

A systematic mapping of goal-oriented requirements engineering: reflections, impact, and the way forward

Downloaded from: https://research.chalmers.se, 2026-11-27 10:39 UTC

Citation for the original published paper (version of record):

Horkoff, J., Aydemir, F., Cardoso, E. et al (2026). A systematic mapping of goal-oriented requirements engineering: reflections, impact, and the way forward. Requirements Engineering, 31(2). http://dx.doi.org/10.1007/s00766-026-00476-7

N.B. When citing this work, cite the original published paper.

research.chalmers.se offers the possibility of retrieving research publications produced at Chalmers University of Technology. It covers all kind of research output: articles, dissertations, conference papers, reports etc. since 2004. research.chalmers.se is administrated and maintained by Chalmers Library

(article starts on next page)

Requirements Engineering (2026) 31:20
https://doi.org/10.1007/s00766-026-00476-7

ORIGINAL ARTICLE

A systematic mapping of goal-oriented requirements engineering: reflections, impact, and the way forward

Jennifer Horkoff[1,2] · Fatma Basak Aydemir[3] · Evellin Cardoso[4] · Tong Li[5] · Alejandro Mate[6] · Elda Paja[7] · Mattia Salnitri[8] · Luca Piras[9] · John Mylopoulos[10] · Paolo Giorgini[11]

Received: 4 June 2026 / Accepted: 23 September 2026
© The Author(s) 2026

Abstract

Goal-oriented Requirements Engineering describes the body of work which focuses on goals as the driving force motivating and capturing requirements, often using graphical goal models. A proliferation of work in this area inspired the study: "Goal-oriented requirements engineering: an extended systematic mapping study" first available in 2017, covering 246 publications in GORE up to 2015. In the current article, we have the pleasure of reflecting on this original mapping study. Using bibliometrics, we examine the study's visibility, although such a quantitative view gives us only an approximation of real impact. The bibliometric view shows us that most citations are relatively recent: 85% of the citations have occurred since 2020. Unsurprisingly, the work has primarily been cited in requirements engineering venues, but also related venues such as the Conceptual Modeling conference (ER) and the Information and Software Technology Journal (IST). The work has been cited by a wide variety of authors. We estimate that roughly 40% of the publications citing the original SLR actually focus on goal models (have some variant of "goal" in the title), but that only roughly 53% of citing papers which are not themselves systematic reviews emphasize some form of empirical evaluation, 10% lower than the presence of empirical studies found in the original mapping study. Turning to trends and future directions, we reflect on the limitations of GORE uptake in industry, along with recommendations for continued use in education. We highlight new developments in using Large Language Models (LLMs) to generate goal models, including limitations, and reflect on what opportunities may be lost via full automation of goal model generation. We reflect on the potential role of GORE in the rising field of intent-driven engineering with AI, and the disconnect between much of this work and requirements-focused goal-oriented foundations. Finally, we consider the rising popularity of agentic AI, describing conceptual challenges and opportunities for GORE in terms of monitoring, capturing, and driving agentic system design. Overall, our reflection emphasizes that in the rise of LLMs and Agentic AI, the expressive power of GORE has great potential, and we invite the research community to explore and harness goal modeling for this purpose.

Keywords Goal modeling · GORE · Requirements engineering · LLMs · Agentic AI · Conceptual modeling · Systematic literature mappings

1 The original goal: mapping the research landscape of GORE

In order to create successful software systems, it is often helpful, or even necessary, to explicitly capture the desired functions and qualities of a software system. To do this, we write or model requirements. In industry, we often see that requirements focus on functions and metrics, but more rarely capture the "why", the intentional dimension that motivates key functions and quality constraints. To address this gap and raise the level of requirements knowledge to the socio-technical domain, much focus in requirements engineering research has been placed on the notion of goals, making intentions a first-class requirements artifact, and treating goals as a critical part of requirements knowledge. This frame of mind led to the inception of Goal-Oriented Requirements Engineering (GORE).

Extended author information available on the last page of the article

Published online: 12 October 2026

To this end, various prominent goal modeling frameworks such as iStar (originally *i**) [1, 2], KAOS [3], GRL [4], and Tropos [5] were developed by pioneers in this area, starting in the 90 s and inspired by previous foundational work in languages such as RML (Requirements Modeling Language) [6]. As a brief overview of goal modeling, most goal modeling frameworks contain goals, the ability to refine goals into AND or OR trees, some form of task or operationalization, and a notion of agents. KAOS leans towards formally and clearly defined goals, while iStar supports the notation of a softgoal or quality goal, a qualitative objective. iStar emphasizes dependencies between agents, while Tropos goes further into the agentic paradigm. GRL pairs goal models with use case maps for a complementary process view. Overall, while different frameworks have slightly different concepts and emphasize different things, the main modeling capabilities are similar.

The overall body of GORE frameworks spawned and inspired a wide breadth of follow-up in GORE: extensions, formal translation and analysis, reasoning, visualizations, tools, case studies, user studies, and more. Although these efforts produced a rich and varied body of work, by 2015, it was our impression that this body of work was growing fragmented, often somewhat abstract, sometimes unevaluated, and was generally moving beyond even what an expert in this area may hope to read, understand, and use. When reviewing and reading papers, those in the GORE community felt that there was a lot of repetition, "re-inventing the wheel" and work that was introduced and not picked up by others in the community. These observations inspired past authors to conduct a systematic mapping study to understand the research landscape of GORE at that point in time [7, 8].

As such, in 2019, a systematic mapping study on goal-oriented requirements engineering (GORE) was published in the Requirements Engineering Journal, first appearing online in 2017 [8]. This work was an extension of a 2016 paper in the IEEE International Requirements Engineering Conference (RE'16) [7], and continued the efforts of many of the same authors in publishing mappings and systematic literature reviews to get a handle on the vast space of GORE research [9, 10]. In this article, we have the privilege of reflecting on this work, including the original motivations and findings (Sec. 1), the impact of the study through bibliometrics (Sec. 2), and how the space of GORE can and should be adapted and applied in an AI-dominated future (Sec. 3).

1.1 Mapping study methodology

To design the study, the authors made use of the latest and increasingly popular advice by Kitchenham and her colleagues [11] regarding Systematic Literature Reviews (SLRs), as well as work with guidelines for mapping studies specifically in Software Engineering by Peterson et al. [12]. A team of ten researchers was formed, with eight taking active part in the paper searching, coding, and synthesis. The authors were driven by eight research questions, four focusing on general bibliometrics (venues, authors, citations), and the other four attempting to get an overview of the field: whether publications built on existing work, the type of empirical evaluation in the covered papers, the general topic or problem the papers covered, and what GORE frameworks were used. The authors' analysis also showed trends over time for most of these factors.

Methodologically, as had been done in their previous studies [9, 10], the authors started with a set of known GORE-related papers and snowballed to a set of 110 papers. However, it seemed that the set of papers was not converging as part of the snowballing process. Instead, the authors switched tactics to a systematic search, defining a search string based on the research questions, and searching prominent meta-databases at the time: Scopus, Web of Science, and Google Scholar. Initial searches found 966 publications, but many publications had a low number of citations, so the authors focused on the 394 papers that had three or more citations at the time of the search (2015). The process excluded 104 papers during initial screenings for being out of scope, leaving 246 papers in the mapping study.

Seven authors went through rounds of qualitative coding, inspired by grounded theory [13], iteratively refining their coding scheme, and measuring inter-coder reliability (ICR) using Krippendorff's alpha. The authors found it challenging to come to a sufficient level of agreement over codes (0.67+). In order to get sufficient values, the authors had to go through a number of iterations, have frequent rounds of discussion, define and merge codes, drop contentious codes (e.g., illustrative example, social modeling), and drop a human coder who had less experience with goal modeling. In retrospect, many of the codes expressing paper types and topics were challenging to apply reliably, e.g., distinctions such as proposal vs. extension, variability vs. reasoning were difficult to make, as these terms had a high level of subjectivity, even with definitions, and papers often had multiple objectives or were difficult to understand. This entire coding process took the authors nearly two months.

At the time, the authors were not aware of any tools to explicitly support systematic mappings or qualitative coding distributed among many coders. Thus, the authors created custom solutions. The authors captured paper coding in a MySQL database with an HTML and PHP front-end. Given the presence of a business intelligence technology expert on the team, the authors used these technologies to detect inconsistencies in our survey data and conduct automated summaries of our data, building a data warehouse.

The multidimensional schema of the data warehouse was designed to ease the comparison of data across dimensions and hierarchies, including Time, Venue, Authors and Institutions, as well as author-added Tags. These elements made the analysis as flexible and dynamic as possible.

1.2 Results

To summarize the key results of the SLR, most papers (91.8%) focused on proposing something new, although such papers could also include an integration (45.5%), extension (41.9%), or formalization (40.2%). Empirical evaluation was dominated by case studies (53%) and scalability tests (27%), but only 63.4% of papers included any sort of evaluation (beyond a motivating example or evaluation argument). Note that even starting in 2015, the authors tried to be discerning about what was and was not a case study, now well-reflected in the work of Wohlin [14], as illustrative examples labeled as case studies were common. Even so, the definition of a case study from 2016 was not as rigorously defined in the mapping study compared to this later work by Wohlin, as the mapping study focused on realism, but not check for multiple sources of evidence.

The mapping study examined popular topics, finding that agents and reasoning were popular topics at the time. The most popular named GORE frameworks were iStar (13.4%) and KAOS (12.6%), but most papers used GORE generally without committing to a framework (21.5%) or used their own new framework (28.9%). Papers appeared in many different venues, but most prominently in the RE conference or journal. The authors noted large citation differences at the time between Google Scholar, Scopus and Web of Science, but the papers with the most citations were generally the same in all databases.

To evaluate author networks and paper citations, the authors created a number of visualizations using different tools. A tool was created to allow interactive visualization of citation networks .[1] The tool used Scopus APIs to find papers related to GORE references, organized the different reference connections between papers, was able to support the analysis of included paper citations and to create a resulting visual *author citation network* (shown in Fig. 16 of[8]). This visualization provided an interconnected representation of citation relationships among authors and allowed us to see clusters of citations, including outliers, i.e., GORE work that did not cite other goal work. Other visualizations captured co-author relationships and showed similar patterns, clusters of co-authorship, with occasional connections to other clusters.

Several points of analysis focused on trends over time, generally showing that the number of papers grew roughly linearly from 1998 to 2011, then appeared to drop, although some of that drop would be due to the paper search cutoff date in 2015.

To some degree, the analysis revealed a well-interconnected and strong GORE community. However, there were also a number of papers weakly connected to the community, appearing isolated, or related to "parallel paths". Authorship and citation tended to cluster around the type of goal framework used. One could, for instance, identify a primary cluster related to the NFR framework[15–17], including extensions of this framework for different domains, e.g., acceptance requirements[18–21] security and privacy requirements[22–25]. Other clusters related to i* [26] or GRL[27]. One cluster captured Tropos[28, 29] and its extensions, e.g., Secure Tropos, SecTro[30, 31], Privacy SecTro[22, 23, 25, 32, 33], GRL[27]. One could see links between these clusters, in particular, KAOS and i*-family authors had a large number of interconnected citations, despite not co-authoring with each other[34].

Generally, although custom-made, the interactive author network and other visualizations and tools were useful for exploring relationships and identifying influential works and communities[34].

At a high level the authors concluded that the GORE research landscape had a focus on new solutions without extensive evaluation and follow-up. At the time the authors hypothesized that this was due to the complexity and relative newness of RE as a research field. RE as a discipline could be seen as special due to the socio-technical divide, unlike other disciplines like AI (at the time) or databases which tend to abstract out human users.

The authors initially had ambitions to do a more high-level analysis with the mapping, looking at relationships between papers in terms of extensions, evaluations, mappings, implementations. However, after a few rounds of manual coding over citations, the authors' inter-coder agreement for these labels were abysmal (0.14). Although interesting, such coding was difficult due to the difficulty in understanding and agreeing upon the relationship between papers.

Overall, the entire process took the authors a year, with the first version of the systematic mapping published in the 2016 edition of the Requirements Engineering conference [7], and an expanded version published in the requirements engineering journal [8], published in 2019, but first appearing online in 2017. Between the two publications, the authors added new research questions and new analysis, but did not expand or update the set of papers.

[1] https://www.cs.toronto.edu/goreslm/CitationNetwork.html. Please give time for the graph to load.

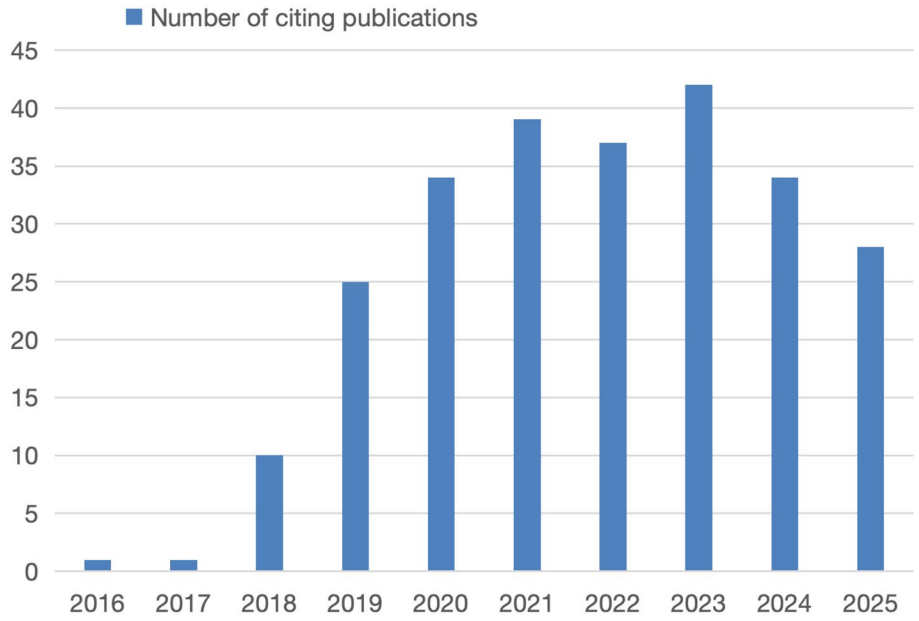


Fig. 1 Annual distribution of publications citing the REJ paper [8]

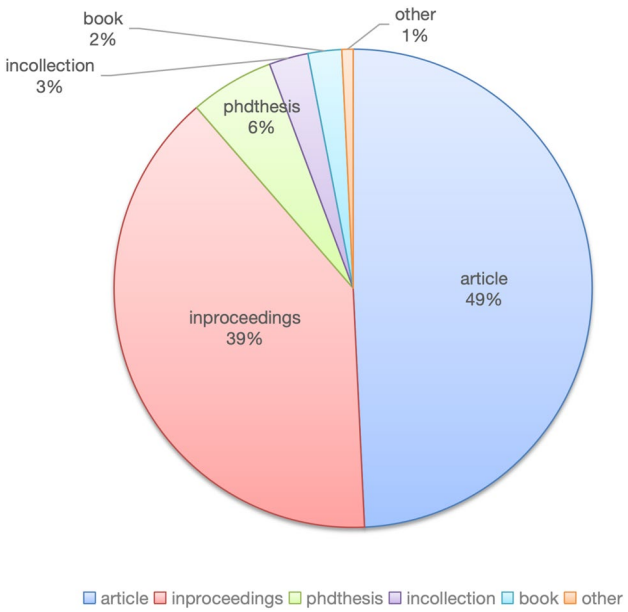


Fig. 2 Distribution of citing publications by type

2 Mapping study impact: a bibliometric view

This section reflects on the impact of the original mapping using bibliometrics and a lightweight analysis of citing papers. However, we note that bibliometric analysis does not offer an in-depth indication of influence, but only of visibility and citations. As of May 2026 according to Google Scholar, there have been 264 papers citing the mapping paper. We examine these papers both bibliometrically and using lightweight snowballing exercises to gauge the presence of empirical studies and to understand if these papers focus on goal models or reference the mapping paper in a more general context, e.g., systematic reviews in requirements engineering.

We can examine the temporal distribution of citations (Fig. 1). Among the citing documents with identifiable publication years, 251 papers were published between 2016 and 2025. Since the citation data for 2026 are incomplete, with only two citations currently recorded, 2026 was excluded from this analysis. Notably, 214 of these citations occurred during 2020–2025, accounting for approximately 85.3% of the dated citations. This indicates that the paper has continued to attract substantial attention in recent years and that its relevance has grown or remained strong over time.

We look at the types of papers citing the mapping study (Fig. 2). Journal articles account for 130 citations, representing 49.2% of all citing documents, while conference papers account for 104 citations, or 39.4%. Together, journal and conference papers make up 234 citations, corresponding to 88.6% of the total citation set. This shows that the paper is mainly cited in peer-reviewed scholarly venues rather than in other informal sources. We also examine at a high level if the citing papers are primary or secondary studies, finding that 34 out of the 264 publications citing the mapping are identified as secondary studies, including systematic literature reviews and mapping studies. The prevalence of secondary studies indicates that the paper has contributed to other secondary studies and consolidation efforts within the field.

Looking at venues where the citing papers are published, we see that frequently appearing venues include the *Requirements Engineering* journal, the *IEEE International Requirements Engineering Conference (and workshops)*, the *International Conference on Conceptual Modeling*, and

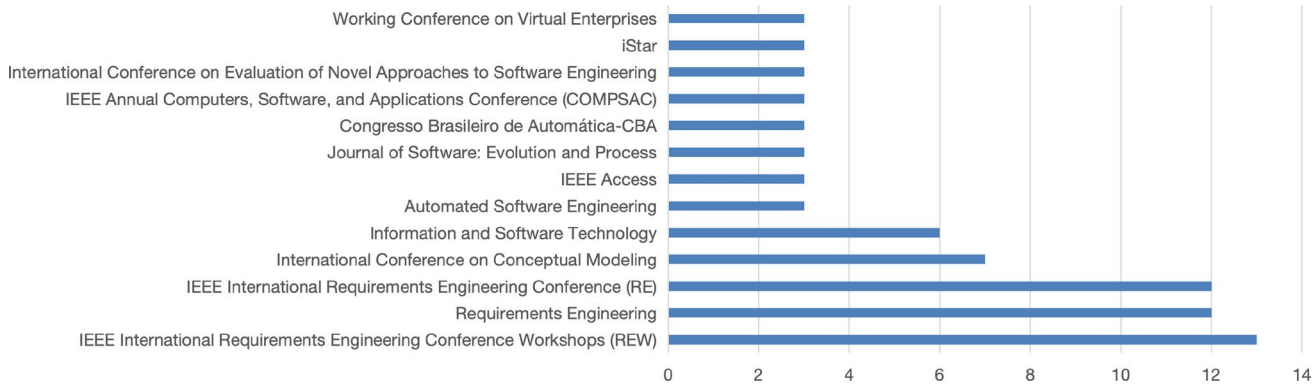


Fig. 3 Top venues

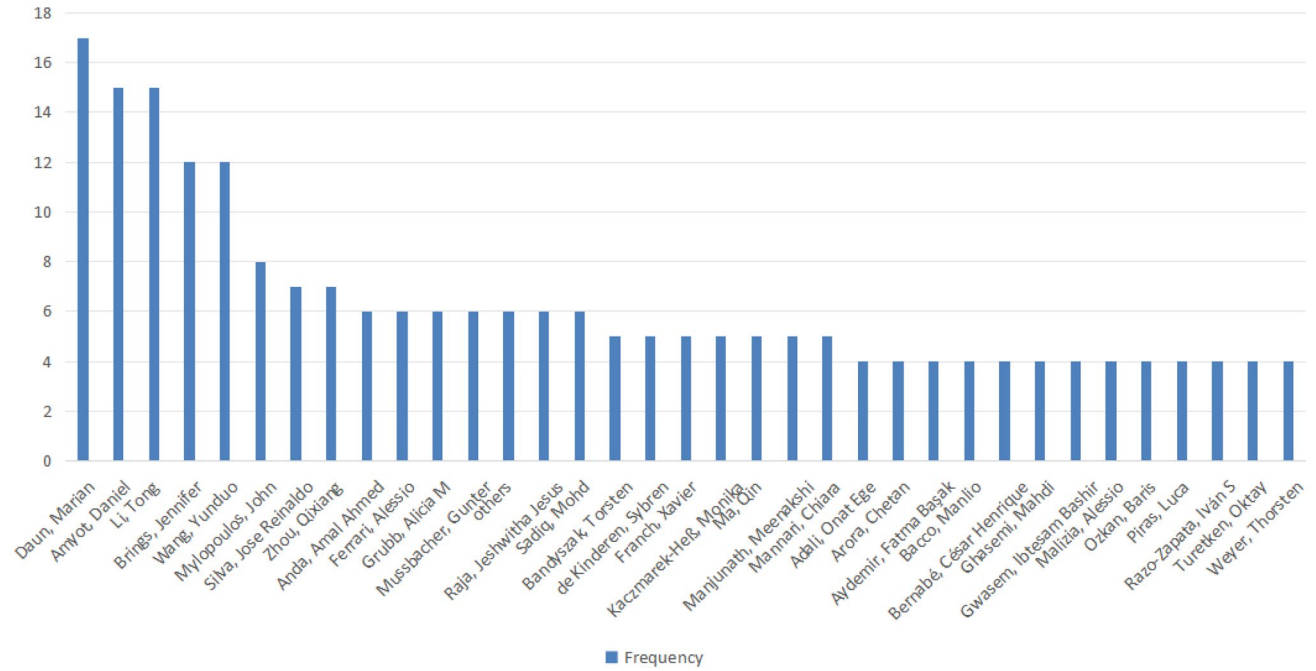


Fig. 4 Top citing authors

the *Information and Software Technology (IST)* journal. These venues demonstrate the paper's strong relevance to the core communities of requirements engineering and software engineering (Fig. 3), but also show wider impact to more generic SE venues like IST.

We examine which authors cited the mapping paper (Fig. 4). We can note that several authors have cited the survey paper multiple times, but also that there is a long tail of authors with citation counts of four to five. We note that the top 35 citing authors in Fig. 4 contain only three of the 10 authors of the original paper, pointing away from a pattern of extensive self-citations. Specifically, the citation records show that the paper has been repeatedly cited by a broad range of researchers, indicating a wide diffusion.

Finally, we conducted a lightweight analysis of the 264 papers to explore two different questions: are the citing papers themselves focusing on goal models? And do subsequent studies operationalize one of the recommendations proposed in the original paper: the need for stronger empirical evaluation within the area?

To answer the first question, we do a simple count over the titles of the citing papers: 82/264 contain the term "goal", 20 contain "iStar", 2 contain "KAOS", 6 contain "GRL", and 1 contains "Tropos". This gives a very rough indication that 100/264 of the papers citing the mapping study are likely specifically about goal modeling. For example, the paper "Goal-oriented email stream classifier with a multi-agent system approach" [35] is likely to be goal-model focused, while the paper "Towards a Solution Proposal to Agile Quality Requirements Challenges in Large-scale Projects" [36] is likely not focusing on goal modeling. Although we can get a rough estimate of the focus on goal modeling in citing

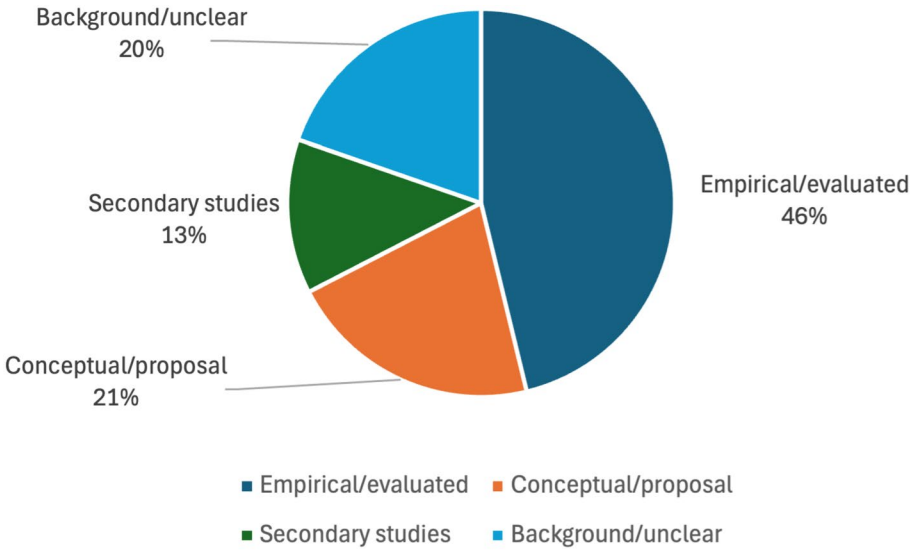


Fig. 5 Distribution of citing papers per contribution category

papers, it is still difficult to classify the influence the original mapping study had on these related pieces of work. Just because a paper may focus on goal modeling does not mean the original mapping paper had a significant influence. As mentioned, past experience in the original mapping study has shown us that classifying references between papers in a consistent way is challenging. Similarly, measuring the degree of impact of a reference on a paper is also challenging, as often authors are not very explicit in describing the impact of previous work on their work, particularly for systematic reviews. Exceptions exist, particularly for other systematic reviews, for example, Mavin et al. [37] used the database created in the survey to further analyze the status of the goal-modeling research field evaluating adoption of goal-based modeling languages by the industry.

To tackle the second question, we adopted an exploratory screening approach based on titles and selectively retrieved abstracts rather than performing a full systematic literature review. Abstracts were retrieved selectively for papers where the contribution type could not be reliably inferred from the title alone. Based on the combined title and abstract analysis, 122 papers out of the 264 citing publications (46%) mentioned some form of empirical evaluation activity (see Fig. 5), including case studies, experiments, industrial validations, feasibility studies, qualitative studies, prototype evaluations, or other empirical investigations. However, as mentioned, 34 (13%) of the publications citing the mapping were identified as secondary studies, so focusing on the remaining 230 studies, we estimate that 122/230 (53%) of the non-systematic review studies citing the original mapping paper conduct empirical evaluation, or at least mention such a study in their title and abstract. It is interesting to note that 53% is lower than the original number in the mapping, with 63.4% having some form of empirical evaluation. This is despite the increased focus on empiricism as highlighted by efforts such as the ACM SIGSOFT Empirical Standards for Software Engineering [38]. This finding could mean that fewer studies using GORE are using empirical evaluation compared to the studies included in the original mapping, or it could just be an indication that any empirical efforts are not emphasized in the title and abstract. Future work should conduct a more rigorous SLR or mapping to validate if the trend found in our lightweight snowballing analysis holds.

3 GORE: current trends and future directions

This section covers areas of current work and outlines possible future directions related to GORE, much of which is inspired by the rise of AI and LLMs.

3.1 GORE in industry

Although not explicitly covered by the original mapping, it is interesting to reflect on GORE use in practice and education. Addressing practical uptake, as mentioned, a year after the original version of the mapping paper appeared in the RE conference, Mavin et al. [37] used the provided database of papers to address the limited adoption of goal-based modeling languages by industry. Specifically, the title of the paper was "Does goal-oriented requirements engineering achieve its goal?", and during the paper presentation, Mavin revealed his t-shirt bearing the word "no". Mavin et al. [37] argued that GORE does not achieve its goal due to a lack of industrial uptake. To some extent the original mapping results support these results by showing many new ideas and only a partial emphasis on empirical evaluation. Our updated snowballing exercise similarly shows that 21% of

papers citing the mapping paper contain only new concepts and a proposal, with another 20% not clearly containing evaluation (Fig. 5). Exceptions to these statistics do exist. The KAOS Objectiver tool supporting goal hierarchies has several industrial customers.[2] Similarly, a recent long-term industrial case study applied KAOS to analyze law enforcement systems in the UK [39]. iStar has been applied in industrial cases since 2015 [40], and other specialized extensions of iStar have been applied for the comparison of AI solutions for software development [41], a gas turbine manufacturer [42], Dashboard design [43] and process simulation in the paper industry [44]. Despite these examples, there are challenges which inhibit industrial adoption.

Challenges: Barriers to Industrial Adoption. Our experiences indicate that goal model usage usually ends when the researcher stops working directly with the company. This is often due to several factors: the complexity of the models, the difficulty in keeping such models up to date, and the lack of integration of goal modeling with companies' tooling and ways of working, including agile processes such as SAFe, and project or requirements management software such as Jira. Even those companies open to modeling and model-driven practices are typically using UML-style or Simulink models (in our experience). The main point is that the gap between goal modeling and these popular tools and approaches is significant, and we see companies are typically not willing to bridge this gap without researcher help.

Future directions: Many of these challenges can be addressed by the proficiency of LLMs: quickly developing new tool support or transformations for GORE, and using auto-generated tooling to help manage goal model complexity and to help keep models up to date. However, these observations on goal modeling in industry are anecdotal, based on our experiences – both these observations and the ability of LLMs to solve such problems should be treated with more depth in further empirical work.

3.2 GORE in education

Despite the lack of widespread adoption of GORE in industry, we see some continued evidence of efforts to use or report on goal modeling in teaching. The iStar Teaching Workshop (iStarT) saw several instances starting in 2015, around the time the first version of the mapping study was published [45]. We can see further work that explores or reports on experiences in teaching goal modeling, or using goal modeling in teaching [46, 47]. GORE can also be used in teaching beyond RE, for example, Franch et al. propose an iStar-based approach to teach data structures [48].

Challenges: pervasiveness and effectiveness of GORE education. Without a more in-depth study, we cannot know how common it is to include goal modeling in university education. This is tied to a more general question about the prevalence of teaching requirements engineering topics in general [49], an interesting topic out of scope for this reflection. However, we can give some possible explanations for the seeming contrast between industrial use of goal models and teaching practices. Our informal snowballing exercise estimates that roughly 100 papers citing the original mapping study focus on goal models, indicating that goal modeling is still an active topic in research. If so, it is not surprising to see it appearing as a teaching topic, as most university education is driven by researchers, and researchers tend to teach what they know.

Future directions: We believe it is still useful to use GORE as part of teaching in software or requirements engineering, mainly as the concepts and relationships in GORE force students to think in unique ways that are not supported by other frameworks or approaches [50]. Namely, goal modeling, in particular, iStar/GRL/Tropos, focuses students to think explicitly of qualities and actors, dependencies between actors, and a space of possible requirements. In our experience, introducing such ideas helps to drive home some key concepts in RE teaching: the social and human-driven nature of domains, making choices in defining requirements, and focusing on system quality. However, further work is needed to understand the scope and utility of teaching goal model concepts, and if our intuitions on the topic based on years of experience are correct. SE education is going through a time of immense change due to AI, but in this space we believe that intentional thinking is a key SE education topic, which can be operationalized through teaching goal models.

3.3 LLM use for GORE

Recent advances in Natural Language Processing (NLP) led to the development of LLMs, enabling text-based generative AI [51]. LLMs are models trained on large textual corpora to learn probability distributions of natural language and perform tasks such as text generation, translation, and approximate reasoning through prompt engineering strategies [52]. The advent of LLMs has changed many aspects of practice and research related to GORE, including software engineering [53], requirements engineering [54], and conceptual modeling [55]. More specifically, LLMs have started to be employed to support GORE activities. They have been primarily used for goal modeling tasks such as model extraction and generation. In this context, goals and goal models are usually extracted by analysing user stories[56], user reviews of software[57], or user interviews [58]. In

[2] https://www.objectiver.com/

more recent work, Li et al. create a custom tool with a BERT backend to support goal modeling, with positive results found in realistic evaluations [59].

LLMs have also been employed to identify traceability links between requirements and goal models. For instance, one study leverages LLMs, via iterative prompt engineering activities applied over User Stories, to generate GRL models, reporting promising results with models that are understandable to users and soft goals extracted that otherwise would not have been identified by novices[56]. A further study focuses on identifying traceability links between security requirements and goal models[60]. The requirements considered are expressed in natural language, while the models are expressed using GRL; based on these inputs, the LLM is able to discover links between the two representations, showing positive results[60].

Challenges: limitations in generating goal models. While the support provided by LLMs is valuable for GORE activities, important limitations remain. The quality of the output strongly depends on the prompting strategy used, while the reuse of the same prompt across different domains often leads to low quality results; depending on the domain, the accuracy of the generated results can vary substantially, especially for domains characterised by limited public knowledge on which the LLM has been trained [56]. An additional challenge concerns the problematic automatic visual management of generated goal models, with layouts hard for humans to parse [59]. Some challenges can be attributed to the complexity and relative scarcity of goal models in the LLM training corpora, compared to more popular and simpler models like use cases or class diagrams. As a result of these challenges, often human expertise remains necessary to refine, correct, and re-arrange the generated models.

Future directions: The overall ability of LLMs to aid in goal modeling can be improved with specific interventions such as fine-tuning, providing context or memory interventions, designing prompts, iteration over outputs, and the use of agents to collectively produce, critique and iterate over models, and clustering inputs semantically as applied in [57, 58]. Additional work is needed to continue to develop methods to auto-generate candidate goal models from initial input, including interviews, reviews, and other forms of text, working to overcome existing limitations. The role and cognitive capabilities of humans must be accounted for. It is interesting to explore whether the limitations of LLM-generated goal models can be overcome via clever engineering, for example, through better pre-training or fine-tuning, multiple overlapping LLMs, and iterative prompting processes, or whether some limitations are fundamental limitations of LLMs. With such techniques, we expect to see significant improvement in this space.

Beyond supporting goal model extraction and generation, LLMs may also help address long-standing cognitive scalability challenges associated with GORE. Goal models often become difficult to navigate and maintain as their size and complexity increase, limiting their practical adoption in industrial contexts. In this respect, LLMs could support the generation of contextualized views of goal models, help stakeholders focus on relevant model fragments, provide explanations of modeling decisions, and bridge technical modeling concepts with domain-specific terminology used by practitioners. Such capabilities may improve the accessibility and usability of goal models for non-expert stakeholders.

Challenges: the human cost of goal model generation. Although promising, auto-generating goal models introduces important trade-offs: while LLM-supported modeling may lower the entry barrier and improve industrial uptake, excessive automation may reduce some of the benefits of learning, negotiation, and shared understanding traditionally associated with collaboratively constructing goal models.

Specifically, we note that using LLMs to generate goal models removes or eliminates the learning, coordination, and communication benefits of going through the modeling process manually. Considering the famous George Box quote: “all models are wrong, but some are useful”, the usefulness of goal models is often not in the final product, but in the harmonizing process of collaboratively creating such models, often discussed in the field of participatory modeling [61]. The benefits of modeling, particularly socio-technical goal modeling, are not insomuch as the final model, but the process of modeling.

Future directions: We should strive to better understand the cognitive benefits of the goal modeling process and strive to retain these benefits as part of LLM-assisted goal model generation. We recommend that new techniques avoid the trap of over-automating the creation of an artifact which is unverified and ultimately less useful than one curated by stakeholders and experts. No doubt LLMs and agents can generate goals, but it is the goals of real users that actually matter.

3.4 Intent-driven engineering with AI

LLMs receive instructions through natural language prompts, which can be entered dynamically, supplied through scripts, or received from other AI agents or tools. The ability of LLMs such as Claude Code[3] to generate working code and even complex code is disrupting the software industry, including processes, ways of working, and the workforce. As more and more code is auto-generated,

[3] https://www.anthropic.com/product/claude-code

the focus of software development is less on the code itself, and more on the prompts that provide instructions for generating code, as well as other artifacts like memory and context [62]. In this way, the prompts become the requirements, whether or not they are explicitly treated or formatted as requirements artifacts. Thus, while code is re-generated and dynamic, prompts and requirements remain static. Informally, the authors can see a rise in the prominence and focus on requirements engineering in this space from researchers or practitioners who previously were not RE-focused.

In fact, not only are requirements prominent in LLM-driven code generation, there is an acknowledgment that prompts should focus less on how to solve a problem with code, and more so on the intents behind the code – exactly what we and other authors have been arguing for decades with goal modeling. Thus, there is a rise in so-called "Intent Engineering", or "Intent-driven", "Intent-based" approaches for LLMs, for example as in [63–65].

Challenges: GORE underutilized in intent-driven AI. However, such approaches are mainly only recognizing the importance of intents in driving LLM-generated code or agentic systems, and not using the full expressive power of GORE, including qualities, goal decomposition, contributions, and trade-offs. Prompt templates help to make prompting more systematic and effective, but often only cover goals/objectives/directions/instructions in a shallow way with single statements [66].

Future directions: As the field of intent engineering grows in popularity, the question is whether this line of research will utilize the extensive body of work in GORE, ignore it, or invent parts of it in parallel. One open question is whether the rich semantics of goal modeling is actually useful for LLMs and agentic systems. Do they perform better if given qualities, alternatives, goal decompositions, etc., beyond just simple statements of intent? If so, systematic methods to elicit and structure such information will be developed. More empirical research in this area is needed.

3.5 GORE for agentic AI

GORE techniques belong to the symbolic (or classical) multiagent tradition in Artificial Intelligence, relying on explicit agent representations grounded in logic, and algorithmic decision-making for achieving goals [67], LLM reasoning, on the other hand, is performed internally by the language model in a sort of black box, limiting its ability to react dynamically to the environment and continuously incorporate previously acquired knowledge, which may lead to errors and hallucinations along its reasoning processes [68].

To cope with this limitation, the Agentic AI paradigm functionally enhances language models, by integrating them with intelligent agents embedded with mechanisms for planning complex sequences of actions, displaying social and environment-aware behavior in dynamic environments according to their internal states, enabling goal-oriented behaviors and greater operational autonomy [68, 69]. Furthermore, they typically perform their tasks embedded in a human-centered context, orchestrating tools, interacting with external data sources, and collaborating asynchronously with other agents in real time [67]. This paradigm shift led to the development of a multitude of Agentic frameworks and multiagent platforms.

Several current agentic frameworks take goals or goal-related information as input. For example, in AutoGPT, the user provides a high-level user prompt (composed of a user's goal, optional context and constraints), which is then broken down into a sequence of tasks (plan), analyzed by a sequence of other agents. [70]. CrewAI organizes agents as a "crew", with each agent having a role, backstory, tools and local objective [67]. The user specifies a global goal which is broken into subgoals and pursued through role-based coordination and delegation among agents. AutoGen is an open-source framework developed by Microsoft in which goals are achieved through multi-agent conversations (dialogue, negotiation and iterative refinement) and integration with external tools and LLM backends [67, 71]. In LangGraph, goals are encoded as nodes, states and transitions (i.e., state machines), with the stateful graph nodes preserving context between agent transitions in the graph-based structure and thus enabling agents to follow workflows [67].

Taken together, these frameworks reveal different strategies for operationalizing goals in Agentic AI systems. While frameworks such as AutoGPT emphasize planning and task decomposition, AutoGen relies more heavily on conversational coordination among agents. These differences suggest that goal handling has become a central architectural concern in contemporary agentic systems, although goal modeling remains implicit.

Challenges: extending concepts and use of goal models. The emergence of Agentic AI systems suggests the need to extend goal-oriented and agent-oriented methodologies toward the design and governance of adaptive AI agents. As shown in the results of the original mapping study, agents have long been a popular topic in goal modeling, but agentic technology at the time was fairly limited. In the last few years, through the rise of Agentic AI, we have gained the technical capability to realistically deploy functional and powerful agents. Traditional approaches such as Tropos and iStar were largely developed under the assumption that agents were designed to fulfill their goals through explicitly modeled reasoning structures and behaviors. Goal models defined the core of agents, namely their intentions and the actions through which such intentions were pursued. By contrast, many contemporary AI agents are not fully

designed to fulfill their goals; rather, they are incrementally trained to do so through interaction, feedback, and learning. As a consequence, both the operational interpretation of goals and the behavior through which goals are pursued may evolve over time.

The characteristics of adaptive AI systems further challenge several assumptions on which classical goal-oriented and agent-oriented methodologies were originally built. Traditional approaches such as Tropos were largely grounded on symbolic models of agency and assumed that goals, reasoning structures, and operational behavior could be specified explicitly at design time and would remain relatively stable once deployed. These assumptions become increasingly fragile in systems whose behavior is probabilistic, context-dependent, adaptive, and only partially predictable. Previous work has extended goal modeling to account for this uncertainty associated with AI, including associated experimental evidence [72], but such an approach has not yet been applied to continuously evolving Agentic systems.

In addition, several conceptual challenges emerge from the shift to Agentic AI. First, reasoning processes are no longer fully explicit or inspectable, since behavior increasingly emerges from neural inference and interaction history rather than from predefined plans and rules. Second, goals themselves may no longer admit stable interpretations over time. The operational meaning of a goal may evolve dynamically depending on context, feedback, and interaction, potentially leading to forms of goal drift in which system behavior progressively diverges from the original stakeholder intentions. Third, the traditional separation between design time and run time becomes blurred, because planning, negotiation, and decision-making increasingly occur dynamically during operation rather than being fully anticipated during design. Finally, adaptive AI systems may exhibit forms of autonomy that exceed predefined behavioral boundaries, raising new concerns regarding accountability, governance, and alignment with organizational objectives.

Application of GORE to the new agentic context also requires revisiting the notion of delegation. In classical Tropos, for example, delegation mainly concerned the assignment of goals and responsibilities among actors. In adaptive AI systems, delegation increasingly involves delegation of reasoning authority, namely the authority to operationalise goals at run time under context-dependent conditions. At the same time, responsibility for the outcomes of delegated decisions may still remain with human or organisational actors. This distinction may become central for future approaches to AI alignment, accountability, and governance.

A further implication concerns autonomy. Classical agent-oriented methodologies typically treated autonomy as a relatively stable design-time property. In adaptive AI systems, autonomy may instead need to be dynamically modulated depending on context, uncertainty, risk, organisational policies, or observed behaviour. Goal models may therefore provide the intentional grounding needed to justify and control such adaptations over time.

These developments also raise broader questions concerning the relationship between symbolic intentional models and neural representations. Classical GORE approaches rely on explicit representations of goals, dependencies, and rationales, whereas many capabilities of contemporary AI systems are encoded implicitly within neural models acquired through training. A key research challenge may therefore be understanding how explicit goal-oriented representations and neural representations can complement each other: the former providing intentional structure, traceability, and governance, and the latter providing adaptive and context-sensitive operational capabilities.

Future directions: These challenges suggest the need to revisit and extend methodologies such as Tropos in order to account for both agents whose behavior is fulfilled by design and agents whose behavior is fulfilled through learning. One possible direction is to distinguish between goals that can be operationalized through predefined reasoning structures and goals whose operationalization depends on adaptive interpretation and incremental learning. In this perspective, goals may no longer be viewed only as design-time specifications of intended behavior, but also as normative and evolving reference points against which adaptive behavior can be evaluated and governed over time.

This reinterpretation has important implications for GORE. First, it suggests that goal models should not be viewed only as static design artefacts, but also as persistent governance structures that remain relevant after deployment. Many current agentic AI systems already rely implicitly on notions that are central to GORE, including goals, subgoals, delegation, trade-offs, planning, coordination, and dependency management among multiple actors and tools. However, in most current approaches these concepts remain embedded in prompts, workflows, or emergent behaviors, rather than being explicitly modeled and analyzed. As a consequence, current agentic frameworks often lack explicit representations of stakeholder intentions, rationale, responsibilities, and quality trade-offs. Explicit representations of goals, softgoals, dependencies, and delegation relationships can support monitoring, escalation, accountability, and controlled adaptation throughout the operational lifecycle of AI systems. In particular, GORE can help structure reasoning about why an agent should behave in a certain way, under which organisational constraints, and on whose behalf. This becomes particularly important as AI systems increasingly operate autonomously in organisational and socio-technical settings where decisions involve conflicting goals, constraints, risks, and accountability concerns.

As a concrete example towards goal models in agentic monitoring, a recently funded project[4] will focus on using Agentic AI artifacts like prompts and scripts to auto-generate a goal model capturing the intended structure and intentions of an agentic system using goal models. Existing goal model analysis procedures [73, 74] will be applied, using evidence and metrics, for example, extracted from tools like Langtrace.ia,[5] to reason over current goal satisfaction. The study plans to start with goal modeling notations like iStar or Tropos, but then to evaluate and extend the concepts and visualizations best suited for agentic system modeling. Aspects and concepts from existing AI or ML requirements modeling approaches (e.g., RM4ML [75]) will be considered as part of model design. Further directions could work less towards bottom-up monitoring of existing agentic systems, and more towards model- and intention-driven design of such systems through goal modeling.

A further example explicitly adjusts goal-oriented concepts for agentic AI. More specifically, goal models defined the core of agents, their intentions and the actions through which they pursued these intentions. Agents in iStar 2.0 consisted of roles (e.g., cashier) or physical agents (e.g., Paolo). The advent of LLMs has created a new class of agents, call them agentives, that are not designed to fulfill their goals. Rather, they are trained to do so. Agentives start with a generic LLM and are incrementally trained to perform specialized tasks, for example answering user manual questions about a particular car model, such as "How do I change the time on the car clock when we move to daylight time?" For their training, agentives need an oracle, human or agent, that corrects when a response is wrong or incomplete.

In Bresciani et al., the authors of the Tropos methodology [5] propose an extension to Tropos that recognizes both agents-by-design (aka g-agents) and agentives (l-agents); also goals that are fulfilled by design (aka d-goals) and goals that are fulfilled through learning (aka l-goals), such as the user manual goal above [76]. The key claim of the new Bresciani et al. paper is that the addition of two concepts to the Tropos ontology, l-agents and l-goals, suffices to introduce into the Tropos methodology LLM concepts for the design and training of future agent-oriented systems that can account for traditional agents as well as agentives.

Taken together, these directions suggest that GORE is not becoming less relevant in the era of Agentic AI. On the contrary, its role may evolve from supporting mainly design-time reasoning toward providing explicit intentional foundations for alignment, governance, accountability, and controlled evolution of adaptive AI systems.

[4] Intentional Observation and Monitoring for Agentic AI Systems (IOMAS), funded by WASP (Wallenberg AI, Autonomous Systems and Software Program) Academic Doctoral Student Projects.

[5] https://www.langtrace.ai/

4 Reflections and conclusions

In this work, we have reflected on the GORE systematic mapping study from ten years ago [8], including motivations, method, and results. We have used bibliometrics to get a quantitative view of the visibility of the paper, although measuring the impact of this and other SLRs is challenging. Overall, we believe that the paper has had an impact on subsequent GORE work, in part by consolidating this vast body of research. The intention of the original paper was to make the breadth of goal modeling research more apparent to support new research directions, and avoid further papers addressing the same challenges and directions. It is difficult to assess if this intention was achieved without a follow-up systematic mapping study, but, informally, we have seen fewer of the same types of familiar goal model papers (e.g., reasoning), and more papers moving towards capturing AI or ML (e.g., [72]). This observation may be a direct result of the mapping study, the rise of AI, or a shift in membership in the GORE community, but we believe that our paper has made a contribution to this shift to new directions. Furthermore, given that many papers citing the original mapping study are not obviously about goal modeling, it is possible that the paper has had a broader impact, particularly on other mapping studies and SLRs.

Since the publication of the original SLR, there have been a number of key developments in and around requirements engineering that affect the perception and use of GORE. Specifically, GORE has faced and continues to face barriers to industrial adoption, but we believe such barriers may be mitigated by LLM support for goal modeling, or by a better fit between GORE and SE practices, with the rise of agentic systems. We use our experience to argue for the utility of GORE in SE education, and invite more studies evaluating whether the perceived benefits of teaching GORE hold. We have highlighted the growing popularity of LLM-generated GORE, but we emphasize how important it is to better understand and preserve the cognitive and human benefit of goal modeling. We have also reflected on growing trends in Intent-driven engineering and emphasize the potential for GORE to play a more active role in this space.

This reflection has placed a lot of focus on the rise of agentic AI, given the synergies between agent-oriented goal modeling frameworks and agentic systems. A general challenge is to extend the use cases, frameworks, and concepts of current GORE approaches for agentic AI systems, including using goal models for dynamic monitoring, and revisiting notions of delegation, symbolic reasoning and autonomy.

Reflecting on the state of systematic reviews in GORE, since the publication of the mapping, related mappings or SLRs have been published, but with a narrower focus, for example, self-adaptation [77], agile requirements modeling

with goal models [78], or as part of AI-specific mapping studies [79]. However, we have not seen a large-scale generic goal modeling review or mapping using new data. As such, we believe it would be very valuable to have an updated version of the review, or a similar review, to understand how GORE has evolved and been applied. Such a review may be particularly useful in light of the increased applicability of GORE in LLMs and agentic systems, to highlight and emphasize the continued work in this area. Reflecting on the original mapping indicates that the time may be right for an updated view of GORE in the last ten years. We very much invite the GORE community to conduct such a study.

Overall, we see great potential for using GORE concepts and frameworks to direct the efforts of LLMs and AI agents in generating and monitoring modern software systems. We invite not only the GORE community, but the wider RE, SE, and AI Engineering communities to build on the rich body of work as summarized in the original mapping study, and in any future mapping study or SLR, as well as the many related studies which followed. Without doing so, we risk – again – reinventing the wheel, this time within the body of the emerging agentic AI paradigm, precisely what the original mapping study was trying to avoid.

Acknowledgements The original mapping was supported by several project: ERC advanced grant 267856, "Lucretius: Foundations for Software Evolution," urlhttp://www.lucretius.eu an ERC Marie Skodowska-Curie Intra European Fellowship (PIEF-GA-2013-627489), a Natural Sciences and Engineering Research Council of Canada Postdoctoral Fellowship (Sept. 2014 - Aug. 2016), by the European Union's Horizon 2020 research and innovation program under Grant Agreement No. 653642– VisiON, and by the APOSTD grant (APOSTD/2014/064) from the Generalitat Valenciana. This project has received funding from the SESAR Joint Undertaking under Grant Agreement No. 699306 under European Union's Horizon 2020 research and innovation program. Blinded for review.

Author contributions Jennifer Horkoff organized the manuscript preparation, added several sections, edited and merged together submitted text, and submitted. All other authors did the following: Provided ideas for the manuscript preparation, provided paragraphs, sentences, figures or references for the manuscript, and reviewed the final manuscript.

Data availability Results of our lightweight snowballing exercises are available upon request. However, we hope these results will be superseded by a more rigorous and complete future systematic mapping study or literature review on GORE.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

References

1. Yu ES (1997) Towards modelling and reasoning support for early-phase requirements engineering. Proceedings of ISRE'97: 3rd IEEE international symposium on requirements engineering. IEEE, pp. 226–235
2. Dalpiaz F, Franch X, Horkoff J (2016). arXiv:1605.07767 arXiv preprint istar 2.0 language guide
3. Dardenne A, Van Lamsweerde A, Fickas S (1993) Goal-directed requirements acquisition. Sci Comput Program 20(1–2):3–50
4. Amyot D (2003) Introduction to the user requirements notation: learning by example. Comput Netw 42(3):285–301
5. Bresciani P, Perini A, Giorgini P, Giunchiglia F, Mylopoulos J (2004) Tropos: An agent-oriented software development methodology. Auton Agent Multi-Agent Syst 8(3):203–236
6. Greenspan SJ, Borgida A, Mylopoulos J (1986) A requirements modeling language and its logic. Inf Syst 11(1):9–23
7. Horkoff J, Aydemir FB, Cardoso E, Li T, Maté A, Paja E, Salnitri M, Mylopoulos J, Giorgini P (2016) Goal-oriented requirements engineering: a systematic literature map. IEEE 24th International Requirements Engineering Conference (RE). IEEE, pp. 106–115
8. Horkoff J, Aydemir FB, Cardoso E, Li T, Maté A, Paja E, Salnitri M, Piras L, Mylopoulos J, Giorgini P (2019) Goal-oriented requirements engineering: an extended systematic mapping study. Requirements Eng 24(2):133–160
9. Horkoff J, Li T, Li F-L, Salnitri M, Cardoso E, Giorgini P, Mylopoulos J, Pimentel J (2014) Taking goal models downstream: a systematic roadmap. In: 2014 IEEE eighth international conference on research challenges in information science (RCIS), pp. 1–12. IEEE
10. Horkoff J, Li T, Li F-L, Salnitri M, Cardoso E, Giorgini P, Mylopoulos J (2015) Using goal models downstream: a systematic roadmap and literature review. Int J Inf Syst Model Design 6(2):1–42
11. Brereton P, Kitchenham BA, Budgen D, Turner M, Khalil M (2007) Lessons from applying the systematic literature review process within the software engineering domain. J Syst Softw 80(4):571–583
12. Petersen K, Feldt R, Mujtaba S, Mattsson M (2008) Systematic mapping studies in software engineering. In: 12th international conference on evaluation and assessment in software engineering (EASE). BCS Learning & Development
13. Charmaz K (2015) Grounded theory. Qualitative psychology A practical guide to research methods 3(2015):53–84
14. Wohlin C (2021) Case study research in software engineering-it is a case, and it is a study, but is it a case study? Inf Softw Technol 133:106514
15. Mylopoulos J, Chung L, Nixon B (1992) Representing and using nonfunctional requirements: a process-oriented approach. IEEE Trans Software Eng 18(6):483–497
16. Li F, Horkoff J, Mylopoulos J, Liu L, Borgida A (2013) Non-functional requirements revisited. In: iStar, pp. 109–114
17. Li FL, Horkoff J, Mylopoulos J, Guizzardi R, Guizzardi G, Borgida A, Liu L (2014) Non-functional requirements as qualities, with a spice of ontology. In: 22nd Int. requirements engineering conference (RE), pp. 293–302. IEEE
18. Piras L, Dellagiacoma D, Perini A, Susi A, Giorgini P, Mylopoulos J (2019) Design thinking and acceptance requirements for designing gamified software. Int. Conf. on Research Challenges in Information Science (RCIS). IEEE,

19. Piras L, Paja E, Giorgini P, Mylopoulos J (2017) Goal models for acceptance requirements analysis and gamification design. In: Int. Conf. on Conceptual Modeling (ER). Springer
20. Piras L, Paja E, Cuel R, Ponte D, Giorgini P, Mylopoulos J (2017) Gamification solutions for software acceptance: a comparative study of requirements engineering and organizational behavior techniques. 11th IEEE international conference on research challenges in information science (RCIS). IEEE
21. Piras L (2018) Agon: a gamification-based framework for acceptance requirements. Trento, Italy (PhD thesis)
22. Alkubaisy D (2021) ConfIs: a tool for privacy and security analysis and conflict resolution for supporting GDPR compliance through privacy-by-design. In: 16th International Conference on Evaluation of Novel Approaches to Software Engineering (ENASE). Springer, pp. 78–93
23. Piras L, Al-Obeidallah MG, Pavlidis M, Mouratidis H, Tsohou A, Magkos E, Praitano A, Iodice A, Crespo BGN (2020) DEFeND DSM: a data scope management service for model-based privacy by design GDPR compliance. International conference on trust, privacy and security in digital business (TrustBus). Springer, pp. 78–93
24. Al-Obeidallah M, Piras L, Iloanugo O, Mouratidis H, Alkubaisy D, Dellagiacoma D (2023) Goal-modeling privacy-by-design patterns for supporting GDPR compliance. In: International conference on software technologies (ICSOFT)
25. Piras L, Al-Obeidallah MG, Praitano A, Tsohou A, Mouratidis H, Crespo BG, Bernard -N, Fiorani JB, Magkos M, Sanz E, AC (2019) DEFeND architecture: a privacy by design platform for GDPR Compliance. International conference on trust and privacy in digital business (TrustBus). Springer, pp. 78–93
26. Yu ES (1997) Proceedings of ISRE'97: 3rd IEEE international symposium on requirements engineering. IEEE, pp. 226–235 (Towards Modelling and Reasoning Support for Early-Phase Requirements Engineering)
27. Amyot D (2003) Introduction to the user requirements notation: learning by example. Comput Netw 42(3):285–301
28. Giorgini P, Kolp M, Mylopoulos J, Pistore M (2004) The tropos methodology: an overview. Methodologies and software engineering for agent systems: the agent-oriented software engineering handbook. pp. 89–106
29. Bresciani P, Perini A, Giorgini P, Giunchiglia F, Mylopoulos J (2004) Tropos: An agent-oriented software development methodology. Auton Agent Multi-Agent Syst 8(3):203–236
30. Mouratidis H, Giorgini P (2007) Secure tropos: a security-oriented extension of the tropos methodology. Int J Software Eng Knowl Eng
31. Mouratidis H (2011) Secure software systems engineering: the secure tropos approach. JSW
32. Piras L, Al-Obeidallah MG, Pavlidis M, Mouratidis H, Tsohou A, Magkos E, Praitano A (2021) A data scope management service to support privacy by design and GDPR compliance. J Data Intelligence
33. Alkubaisy D et al (2022) A framework for privacy and security requirements analysis and conflict resolution for supporting GDPR compliance through privacy-by-design. Communications in Computer and Information Science (CCIS). Springer
34. Horkoff J, Aydemir FB, Cardoso E, Li T, Maté A, Paja E, Salnitri M, Piras L, Mylopoulos J, Giorgini P (2017) Goal-oriented requirements engineering: an extended systematic mapping study. Require Eng J 1–28
35. Hojas-Mazo W, Moreno-Espino M, Martínez JVB, Pérez FM, Fonseca IL (2021) Goal-oriented email stream classifier with a multi-agent system approach. Int J Adv Comput Sci Appl 12(9):575–580
36. Alsaqaf W, Daneva M, Wieringa R (2023) Towards a solution proposal to agile quality requirements challenges in large-scale projects. 2nd International Workshop on Agile Methods for Information Systems Engineering, Agil-ISE 2023. CEUR, pp. 30–35
37. Mavin A, Wilkinson P, Teufl S, Femmer H, Eckhardt J, Mund J (2017) Does goal-oriented requirements engineering achieve its goal? In: 2017 IEEE 25th International Requirements Engineering Conference (RE), pp. 174–183. IEEE
38. Ralph P, Ali NB, Baltes S, Bianculli D, Diaz J, Dittrich Y, Ernst N, Felderer M, Feldt R, Filieri A (2020) Empirical standards for software engineering research.arXiv:2010.03525 arXiv preprint
39. Alrajeh D, Nowack V, Benjamin P, Thomas K, Hobson W, Munoz CG, Hamilton-Giachritsis C, Kloess J, Woodhams J, Butler D (2025) Data-dependent goal modelling for ml-enabled law enforcement systems. Software Engineering in Society (SEIS)
40. Horkoff J, Hammouda I, Lindman J, Debbiche J, Freiholtz M, Liao P, Mensah S, Strömberg A (2017) Goals, workflow, and value: Case study experiences with three modeling frameworks. IFIP Working Conference on The Practice of Enterprise Modeling. Springer, pp. 96–111
41. Maté A, Trujillo J, García F, Serrano M, Piattini M (2016) Empowering global software development with business intelligence. Inf Softw Technol 76:81–91
42. Barrera JM, Reina-Reina A, Lavalle A, Maté A, Trujillo J (2024) An extension of istar for machine learning requirements by following the prise methodology. Computer Standards Interfaces 88:103806
43. Lavalle A, Maté A, Santos MY, Guimarães P, Trujillo J, Santos A (2025) A methodology for the systematic design of storytelling dashboards applied to industry 4.0. Data Knowledge Eng 156:102410
44. Sousa V, Lavalle A, Maté A, Vieira A, Santos MY (2026) Extending istar for synthetic data generation and simulation modeling for industry 5.0. International working conference on requirements engineering: foundation for software quality. Springer, pp. 55–64
45. Horkoff J, Lockerbie J, Franch X, Yu E, Mylopoulos J (2015) Report on the first international i* teaching workshop (istart) June 9th, co-located with the conference on advanced information systems engineering (caise'15), stockholm, sweden. ACM SIGSOFT Software Engineering Notes 40(6):24–27
46. Li T, Zhou Q, Wang Y, Xiong H, Liu W, Ge N (2025) Understanding the challenges and requirements for facilitating istar learning: An empirical study with istar learners. Inf Software Technol 107839
47. Elbishbishy S, Raja JJ, Brings J, Daun M (2026) Improving students' ability to write requirements by teaching goal modeling
48. Franch X, Ruiz M (2021) Goal-oriented models for teaching and understanding data structures. International conference on conceptual modeling. Springer, pp. 227–241
49. Horkoff J (2018) The influence of agile methods on requirements engineering courses. IEEE 8th international workshop on requirements engineering education and training (REET). IEEE, pp. 11–19
50. Bennaceur A, Lockerbie J, Horkoff J (2015) On the learnability of i*: Experiences from a new teacher. iStarT@ CAiSE. pp. 43–48
51. Xi Z, Chen W, Guo X, He W, Ding Y, Hong B, Zhang M, Wang J, Jin S, Zhou E (2025) The rise and potential of large language model based agents: A survey. SCIENCE CHINA Inf Sci 68(2):121101
52. Brown T, Mann B, Ryder N, Subbiah M, Kaplan JD, Dhariwal P, Neelakantan A, Shyam P, Sastry G, Askell A (2020) Language models are few-shot learners. Adv Neural Inf Process Syst 33:1877–1901
53. He J, Treude C, Lo D (2025) Llm-based multi-agent systems for software engineering: Literature review, vision, and the road ahead. ACM Transactions on Software Engineering and Methodology 34(5):1–30

54. Cheng H, Husen JH, Lu Y, Racharak T, Yoshioka N, Ubayashi N, Washizaki H (2026) Generative ai for requirements engineering: A systematic literature review. Software: Practice and Experience 56(2), 141–170
55. Storey VC, Pastor O, Guizzardi G, Liddle SW, Maaß W, Parsons J, Ralyté J, Santos MY (2025) Large language models for conceptual modeling: Assessment and application potential. Data & Knowledge Engineering 160:102480
56. Siddeshwar V, Alwidian S, Makrehchi M (2024) A Comparative Study of Large Language Models for Goal Model Extraction. Proceedings of the ACM/IEEE 27th International Conference on Model Driven Engineering Languages and Systems. pp. 253–263
57. Ren S, Nakagawa H, Tsuchiya T (2024) Harnessing llm conversations for goal model generation from user reviews. ICAART (3). pp. 385–392
58. Sharfuddin A, Breaux T (2025) Generative goal modeling. 2025 IEEE 33rd International Requirements Engineering Conference (RE). IEEE, pp. 92–103
59. Li T, Zhou Q, Dong F, Zhang T, Wang Y (2026) An interactive and ai-enhanced framework for semi-automatically generating istar goal models. Autom Softw Eng 33(2):41
60. Hassine J (2024) An LLM-based Approach to Recover Traceability Links Between Security Requirements and Goal Models. Proceedings of the 28th International Conference on Evaluation and Assessment in Software Engineering. pp. 643–651
61. Gutschmidt A, Verbruggen C, Snoeck M (2025) A study on the impact of the level of participation in enterprise modeling. Software And Systems Modeling 1–16
62. Xu W, Liang Z, Mei K, Gao H, Tan J, Zhang Y (2026) A-mem: Agentic memory for llm agents. Adv Neural Inf Process Syst 38:17577–17604
63. Yoon J, Feldt R, Yoo S (2024) Intent-driven mobile gui testing with autonomous large language model agents. 2024 IEEE Conference on Software Testing, Verification and Validation (ICST). IEEE, pp. 129–139
64. Wang Z, Lin S, Yan G, Ghorbani S, Yu M, Zhou J, Hu N, Baruah L, Peters S, Kamath S (2025) Intent-driven network management with multi-agent llms: The confucius framework. Proceedings of the ACM SIGCOMM 2025 Conference. pp. 347–362
65. Qin X, Sun M, Dai J, Ma P, Cao Y, Zhang J, Wang J, Xu X, Zhang P, Niyato D (2025) Generative ai meets wireless networking: An interactive paradigm for intent-driven communications. IEEE Transactions on Cognitive Communications and Networking
66. Mao Y, He J, Chen C (2025) From prompts to templates: A systematic prompt template analysis for real-world LLMapps. Proceedings of the 33rd ACM International Conference on the Foundations of Software Engineering. pp. 75–86
67. Derouiche H, Brahmi Z, Mazeni H (2025) Agentic ai frameworks: Architectures, protocols, and design challenges. arXiv:2508.10146 arXiv preprint
68. Yao S, Zhao J, Yu D, Du N, Shafran I, Narasimhan K, Cao Y (2022) React: Synergizing reasoning and acting in language models.arXiv:2210.03629 arXiv preprint
69. Xie T, Zhou F, Cheng Z, Shi P, Weng L, Liu Y, Hua TJ, Zhao J, Liu Q, Liu C et al (2023) Openagents: An open platform for language agents in the wild. arXiv preprint arXiv:2310.10634
70. Belcic I (2026) What is AutoGPT?. https://www.ibm.com/br-pt/think/topics/autogpt Accessed 2026-05-17
71. Wu Q, Bansal G, Zhang J, Wu Y, Li B, Zhu E, Jiang L, Zhang X, Zhang S, Liu J (2024) Autogen: Enabling next-gen llm applications via multi-agent conversations. In: First Conference on Language Modeling
72. Ishikawa F, Matsuno Y (2020) Evidence-driven requirements engineering for uncertainty of machine learning-based systems. 2020 IEEE 28th International Requirements Engineering Conference (RE). IEEE, pp. 346–351
73. Horkoff J, Yu E (2016) Interactive goal model analysis for early requirements engineering. Requirements Eng 21(1):29–61
74. Amyot D, Ghanavati S, Horkoff J, Mussbacher G, Peyton L, Yu E (2010) Evaluating goal models within the goal-oriented requirement language. Int J Intell Syst 25(8):841–877
75. Yang Y, Zeng B, Gao J (2025) Rm4ml: requirements model for machine learning-enabled software systems. Requirements Eng 30(1):1–33
76. Bresciani P, Perini A, Giorgini P, Giunchiglia F, Mylopoulos J (2026) Tropos+: Tropos Extended in the Days of Large Language Models. Conceptual Modelling Conference 2026 (submitted). Submitted for publication
77. Andaa AA, Amyot D (2022) Self-adaptation driven by sysml and goal models–a literature review. e-Informatica Software Engineering Journal **16**(1)
78. Muelas-Muelas F-I, Narváez-Jiménez L-C, Pardo-Calvache C-J (2022) Agile functional requirements modeling from goal-oriented requirements engineering: A systematic mapping study. Facultad de Ingeniería 31(62):
79. Habiba U-E, Haug M, Bogner J, Wagner S (2024) How mature is requirements engineering for ai-based systems? a systematic mapping study on practices, challenges, and future research directions. Requirements Eng 29(4):567–600

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Authors and Affiliations

Jennifer Horkoff[1,2] · Fatma Basak Aydemir[3] · Evellin Cardoso[4] · Tong Li[5] · Alejandro Mate[6] · Elda Paja[7] · Mattia Salnitri[8] · Luca Piras[9] · John Mylopoulos[10] · Paolo Giorgini[11]

✉ Jennifer Horkoff
jennifer.horkoff@gu.se

Fatma Basak Aydemir
f.b.aydemir@uu.nl

Evellin Cardoso
evellin@ufg.br

Tong Li
litong@bjut.edu.cn

Alejandro Mate
amate@dlsi.ua.es

Elda Paja
elpa@itu.dk

Mattia Salnitri
mattia.salnitri@unibg.it

Luca Piras
l.piras@fbk.eu

John Mylopoulos
jmylopou@eecs.uottawa.ca

Paolo Giorgini
paolo.giorgini@unitn.it

1 University of Gothenburg, Gothenburg, Sweden

2 Chalmers University of Technology, Gothenburg, Sweden

3 Utrecht University, Utrecht, Netherlands

4 Universidade Federal de Goiás, Goiânia, Brazil

5 Beijing University of Technology, Beijing, China

6 University of Alicante, Alicante, Spain

7 IT University of Copenhagen, Copenhagen, Denmark

8 University of Bergamo, Bergamo, Italy

9 Fondazione Bruno Kessler, Trento, Italy

10 University of Ottawa, Ottawa, Canada

11 University of Trento, Trento, Italy