient and B the Learnier (mnemonically rhyming with recipient and learner, but distinct from them in meaning). The link indicates a relationship between two Actors relative to each other, without presupposing the properties of A or B. Analysis is conducted through intentional dependencies explicitly defined between the two Actors. This approach leaves the specification of how learning takes place entirely open, so as to accommodate all kinds of human and machine learning.

The Learnient can be thought of as the user of the results from Learnier, integrating the learning into its own activities, the “doing”. Both Actors, being strategic actors with agency, make choices according to their own best interests. The Learnient would assess how Learnier’s way of learning would be beneficial or problematic, while the Learnier may adjust its practices to respond to Learnient’s needs and concerns if that suits its own interests.

A wellness app that learns from user behavior to improve its effectiveness could be modeled as having a learning Role (the Learnier side) separate from the doing Role (the Learnient side), so that their dependencies to and from other actors can be separately identified and analyzed.

As in the case of development, RE analysis can help redistribute the kinds of choices open to each side. For example, the Learnier may restrict what is learned to a narrower context, or leave more room for interpretation or adaptation by the Learnient. More detailed analysis can be conducted by elaborating on sub-Actors on each side. A drill-down into learning can reveal where biases originate, and how they might be mitigated by other Actors during development or use. Elaboration on the doing or usage side might uncover how what is learned could be misused.

We treat learning as similar to development in that in both cases, the beneficiary, the Learnient and the Developient, gain capability. We provide the two links (LEARNS-THRU and DEVELOPS-THRU) to correspond to the distinction between software systems gaining capabilities through machine learning from data versus through explicit programming of encoded knowledge, respectively. The former is inductive and relies on instances and examples. The latter is deductive and relies on concepts and theories. In the case of humans, they correspond to learning through experience versus through descriptions or instructions. A person who follows a fitness exercise regime based on explicitly articulated instructions and schedule would be considered to have undergone "development", whereas if he acquires a pattern of regular workouts through the habituation, we would consider that to be "learning".

The LEARNS-THRU and DEVELOPS-THRU links can be used together, and can also be applied to humans and machines in any combination. This allows software systems created through a mix of machine learning and programming to be analyzed in their human social context, including, for example, human-AI collaboration in a business process, or AI-enabled enterprises. They can also be used to examine alternative methods and associated tools for creating ML and AI solutions, compare their relative strengths and shortcomings for different types of applications, and to trace problems and issues to their contributing sources through the network of dependencies, and to locate Actors that are responsible, or capable of remedial actions. Such analysis would be important for achieving trustworthy and responsible AI.

For instance, to generate effective reminders to nudge a busy user to get active from time to time, the wellness app could use ML models learned from that user's activity patterns, combined with best practice heuristics developed by health experts. The Reminding function (a Role PLAYed by WellnessApp) LEARNS-THRU Reminder-ML-process, and DEVELOPS-THRU AppDevelopment process, which has a dependency on HealthBehaviorExperts for knowledge about effective nudging.

Pivot 6 – From detached to empirically grounded Actors

In the age of AI, systems rely heavily on diverse types and sources of data to respond to changing conditions in the world. Many software systems are created by observing actual results from repeated experimentation, rather than by applying well-worn techniques with predictable results. These include online experiments like A/B testing, and algorithm selection and hyperparameter tuning in machine learning [21].

We introduce the concept of Sensors and Actuators in an Actor's intentional model so that it can refer to what happens to the world. They are tagged as such in the literals (names) of intentional elements (Task, Resource, etc.), and serve as interface points to the non-intentional, causal world.

A Sensor variable is an input to an Actor from a causal world, i.e., the Actor can decide what to do based on the value of the sensed variable. They may appear as a Resource in an i* model, or as a Situation or Indicator following BIM [28].

An Actuator variable is an output to the world. The Actor can manipulate the value of the variable, through an i* Task.

In cases where models about the causal world exist to complement the intentional models of strategic Actors, Sensors and Actuators would refer to named elements that are variables in those models, e.g., in causal loop diagrams of system dynamics, or in systems of differential equations.

During RE, one would analyze vulnerabilities and opportunities in the ways that an Actor interacts with the non-intentional world, in addition to the intentional relationships it has with other Actors. Feedback and adaptive architectures are important in many contexts, including for example, continuous software engineering, devOps, MLops and the data-driven enterprise [21]. Observations and interaction with the non-intentional empirical world are also essential for human learning.

The wellness app could sense and respond to its application context (health results, healthcare team actions, user behavior, etc.) in many different ways. By extending intentional modeling with sensors and actuators that interact with the causal world, the RE analyst would be able to analyze the implications of alternative sense-and-respond configurations from the viewpoint of strategic interests of various stakeholders.

Pivot 7 – From atemporal to temporally-contextualized Actors

When Actors make choices and face trade-offs, they do so within a temporal context. Short term decisions may be less well informed and hurried. Decisions for the longer term may face greater uncertainty and psychologically feel more distant. Trade-offs may need to be made between short term effects and long term consequences.

When multiple Actors relate to each other intentionally, it is important to recognize that each Actor may operate on a different time scale or specific time frame. One might respond differently to an urgent complaint from a long-standing customer, compared to an inquiry from a new customer about a possible project next year. Decisions made in the context of the Project Manager Role may be in the time frame of each project, whereas the Agent playing the Role may include career-long considerations in its decisions. The wellness app user and various healthcare team members, in their multiple roles, would each be making choices and trade-offs in their respectively different temporal frames.

For the purpose of RE modeling and analysis, we associate each Actor with a temporal frame. A temporal frame may define a finite time horizon or a recurring cycle with a certain frequency. It may be delineated in absolute time or by start and end conditions. It may also define a granularity so as to filter out high frequency noise, and exclude variations that are too slow to be considered significant. Iterative cycles may aim to converge towards a terminating condition, based on sensing and actuating within each cycle.

Pivot 8 – Beyond goals – values, norms, and emotions

The goal-oriented agent abstraction from early AI [44] provided the inspiration for the i* Actor. This exclusive focus on goal-guided behavior is an oversimplification that is no longer adequate for today's RE modeling needs.

In the age of AI, software systems increasingly aim to appear to behave in human-like ways, for smoother interactions and to achieve desired effects and to gain acceptance. Software systems not only can recognize human emotions and values and respond accordingly, many applications aim to trigger emotions and shape values. Furthermore, as AI derives its behavior by learning from humans, they will embody human biases and emotions.

Humans also need to be modelled much more richly than in the past, since we interact with machines increasingly on human-like terms,. For humans, the default mode of interaction is reactive and intuition based, which is fast and automatic, rather than deliberative goal-oriented thinking, which is slow and effortful [36, 20]. Values, which are deeply held beliefs, can override strategic interests.

We further extend the Actor abstraction that we have proposed so far to include behaviors that are driven by values and emotions, besides goals and interests. As before, this extension applies to the generalized Actor, encompassing humans and machines on any scale, with development and learning abilities, and so on.

In the age of AI, while considering technology solutions, stakeholders want to know whether and how their values and emotions will be detected and exploited or manipulated and for what purpose, and whether their interests will be enhanced or compromised. They would want to explore alternative options for leveraging values and emotions for positive gains while minimizing vulnerabilities.

By modeling sociotechnical configurations in terms of strategic dependencies among Actors with values and emotions, RE analysis can assess the implications of alternative configurations for various stakeholders. They can explore opportunities and inquire how and where things can go wrong, and what mitigation strategies might be sufficient. The effectiveness of wellness and healthcare applications could be significantly affected by differences and variations in cultural and professional values, as well as in emotional responses, among the many roles and participants.

How values and emotions can best be represented for the purpose of RE modeling will need to be explored. While values might be treated as a kind of goal [58], one would also want to be able to analyze how values get transmitted across Actors. Values could potentially be treated as a kind of belief that gives positive or negative support to different ways for achieving goals, as in the concept of Claim in the NFR framework [14], and similar to other argumentation frameworks. By extending i* with a Belief Dependency, the effects of beliefs can be propagated across Actors using the same qualitative reasoning procedures as for goal reasoning [29, 4].

Pivot 9 – From requirements on machines to augmentation of selves

In the conventional view, requirements specify the relationship between the target machine and its environment [10]. In Pivot 1, we proposed to shift perspective to think of RE as guiding strategic actors in their search for reconfigurations of relationships that would advance their strategic interests. In that process, we had assumed that the Actors themselves remain unchanged. A target technology system is treated as a newly introduced strategic actor in their midst, resulting in reconfigured relationships among all Actors.

Today's AI-enabled systems have the potential to engage with humans much more deeply and can have far ranging impacts. We thus propose to take a further step, to think of the requirements process as guiding Actors to examine and transform their selves, incorporating technology into the self where appropriate. We will refer to this vision of RE as aaI* (or aaIstar), short for "Augmenting Aspiring Selves," to highlight the RE task as that of helping stakeholders to examine and augment their selves, and to search for or create Agents that aspire to realize the ideals defined by Roles. The term "augmenting" rather than "augmented" is used to suggest an ongoing and possibly never-ending process.

Augmenting the self by "incorporating" another autonomous Actor (such as a black box AI agent) is analogous to a company acquiring another company, and is subject to similar integration challenges, including alignment of interests and values, and dealing with potential third party allegiances of constituent actors.

An analysis of the self, as applied to all relevant stakeholders, regardless of how they are constituted, is part of the journey in self-transformation, and so is the exploration and search for technology solutions. The notion of "self" is applicable to each Actor, which, being a generalized abstraction (Pivot 2), can be abstract (Role) or realizational (Agent), can be any combination of humans and machines, and can be composite and potentially analyzable in terms of constituent Actors (Pivot 3).

During RE inquiry, each Actor would ask of itself: (from Pivot 1) How are my possibilities for action enabled and/or constrained by strategic dependencies with other Actors? (from Pivot 8) How are my values and beliefs influenced and shaped, and by whom (belief dependencies)? Am I attempting to shape other Actors beliefs or manipulate their emotions? (Pivot 4 and 5) How did I come to be? Who and what do I depend on for my development and learning? (Pivot 6) How am I acting on the causal world and sensing the outcomes? (Pivot 7) What temporal frames do I operate in? How am I situated temporally in relation to other Actors?

This self-reflection provides the basis for exploring alternative configurations of relationships with other Actors, and the boundary of the self – How else can things be? What would be a more desirable self? The notion of agency refers to the ability to make choices to respond to or control the environment [6]. Given the opportunity, Actors would seek to strengthen or augment their agency, leveraging technology if necessary.

Taking advantage of the generality of the Actor abstraction (Pivot 2), the analysis of self and exploration of potential future selves can be conducted at an abstract technology-independent level (in terms of Roles), avoiding the common pitfall of favoring particular technology solutions prematurely at the expense of considering further alternatives. Envisioning possibilities at an abstract level transcends material limitations inherent in realizational Agents, such as humans and specific classes of machines. Technology alternatives can be considered in terms of various specialized classes of Agents capable of PLAYing the Roles.

The general Actor abstraction also allows the examination of self to be conducted at multiple levels of granularity. A composite Actor (Pivot 3) can be examined as a single autonomous actor. The same analysis can be applied to each of its constituent Actors, and so forth recursively. Opportunities for self-transformation and augmentation may exist for many of these Actors, sometimes at the expense of others[9]. RE methods are needed to help navigate and manage the analysis, as in the case of complex organizations of human and technology elements. In the age of AI, RE processes are ongoing cycles of transformation, resulting from dialogue and negotiation between what is desirable (Roles) and what is possible (Agents).

In the case of the wellness app, Actors to be considered might include the health conscious person's Roles and realizational Agents pertaining to his health and fitness, family and social life, work life, and those pertaining to persons he interacts within these roles, Actors associated with his healthcare providers, and the developers and service providers of the app and platform, and perhaps regulators and policy makers. The RE analyst would work with stakeholders to determine the breadth and depth to which the network of dependencies and associations among Actors should be pursued, to the extent that their decisions would be affected. The characteristics of these Actors may depend on the local culture, demographics, and regulatory and competitive environments.

5 Discussion and Related Work

Requirements engineering bridges the needs and wants of users and stakeholders on the one hand, and the capabilities of technology systems on the other. Much attention is currently being paid to how RE techniques can encompass the distinctive characteristics of recent AI technologies, under the rubric of RE4AI [1]. As AI becomes widely incorporated into all kinds of systems, RE techniques must be able to analyze how systems made up of technologies of all sorts, AI or otherwise, affect people's lives, so that stakeholders can make informed choices.

We have proposed to model complex sociotechnical systems using a unified concept of Actor, sufficiently abstract and generalized to encompass sociotechnical actors of any composition, while specializable through IS-A hierarchies to distinguish among humans and machines of various types and characteristics.

The Actor abstraction can be seen as a further evolution of the object abstraction. Objects localize and encapsulate data and behavior, whereas Actors localize intentionality and choice. Objects collaborate by exchanging messages. Actors, through intentional dependencies, expand the space of options for each other, but also constrain each other's freedom.

The social science literature provides an extremely rich reference source for developing and refining the Actor abstraction. The concept of agency has been extensively explored (e.g., [19, 6]). Social psychologists distinguish among individual self, relational self, and collective self [54]. Many theories of identity have been

[9] As each actor seeks to augment its self, it may be considered harmful or malevolent by some other actors. The modeling framework makes no assumptions about which goals are beneficial or deleterious. The analysis is conducted from the viewpoint of each stakeholder.

developed [48]. The concept of role appears in many forms in various theories. The Role concept in i* is a simplification. The imperfect realization of a Role by an Agent may be compared to the concepts of identity standard and identity verification in identity theory and affect control theory [23]. William James [33] distinguished between the self as subject (I) and as object (me), and hinted at a notion of extended self that can include artifacts. The “extended self” concept has been applied to modern marketing [8] and has been discussed in the context of virtual assistants [38].

Psychologists recognize that emotion is a more fundamental driver of human behavior, whereas deliberative thought and conscious decision making is cognitively demanding and slow [20]. It has been proposed that the fast and slow thinking of Systems One and Two [36] be viewed as separate minds [2]. RE researchers have drawn attention to the importance of including values and emotions in requirements analysis [55, 32]. Values have been treated as goals [50]. Emotions are especially important in RE for security [63], gaming, gamification, persuasive technology, and affective computing.

Time frames and horizons are important factors in organizational decision making [47]. Technical debt in software engineering is a manifestation of intertemporal choice and trade-offs [7].

There is much work on RE for adaptive systems, including goal- and agent-oriented approaches [39, 5]. The current proposal suggests to provide only interface points to dynamic models, allowing for different types of system dynamics modeling to be used to complement the intentional modeling of sociotechnical Actors.

The proposal to model learning and doing, as well as development and use, as intermingled processes draws inspiration from the modeling of DevOps [16], and from the treatment of variability in software engineering, where the binding time of variants can be allocated to different points along the multiple stages of development [56]. The proposal is distinctive in aiming to encompass both machines and humans, and learning and development together in a single modeling framework.

Reconceptualizing RE as self-reflection and reinvention can be seen as a generalization of the call for businesses to rethink who and what they are as they pursue digital transformation and business model innovation [57, 37]. By asking themselves repeatedly – “what business am I in?” [51], an online book seller could end up becoming a e-commerce platform and a technology company; a DVD rental company could become a streaming service then a major movie producer, and an athletic footwear company can also become a fashion leader. RE using a generalized Actor abstraction would suggest self-inquiry and prospecting for opportunities at multiple levels at the same time, from individuals to groups to organizations, each constituting of multiple selves, with interactions and dependencies among them and across levels.

The notion of augmented intelligence has been a long running theme in connection with AI, advocating the use of machine intelligence to complement and amplify human capability rather than to compete with or supplant human intelligence [68, 69]. The proposal of the augmenting self, as articulated in Pivot 9, may be seen as a generalization of the notion of augmentation adapted for the purpose of RE. When applied to a generalized Actor abstraction, augmentation refers to the strengthening or expansion of the agency of an Actor, without prior assumptions about eventual realization, and applicable to Actors of any size or scale (Pivot 2). Agency may be interpreted as the capacity of the self to control the environment [6]. The proposed notion of the augmenting self thus does not rely on any conception of intelligence, and is independent of prior assumptions about human or machine capabilities. Using this abstraction for requirements modeling avoids prejudging relationships among humans and machines, allowing alternative configuration of human and machine elements to be explored and determined during RE.

The current proposal aims to benefit from the extensive and diverse literature on these topics to contribute towards a unified requirements modeling framework centered around the core abstraction of Actor.

6 Conclusions and Future Work

As AI continues to gain ground in the computing landscape [43], further accelerating the already massive transformations of the digital era, many brace for the oncoming onslaught of the new technology with anticipation and trepidation. Will I become more productive and valuable, or will I become redundant? How will I share responsibilities and credit with my AI coworkers? Will I have meaningful relationships with others when interactions are mediated through AI? Will my agency be diminished if my life is increasingly taken over by technology? As a business leader, will my investments in advanced technologies continue to be complemented by human ingenuity for continual organizational renewal and sustainability?

RE does not aim to answer the question – will AI be a boon to humanity, or will it diminish human agency in general? But RE techniques can guide each stakeholder to ask, for a specific application setting – which of my constituent selves will a proposed technology intervention augment or compromise? RE can guide the systematic search for alternative configurations of relationships that can best advance overall interests while mitigating against potential vulnerabilities.

This proposal has suggested the Actor abstraction as a core ontological construct for modeling humans and machines alike, through a generalized concept of Actor. The skeletal ontology outlined in this paper is only a first step in a vision, towards a modeling-based RE framework and methodology. The ontology serves as a basis for developing a language notation and semantics, reasoning techniques and algorithms and supporting tools. The framework will also need to be supplemented with catalogs of reusable knowledge, such as classes of commonly seen Roles and technology Agents, as well as patterns and anti-patterns of sociotechnical configurations of Actors.

There are many technical issues to be considered, including, for example, whether association links among Actors such as PLAYS, DEVELOPS-THRU, and LEARNS-THRU should include intentional semantics to support reasoning, and whether structuring relationships well-developed in conceptual modeling for objects – generalization-specialization (IS-A), classification-instantiation (instance-of), and aggregation/ composition (part-whole) – need to be adapted for intentional autonomous Actors. Beyond expressiveness and analytical power, there are the issues of cognitive complexity and learnability for eventual practical adoption. One might consider, for example, incremental or selective adoption of the pivots for different application settings.

Great strides have been made in recent years in advancing RE concepts and techniques. This proposal builds on foundations in goal-oriented and agent-oriented RE. There is much opportunity to consider how to take advantage of existing research results to give substance to this current proposal, including the considerable bodies of work in applying, adapting, and extending the i* framework, e.g., [24, 31, 14, 25, 3, 67].

When a technology is first introduced, it tends to be used to automate familiar tasks, for greater efficiency, lower costs, and higher productivity [61]. As we gain better understanding of the capabilities and limitations of the technology, we begin to recognize the different ways in which it can reshape our lives and relationships, eventually enabling us to reconceptualize who we are, and what we can become [13, 27, 34, 11].

Acknowledgements I would like to thank all those who provided invaluable suggestions and feedback during the preparation of this paper. I particularly want to thank Gerhard Lakemeyer, Alistair Sutcliffe, Alexei Lapouchnian, and anonymous reviewers for their detailed reviews and constructive suggestions for improving the manuscript. Any errors, omissions, or shortcomings in this paper are entirely my own responsibility.

References

[1] Ahmad, K., Abdelrazek, M., Arora, C., Bano, M., & Grundy, J. (2023). Requirements engineering for artificial intelligence systems: A systematic mapping study. Information and Software Technology, 158, 107176.

[2] Alós-Ferrer, C., & Strack, F. (2014). From dual processes to multiple selves: Implications for economic behavior. Journal of Economic Psychology, 41, 1-11.

[3] Amyot, D., Akhigbe, O., Baslyman, M., Ghanavati, S., Ghasemi, M., Hassine, J., ... & Yu, E. (2022). Combining goal modelling with business process modelling: Two decades of experience with the user requirements notation standard. Enterprise Modelling and Information Systems Architectures (EMISAJ), 17, 2-1.

[4] Amyot, D., Ghanavati, S., Horkoff, J., Mussbacher, G., Peyton, L., & Yu, E. (2010). Evaluating goal models within the goal-oriented requirement language. International Journal of Intelligent Systems, 25(8), 841-877.

[5] Anda, A. A., & Amyot, D. (2021). Self-Adaptation Driven by SysML and Goal Models--A Literature Review. e-Informatica Software Engineering Journal, 16(1), 220101.

[6] Bandura, A. (2001). Social cognitive theory: An agentic perspective. Annual review of psychology, 52(1), 1-26.

[7] Becker, C., Chitchyan, R., Betz, S., & McCord, C. (2018, May). Trade-off decisions across time in technical debt management: a systematic literature review. In Proceedings of the 2018 International Conference on Technical Debt (pp. 85-94).

[8] Belk, R. W. (2013). Extended self in a digital world. Journal of consumer research, 40(3), 477-500.

[9] Benis, A., Tamburis, O., Chronaki, C., & Moen, A. (2021). One digital health: a unified framework for future health ecosystems. Journal of Medical Internet Research, 23(2), e22189.

[10] Bennaceur, A., Tun, T. T., Yu, Y., & Nuseibeh, B. (2019). Requirements engineering. Handbook of software engineering, 51-92.
[11] Bingley, W. J., Haslam, S. A., Steffens, N. K., Gillespie, N., Worthy, P., Curtis, C., ... & Wiles, J. (2023). Enlarging the model of the human at the heart of human-centered AI: A social self-determination model of AI system impact. New Ideas in Psychology, 70, 101025.
[12] Callaghan, S., Lösch, M., Pione, A., & Teichner, W. (2021). Feeling good: The future of the $1.5 trillion wellness market. McKinsey & Company, 8.
[13] Carter, M., & Grover, V. (2015). Me, my self, and I (T). MIS quarterly, 39(4), 931-958.
[14] Chung, L., Nixon, B. A., Yu, E., & Mylopoulos, J. (2000). Non-functional requirements in software engineering. Springer Science & Business Media.
[15] Coad, P., Yourdon, E., & Coad, P. (1991). Object-oriented analysis (Vol. 2). Englewood Cliffs, NJ: Yourdon press.
[16] Colantoni, A., Berardinelli, L., & Wimmer, M. (2020). DevopsML: Towards modeling devops processes and platforms. In Proceedings of the 23rd ACM/IEEE International Conference on Model Driven Engineering Languages and Systems: Companion Proceedings (pp. 1-10).
[17] DeMarco, T. (1979). Structure analysis and system specification. In Pioneers and Their Contributions to Software Engineering: SD&M Conference on Software Pioneers, Bonn, June 28/29, 2001, Original Historic Contributions (pp. 255-288). Berlin, Heidelberg: Springer Berlin Heidelberg.
[18] Dennett, D. C. (1989). The intentional stance. MIT press.
[19] Emirbayer, M., & Mische, A. (1998). What is agency?. American journal of sociology, 103(4), 962-1023.
[20] Evans, J. S. B., & Stanovich, K. E. (2013). Dual-process theories of higher cognition: Advancing the debate. Perspectives on psychological science, 8(3), 223-241.
[21] Fabijan, A., Dmitriev, P., Olsson, H. H., & Bosch, J. (2017, May). The evolution of continuous experimentation in software product development: from data to a data-driven organization at scale. In 2017 IEEE/ACM 39th International Conference on Software Engineering (ICSE) (pp. 770-780). IEEE.
[22] Feather, M. S. (1987). Language support for the specification and development of compo-site systems. ACM Transactions on Programming Languages and Systems (TOPLAS), 9(2), 198-234.
[23] Finch, J. K., & Stryker, R. (2020). Competing identity standards and managing identity verification. Identity and Symbolic Interaction: Deepening Foundations, Building Bridges, 119-148.
[24] Franch, X., Susi, A., Annosi, M. C., Ayala, C., Glott, R., Gross, D., Kenett, R., & Siena, A. (2013). Managing risk in open source software adoption. In International Conference on Software Engineering and Applications (Vol. 2, pp. 258-264). SciTePress.
[25] Gonçalves, E., Castro, J., Araújo, J., & Heineck, T. (2018). A systematic literature review of iStar extensions. Journal of Systems and Software, 137, 1-33.
[26] Hermes, S., Riasanow, T., Clemons, E. K., Böhm, M., & Krcmar, H. (2020). The digital transformation of the healthcare industry: exploring the rise of emerging platform ecosystems and their influence on the role of patients. Business Research, 13, 1033-1069.
[27] Hernández-Orallo, J., & Vold, K. (2019). AI extenders: the ethical and societal im-plications of humans cognitively extended by AI. In Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society (pp. 507-513).
[28] Horkoff, J., & Yu, E. (2016). Interactive goal model analysis for early requirements engineering. Requirements Engineering, 21, 29-61.
[29] Horkoff, J., Barone, D., Jiang, L., Yu, E., Amyot, D., Borgida, A., & Mylopoulos, J. (2014). Strategic business modeling: representation and reasoning. Software & Systems Modeling, 13, 1015-1041.

[30] IEEE/ISO/IEC 29148-2018 International Standard - Systems and software engineering -- Life cycle processes -- Requirements engineering
[31] Ingolfo, S., Jureta, I., Siena, A., Perini, A., & Susi, A. (2014). Nomos 3: Legal compliance of roles and requirements. In Conceptual Modeling: 33rd International Conference, ER 2014, Atlanta, GA, USA, October 27-29, 2014. Proceedings 33 (pp. 275-288). Springer International Publishing.
[32] Iqbal, T., Anwar, H., Filzah, S., Gharib, M., Mooses, K., & Taveter, K. (2023, May). Emotions in requirements engineering: A systematic mapping study. In 2023 IEEE/ACM 16th International Conference on Cooperative and Human Aspects of Software Engineering (CHASE) (pp. 111-120). IEEE.
[33] James, W. (1890). The consciousness of self.
[34] Jarrahi, M. H., Kenyon, S., Brown, A., Donahue, C., & Wicher, C. (2023). Artificial intelligence: A strategy to harness its power through organizational learning. Journal of Business Strategy, 44(3), 126-135.
[35] Jennings, N. R. (2000). On agent-based software engineering. Artificial intelligence, 117(2), 277-296.
[36] Kahneman, D. (2011). Thinking, fast and slow. Macmillan.
[37] Majchrzak, A., Markus, M. L., & Wareham, J. (2016). Designing for digital transformation. MIS quarterly, 40(2), 267-278.
[38] Mirbabaie, M., Stieglitz, S., Brünker, F., Hofeditz, L., Ross, B., & Frick, N. R. (2021). Understanding collaboration with virtual assistants–the role of social identity and the extended self. Business & Information Systems Engineering, 63, 21-37.
[39] Morandini, M., Penserini, L., Perini, A., & Marchetto, A. (2017). Engineering requirements for adaptive systems. Requirements Engineering, 22(1), 77-103.
[40] Mylopoulos, J. (1998). Information Modeling in the Time of the Revolution. Information systems, 23(3-4), 127-155.
[41] Mylopoulos, J., Chung, L., & Yu, E. (1999). From object-oriented to goal-oriented requirements analysis. Communications of the ACM, 42(1), 31-37.
[42] Méndez Fernández, D., Christiansson, M. T., & Wieringa, R. (2016). Naming the pain in requirements engineering: Contemporary problems, causes, and effects in practice. Journal of Empirical Software Engineering, 1-41.
[43] Nestor Maslej, Fattorini, L., Perrault, R., et al. (2024) “The AI Index 2024 Annual Report,” AI Index Steering Committee, Institute for Human-Centered AI, Stanford University, Stanford, CA, April 2024.
[44] Newell, A. (1982). The knowledge level. Artificial intelligence, 18(1), 87-127.
[45] Object Management Group Inc. (2008). “Software & Systems Process Engineering Meta-Model Specification. Version 2.0.,” OMG Std.
[46] Object Management Group Inc. (2011). Business Process Model and Notation (BPMN) Version 2.0. OMG Specification.
[47] Orlikowski, W. J., & Yates, J. (2002). It's about time: Temporal structuring in organizations. Organization science, 13(6), 684-700.
[48] Owens, T. J., Robinson, D. T., & Smith-Lovin, L. (2010). Three faces of identity. Annual Review of Sociology, 36, 477-499.
[49] Palomares, C., Franch, X., Quer, C., Chatzipetrou, P., López, L., & Gorschek, T. (2021). The state-of-practice in requirements elicitation: an extended interview study at 12 companies. Requirements Engineering, 26, 273-299.
[50] Perera, H., Mussbacher, G., Hussain, W., Shams, R. A., Nurwidyantoro, A., & Whittle, J. (2020). Continual human value analysis in software development: A goal model based approach. In 2020 IEEE 28th International Requirements Engineering Conference (RE) (pp. 192-203). IEEE.
[51] Porter, M. E., & Heppelmann, J. E. (2014). How smart, connected products are transforming competition. Harvard business review, 92(11), 64-88.

[52] Ross, D. T., & Schoman, K. E. (1977). Structured analysis for requirements definition. IEEE transactions on Software Engineering, (1), 6-15.
[53] Russell, S., & Norvig, P. (1995). A modern, agent-oriented approach to introductory artificial intelligence. ACM SIGART Bulletin, 6(2), 24-26.
[54] Sedikides, C., & Brewer, M. B. (2015). Individual self, relational self, collective self. Psychology Press.
[55] Sutcliffe, A., Sawyer, P., & Bencomo, N. (2022). The Implications of 'Soft' Requirements. In 2022 IEEE 30th International Requirements Engineering Conference (RE) (pp. 178-188). IEEE.
[56] Svahnberg, M., Van Gurp, J., & Bosch, J. (2005). A taxonomy of variability realization techniques. Software: Practice and experience, 35(8), 705-754.
[57] Teece, D. J. (2010). Business models, business strategy and innovation. Long range planning, 43(2-3), 172-194.
[58] Thew, S., & Sutcliffe, A. (2018). Value-based requirements engineering: method and experience. Requirements engineering, 23, 443-464.
[59] Turner, J. C., & Reynolds, K. J. (2011). Self-categorization theory. Handbook of theories in social psychology, 2(1), 399-417.
[60] Van Lamsweerde, A. (2001). Goal-oriented requirements engineering: A guided tour. In Proceedings Fifth IEEE Int. Symp. on Requirements Engineering (pp. 249-262).
[61] Venkatraman, N. (1994). IT-Enabled Business Transformation: From Automation to Business Scope Redefinition. Sloan Management Review 35(2), 73-87.
[62] Wooldridge, M. (2009). An introduction to multiagent systems. John Wiley & sons.
[63] Yasin, A., Fatima, R., Liu, L., Wang, J., Ali, R., & Wei, Z. (2021). Understanding and deciphering of social engineering attack scenarios. Security and Privacy, 4(4), e161.
[64] Yu, E. (2001). Agent orientation as a modelling paradigm. Wirtschaftsinformatik, 43, 123-132.
[65] Yu, E. (2002). Agent-Oriented Modelling: Software versus the World. In: Wooldridge, M.J., Weiß, G., Ciancarini, P. (eds) Agent-Oriented Software Engineering II. AOSE 2001. LNCS, vol 2222. Springer, Berlin, Heidelberg.
[66] Yu, E. S. (1997). Towards modelling and reasoning support for early-phase requirements engineering. In Proceedings of ISRE'97: 3rd IEEE International Symposium on Requirements Engineering (pp. 226-235). IEEE.
[67] Yu, E., Giorgini, P., Maiden, N., & Mylopoulos, J. (2011) Social Modeling for Requirements Engineering. MIT Press.
[68] Engelbart, D. C. (1962). Augmenting human intellect: a conceptual framework SRI Summary Report AFOSR-3223.
[69] Raisch, S., & Krakowski, S. (2021). Artificial intelligence and management: The automation–augmentation paradox. Academy of Management Review, 46(1), 192-210.