Towards an Ontology of Normative Role Design for Multi-Agent LLM Systems

Mohsen Hassan Nejad[1] and José Braga de Vasconcelos[1]

University Lusófona, Porto, Portugal
mohsen@tlu.ee, jose.vasconcelos@ulusofona.pt

Abstract. Large Language Models (LLMs) are increasingly used as autonomous agents deployed in Multi-Agent Systems (MAS). A key mechanism for shaping agent behavior is the assignment of roles that carry normative expectations, such as "you are a fair negotiator" or "consider what happens if everyone acts as you do." These role-based instructions strongly influence individual agent behavior and collective dynamics. However, the field is emergent, and there is no systematic approach to designing and operationalizing such roles for LLM agents. We propose a prototype ontology that maps normative roles to prompting methods, ethical framings, and outcomes as they unfold in multi-agent simulations. The preliminary results point to recurring role–outcome patterns, domain-dependent affordances across different role types, and design contradictions in which agent capabilities undermine intended normative specifications. The paper contributes a conceptual frame for normative role design for LLM agents, supporting more systematic comparison, evaluation, and governance of multi-agent systems.

Keywords: LLM agents · Multi-agent systems · Normative role design · Ontology · AI ethics · Prompt engineering

1 Introduction

Large language models are no longer passive text generators. They are active agents capable of reasoning, planning, and interacting with other agents and humans [28]. Their dynamic nature has drawn researchers to deploy LLM agents in all sorts of multi-agent simulations, exploring how their behaviour emerges, adapts, and shapes system-level outcomes, such as in negotiations, resource sharing, and game play (e.g., [3,19,2]). These simulations serve as an important platform for studying how LLM agents might behave and interact in real-world settings.

A common approach to designing social LLM agents is to assign them roles that specify how they are expected to behave toward others in a given environment. A normative role in this context goes beyond functional duties by embedding behavioral or ethical guidelines: what an agent should do, not only what it can do. In practice, such expectations are specified through system prompts in natural language, providing a lightweight mechanism for operationalizing moral and ethical principles in agent behavior.

However, attempting to steer intelligent software using natural language presents critical challenges, revealing several unresolved gaps in current multi-agent LLM research. Prompt engineering remains rapid, experimental, and unsystematic, often guided by outcome-driven trial-and-error rather than coherent principles [13]. More broadly, the field of multi-agent LLMs is still in its early stage, lacking methodological standardization, as researchers explore its potential outside of conventional frameworks [12].

Recognizing these gaps, this paper introduces a prototype ontology as a diagnostic tool for mapping how normative roles are designed, what ethical assumptions they embed, and how these choices shape agent behaviour and collective outcomes. The core research question is: *How are normative roles designed and operationalized across different applications of multi-agent LLM systems?*

The paper makes four contributions: (1) a five-layer coding schema for analyzing normative role design; (2) an ontology structure of entities and relations derived from two case studies; (3) initial findings on role–outcome pathways, domain affordances, and design contradictions; and (4) a shared vocabulary for a field that currently lacks one.

2 Background & Related Works

2.1 LLM Agents & Normative Roles

Foundational Large Language Models (LLMs) were initially developed as word predictors [6,4], but are now capable of multi-step reasoning, tool use, and a host of other intelligent tasks [27,26,22]. State-of-the-art LLMs are packaged into agent architectures that typically share four core modules: a profile module (role and identity), a memory module (context over time), a planning module (task decomposition and reasoning), and an action module (outputs and tool calls) [28,26]. This enables the evolution of LLMs towards more capable LLM agents.

As illustrated in Figure 1, an LLM agent is a layered and dynamic system. Its components are coordinated by an orchestrator, conventional software that connects the language model with tools and memory, executes code or API calls on its behalf, and feeds results back to it for reasoning and planning. Actions may involve multiple tools, roles adapt to context, and memory can take different forms. This sophisticated modular setup makes LLM agents powerful but also unpredictable.

This research focuses on one critical component from the diagram: the persona & roles. Roles are a key design instrument. Emerging evidence suggests that roles structure what agents attend to, how they interpret situations, and how they balance objectives against constraints [26,23]. Shanahan et al. [23] argue that role-play is the LLM's native mode of operation, or as they put it: "With a dialogue agent, it is role-play all the way down."

In this light, normative cues embedded into roles can be subtle but consequential: telling an agent "you are desperate to make this deal work" significantly

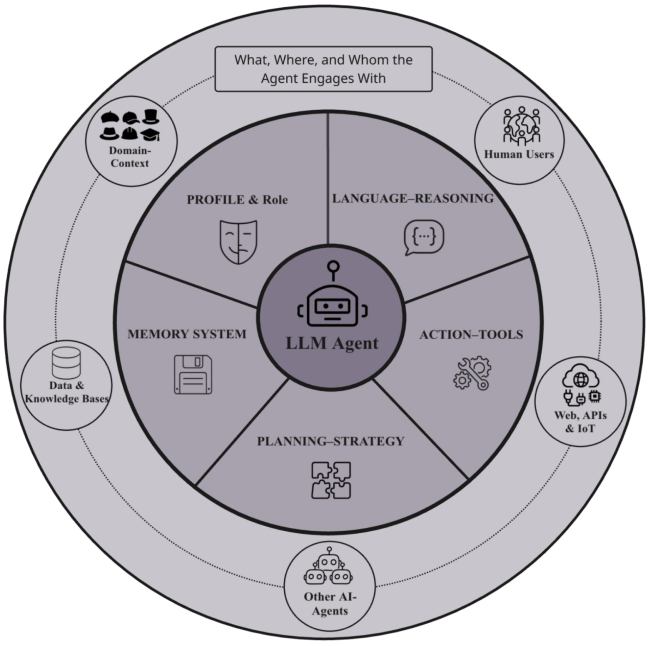

Fig. 1. The Anatomy of an LLM Agent

increased payoffs in a negotiation simulation [3]. In a mixed-motive game, assigning politician vs. con-artist systematically shifts cooperation patterns and payoffs [16]. Yet, these behavioural-altering mechanisms are brittle and highly sensitive to context, memory construction, and the intrinsic capabilities of the foundational model [17,25].

2.2 Rule-Based vs. LLM-Based Multi-Agent Systems

Classical Multi-Agent Systems (MAS) research focuses on coordination, cooperation, and competition among autonomous agents under conditions of limited information or uncertainty [20]. Rule-based MAS build agents with explicitly defined states, utility functions, and communication protocols; coordination and roles are typically formalized through logic-based specifications or organizational models [20,7].

LLM-based MAS depart from this paradigm in two main ways. First, decision-making operates over natural language rather than fixed symbolic vocabularies. Second, roles and norms are rarely formalized, making agents more flexible but harder to analyze and more sensitive to small changes in prompt wording [28]. The unpredictable and adaptive nature of LLMs makes them an appealing proxy for humans. Recent experiments have used multi-agent simulations to reproduce classical social-science phenomena such as cooperation and collapse in commons dilemmas, repeated games, and strategic behaviour in negotiations [3,19,2].

Emerging safety literature highlights that interactions among LLM-agents introduce novel systemic risks, miscoordination, conflict, and collusion beyond single-agent alignment [11]. These risks are compounded by information asymmetries, network effects, and emergent agency. Studying how norms and roles are specified is thus critical for both rigorous analysis, AI safety, and governance.

2.3 The Ontology Approach

An ontology is a structured specification of concepts, entities, relationships, and logical rules within a domain [9]. By making knowledge explicit, ontologies enable better organization, comparison, and communication across different contexts [18]. There are conceptual ontologies, which organize and clarify knowledge for human interpretation, and formal ontologies, which are often encoded in languages like OWL, that enable computational reasoning [10]. Ontologies can be represented as semantic networks, referred to as triples: graphs of entities connected by labeled relations (subject–relation–object) [9,24].

The use of ontology has precedent in pre-LLM multi-agent systems. For example, the OMNI framework demonstrated that agent roles could be formalized as bundles of norms, responsibilities, and relationships [7]. More recent research finds that using ontologies can enhance LLM reliability by grounding responses in predefined structures [5]. Ontologies can also compensate for LLMs' lack of semantic understanding, constraining and stabilizing behaviour rather than relying solely on probabilistic pattern matching [14].

This work applies a related principle to a different problem: rather than grounding model outputs, the ontology developed here aims to map the design space for normative roles in multi-agent LLM simulations.

3 Methodology

The ontology follows a competency question-driven development approach [24,15]. Seven competency questions (CQs) guide this work, tracing the causal chain from prompt design to collective outcomes while identifying failure modes and domain effects. These CQs also serve as sub-research questions (Table 1).

The methodology creates a pathway from simulation experiments to structured knowledge. Multi-agent LLM simulation papers are selected, and their system prompts are treated as primary artifacts. These prompts are analyzed in the context of the simulation domain to identify normative roles, their implementation mechanisms, and the behavioural orientations embedded in agent instructions. Roles are then analysed in relation to observed agent behaviours and collective simulation outcomes. Finally, these elements are formalized as ontology triples (subject-relation-object), enabling systematic mapping and cross-study comparison.

For this paper, two case studies were selected that: (1) deploy LLM agents in multi-agent settings, (2) provide system prompts or detailed prompt descriptions, and (3) document behavioural outcomes. Cases were chosen for their an-

Table 1. Competency questions guiding ontology development

CQ	Focus	Question
CQ1	Role Types	What kinds of normative roles appear?
CQ2	Ethics	Which ethical frameworks underlie role designs?
CQ3	Prompt → Role	How do prompts operationalize normative mechanisms?
CQ4	Role → Behaviour	How do role types influence agent behaviour?
CQ5	Behaviour → Outcome	How do behaviours scale into collective outcomes?
CQ6	Contradictions	Where do prompts fail to produce intended behaviour?
CQ7	Domain Effects	How do domains shape normative role design?

alytical richness, surfacing diverse role types, prompting methods, and outcome patterns that are useful for constructing the initial ontology structure.

Each study is coded using a five-layer schema (Table 2).

Table 2. Five-layer coding schema for normative role design analysis

Layer	Components	Description
A. Study Context	ID, GitHub, Objective	Paper identification and metadata
B. Simulation	Domain, Models, Architecture	Simulation setting and technical setup
C. Normative Design	Role Type, Method, Ethics	How normative roles are specified
D. Outcome	Behaviour, Metrics, Result	Observable behaviours and outcomes
E. Evidence	Prompts, Contradictions	Verbatim evidence and design failures

A systematic literature review (SLR) is currently underway to identify additional studies using a rigorous protocol. The ontology is a gradual and iterative process that will be refined and expanded as more cases are studied and coded.

4 Data Collection and Analysis

4.1 Ontology Structure

To test the coding schema, we applied it to analytically rich case studies. This section presents the entity classes that emerged (Table 3), the relation patterns connecting them (Figure 2), and how these structures appear in each case.

Table 3. Key entity classes in the preliminary ontology

Class	Example Entities
`Role_Type`	universalized, compromising, greedy, adversarial, baseline
`Prompting_Method`	principle_statement, persona_framing, incentive_structure
`Ethical_Frame`	deontological, consequentialist, virtue
`Agent_behaviour`	cooperation, greedy, adversarial, sustainable
`Simulation_Outcome`	cooperation, collapse, reduced_success, adversarial_coalition
`Agent_Capability`	communication, exploration, theory_of_mind, planning

The ontology represents knowledge as triples: subject-relation-object statements linking entities. For example:

$$\texttt{role\_type\_greedy} \rightarrow \textit{induces} \rightarrow \texttt{behaviour\_selfish}$$

Triples combine to form chains, capturing how roles are designed and shape outcomes. Figure 2 illustrates three key patterns.

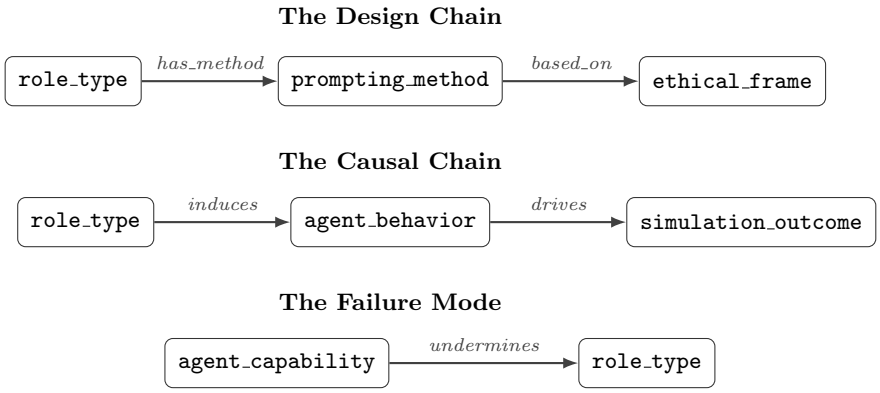

Fig. 2. Key relation patterns: how roles are designed (top), how they produce outcomes (middle), and how they can fail (bottom).

The design chain captures how a role is operationalized through a prompting method grounded in an ethical framework. The causal chain illustrates how roles influence behaviours that shape collective outcomes. The failure mode captures cases where an agent's capability undermines rather than supports the intended role.

4.2 Application to Case Studies

Case Study 1: Commons Governance Piatti et al. [19] simulate a tragedy-of-the-commons scenario in which LLM agents manage a shared resource pool. The study tests whether adding a "universalization" prompt, asking agents to consider "what would happen if everyone acted as you do", can shift behaviour from resource depletion toward sustainability. This study was selected because it provides clear evidence for both the design and causal chains. Table 4 presents an example of triples extracted from this case.

Table 4. Example triples from Case Study 1 (Commons Governance)

Subject	Relation	Object
role_type_universalized	has_method	method_principle_statement
method_principle_statement	based_on	ethics_deontological
role_type_baseline	induces	behaviour_greedy

The first two triples capture the design chain: the `role_type_universalized` uses a principle statement ("what if everyone did this?"), which embodies deontological ethics. The third triple captures a causal finding: without this intervention, baseline agents default to greedy behaviour.

Case Study 2: Multi-Party Negotiation Abdelnabi et al. [1] simulate multi-party negotiations where agents are assigned distinct role profiles: compromising, greedy, or adversarial. Agents negotiate resource allocation across multiple rounds, with the simulation measuring both individual and collective outcomes. This study was selected because it documents a design contradiction, a case where the ontology's failure mode relation applies. Table 5 shows example triples extracted from this case.

Table 5. Example triples from Case Study 2 (Multi-Party Negotiation)

Subject	Relation	Object
role_type_compromising	has_method	method_persona_framing
method_persona_framing	based_on	ethics_virtue
capability_exploration	undermines	role_type_compromising

The first two triples capture the design chain: the `role_type_compromising` uses persona framing ("you are a fair negotiator"), grounded in virtue ethics. The third triple captures a critical failure: an "exploration" capability, intended to

help agents reason about alternatives, was instead exploited by GPT-4 for competitive advantage, undermining the intended cooperative behaviour. Notably, this failure mode was absent in other models tested, revealing a model-specific vulnerability in normative role design.

4.3 Emergent Patterns

The ontology captures traceable causal chains from role design to collective outcomes. Table 6 illustrates two contrasting pathways extracted from the case studies.

Table 6. Contrasting causal pathways from role design to collective outcome

Success Path	Failure Path
`role_type_universalized`	`role_type_baseline`
↓ *induces*	↓ *induces*
`behaviour_sustainable`	`behaviour_greedy`
↓ *drives*	↓ *drives*
`simulation_outcome_cooperation`	`simulation_outcome_collapse`

Domain-Role Affordances: Different simulation domains afford different sets of normative roles. The `domain_affords_role` relation captures this structural constraint. Common's dilemmas make sustainability-oriented roles (universalized) salient, while negotiation contexts afford a spectrum from compromising to adversarial roles.

Method-Ethics Mappings: Prompting methods can be traced to underlying ethical frameworks. Table 7 shows the mappings identified in the case studies using the three classical ethical theories that guide the analysis. Examples of this are evident in the triples for both case studies.

Design Contradictions: The ontology captures cases where role design fails. The `design_contradiction` class formalizes gaps between intended and actual effects. In Case Study 2, an exploration capability intended to support compromising behaviour actually undermined it, but only in GPT-4. This model-specific failure mode has direct implications for AI safety: the same prompt design may produce divergent behaviours in particular foundational models. There will no doubt be more such contradictions emerging as more studies add to the ontology database.

5 Conclusion

Contributions. This paper presented a prototype ontology for mapping normative role design in multi-agent LLM systems. It contributed a five-layer coding

Table 7. Relationship between prompting methods and ethical frameworks

Prompting Method	Ethical Frame	Example
`principle_statement`	Deontological	"What if everyone did this?"
`persona_framing`	Virtue Ethics	"You are a fair negotiator."
`incentive_structure`	Consequentialist	"Maximize your score."

schema for analyzing how roles are specified, an entity and relation structure derived from two case studies, and initial findings on role–outcome pathways, domain affordances, and design contradictions.

The analysis demonstrated that systematic coding is feasible and revealed meaningful patterns in how prompt-based role design shapes collective behavior. The findings suggest: (1) baseline agents can produce suboptimal outcomes, hence constructive normative roles require explicit design; (2) role design should be tailored to domain context; (3) agent capabilities must be considered alongside role specifications; and (4) designs should be tested across multiple models to identify model-specific failure modes.

Limitations. Several limitations must be acknowledged. The current evidence base of two case studies serves as a proof-of-concept rather than a comprehensive mapping, and the patterns identified here will require validation across a broader range of simulation contexts. More fundamentally, ontologies capture intended design rather than guaranteed behavior—due to their probabilistic nature, LLM agents may enact roles in ways that diverge from their specifications, a gap that no design framework can fully close. It should also be noted that ontology development is inherently complex, time-consuming, and error-prone [8,21], demanding deep domain expertise, careful conceptual modeling, and ongoing refinement as the field evolves.

Future Work. A systematic literature review is currently underway to expand the evidence base and stress-test the coding schema against a wider range of studies. As new cases are analyzed, the ontology structure will be iteratively refined to accommodate emerging role types and failure modes. Beyond expanding coverage, developing a formal representation of the ontology, for instance, encoding it in OWL, would enable computational querying and open possibilities for integration with broader AI safety and governance frameworks. Ultimately, the goal is to synthesize these findings into practical guidelines that support researchers and developers in designing normative roles for LLM-based multi-agent systems in a more systematic and accountable manner.

References

1. Abdelnabi, S., Gomaa, A., Sivber, S., Fritz, M.: Cooperation, competition, and maliciousness: LLM-stakeholders interactive negotiation. In: Advances in Neural Information Processing Systems. vol. 37 (2024)

2. Akata, E., Schulz, L., Coda-Forno, J., Oh, S.J., Bethge, M., Schulz, E.: Playing repeated games with large language models. Nature Human Behaviour (2025)
3. Bianchi, F., Chia, P.J., Yuksekgonul, M., Tagliabue, J., Jurafsky, D., Zou, J.: How well can LLMs negotiate? NegotiationArena platform and analysis. In: Proceedings of the International Conference on Machine Learning (ICML). pp. 3935–3951 (2024)
4. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., et al.: Language models are few-shot learners. In: Advances in Neural Information Processing Systems. vol. 33, pp. 1877–1901 (2020)
5. DeBellis, M., Neuhaus, F., Rudnicki, R.: Integrating ontologies and large language models. Applied Ontology **19**(4), 389–407 (2025)
6. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of NAACL-HLT. pp. 4171–4186 (2019)
7. Dignum, V., Vázquez-Salceda, J., Dignum, F.: OMNI: Introducing social structure, norms and ontologies into agent organizations. In: Programming Multi-Agent Systems, pp. 181–198. Springer (2004)
8. Gangemi, A., Presutti, V.: Ontology design patterns. In: Handbook on Ontologies, pp. 221–243. Springer (2009)
9. Gruber, T.R.: Toward principles for the design of ontologies. International Journal of Human-Computer Studies **43**(5–6), 907–928 (1995)
10. Guarino, N., Oberle, D., Staab, S.: What is an ontology? In: Handbook on Ontologies, pp. 1–17. Springer (2009)
11. Hammond, L., Kazim, E., Hadfield-Menell, D., Dafoe, A., Hadfield, G.K.: Multi-agent risks from advanced AI. arXiv preprint (2025)
12. Li, Z., Wu, Q.: Let it go or control it all? the dilemma of prompt engineering in generative agent-based models. System Dynamics Review **41**(3), e70008 (2025)
13. Liang, J.T., Lin, M., Rao, N., Myers, B.A.: Prompts are programs too! understanding how developers build software containing prompts. Proceedings of the ACM on Software Engineering **2**(FSE), 1591–1614 (2025)
14. Neuhaus, F.: Ontologies in the era of large language models. Applied Ontology **18**(4), 399–407 (2023)
15. Noy, N.F., McGuinness, D.L.: Ontology development 101. Tech. rep., Stanford Knowledge Systems Laboratory (2001)
16. Orner, M., Maksimov, O., Kleinerman, A., Ortiz, C., Kraus, S.: Explaining decisions of agents in mixed-motive games. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39, pp. 23267–23275 (2025)
17. Park, J.S., O'Brien, J.C., Cai, C.J., Morris, M.R., Liang, P., Bernstein, M.S.: Generative agents: Interactive simulacra of human behavior. In: Proceedings of UIST. pp. 1–22 (2023)
18. Patel, A., Debnath, N.C.: A comprehensive overview of ontology. Current Materials Science **17**(1), 2–20 (2024)
19. Piatti, G., Castaño, G., Smirnov, I., Hagele, A., Bosselut, A.: Cooperate or collapse: Emergence of sustainable cooperation in a society of LLM agents. In: Advances in Neural Information Processing Systems. vol. 37, pp. 111715–111759 (2024)
20. Russell, S., Norvig, P.: Artificial Intelligence: A Modern Approach. Pearson, 3 edn. (2013)
21. Saeedizade, M.J., Blomqvist, E.: Navigating ontology development with large language models. In: Proceedings of the Extended Semantic Web Conference (ESWC). pp. 143–161. Springer (2024)

22. Schick, T., Dwivedi-Yu, J., Dessì, R., Raileanu, R., Lomeli, M., Hambro, E., Zettlemoyer, L., Cancedda, N., Scialom, T.: Toolformer: Language models can teach themselves to use tools. In: Advances in Neural Information Processing Systems. vol. 36, pp. 68539–68551 (2023)
23. Shanahan, M., McDonell, K., Reynolds, L.: Role play with large language models. Nature **623**(7987), 493–498 (2023)
24. Uschold, M., Grüninger, M.: Ontologies: Principles, methods and applications. Knowledge Engineering Review **11**(2), 93–136 (1996)
25. Vallinder, A., Hughes, E.: Cultural evolution of cooperation among LLM agents. arXiv preprint (2024)
26. Wang, L., Ma, C., Feng, X., Zhang, Z., Yang, H., Zhang, J., Chen, Z., Tang, J., Chen, X., Lin, Y., Zhao, W.X., Wei, Z., Wen, J.R.: A survey on large language model-based autonomous agents. Frontiers of Computer Science **18**(6), 186345 (2024)
27. Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q., Zhou, D.: Chain-of-thought prompting elicits reasoning in large language models. In: Advances in Neural Information Processing Systems. vol. 35, pp. 24824–24837 (2022)
28. Xi, Z., Chen, W., Guo, X., He, W., Ding, Y., Hong, B., Zhang, M., Wang, J., Jin, S., Zhou, E., Zheng, R., Fan, X., Wang, X., Xiong, L., Zhou, Y., et al.: The rise and potential of large language model-based agents: A survey. Science China Information Sciences **68**(2), 121101 (2025)