



Large language models for conceptual modeling: Assessment and application potential



Veda C. Storey^a*, Oscar Pastor^b, Giancarlo Guizzardi^c, Stephen W. Liddle^d, Wolfgang Maas^e, Jeffrey Parsons^f, Jolita Ralyte^g, Maribel Yasmina Santos^h

^a Georgia State University, Atlanta, GA, USA

^b Universitat Politècnica de València, Spain

^c University of Twente, The Netherlands

^d Brigham Young University, Provo, UT, USA

^e Saarland University and German Research Center for Artificial Intelligence, Germany

^f Memorial University of Newfoundland, Canada

^g University of Geneva, Switzerland

^h ALGORITMI Research Center, University of Minho, Portugal

ARTICLE INFO

Keywords:

Conceptual modeling

Large language models

Artificial intelligence

Semantics

ABSTRACT

Large Language Models (LLMs) are being rapidly adopted for many activities in organizations, business, and education. Included in their applications are capabilities to generate text, code, and models. This leads to questions about their potential role in the conceptual modeling part of information systems development. This paper reports on a panel presented at the *43rd International Conference on Conceptual Modeling* where researchers discussed the current and potential role of LLMs in conceptual modeling. The panelists discussed applications and interest levels and expressed both optimism and caution in the adoption of LLMs. Suggested is a need for much continued research by the conceptual modeling community on LLM development and their role in research and teaching.

1. Introduction

The history of humankind is a long story centered on the development of tools and technology to improve the human condition. Writing, for example, evolved from proto-writing systems using mnemonics and symbols, to hieroglyphic or cuneiform types of systems, to alphabet-based systems for representing words of spoken languages. Tools for writing likewise evolved across the millennia, from rocks to engraving tools to printing presses, typewriters, and word processing systems. Each area of human interest shows similar development trends, which continues to be evident in our modern, digital age. The digitalization of society has involved the continuous movement of tasks and processes to digital forms, resulting in many challenges related to data management, interoperability, and the role of digital agents.

Data management focuses on how to identify, capture, represent, store, and manipulate the data needed for an application. During the development process, it is important to represent the real world in a way that is understandable to developers, designers, and users. Entity-relationship modeling [1], which is a significant component of the field of conceptual modeling, emerged over five decades ago and remains a popular form of representation for a conceptual model. Conceptual modeling has also evolved as topics

* Corresponding author.

E-mail address: vstorey@gsu.edu (V.C. Storey).

have emerged in design methodologies, requirements, and ontology [2]. Work on the advancement of conceptual modeling topics has been reported in various outlets, including conferences and journal publications [3–9].

The growth of Large Language Models (LLMs) is now affecting software and information systems engineering practice in a way that many researchers consider disruptive. There are concerns over whether LLMs are over-hyped, or are truly a disruptive technology with the potential to affect many applications. In this “wave” of technological advancements, there are many relevant discussions regarding how LLMs can generally influence information systems design and development. Similar to how conceptual modeling efforts adapt to other topics, we are particularly interested in identifying the value LLMs might provide to the conceptual modeling community, along with the risks and concerns that might arise.

In a broader sense, we are interested in the role of artificial intelligence (AI) in conceptual modeling activities. Our discussion is structured around three main questions.

- What is the current state of the art?
- What is the potential impact of LLMs on conceptual modeling and conceptual modelers?
- What are other challenges and issues regarding the future of LLMs and conceptual modeling research?

This panel paper explores LLMs and their potential applications in conceptual modeling, highlighting the various ways in which they may shape the field and suggesting future research directions. We aim to provide insights that clarify how the emergence of LLMs could influence conceptual modeling practice.

To achieve its purpose, the paper is organized as follows. Section 2 presents the questions that guide the assessment of the impact of LLMs in conceptual modeling and modelers. Section 3 provides an overview of the state of the art. Section 4 highlights the challenges and issues of LLMs when managing abstractions and identifying concepts. Section 5 discusses the impact of LLMs on conceptual modeling and conceptual modelers. Section 6 shows the potential use of LLMs in conceptual modeling. Section 7 addresses the future of LLMs and conceptual modeling, with Section 8 highlighting potential research topics. Section 9 then concludes the paper.

2. LLMs in conceptual modeling: Assessment questions

LLMs are deep learning models. They have been trained on very large datasets and have been shown to perform quite well at language processing tasks. They generate new text by predicting word sequences, based on the input they receive, ([10], p. 2). Generative artificial intelligence (GenAI) tools, such as ChatGPT, are based on LLMs and algorithms. They create content that is text-based, along with music, software source code, and images, among other representation forms. Their use has created much enthusiasm and experimentation with applications across a wide range of fields. However, caution is warranted. Saba notes that “... due to their sub-symbolic nature, whatever ‘knowledge’ these models acquire about how language works will always be buried in billions of microfeatures ... none of which humans can meaningfully interpret. This is also the reason why explainability can never be attained in such models ...” ([11], p. 3). Thus, the use of LLMs suggests exercising care when exploring these topics together, while attempting to gain insights into the various ways in which LLMs might influence conceptual modeling, positively or negatively.

We are first interested in the *state of the art*. Are LLMs a threat for conceptual modeling or will they become an effective assistant to better create conceptual models? What influences are LLMs already having on conceptual modeling? Are LLMs making humans irrelevant because LLMs can perform the modeling for us? Are LLMs improving conceptual modeling practices? If so, how? Is the potential of LLMs overstated or understated?

An assessment of the *impact on conceptual modeling and conceptual modelers* raises the following questions. Can LLMs help to address conceptual modeling challenges that have been unachievable or difficult to achieve without them? Conversely, could LLMs make individuals ‘less intelligent’, leading to homogenized outputs that overlook critical nuances in complex challenges? Are LLMs capable of ‘reasoning’ (as suggested in the latest round of claims about them)? And, if so, how much reasoning? Can we use LLMs to produce initial drafts of a conceptual model? What are the *challenges* associated with using LLMs in conceptual modeling activities—for both research and education? Finally, as with any new or emerging technology, questions regarding concerns about the future deserve attention.

The purpose of this paper is to share the panelists’ views on the diverse dimensions of LLMs. For instance, what could be the implications for us as conceptual modeling researchers? How will LLM use affect conceptual modeling present and future practice? Can a prompt engineering perspective provide a new type of programming practice for conceptual modeling?

We present the ideas and insights shared during the panel related to these main topics. Other questions and discussion topics that emerged from considering the potential impact of LLMs on conceptual modeling are also reported. In this way, we intend to bring attention to, and consideration for, the potential of LLMs in conceptual modeling.

3. State of the art

Transformers [12] and diffusion models [13] are the predominant architectures in GenAI, providing robust frameworks for high-dimensional data processing and generative tasks. These models have proven effective across various applications, from natural language processing to image generation, solidifying their role as foundational designs. Furthermore, self-supervised learning (SSL) [14] and causal representation learning [15] have emerged as critical methodologies, allowing models to utilize unlabeled data more efficiently, while incorporating causal structures for improved generalization and robustness in later tasks. Advances in state-space models [16] and quantum machine learning (QML) [17] are promising developments offering new perspectives and potential capabilities that could transform machine learning.

3.1. Integration of conceptual modeling and AI

For conceptual modeling and artificial intelligence (AI), generally, there have been efforts to embrace these complementary activities with promising conceptual modeling research extending in both directions (the impact of AI on conceptual modeling and the impact of conceptual modeling on AI). However, there are still concerns, with some research recognizing its potential to enhance modeling practices. More skeptical concerns remain over AI's inherent complexity and potential over-reliance. Notably, experimental studies are being conducted to investigate the reciprocal influence between conceptual modeling and AI, examining how AI methods can benefit from conceptual modeling frameworks and vice versa [18]. This bidirectional exploration suggests that AI may be structured more effectively through conceptual modeling principles, whereas conceptual modeling practices may gain new analytical tools and automation capabilities through AI integration, creating a dynamic that could accelerate innovation in both fields.

3.2. LLMs in conceptual modeling tasks

LLMs are being used in different conceptual modeling tasks to assess how they can support conceptual modeling activities or automate some of these activities. As seen in [Table 1](#), the analysis of results usually includes baseline models or specific use cases, with *F-measure*, *Precision*, and *Recall* measures used to quantify performance when applicable.

To compare the performance of GPT3.5 and GPT4 for fully automated domain modeling, Chen et al. [19] used prompt engineering techniques on a dataset that includes ten different domain modeling examples, including reference solutions created by modeling experts. The authors highlight that, although LLMs show domain understanding capabilities, they do not yet allow full automation. The best results were achieved in class and attribute generation, with worse results for relationship generation. Entity relation extraction is also addressed in [20], based on unlabeled text data. Here, in-context learning guides the results towards a specified format to help obtain structured data.

For automating requirements engineering tasks, LLMs are used to extract domain models from product logs and user stories [21]. The results are compared with state-of-practice tools and Natural Language Processing (NLP) approaches. To evaluate the extracted domain models, a ground truth with annotated user stories was prepared, which was complemented by running Visual Narrator on top of the dataset to obtain a baseline from the state of practice. For the comparison with the NLP approach, Conditional Random Fields (CRF) were used to transform a user story into a sequence of tuples. The obtained results point out that the CRF approach, designed with the help of NLP experts, outperforms GPT3.5 results, although these are better than the state of practice.

Fill et al. [22] present an experiment conducted using ChatGPT, namely GPT-4, to explore potential applications for generating and interpreting conceptual models. In particular, the work explores how to generate and interpret Entity-Relationship (ER) diagrams, Business Process (BPMN) diagrams, UML class diagrams, and Heraklit models, highlighting how modeling applications could be realized. The conducted experiments were based on prompts that define the context and include a variable for a concrete task definition. Although the results show some correct results, they still require experience in modeling to validate them. The researchers concluded that LLMs can quickly generate a first draft result, which can be refined with a stepwise approach [22].

Cámará et al. [23] focus on UML class diagrams, enriched with OCL (Object Constraint Language) constraints, for modeling tasks and assisting modelers. They identify syntactic and semantic deficiencies, and the lack of consistency, when using ChatGPT for software model development, concluding that ChatGPT's capabilities in this task are not as good as those for code generation. Moreover, the variability and inconsistency of the resulting models, for the same prompts, were high, hindering the repeatability and reproducibility of those results. On a positive note, the authors highlight the results obtained for the OCL expressions and constraints.

Addressing the extraction of information from textual process descriptions for process models with activities and actors as process elements, Neuberger et al. [24] show how adequate prompt strategies for LLMs can outperform state-of-the-art machine learning approaches. Three different datasets and eight LLMs were used, contributing to a broad evaluation. The number of example texts, the specificity of the definitions, and the rigor of the format instructions contribute to better results. The Tree-of-Thoughts strategy proposed by [25] also acknowledges the relevance of the prompting strategy, allowing the decomposition of the modeling process into several sub-steps. In each step, specific prompts are used to optimize the quality of the LLM output.

Moving towards automation of domain modeling with LLMs and the collaboration of human experts, Prokop et al. [26] present an assistant that provides modeling suggestions based on a given textual description of the domain. This assists in the identification of the classes, attributes, and associations.

More recently, Rizzi [27] addresses the conceptual design of multidimensional cubes that integrate measures, dimensions, and attributes. Designers usually obtain these models manually in close collaboration with end-users. The results show that the used prompts can improve the refinements, although involvement of designers is still needed to ensure the validity of the refined models.

3.3. Emerging developments

LLMs are known for their ability to make predictions based on text. There are also many existing useful AI and non-AI models. Additionally, the ubiquitous transformer models are diffusion models. Cell-supervised learning is becoming more prevalent. Cell-supervised learning is a type of self-learning model that occurs in machine learning. It learns from the characteristics of the data itself. It is an energetic kind of field, and worthy of consideration. State-space models are also promising developments, as is work on quantum machine learning, which requires the learning of new mathematics, such as that undertaken by theoretical physicians working in this area. [28].

Table 1
LLMs in conceptual modeling tasks.

Work	Conceptual modeling task	LLM	Evaluation	Metrics
[19]	Domain Models (Classes, Attributes, Relationships)	GPT3.5, GPT-4	Baseline models provided by domain experts	F-measure (F1), Precision, Recall
[20]	Entity Relation Extraction	ChatGLM	Baseline models	F-measure (F1), Precision, Recall
[21]	Domain Models	GPT-3.5	Ground truth with annotated user stories and a baseline from the state of practice	F-measure (F1)
[22]	ER Diagram, BPMN Diagram, UML Class Diagram, Heraklit Diagrams	GPT-4	Domain expert	- - - - -
[23]	UML Class Diagram (with OCL)	ChatGPT	Domain expert	- - - - -
[24]	Process Model (Mention Detection — activities, objects, actors; Entity Resolution; Relation Extraction)	GPT-4o, GPT-4-2024-04-09, GPT-4-0125-preview, GPT-3.5-0125, Claude 3 Opus, Claude 3 Sonnet, Llama 3 70B Instruct, Qwen1.5 72B Chat	Baseline models	F-measure (F1), Precision, Recall
[25]	UML Class Diagram	GPT-4	Baseline models	F-measure (F1), Precision, Recall
[26]	UML Class Diagram	Mixtral-8 × 7B, Llama-3-70B (both with and without RAG)	Baseline models	F-measure (F1), Precision, Recall
[27]	Dimensional Fact Model	GPT-4o	Five use cases with increasing difficulties	- - - - -

4. Challenges and issues

A compelling set of relevant points explores the nature of LLMs and how they manage abstraction and identify concepts.

4.1. Nature of LLMs

LLMs are primarily text-based. Together with the fact that LLMs make statistical predictions, this implies that LLMs cannot effectively manage semantics. They do, however, simulate semantics to a limited extent. Generally, humans as modelers internalize numerous external relationships. Therefore, to capture real-world semantics, a referential mapping to this human internalization is required. This leads to a problem of semantic interpretability. These models need to be explanatory, not just descriptive, but it is not clear how they can obtain explanations out of descriptions. Many conceptual models are not very effective, so it does not seem reasonable to extract something that is an interpretation of the real world from them. A bigger issue is to identify what is needed to create novel conceptual models for new domains. This might require abstracting from an application or reducing a phenomenon to one that modelers can understand.

There are some expectations for semantic-driven systems engineering [29]. Since intelligence, generally, is “knowing what to do when we don’t know what to do”, we can reduce general patterns to patterns of things we know.

4.2. Abstractions and concepts

Conceptual modeling is heavily based on abstraction. An important consideration is whether the conceptual modeling community can delegate activities, such as abstracting and modeling, to systems based on LLMs. For example, using a dataset as input, systems such as ChatGPT can extract a class diagram that identifies a reasonable set of concepts. It might be possible for the conceptual modeling community to delegate abstraction and modeling, but under human supervision (at least given the current state of the technology). One possibility is that LLMs can be serve as effective and helpful modeling assistants. For example, suppose we were to instruct ChatGPT to create a data model of a data warehouse system. It might be able to extract appropriate and relevant constructs, facilitating and improving the process of creating the “right” model. This aspect was reinforced by stating that LLMs can be viewed as an enhancement of conceptual modeling, but not as a replacement for humans.

Following this argumentation, it is possible to use and explore LLMs, but we need to identify the risks. LLMs can help process large amounts of text. The extraction of entities and relationships can help to identify concepts and goals. LLMs can also help to organize ideas and detect patterns, which could be a source of expertise. However, the response generation is based on probabilistic models, raising questions as to whether it is acceptable. Is there any bias? What is the quality of the data used for the inference? Issues of importance are abstractions and relevance in conceptual modeling, which require a need for a deep understanding of the application domain and problem. For now, they still require oversight. Thus, the risk is whether to “fully trust” an LLM or whether it should even be trusted at all. Given the limitations, it is likely that LLMs can provide assistance rather than replacement.

Furthermore, LLMs have often been described as a moving target, making it challenging to describe the state of the art. The first generation of LLMs raised many concerns about the problem of hallucination. Retrieval Augmented Generation (RAG) and RAG Fusion are now being used to verify knowledge sources. There are recent claims that LLMs have some capability to “reason”. By dividing a problem into multiple steps in what is sometimes called “chain-of-thought” prompting, and then using reinforcement learning, LLMs can better tackle complex tasks in a simulated “reasoning” approach.¹ Fundamentally, though, what does it mean for a tool, which is based on statistical word prediction, to claim to be able to reason? A related issue is understanding what role these reasoning LLM skills will have in conceptual modeling.

An interesting observation is the connection with the history of communication and representation, emphasizing that the correct use of a new invention is crucial to obtain an adequate evolution. LLMs may be another link in the sequence of tools that have been developed over time. Although current versions of LLMs have claims of reasoning, they really only make predictions. Nevertheless, they can be treated as a tool. Since LLMs have ingested many conceptual models, it should be possible to evaluate them. Productivity might also be obtained in other ways; for example, as chatbots to provide students with tutoring capabilities. Even teaching students how to judge the value of the output could, optimistically, be a useful advancement. It is difficult, of course, because LLMs cannot deal directly with the semantics of the constructs used in the conceptual model representation.

The use of LLMs in conceptual modeling ranges from enthusiasm to awareness to apprehension of risk:

- *Enthusiasm.* Conceptual modeling is time-consuming and difficult because it involves both the abstraction of concepts from the real world and their representation. The positive side of LLMs is that they are rather excellent at processing language-based content. If the input available for building a conceptual model is in the form of text, LLMs can certainly increase the efficiency of the conceptual modeling process by: extracting entities, relationships, and attributes for domain modeling; goals and actors for goal modeling; events and activities for process modeling, etc. They can also contribute to organizing ideas, suggesting concepts (and constructs), detecting patterns, and generating initial models.
- LLMs enable tasks to be scaled in a way that was previously impossible. They can analyze large quantities of text or data much faster than human modelers and help address scalability issues in large applications. They can also be considered as a source of domain expertise because they generate information from multiple and varied data sources.
- *Awareness.* We need to be aware that the methods used by LLMs to generate information, and therefore conceptual models, are based on probabilistic prediction, which is highly dependent on the quality of the data used to train them. Biased or poor-quality data can lead to incorrect and inconsistent predictions. While LLMs are likely to assist in making conceptual modeling more efficient, they require human oversight to ensure accuracy and domain-specific relevance. Such oversight is required.
- Conceptual modeling relies on abstraction mechanisms, logical reasoning, domain knowledge, context assessment and, especially, the determination of what is and what is not relevant to the current modeling scope and situation. LLMs are not in a position to do this. Current machine learning approaches do not enable LLMs to acquire human-level logical reasoning and/or critical thinking. LLMs are unable to plan or forecast the future, decide on development and innovation strategies, express wishes, or invent. An interesting challenge is to analyze the scope and limits of their evolution to make the reasoning challenge achievable.
- *Risk.* Misunderstanding the limitations of LLMs leads to over-reliance on LLMs and, therefore, a lack of monitoring. If model generation is not carefully controlled and validated by modeling experts, the result is situations where the models generated are incorrect or inconsistent.
- The over-reliance on LLMs could also gradually lead to an under-appreciation of human expertise and judgment. Furthermore, we are trying to explain a new phenomenon. This is not something we can easily perform at scale.

Thus, we can conclude that LLMs cannot yet replace human modelers, especially with current prediction algorithms and methods. However, they are powerful enough to improve conceptual modeling practice. LLMs do not prevent humans from modeling, but rather may change the manner in which humans engage with the modeling process. Instead of eliminating the need for human modelers, LLMs shift the focus of their role from tedious and repetitive tasks to more high-level, supervisory, and creative activities.

5. Potential impact of LLMs on conceptual modeling and conceptual modelers

There are significant potential challenges that LLMs can help to address in conceptual modeling. In today’s digital world, there are vast amounts of unstructured data to be processed. Certainly an automated system can do that more efficiently and probably more effectively than humans can. Humans communicate in natural language. Since LLMs are based on natural language and excel at natural language tasks, they might be able to help generate conceptual models from natural language text. They could also help to integrate cross-domain knowledge; that is knowledge from different application domains for which it might be desirable to integrate to address a problem.

The list of relevant conceptual modeling challenges that LLMs can likely help to address includes:

- Processing vast amounts of unstructured data

¹ See, e.g. <https://www.vox.com/future-perfect/400531/ai-reasoning-models-openai-deepseek> and <https://sebastianraschka.com/blog/2025/understanding-reasoning-lm.html>, retrieved 6 May 2025.

- Generating models from natural language
- Integrating cross-domain knowledge
- Maintaining models for dynamic systems
- Improving communication by supporting the building of glossaries and taxonomies

As adoption of LLMs continues at a rapid pace, we face the challenge of over-reliance to the point where we can lose our habit of independent thinking. Then, the results obtained will appear similar to those obtained by all others who engage with an LLM for a similar task. This becomes a challenge of over-reliance instead of using LLMs to reduce cognitive load. For teaching, perhaps we should include some techniques that use LLMs and build conceptual modeling approaches around them. That would result in a shift in approaches to conceptual modeling.

There have been a number of reported examples of ChatGPT being incorrect in its output. For example, some researchers reported that an AI-powered transcription tool used in hospitals invents things no one ever said. To be concrete, suppose we consider the potential impact of LLMs on the development (and possibly, use) of conceptual modeling.

Scenario 1 — Assistant. An assistant can be used to summarize research efficiently and effectively to “automate tedious tasks”. It could generate an initial set of entities, relationships, and attributes from a description of an application domain. It might help in coding; for example, writing Structured Query Language (SQL) queries for a specific application.

Scenario 2 — Adversary. LLMs might be able to make suggestions for conceptual model development. This requires some innovation and creativity in observing an output and then relating it to a given problem.

Text generation from massive training data can produce perspectives other than the one in the mental model of the conceptual modeler. Such exposure might lead a modeler to rethink how something might be modeled differently, leading to alternative solutions.

Scenario 3 — The “Borg”. The modeler will be assimilated; resistance is futile. The modeling challenge will be addressed by machines.

The impact of AI models in general and specifically LLMs on conceptual modeling raises critical questions about symbol grounding and the alignment between human-made and AI-generated models. Human-defined conceptual models are built by experts who structure knowledge based on cognitive understanding and theoretical frameworks, emphasizing semantic clarity and interpretability. These models are essential in areas like ontology engineering and information systems, where human-readable interpretations are crucial. In contrast, data-derived models generated by LLMs rely on statistical patterns from large datasets, producing relationships that may be mathematically sound, but lack human-centered interpretability. This divergence can lead to a “conceptual alignment problem”, where data-derived models may misrepresent domain-specific knowledge due to their reliance on data correlations, rather than cognitive anchors [30]. Addressing this discrepancy requires hybrid approaches that combine human oversight with machine learning to ensure semantic alignment.

Recent advances in representation learning and concept learning provide pathways to improve the alignment and reduce biases in data-derived conceptual models. Techniques such as self-supervised and causal representation learning focus on capturing meaningful, underlying features of data, thereby enhancing interpretability by aligning models with human-understandable concepts [15,31]. Concept learning further supports this alignment by structuring model outputs around recognizable entities and relationships, promoting learning that mirrors human reasoning processes. These methods help ensure that models not only reflect statistical regularities, but also respect the semantic clarity inherent in human-defined models, reducing the risk of opaque or counterintuitive AI applications. Integrating these approaches into data-derived frameworks is a promising step towards creating AI systems that are both powerful and interpretable, resulting in more effective and trustworthy human–AI collaboration in high-stakes fields, such as healthcare and finance.

Humans use mental models and anchor their thinking in a situation. A person has an intuitive perception about how they act and the consequences of their actions [32]. Large AI models are anchored in data. For example, sensor data is anchored in physics. Expert conceptual modelers can craft good conceptual models using, for example, good bottom-up approaches to conceptual model creation (e.g., [33,34]).

Will it be feasible to have approaches that are data-driven or data-identified that could be merged with human conceptual modelers? This could provide great potential, but a conceptual alignment problem could occur between the understanding of machines — which relies on explainable AI (XAI) — and humans [30]. Such alignment is generally needed between the AI tools, such as machine learning tools, and humans.

LLMs cannot abstract in the same way as humans, although they can mimic the things that humans say. Figuring out how to prompt the LLMs then becomes a large task to obtain results that are consistent with a human. Steps involved include testing temperature settings, using retrieval augmented generation (RAG) to optimize the output from LLMs, and breaking problems into small pieces and then sequencing. LLMs may require significant training and sophisticated techniques to obtain good prompts. Good prompts can guide the generation of correct and complete conceptual models, generating a prompt engineering discipline that is going to be essential for the correct use of LLMs in conceptual modeling practice, as discussed below.

6. Potential use of LLMs in conceptual modeling

One possible use of LLMs to support conceptual modeling is the identification of the data model behind one or more datasets. This may be particularly useful within contexts that have high complexity of the data sources, could provide an overview of the domain, and highlight the corresponding entities and relationships. We use a concrete example to illustrate this idea, with the intention

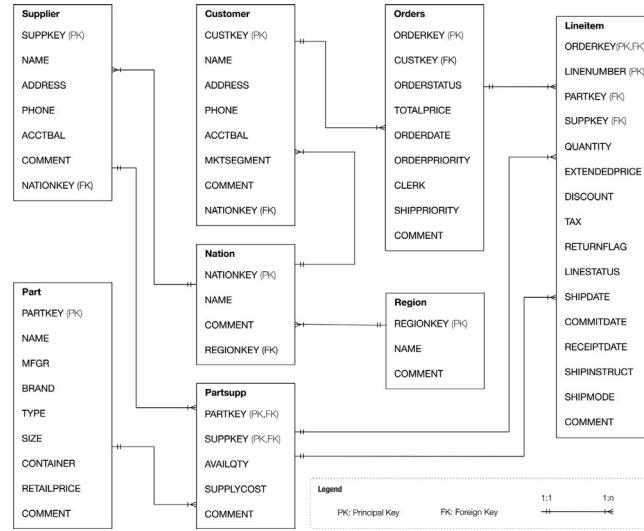


Fig. 1. TPC-H data model.

Source: Adapted from: [35].

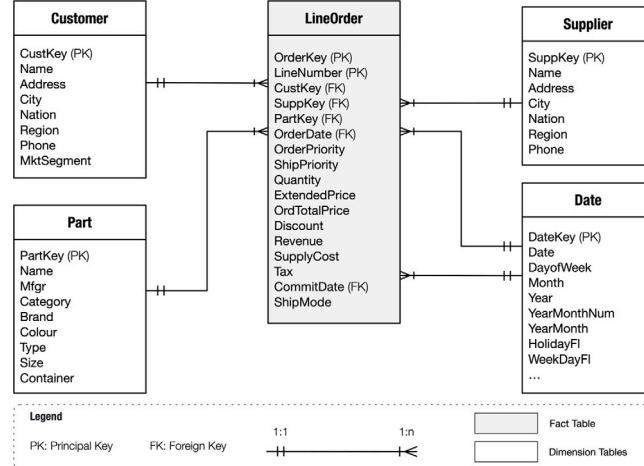


Fig. 2. SSB data model.

Source: Adapted from: [36].

of showing how an intelligent LLM-guided iterative interaction can make it feasible to achieve effective and efficient conceptual modeling assistance.

Three tasks and two datasets are considered. The first task intends to highlight the inference of the conceptual model of a normalized dataset, identifying the entities and their relationships. In this task, the first dataset includes data from the TPC-H Benchmark [35], a decision support benchmark that includes a set of business-oriented tables, such as sales, customers, and suppliers, modeling a business that manages, sells, and distributes products. The design pattern associated with the TPC-H dataset is a third normal form representation, with 8 tables, *Customer*, *Orders*, *Lineitem*, *Part*, *Supplier*, *Partsupp*, *Nation*, and *Region*, as shown in Fig. 1.

The second task shows the inference of a conceptual data model of a de-normalized dataset, highlighting a different design pattern, namely the one followed in the dimensional design of a data warehousing system. For this, the second dataset includes data from the Star Schema Benchmark (SSB) [36], which is based on the TPC-H dataset but adopts a different design pattern, as mentioned, a star schema for a data warehouse. The SSB dataset includes one fact table, *Lineorder*, and four dimension tables, *Customer*, *Supplier*, *Part*, and *Date*, as shown in Fig. 2.

The third task assumes the proposal of a conceptual model for a data warehouse, but taking as input the TPC-H dataset and evolving its design pattern to a model based in fact tables and dimension tables. In this case, the inference task must be guided by the modeling concepts and principles.

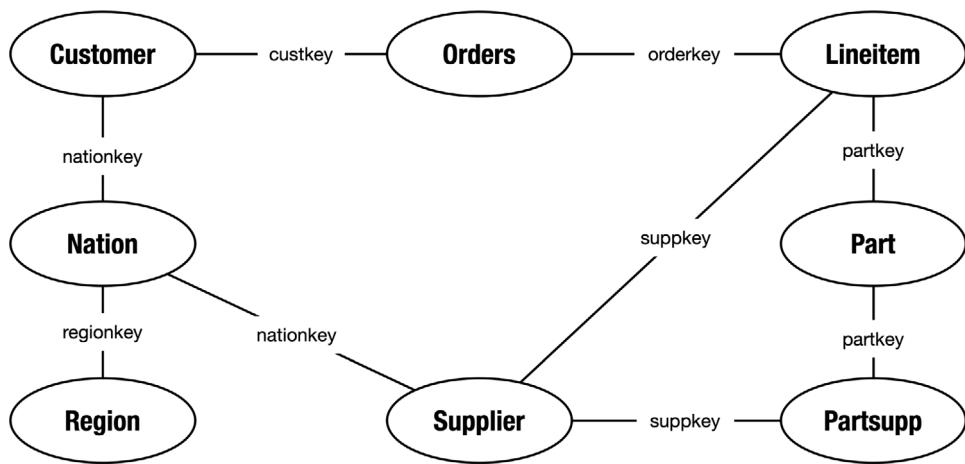


Fig. 3. TPC-H data.
Source: Model suggested by ChatGPT.

The selection of these two datasets is justified by the need to ensure the replicability of the results as the datasets are openly available, although the particular prompts and LLM chosen will influence the results obtained. Additionally, as two different design patterns are used, the findings can address and discuss these differences.

For both tasks and datasets, ChatGPT 4o mini (ChatGPT) was used for the inference of the corresponding conceptual models, guiding this process with some of the prompts next presented.

6.1. Task 1: Inference of the TPC-H model

Starting the inference process, data from the TPC-H dataset was made available to ChatGPT, along with the following prompt:

Consider the following dataset. The file contains different tables, one in each tab of the Excel file. Can you infer the corresponding data model, identifying the entities and the relationships among them?

The result obtained includes the eight entities and most of the relationships present in the original model (Fig. 1). In this process, ChatGPT acknowledges that ... *this structure suggests a classic transactional data model, often found in order management systems, with well-defined relational links across key entities such as orders, customers, suppliers, and parts.* However, as seen in Fig. 3, **Lineitem** is directly related to the entities **Supplier** and **Part**, and not to the **Partsupp** entity, introducing a cycle and with the corresponding redundant relationships.

Attempting to improve the obtained result, in the second iteration, ChatGPT was asked about this cycle:

Your proposal includes a loop between **Lineitem**, **Supplier**, **Part** and **Partsupp**. Could this loop be avoided? Is it advantageous, or not, to avoid this loop?

The answer pointed out the advantages of avoiding cycles, such as *simpler querying and maintenance, and reduced redundancy*. However, the two proposed models, as seen in Fig. 4, introduce additional problems.

- Approach 1 (Fig. 4(a)). Removing the relationship from **Lineitem** to **Supplier** will prevent identifying the **Supplier**, as a **Part** may have different suppliers.
- Approach 2 (Fig. 4(b)). Removing the **Partsupp** table excludes from the model useful data such as the supply cost and the available quantity, which ChatGPT suggested to include into **Supplier** or **Part**. However, both attributes depend on the two entities.

In the last iteration, the prompt explicitly queried the linking of **Lineitem** to **Partsupp**, and this to **Part** and to **Supplier**:

Why not having **Lineitem** linked to **Partsupp** (with the **partkey** and **suppkey**), and then **Partsupp** is linked to **Supplier** and also to **Part**?

The result, as expected, was the identification of the advantages of this approach, such as *reduces redundancy, simplifies data integrity, optimizes query performance and represents the logical data flow*.

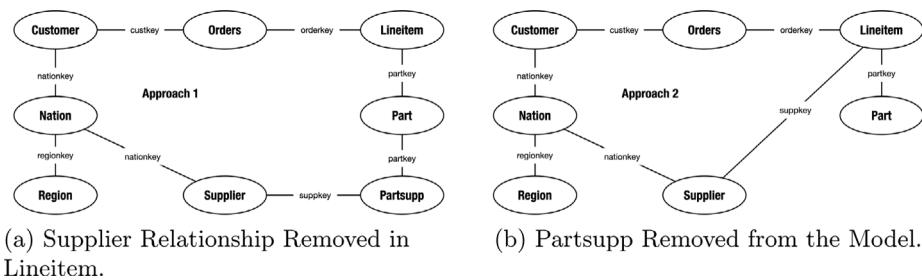


Fig. 4. ChatGPT proposals for the TPC-H data model (avoiding cycles).

6.2. Task 2: Inference of the SSB model

For this dataset, the interaction with ChatGPT started with the same prompt as the previous dataset, asking for the entities and relationships. ChatGPT associated one entity to each table in the dataset, which corresponds to the original model (Fig. 2), and highlighted that *this model outlines a schema that reflects the relationships of customers, suppliers, parts, and order details tied to specific dates*, without mentioning any specific design pattern, as happened in the previous dataset.

ChatGPT answer:

...

Lineorder

Attributes: orderkey (Primary Key), linenum, custkey (Foreign Key to Customer), partkey (Foreign Key to Part), suppkey (Foreign Key to Supplier), orderdate (Foreign Key to Date), orderpriority, shippriority, quantity, extendedprice, ordtotalprice, discount, revenue, supplycost, tax, commitdate, shipmode.

...

Customer - Lineorder: Each customer can have multiple line orders, establishing a one-to-many relationship between Customer and Lineorder.

Supplier - Lineorder: Each supplier can supply parts for multiple line orders, establishing a one-to-many relationship between Supplier and Lineorder.

Part - Lineorder: Each part can be ordered in multiple line orders, creating a one-to-many relationship between Part and Lineorder.

Date - Lineorder: Each order date in Lineorder corresponds to a specific date in the Date table, representing a one-to-many relationship between Date and Lineorder.

Analyzing the ChatGPT proposals for the relationships and keys, it is relevant to highlight that:

- The *OrderKey* attribute was suggested as the primary key of the *LineOrder* table, although this fact table includes several items by order constraining the use of this attribute as the primary key of this table.
- The *Date* dimension presents two relationships with the *LineOrder* table, one for the *OrderDate* and the other for the *CommitDate*, although only the first one was identified.

Both situations suggest that the design pattern of a star schema was not identified, as *OrderKey* is associated with the concept of Degenerate Dimensions and the different relationships to *Date* represent the concept of Different Roles for Dimensions [37].

The second iteration for the SSB dataset explicitly mentioned the design pattern of a star schema, asking ChatGPT to review the suggested primary keys and relationships. The result was the same.² Therefore, the third iteration required the proposal of a constellation schema, assuming two fact tables at different levels of detail, one for *Order* and the other for *LineOrder*, assuming that one order can have one or more line orders. The proposal should include the attributes of each fact table and the associated dimensions (and their attributes):

Propose a constellation schema for this dataset, assuming two fact tables at different levels of detail, one for Order and the other for Lineorder, assuming that one order can have one or more line orders. The proposal should include the attributes of each fact table and the associated dimensions (and their attributes).

² It is important to highlight the size of the dataset. As the dataset can be generated at different scale factors or even as a small sample of the data, the keys could be influenced by the available data.

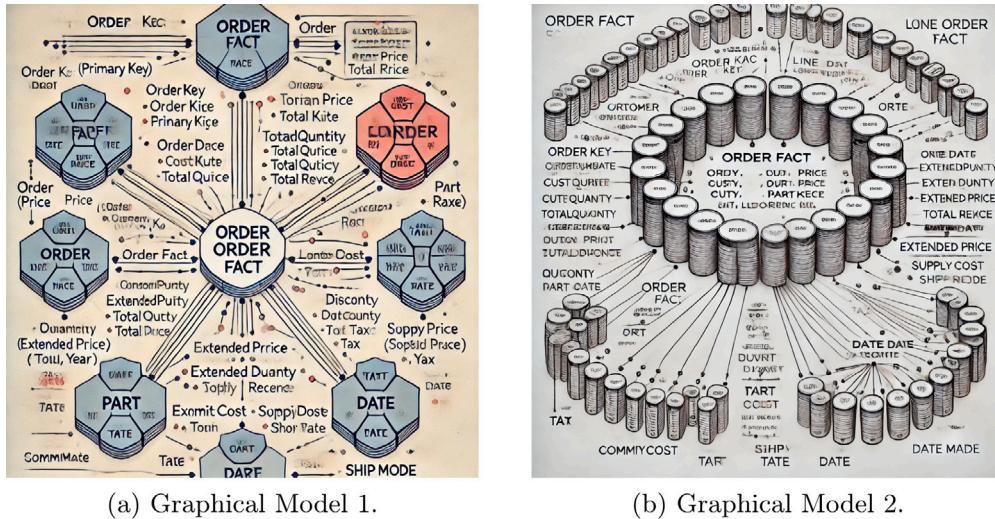


Fig. 5. ChatGPT models for the constellation schema.
Source: Images by ChatGPT.

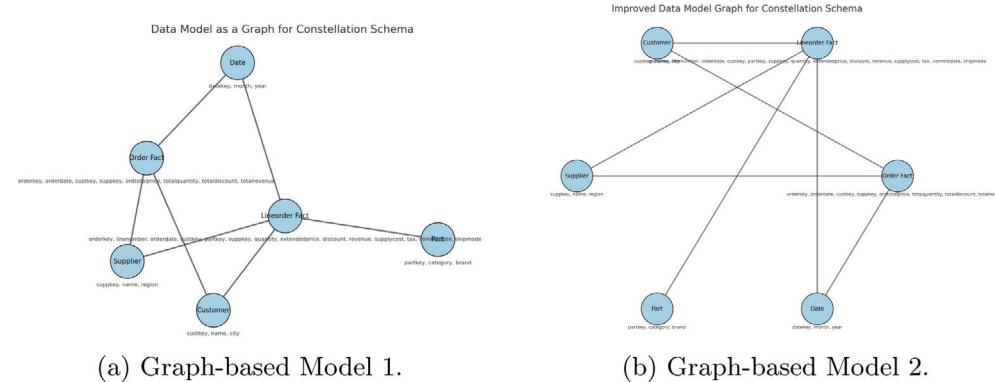


Fig. 6. ChatGPT models in a graph-based format.
Source: Images by ChatGPT.

This prompt guided the identification of the two fact tables and a data model that extends the previous one. The graphical representation of the proposal was a challenge as several answers rendered images not aligned with an entity-relationship diagram (Fig. 5), as requested. Prompting for a graph-based representation helped in the visualization of the model (Fig. 6), although the second visualization, which included a request to avoid overlapping between edges or nodes (Fig. 6(b)), perversely includes more crossing edges than the previous one (Fig. 6(a)).

6.3. Task 3: From the TPC-H to the SSB model

As the SSB data model [36] was proposed from the TPC-H [35], ChatGPT was prompted to propose a data warehouse model from the TPC-H dataset, first without any constraints on the number of fact tables.³ The result presents several problems (Fig. 7):

- Instead of an *Order* fact table, a *Sales* table was proposed;
- *Nation* and *Region* were proposed as two autonomous dimensions, without any denormalization (a common practice to increase performance in data warehousing systems);
- In a star schema, for the design pattern selected by ChatGPT, the dimension tables must establish relationships with the fact table, but not among them. (Relationships between dimension tables are possible in a snowflake schema [37]).
- An *Order* dimension table was proposed when dimensions are not meant to store dynamic data;

³ The proposal could go for a constellation schema with three fact tables, as documented in [38].

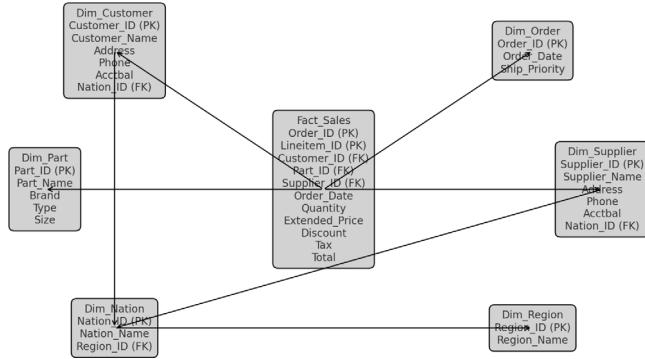


Fig. 7. ChatGPT star schema proposal for the TPC-H dataset.

Source: Image by ChatGPT.

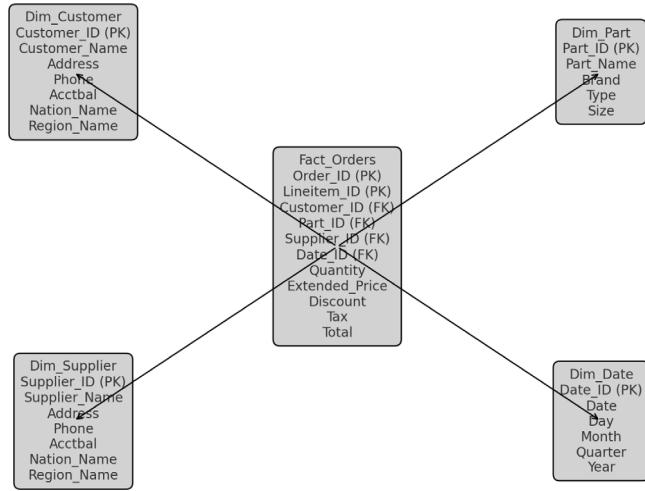


Fig. 8. ChatGPT corrected proposal for the TPC-H dataset.

Source: Image by ChatGPT.

- No *Date* dimension was included in the data model, a core dimension in a typical data warehousing system.

After prompting all these issues, the star schema was corrected, removing relations between dimensions, denormalizing dimensions, adding the *Date* dimension, and clarifying the fact table, as can be seen in Fig. 8.

These examples (presented as tasks above) and the discussion present an approach for achieving a successful use of LLMs for conceptual modeling practice by using LLMs as “intelligent” assistants. In this scenario, the correct generation of prompts becomes a crucial task for ensuring a promising, modern prompt engineering-based conceptual modeling practice.

7. Future of LLMs and conceptual modeling

As the adoption of LLMs increases, there will be continued interest in applying them to conceptual modeling, assessing the results, and incorporating them into conceptual modeling research and practice. Various types of issues arise, with some of the most notable summarized below, focusing on: adoption risks and challenges, human-level intelligence, educational impacts, adoption in practice, AI model research, incorporating into educational material, potential risks, and exploration of LLMs use as a new, relevant tool. Our aim is to report on a significant set of ideas, each of which provides interesting insights around the central topic of this work: how LLM can affect CM practice.

7.1. Adoption risks and challenges

Any new and evolving technology has adoption risks. For LLMs, there might be an over-reliance on them, leading to de-skilling in professional fields or the loss of organizational knowledge [39,40]. The challenges related to the adoption of LLMs in conceptual modeling are technical, methodological, and ethical in nature. LLMs face semantic and syntactic limitations because they are trained

Impact on educational practice in CM

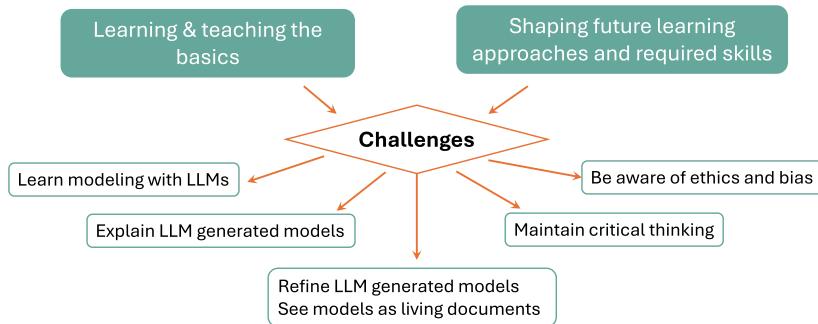


Fig. 9. Impact on education.

in natural language processing and may not fully be able to deal with the formal logic required for conceptual modeling. Their processing of domain-specific rules, ontologies, and business processes may also be limited and require extensive human oversight. For designers and conceptual modelers, the cognitive load required to verify and correct models produced by LLMs might even be heavier than creating models from scratch.

7.2. Human-level intelligence

The current state of the art is that AI, generally, cannot reach human levels. Current LLMs process mainly text. However, conceptual modeling is focused on structuring, abstracting, assessing relevance, and performing other human-focused decision-driven activities, making it difficult to build a repository of effective, and useful, models. Furthermore, models generally need to be maintained for consistency. This is challenging in dynamic environments and when a system's technology is evolving. Improved communication may be required to build glossaries and taxonomies.

To support conceptual modeling, advanced LLMs or machine intelligence need to exhibit several characteristics: learn world models from different inputs (descriptions, data, conceptual models, graphics, sensors, videos); have persistent memory; be able to plan actions to satisfy goals; simulate understanding and reasoning in the sense of creating new solutions to unseen problems; and be controllable and safe (by design).

Specifically, for machine learning of conceptual models, LLMs will need to learn from both graphical models and text. This results in a data challenge because large amounts of high-quality models will be needed for the learning. It is necessary to both understand what a “good” model is and then write rules and heuristics for how to recognize such a model. For human learning of conceptual models, humans will need to focus on prompt engineering and how to properly carry it out.

7.3. Educational impact

It is well-known that conceptual modeling activities can be challenging for current and future designers and developers of information systems. Thus, there could be an impact on conceptual modeling education as shown in Fig. 9.

In any educational endeavor, it is necessary to teach students (here, future conceptual modelers) the basic concepts of a field, while shaping future learning approaches as the required skills evolve. Teaching students how to model using LLMs will require an understanding of both conceptual modeling practices and the capabilities of LLMs, which are continually evolving. Explanations associated with the generated models will be necessary for modelers to be able to judge what refinements to a model might be needed. As applications change over time, so must models as they become living documents that represent a real-world application. There are also embedded ethical and bias concerns because LLMs are fundamentally based on existing work.

Maintaining critical thinking is crucial. It is undesirable for the adoption of LLMs to reach the Dunning–Kruger Effect of Mount Stupid [41]. This can happen when students might become careless in their work, with over-reliance on a tool such as ChatGPT and fail to verify the results. Students might understand they are not an expert in conceptual modeling, but based on the results from LLMs, feel confident that they can conduct modeling tasks efficiently. Such over-reliance can lead to a superficial understanding, and de-skilling, as shown in Fig. 10.

7.4. Adoption approach

When using LLMs for conceptual modeling activities, whether by experienced modelers or less experienced ones, such as students, there is a need to continuously check and validate the findings. As discussed above, LLMs can facilitate data processing and the inference of a conceptual model, although design patterns or other model features may not be captured. Furthermore, because multiple prompts may be required to request changes in the findings, there is a risk of losing concepts, attributes, or other model

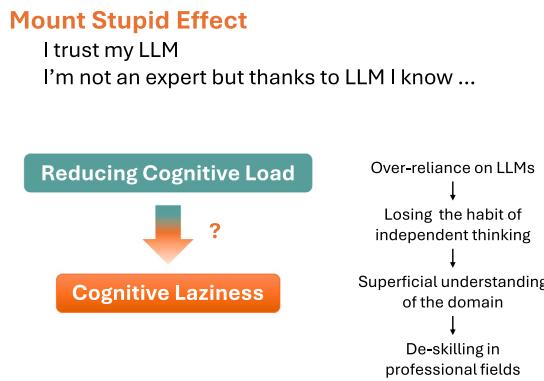


Fig. 10. Impact on conceptual modeling and conceptual modelers.

characteristics as LLMs begin to suggest new ones. This departure from the original context may be beneficial for students to observe alternative possibilities and reason about them, but validation and conformity with modeling design principles must be rigorously evaluated.

7.5. AI models

Research on the future of AI models generally, and specifically on LLMs in conceptual modeling, holds significant potential. Two of the main challenges are interoperability and mental model representation. Interoperability, often a complex problem in information systems, arises from the need for different systems, data formats, and conceptual models to work seamlessly together. AI models can support interoperability through their ability to learn and recognize patterns across varied data sources, and to establish common representations. Through fine-tuning and representation learning, AI models can serve as intermediaries that map and align different conceptual models, facilitating interoperability in multi-system environments [42]. Representation learning involves extracting patterns from data to create representations that are understandable and useful for given tasks. LLMs can, thus, definitively provide a viable solution to achieve those objectives.

Similarly, AI models can contribute to problems related to mental model representations by creating models that reflect human-like understanding, thus bridging the gap between machine representations and human cognitive structures. By creating and enhancing an alignment between AI-driven models and human mental models, AI models can make AI applications more intuitive and accessible for end-users, supporting clear, effective human–AI interactions [43].

The advancement of AI models and their application in conceptual modeling research will require the adoption of novel research methods. Traditional methodologies may not be sufficient to explore the nuanced capabilities of AI models in fields requiring both data-driven and semantically grounded models. New experimental approaches, particularly those combining qualitative analysis with quantitative validation, will be essential for understanding how AI models can best contribute to solving interoperability and representation challenges. Additionally, interdisciplinary research efforts, blending insights from cognitive science, AI, and information systems, will be crucial in refining methods that capture the depth of human conceptual understanding while leveraging the scalability and efficiency of AI models [44]. This will support the technical development of AI models in conceptual modeling and address ethical and practical considerations, ensuring that the deployment of AI models aligns with human-centric values and effective system design.

Representation problems always exist in AI systems. As humans, we have mental models that we use when representing problems. Humans have many sources from which to capture, for example, personal data, such as digital watches to capture health data, which a human may interpret to augment or change their behavior. Now, it might be possible to make or represent mental models that can make decisions for people. The challenge is going to be how to enable the machine to create concepts from raw data. This is a different conceptualization strategy, which explains the need of novel research methods. Current “black box” models cannot explore these in depth. It will require much more research into natural science research methods to investigate this possibility. Structured approaches (e.g., current manipulation of databases) will not suffice.

7.6. Incorporation into educational material

An adequate use of LLMs should reinforce the construction process that is behind learning. Constructivist Learning Theory (CLT) posits that knowledge is constructed (built on other knowledge), so people learn how to learn as they learn, constructing meaning and systems of meaning as they progress. Additionally, it is well-established that learning is an active process, a social activity, contextual, personal, exists in the mind, and has motivation as a key aspect.⁴ An implication of CLT is that understanding and recall improve when a student constructs their own answer through active learning. How can LLM use affect the learning process?

⁴ See <https://www.wgu.edu/blog/what-constructivism2005.html>.

The risk of AI for a learner is that the learner will short-circuit the construction process and arrive at a solution without undergoing meaningful learning. Sparrow et al. [45] carried out several studies, the results of which suggested that, when faced with difficult questions, people expect that they will have access to computers so they have lower rates of recall of the information, but greater memory of where the information is located. The analogy is that if AI becomes a crutch, and we do not understand it deeply enough, we will only be able to produce the desired results with the tool, and only when we have that tool. There are numerous situations where the tool is not accessible or at least is inconvenient to attempt to use (e.g., during a job interview).

An LLM can be used to help with tasks a student seeks to perform, as well as to help them study a subject. If students only focus on the first, and not on the second, then they risk bypassing true learning. With proper prompting guidance, AI can act as a personal tutor in some situations.

The following are suggestions for using AI as a tutor (based on a case study by one of the authors). The first is to pre-load the LLM with a prompt explaining that a student is a beginner, learning a particular subject. The system also needs to be instructed to know that it is expected to act as a tutor and to explain things simply, and step-by-step. In this way, the system can be instructed to tell the student what they want to learn about, not just solutions.

Students should first attempt to solve the problem by challenging themselves with it and studying it for some time. When a student encounters a problem, rather than asking the LLM to resolve it, they should ask the system to explain any significant error without fixing it. Rather than copying and pasting code into the LLM, a student can paste the error message and ask for an explanation. Another strategy is to ask the LLM to explain working code line-by-line to determine whether it provides insights that have not yet been recognized. The LLM can indicate whether there are other ways to solve a given problem and to critique a student's proposed solution. Other potential approaches involve asking the LLM to explain whatever point a student is trying to understand, with follow-up questions and having a conversation, much like a tutor. Alternatively, a student could explain a concept in his or her own words and ask the LLM to correct anything that is incorrect. Also, if given appropriate background information, the LLM could provide practice problems with, for example, a specified difficulty level.

Of course, there are other topics and means for a student to engage with an LLM system. For conceptual modeling, a system might be able to help with identifying concepts that might appear in an application domain, if given enough description of the application. Or an LLM might be asked to explain how, in general, constructs should be applied and then do so in a given situation.

7.7. Learning risks

Three well-known problems of learning are remembering, understanding, and transferring. Learning how to find a correct solution requires combining those three aspects in a way that allows them to gain a deep knowledge of the problem, its different dimensions, and the different alternatives that lead to a valid solution. In this way, Productive Failure is a useful theory because it suggests that we learn better when we first wrestle with how to solve a problem and then either receive instructions on how to solve it or receive a solution [46]. Thus, risk of using LLMs for a student is that they will avoid the failure process, bypassing opportunities for real learning in favor of simply finding a solution.

7.8. LLMs as an important tool

There remain fundamental, basic things we need to understand for progressing work on conceptual modeling, which includes identifying types and patterns. For both abstraction and analytical reasoning, patterns are a foundational notion of dispositions, properties, and other important concepts. They correspond to differences between thinking fast and thinking slow that can guide modelers and developers [47]. Thinking fast involves efforts such as machine learning. Thinking slow recognizes the importance of thoughtful human consideration and cognition. Speed is not the only value to pursue. Rather, it is important to build other models that capture or represent an enhanced, or augmented, understanding of both an application domain and the modeling process.

LLMs can, in some sense, be considered yet another tool. They are currently “shiny and new”, and are becoming very popular in many areas of application. However, the present state of the field is not just a passing popular fad—LLMs are highly capable and improving, and, no doubt, here to stay. There is still much to learn about the best ways to use them, with likely surprises as we continue to learn what they do well. It would be wise to be clear about our expectations of their capabilities. Although it is unlikely that they will replace human judgment, they will most likely be able to help researchers and practitioners in conceptual modeling to be increasingly productive as the tools evolve and along with our experiences in using them.

As a caution, it is important not to repeat the hype of the 1970's around AI where there were far-fetched claims such as “personal humanoid robot butlers could be just around the corner”. At that point, both the hardware and software were far behind what would be needed for that envisioned future. Even with many generations of Moore's Law-style doubling of capability, we need intellectual humility to recognize that there is still much work to be done to lead toward futuristic visions of what could be possible with each new generation of AI. LLMs can be viewed just as another tool in our tool belt that we have yet to fully master. LLMs do not “think” or “reason” in the conventional sense, and we must not expect them to do so.

Table 2
Potential research topics for conceptual modeling with LLMs.

Research topic	Description
Prompt engineering	Provide a way to make conceptual modeling programming feasible
Task allocation	Delegate abstraction and design tasks to LLMs and assess the results for feasibility
Modeling practices	Assess the impact on conceptual modeling practice in software engineering
Ontologies	Address the role of ontologies in LLMs
Risks	Awareness of over reliance and similar outputs
Education	Guide responsible use; evaluate LLMs as tutoring agents

8. Potential research topics

This paper reported on the shared visions of a set of researchers in conceptual modeling who wished to raise awareness of the potential of LLMs in conceptual modeling research, practice, and education. A wide range of issues were raised from the role of LLMs as another tool to support conceptual modeling to actively using conceptual modeling in the generation of the relevant constructs needed to build a conceptual model and whether, eventually, this will lead to the replacement of human input. While reporting these shared visions, future research needs to be considered.

Based on current technology trajectories, it is possible that LLMs could shift the role of conceptual modeling practice in software engineering and the way models are involved in engineering processes. LLMs can generate or process diagrammatic content of various model types such as UML class diagrams, BPMN diagrams, and ER models, with some impressive results suggesting that they are aware of certain abstraction patterns present in such structured visualizations. This seems to indicate that the conceptual modeling community can already delegate some abstraction and design tasks to LLM-based products.

Another significant challenge is to design conceptual model-based prompt engineering processes that will more effectively assist modelers in their design tasks. And there may even be an intriguing opportunity to use LLMs to assist in making conceptual model programming feasible [48]. LLMs are very good at generating certain kinds of source code, so perhaps an LLM could be trained to help with tasks from modeling all the way through to system implementation and maintenance. Thus, perhaps LLMs can significantly impact the role of conceptual modeling practice in software engineering. It may be that the CM community can write the right prompts, delegating design and programming tasks to LLM-based products, and then assess them for their feasibility.

Many potential topics could be investigated for conceptual modeling and the role of LLMs. Significant research topics are highlighted, including work on prompt engineering, task allocation, modeling practices, ontologies, risks, and education, as summarized in Table 2.

9. Conclusion

Conceptual modeling has long been recognized as challenging because humans still have difficulty performing modeling tasks well, despite years of research on conceptual modeling tools, techniques, and approaches. This paper has reported on a panel discussion concerning conceptual modeling and large language models. Specifically, the panel commented on the state of the art, the potential impacts and risks of large language models, and how the conceptual modeling community might embrace them. Future research should focus on a wide range of topics, including modeling impact, task support and applications, risks, education concerns, and prompt engineering.

CRediT authorship contribution statement

Veda C. Storey: Writing – review & editing, Writing – original draft, Supervision, Project administration, Investigation, Conceptualization. **Oscar Pastor:** Writing – review & editing, Writing – original draft, Supervision, Conceptualization. **Giancarlo Guizzardi:** Writing – original draft, Conceptualization. **Stephen W. Liddle:** Writing – review & editing, Writing – original draft, Investigation, Conceptualization. **Wolfgang Maaß:** Writing – original draft, Investigation, Conceptualization. **Jeffrey Parsons:** Investigation, Conceptualization. **Jolita Ralyté:** Writing – review & editing, Writing – original draft, Visualization, Investigation, Conceptualization. **Maribel Yasmina Santos:** Writing – review & editing, Writing – original draft, Formal analysis, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This paper is based on a panel entitled “Conceptual Modeling and Large Language Models” that was held in connection with the 43rd International Conference on Conceptual Modeling, Pittsburg, Pennsylvania, USA, on October 29, 2024. The authors are grateful to the organizing committee for the opportunity to present this panel. We also wish to thank the editor-in-chief, Dr. Carson Woo, for the opportunity to submit this paper to *Data & Knowledge Engineering*, as well as the associate editor, Carlos Ordonex, for his work in handling this paper. Jeffrey Parsons was supported by a research grant from the Natural Sciences and Engineering Research Council of Canada (NSERC) [RGPIN-2020-04916]. Maribel Yasmina Santos was supported by FCT – Fundação para a Ciência e Tecnologia within the R&D Unit Project Scope UID/00319/Centro ALGORITMI (ALGORITMI/UM).

Data availability

No data was used for the research described in the article.

References

- [1] P. Chen, The entity-relationship model—Toward a unified view of data, *ACM Trans. Database Syst.* 1 (1) (1976) 9–36, <http://dx.doi.org/10.1145/320434.320440>.
- [2] J. Akoka, I. Comyn-Wattiau, N. Prat, V.C. Storey, Unraveling the foundations and the evolution of conceptual modeling—Intellectual structure, current themes, and trajectories, *Data Knowl. Eng.* 154 (2024) 102351.
- [3] D. Bork, S.J. Ali, B. Roelens, Conceptual modeling and artificial intelligence: A systematic mapping study, 2023, arXiv preprint [arXiv:2303.06758](https://arxiv.org/abs/2303.06758).
- [4] H. Jaakkola, B. Thalheim, Sixty years—and more—of data modelling, *Inf. Model. Knowl. Bases XXXII* 333 (2021) 56.
- [5] F. Härrer, H.-G. Fill, Past trends and future prospects in conceptual modeling—a bibliometric analysis, in: International Conference on Conceptual Modeling, Springer, 2020, pp. 34–47.
- [6] H.C. Mayr, B. Thalheim, The triptych of conceptual modeling: A framework for a better understanding of conceptual modeling, *Softw. Syst. Model.* 20 (1) (2021) 7–24.
- [7] V.C. Storey, I.-Y. Song, Big data technologies and management: What conceptual modeling can do, *Data Knowl. Eng.* 108 (2017) 50–67.
- [8] V.C. Storey, J. Trujillo, S.W. Liddle, Research on conceptual modeling: Themes, topics, and introduction to the special issue, *Data Knowl. Eng.* 98 (2015) 1–7.
- [9] B. Thalheim, The science and art of conceptual modelling, *Trans. Large-Scale Data- Knowledge-Centered Syst.* 6 (2012) 76–105.
- [10] Y. Chang, X. Wang, J. Wang, Y. Wu, L. Yang, K. Zhu, H. Chen, X. Yi, C. Wang, Y. Wang, et al., A survey on evaluation of large language models, *ACM Trans. Intell. Syst. Technol.* 15 (3) (2024) 1–45.
- [11] W.S. Saba, Towards explainable and language-agnostic LLMs: Symbolic reverse engineering of language at scale, 2023, arXiv preprint [arXiv:2306.00017](https://arxiv.org/abs/2306.00017).
- [12] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, *Adv. Neural Inf. Process. Syst.* 30 (2017).
- [13] J. Sohl-Dickstein, E.A. Weiss, N. Maheswaranathan, S. Ganguli, Deep unsupervised learning using nonequilibrium thermodynamics, 2015, arXiv preprint [arXiv:1503.03585](https://arxiv.org/abs/1503.03585).
- [14] Y. LeCun, Y. Bengio, G. Hinton, A path towards autonomous machine intelligence, 2021, arXiv preprint [arXiv:2103.10078](https://arxiv.org/abs/2103.10078).
- [15] B. Schölkopf, F. Locatello, S. Bauer, N.R. Ke, N. Kalchbrenner, A. Goyal, Y. Bengio, Toward causal representation learning, *Proc. IEEE* 109 (5) (2021) 612–634.
- [16] A. Gu, T. Dao, Mamba: Linear-time sequence modeling with selective state spaces, 2023, arXiv preprint [arXiv:2312.00752](https://arxiv.org/abs/2312.00752).
- [17] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, S. Lloyd, Quantum machine learning, *Nature* 549 (7671) (2017) 195–202.
- [18] W. Maass, V.C. Storey, Pairing conceptual modeling with machine learning, *Data Knowl. Eng.* 134 (2021) 101909.
- [19] K. Chen, Y. Yang, B. Chen, J.A. Hernández López, G. Mussbacher, D. Varró, Automated domain modeling with large language models: A comparative study, in: 2023 ACM/IEEE 26th International Conference on Model Driven Engineering Languages and Systems, MODELS, 2023, pp. 162–172, <http://dx.doi.org/10.1109/MODELS58315.2023.00037>.
- [20] W. Zhao, Q. Chen, J. You, LlmRe: A zero-shot entity relation extraction method based on the large language model, in: Proceedings of the 2023 7th International Conference on Electronic Information Technology and Computer Engineering, 2023, <http://dx.doi.org/10.1145/3650400.3650478>.
- [21] S. Arulmohan, M.-J. Meurs, S. Mosser, Extracting domain models from textual requirements in the era of large language models, in: 2023 ACM/IEEE International Conference on Model Driven Engineering Languages and Systems Companion, MODELS-C, 2023, pp. 580–587, <http://dx.doi.org/10.1109/MODELS-C59198.2023.00096>.
- [22] H.-G. Fill, P. Fettke, J. Köpke, Conceptual modeling and large language models: Impressions from first experiments with ChatGPT, *Enterp. Model. Inf. Syst. Archit.* 18 (3) (2023) 1–15, <http://dx.doi.org/10.18417/EMISA.18.3>.
- [23] J. Cámará, J. Troya, L. Burgueño, A. Vallecillo, On the assessment of generative AI in modeling tasks: An experience report with ChatGPT and UML, *Softw. Syst. Model.* 22 (3) (2023) 781–793, <http://dx.doi.org/10.1007/s10270-023-01105-5>.
- [24] J. Neuberger, L. Ackermann, H. Van Der Aa, S. Jablonski, A universal prompting strategy for extracting process model information from natural language text using large language models, in: W. Maass, H. Han, H. Yasar, N. Multari (Eds.), Conceptual Modeling, ER2024 Proceedings, in: Lecture Notes in Computer Science, vol. 15238, Springer Nature Switzerland, 2024, pp. 38–55, http://dx.doi.org/10.1007/978-3-031-75872-0_3.
- [25] J. Silva, Q. Ma, J. Cabot, P. Kelsen, H.A. Proper, Application of the tree-of-thoughts framework to LLM-enabled domain modeling, in: W. Maass, H. Han, H. Yasar, N. Multari (Eds.), Conceptual Modeling, ER2024 Proceedings, in: Lecture Notes in Computer Science, vol. 15238, Springer Nature Switzerland, 2024, pp. 94–111, http://dx.doi.org/10.1007/978-3-031-75872-0_6.
- [26] D. Prokop, Š. Stenclák, P. Škoda, J. Klímek, M. Nečaský, Enhancing domain modeling with pre-trained large language models: An automated assistant for domain modelers, in: W. Maass, H. Han, H. Yasar, N. Multari (Eds.), Conceptual Modeling, ER2024 Proceedings, in: Lecture Notes in Computer Science, vol. 15238, Springer Nature Switzerland, 2024, pp. 235–253, http://dx.doi.org/10.1007/978-3-031-75872-0_13.
- [27] S. Rizzi, Using ChatGPT to refine draft conceptual schemata in supply-driven design of multidimensional cubes, in: 27th International Workshop on Design, Optimization, Languages and Analytical Processing of Big Data, DOLAP'2025, Co-Located with EDBT/ICDT 2025, 2025.
- [28] W. Maass, A. Agrawal, A. Ciani, S. Danz, A. Delgadillo, P. Ganser, P. Kienast, M. Kulig, V. König, N. Rodellas-Gràcia, et al., QUASIM: Quantum computing enhanced service ecosystem for simulation in manufacturing, KI- Künstliche Intell. (2024) 1–10.
- [29] R. Buchmann, J. Eder, H.-G. Fill, U. Frank, D. Karagiannis, E. Laurenti, J. Mylopoulos, D. Plexousakis, M.Y. Santos, Large language models: Expectations for semantics-driven systems engineering, *Data Knowl. Eng.* (2024) 102324.
- [30] W. Maass, A. Castellanos, M.C. Tremblay, R. Lukyanenko, V.C. Storey, J.S. Almeida, Conceptual alignment method, in: AMCIS, 2023.
- [31] Y. Bengio, A. Courville, P. Vincent, Representation learning: A review and new perspectives, *IEEE Trans. Pattern Anal. Mach. Intell.* 35 (8) (2013) 1798–1828.
- [32] W. Maass, V.C. Storey, R. Lukyanenko, From mental models to machine learning models via conceptual models, in: International Conference on Business Process Modeling, Development and Support, Springer, 2021, pp. 293–300.
- [33] T.J. Teorey, D. Yang, J.P. Fry, A logical design methodology for relational databases using the extended entity-relationship model, *ACM Comput. Surv.* 18 (2) (1986) 197–222.
- [34] V.C. Storey, Relational database design based on the entity-relationship model, *Data Knowl. Eng.* 7 (1) (1991) 47–83, [http://dx.doi.org/10.1016/0169-023X\(91\)90033-T](http://dx.doi.org/10.1016/0169-023X(91)90033-T).
- [35] Transaction Processing Performance Council, TPC-h specification (decision support) standard specification, revision 2.17.2, 2017, URL http://www.tpc.org/tpc_documents_current_versions/pdf/tpc-h.v2.17.2.pdf.
- [36] P. O’Neil, B. O’Neil, X. Chen, The Star Schema Benchmark (SSB), 2009, URL <https://www.cs.umb.edu/~poneil/StarSchemaB.PDF>.
- [37] R. Kimball, M. Ross, *The Data Warehouse Toolkit: The definitive Guide to Dimensional Modeling*, third. ed., John Wiley & Sons, Inc., 2013.

- [38] O. Romero, A. Abelló, Automatic validation of requirements to support multidimensional design, *Data Knowl. Eng.* 69 (9) (2010) 917–942.
- [39] V.C. Storey, Knowledge management in a world of generative AI: Impact and implications, *ACM Trans. Manag. Inf. Syst.* (2025).
- [40] V.C. Storey, W.T. Yue, J.L. Zhao, R. Lukyanenko, Generative artificial intelligence: Evolving technology, growing societal impact, and opportunities for information systems research, *Inf. Syst. Front.* (2024).
- [41] D. Dunning, The Dunning-Kruger effect: On being ignorant of one's own ignorance, in: *Advances in Experimental Social Psychology*, Vol. 44, Elsevier, 2011, pp. 247–296.
- [42] J. Wang, et al., Ontology-based interoperability for data integration, in: *International Conference on Data Engineering*, IEEE, 2019, pp. 231–240.
- [43] A. Tversky, D. Kahneman, Judgment under uncertainty: Heuristics and biases, *Science* 185 (4157) (1974) 1124–1131.
- [44] E.M. Bender, T. Gebru, A. McMillan-Major, S. Shmitchell, On the dangers of stochastic parrots: Can language models be too big? in: *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, ACM, 2021, pp. 610–623.
- [45] B. Sparrow, J. Liu, D.M. Wegner, Google effects on memory: Cognitive consequences of having information at our fingertips, *Science* 333 (6043) (2011) 776–778.
- [46] M. Kapur, Productive failure, *Cogn. Instr.* 26 (3) (2008) 379–424.
- [47] G. Guizzardi, O. Pastor, V.C. Storey, Thinking fast and slow in software engineering, *IEEE Softw.* 40 (6) (2023) 139–142.
- [48] D.W. Embley, S.W. Liddle, O. Pastor, Conceptual-model programming: A manifesto, in: *Handbook of Conceptual Modeling - Theory, Practice, and Research Challenges*, Springer, 2011, pp. 3–16, http://dx.doi.org/10.1007/978-3-642-15865-0_1.

Veda C. Storey is the Tull Professor of Computer Information Systems and professor of computer science at the J. Mack Robinson College of Business, Georgia State University. Her research interests are in intelligent information systems, data management, conceptual modeling, and design science research. She is particularly interested in the assessment of the impact of new technologies on business and society from a data management perspective. Dr. Storey is a member of the steering committee of the *International Conference of Conceptual Modeling* and a recipient of the Peter P. Chen Award. She is also a member of the *Workshop on Information Technologies and Systems* and the AIS Senior Scholars.

Oscar Pastor is a full professor and Internationalization and Technology Transfer Director at the Valencian Research Institute for Artificial Intelligence, Universitat Politècnica de València. He is a professor of methods at the Department of Information Systems and Computing and the director of the Research Centre in Software Production Methods (PROS). Óscar Pastor is the author of over 200 scientific publications in conference proceedings, journals and books and received the Peter P. Chen Award in 2016 for his work on conceptual modeling.

Giancarlo Guizzardi is a full professor of Software Science and Evolution and chair of Semantics, Cybersecurity, and Services at the University of Twente, in The Netherlands. He has been active for nearly three decades in the areas of Applied Ontology, Conceptual Modeling, Enterprise Computing and Information Systems Engineering, working with a multidisciplinary approach in Computer Science that aggregates results from Philosophy, Cognitive Science, Logics and Linguistics. He is the main contributor to the Unified Foundational Ontology (UFO) and to the OntoUML modeling language. He is currently the chair of the steering committee of the International Conference on Conceptual Modeling (ICCM), a member of the steering committees of CAiSE, EDOC, and IEEE CBI, and a member of the Advisory Board of the International Association for Ontology and its Applications (IAOA). Finally, he is an ER fellow.

Dr. Stephen W. Liddle is a professor of information systems at the Marriott School of Management, Brigham Young University, where he holds the J. Owen Cherrington Professorship. Dr. Liddle teaches mobile app development, web development, systems design, and other topics related to the practical application of computer systems and conceptual modeling. He has been a member of BYU's business school faculty since 1995, when he received his Ph.D. in computer science from BYU. His work has appeared in a number of respected journals and conferences. Besides authoring or coauthoring numerous refereed academic papers, Liddle is editor of many conference and workshop proceedings and is coauthor of the book *E-Business: Principles and Strategies for Accountants*.

Prof. Dr. -Ing. Wolfgang Maas holds the Chair of Information Systems for the Service Industry at Saarland University. Previously, Wolfgang Maas led the research department 'Smart Products' at the Institute for Technology Management (ITEM) at the University of St. Gallen (HSG), Switzerland. He was a private lecturer at the University of St. Gallen and professor for media informatics at Furtwangen University (HFU). There he led the research department 'Intelligent Media' (RCIM). In addition to his research activities, he worked as a consultant for Diebold Management Consulting, was managing director of a start-up company for several years and is actively involved as a co-founder of spin-offs. His research area is Information Systems, which includes Ubiquitous Information Systems (UIS) with a focus on Smart Products, the Internet of Things, Intelligent Information Systems, Conceptual Modeling, Electronic Knowledge Markets, and Paid Content. Professor Dr.-Ing. Maas is a renowned researcher in the field of information systems with expertise in data science, service design, design science, recommendation systems and dialog systems. He has technical expertise in artificial intelligence and quantum computing. His research focuses on the application of these skills in various application areas, including Industry 4.0, healthcare, and sports. His work aims to gain insights from data, optimize processes and create innovative solutions at the interface of technology and society.

Jeffrey Parsons earned an undergraduate degree in business from Memorial and a Ph.D. in information systems from the University of British Columbia. His research interests focus on representing information in ways that better match how humans think about objects in their physical or social environment. His research has been published in journals such as *Nature*, *Management Science*, *MIS Quarterly*, *Information Systems Research*, *Journal of the Association for Information Systems*, *Journal of Management Information Systems*, *Communications of the ACM*, *ACM Transactions on Database Systems*, and *IEEE Transactions on Software Engineering*. He holds research grants from NSERC and SSHRC and has served on grant committees for both funding councils.

Jolita Ralyté is a senior researcher and lecturer at the **University of Geneva**, **CUI (Centre Universitaire d'Informatique)** and member of the **Institute of Information on Services Science**. She is the director of the continuing education program **MATIS - Management and Technology of Information Systems**. Her current research interests lie at the crossroads of Digital Transformation, Services Science and Situational Method Engineering. In particular, she works to build exploratory and contributory approaches for the development of transdisciplinary information services and service systems. Jolita currently acts as national representative and Chair of the IFIP TC8 Information Systems.

Maribel Yasmina Santos is a Full Professor in Information Systems and Technologies at the Department of Information Systems, University of Minho, Portugal. She is a Senior Researcher at the ALGORITMI Research Centre and the CCG/ZGDV ICT Innovation Institute, where she leads the data engineering and analytics group. Her research interests include Business Intelligence and Analytics, focusing on Big Data Analytics, including data architectures, data processing, data analysis, and data visualization. Maribel Yasmina Santos was Vice-Dean of the School of Engineering at the University of Minho (2019-2022), Dean of the Pedagogical Council of the School of Engineering at the University of Minho (2010-2014). Maribel was actively involved in AGILE (Association of Geographic Information Laboratories for Europe), serving as Secretary-General (2013-2015) and Member of the AGILE council (2011-2015).