eKNOW 2026

The Eighteenth International Conference on Information, Process, and Knowledge Management

ISBN: 978-1-68558-387-3

May 24 - 28, 2026

Venice, Italy

eKNOW 2026 Editors

Susan Gauch, University of Arkansas, USA

eKNOW 2026

Forward

The Eighteenth International Conference on Information, Process, and Knowledge Management (eKNOW 2026), held between May 24, 2026, and May 28, 2026, in Venice, Italy, continued a series of events covering the complexity of the current systems, the diversity of the data, and the challenges for mental representation and understanding of environmental structure and behavior.

Capturing, representing, and manipulating knowledge was and still is a fascinating and extremely useful challenge from both a theoretical and practical perspective. Using validated knowledge for information and process management and for decision support mechanisms raised a series of questions the eKNOW 2026 conference was aimed at.

The event provided a forum where researchers were able to present recent research results and new research problems and directions related to them. The topics covered aspects from knowledge fundamentals to more specialized topics such as process analysis and modeling, management systems, semantics processing and ontology.

We take here the opportunity to warmly thank all the members of the eKNOW 2026 technical program committee, as well as all the reviewers. The creation of such a high-quality conference program would not have been possible without their involvement. We also kindly thank the authors who dedicated time and effort to contribute to eKNOW 2026. We truly believe that, thanks to all these efforts, the final conference program consisted of top-quality contributions. We also thank the members of the eKNOW 2026 organizing committee for their help in handling the logistics of this event.

We hope that eKNOW 2026 was a successful international forum for the exchange of ideas and results between academia and industry for the promotion of progress in the field of information, process, and knowledge management.

eKNOW 2026 Chairs

eKNOW 2026 Steering Committee
Susan Gauch, University of Arkansas, USA
Samia Aitouche, Laboratory of Automation and Manufacturing (LAP), University of Batna 2, Algeria
Roy Oberhauser, Aalen University, Germany

eKNOW 2026 Publicity Chairs
José Miguel Jiménez, Universitat Politecnica de Valencia, Spain
Francisco Javier Díaz Blasco, Universitat Politècnica de València, Spain
Ali Ahmad, Universitat Politècnica de València, Spain
Sandra Viciano Tudela, Universitat Politecnica de Valencia, Spain
Laura Garcia, Universidad Politécnica de Cartagena, Spain

eKNOW 2026
Committee

eKNOW 2026 Steering Committee

Susan Gauch, University of Arkansas, USA
Samia Aitouche, Laboratory of Automation and Manufacturing (LAP), University of Batna 2, Algeria
Roy Oberhauser, Aalen University, Germany

eKNOW 2026 Publicity Chairs

José Miguel Jiménez, Universitat Politecnica de Valencia, Spain
Francisco Javier Díaz Blasco, Universitat Politècnica de València, Spain
Ali Ahmad, Universitat Politècnica de València, Spain
Sandra Viciano Tudela, Universitat Politecnica de Valencia, Spain
Laura Garcia, Universidad Politécnica de Cartagena, Spain

eKNOW 2026 Technical Program Committee

Rocío Abascal Mena, Universidad Autónoma Metropolitana - Cuajimalpa, Mexico City, Mexico
Malak A. Abdullah, Jordan University of Science and Technology, Jordan
Marie-Hélène Abel, Sorbonne universités - Université de technologie de Compiègne, France
Awais Adnan, Institute of Management Sciences Peshawar, Pakistan
Nitin Agarwal, University of Arkansas at Little Rock, USA
Abdullah Fathi Ahmed, University Paderborn, Germany
Samia Aitouche, University of Batna 2, Algeria
Arnulfo Alanis, Instituto Tecnológico de Tijuana | Tecnológico Nacional de México, Mexico
Mohammad T. Alshammari, University of Hail, Saudi Arabia
Bráulio Alturas, Instituto Universitário de Lisboa (ISCTE-IUL) | ISTAR-Iscte (University Institute of Lisbon), Portugal
Gil Ad Ariely, Lauder School of Government, Diplomacy and Strategy - Interdisciplinary Center Herzliya (IDC), Israel
Mohamed Anis Bach Tobji, ESEN – University of Manouba | LARODEC Laboratory – ISG of Tunis, Tunisia
Mário Antunes, Polytechnic of Leiria, Portugal
Jorge Manuel Azevedo Santos, Universidade de Évora, Portugal
Michal Baczynski, University of Silesia in Katowice, Poland
Zbigniew Banaszak, Koszalin University of Technology, Poland
Basel Bani-Ismail, Oman College of Management and Technology, Oman
Dušan Barać, University of Belgrade, Serbia
Peter Bellström, Karlstad University, Sweden
Hajer Ben Othman, National school of computer science - University of Manouba, Tunisia
Asmaa Benghabrit, Moulay Ismaïl University, Meknès, Morocco
José Alberto Benítez Andrades, University of León, Spain
Julita Bermejo-Alonso, Universidad Isabel I, Spain
Shankar Biradar, Indian Institute of Information Technology Dharwad, India
Karsten Boehm, University of Applied Sciences, Kufstein, Austria

Copyright Information

For your reference, this is the text governing the copyright release for material published by IARIA.

The copyright release is a transfer of publication rights, which allows IARIA and its partners to drive the dissemination of the published material. This allows IARIA to give articles increased visibility via distribution, inclusion in libraries, and arrangements for submission to indexes.

I, the undersigned, declare that the article is original, and that I represent the authors of this article in the copyright release matters. If this work has been done as work-for-hire, I have obtained all necessary clearances to execute a copyright release. I hereby irrevocably transfer exclusive copyright for this material to IARIA. I give IARIA permission or reproduce the work in any media format such as, but not limited to, print, digital, or electronic. I give IARIA permission to distribute the materials without restriction to any institutions or individuals. I give IARIA permission to submit the work for inclusion in article repositories as IARIA sees fit.

I, the undersigned, declare that to the best of my knowledge, the article is does not contain libelous or otherwise unlawful contents or invading the right of privacy or infringing on a proprietary right.

Following the copyright release, any circulated version of the article must bear the copyright notice and any header and footer information that IARIA applies to the published article.

IARIA grants royalty-free permission to the authors to disseminate the work, under the above provisions, for any academic, commercial, or industrial use. IARIA grants royalty-free permission to any individuals or institutions to make the article available electronically, online, or in print.

IARIA acknowledges that rights to any algorithm, process, procedure, apparatus, or articles of manufacture remain with the authors and their employers.

I, the undersigned, understand that IARIA will not be liable, in contract, tort (including, without limitation, negligence), pre-contract or other representations (other than fraudulent misrepresentations) or otherwise in connection with the publication of my work.

Exception to the above is made for work-for-hire performed while employed by the government. In that case, copyright to the material remains with the said government. The rightful owners (authors and government entity) grant unlimited and unrestricted permission to IARIA, IARIA's contractors, and IARIA's partners to further distribute the work.

Table of Contents

Revaluation of Human Experts in AI Systems with Joint Interactive Modelling

Marjolein Deryck and Simon Vandevelde and Joost Vennekens
Dept. of Computer Science, Leuven.AI, Flanders Make
KU Leuven, Jan Pieter De Nayerlaan 5, Sint-Katelijne-Waver, Belgium
e-mail: {marjolein.deryck | s.vandevelde}@kuleuven.be
Dept. of Informatics and Applied Informatics, Federated Labs of AI and Robotics
Vrije Universiteit Brussel, Pleinlaan 9, Etterbeek, Belgium
e-mail: joost.vennekens@vub.be

***Abstract*—Knowledge acquisition is crucial for gathering and managing a company's knowledge, especially when creating systems that support business activities. Typically, this process involves a close collaboration between domain experts and knowledge engineers. While it is traditionally driven by the knowledge engineer, the role of the domain expert has steadily evolved from a mere source of knowledge to that of an active partner. In this paper, we introduce a new knowledge acquisition methodology, named Joint Interactive Modelling, which covers all stages of the knowledge acquisition process while placing the domain expert at the centre of the action. We describe this methodology, the tools developed to support it, and an evaluative case study.**

Keywords-knowledge acquisition; user-centred design; knowledge representation formalisms and methods.

I. Introduction

Knowledge acquisition is often approached as a one-off project, aimed at collecting the knowledge of a domain expert (hereafter referred to as 'the expert(s)') at a specific point in time. Although this view has been partly relaxed by organizing the knowledge acquisition efforts in cycles similar to agile project management, it still inherently fails to recognize the dynamic aspect of knowledge in a learning organization. In such organizations, knowledge creation and communication are part of the day-to-day activities. As a result, knowledge is decentralized and much of the knowledge exchange is not purposely planned. Knowledge acquisition techniques that fail to recognize this, will almost inevitably lead to knowledge models that become quickly deprecated, as well as the systems that are based on it.

Fortunately, the original idea of an expert as a "barrel" of knowledge that should be "tapped" by a Knowledge Engineer (hereafter referred to as KE or engineer) has since shifted towards a more inclusive view in which the expert and the engineer collaborate to formalize the sought-after knowledge. Nevertheless, the engineer still plays a crucial role in the knowledge acquisition activity, as they are the ones that translate the domain knowledge into formal models. This has two major disadvantages. First, it is difficult to keep the knowledge up-to-date, as there is the need to always involve a engineer to update the model. Second, there may be frequent misunderstandings between expert and engineer, as one is not familiar with the modelling formalism and the other is usually not acquainted with the domain.

In this paper, we present Joint Interactive Modelling (JIM), a knowledge acquisition methodology that supports the entire knowledge acquisition process, from elicitation to formalization and prototyping, and places the expert at the centre. The methodology that we present in this paper has been developed over the course of several use cases, where the various requirements of the different use cases prompted us to develop different parts of the methodology. The key contribution of this paper is that we bring all of the different components together into a single coherent description of the whole JIM approach, situate it within the knowledge acquisition literature, and do a first comparative case study comparing traditional interviewing and JIM.

This paper begins with a related work section, where we explain some fundamentals of knowledge acquisition. Next, in Section III, we introduce the JIM method. The tools and methodology to use JIM in practice are described in Section IV. Then, in Section V we describe how we evaluated our method. Finally, the paper concludes in Section VI.

II. Related Work

A. Knowledge acquisition process

There is no single accepted definition of knowledge acquisition, and different authors discern different stages. In this paper, we follow the definition of Leu et al. [1], who identify three steps: i) knowledge elicitation, or the formulation of knowledge by experts, ii) knowledge explication, or the analysis and interpretation of the elicited knowledge by an engineer, and iii) knowledge formalisation, or the modelling of the explicated knowledge in formal models by the engineer. This process can be executed in multiple cycles, during which typically different kinds of models are created [2].

As a first description of the elicited knowledge, the engineer creates a *phenomenon model*. This is a model that is understandable for the expert, and should be validated by them. This model often has the form of a natural language description, possibly enriched with tables or diagrams.After the phenomenon model is validated, the engineer will further engage in analysis and modelling to create the *information model*, that aims to communicate the requirements of the application to the programmer. This model often contains blocks of pseudo-code, entity relationship diagrams, etc. After a third round of analysis and modelling, the *computer model* that forms the application, is created by a programmer. With the creation of each model, there is a risk of misunderstanding.

 ISBN: 978-1-68558-387-3

For knowledge elicitation, a plethora of techniques exist (going into the hundreds according to [1]), from interviews over observation to protocol analysis and more. There exist many differences between them, as well as a variety of ways to classify them. One common classification is based on the differential access hypothesis. This hypothesis states that the elicitation method determines the kind of knowledge one obtains [3]. Hence, methods can be classified according to their information output. One explanation for this is that the elicitation method impacts the reasoning strategy of the expert, who as a result focuses more or less on specific aspects. Another argument states that some knowledge is implicit or tacit, and experts will not be able to verbalise it unless an adapted elicitation method is used [3]. For example, in the field of software development, prototyping is a widely used way of eliciting requirements, because users typically do not have a good idea of what they need prior to seeing and testing the prototype [4]. Prototypes can be divided in throwaway and evolutionary prototypes. Throwaway prototypes have the sole purpose of gathering feedback and are discarded afterwards. Evolutionary prototypes are used when users already have a good idea of what they need, and the prototype gathers additional functionalities or knowledge that are iteratively added [4].

Other classification methods do not solely focus on knowledge elicitation techniques but on knowledge acquisition techniques in general, and classify them according to the knowledge acquisition phase they support. Leu et al. [1] investigated 21 knowledge acquisition methods for their support in each of the three knowledge acquisition phases. Some of the techniques support mainly one phase, such as verbal reports (elicitation), protocol analysis (analysis), or diagramming (representation), whereas other techniques support two phases, such as cognitive demands table (elicitation, analysis) or psychological scaling (analysis, representation). Currently, no method supports all knowledge acquisition phases.

Importantly, none of the classifications elaborates on the role of the expert in the knowledge acquisition process. At the emergence of knowledge acquisition as a separate field, experts were mainly seen as barrels of knowledge, that could be tapped by engineers [1]. The experts played a passive role in the process, by which they were seemingly unaffected.

Soon after, a transactional view on knowledge acquisition was proposed, as it became apparent that the transmission of knowledge is an interactive process in which the expert plays an active role [1]. Engineers may ask questions that the expert cannot answer. By searching the answer, the knowledge of both the engineer and the expert grows [5]. This is also called the co-creation of knowledge.

To go one step further, other researchers envisioned a knowledge acquisition process without any engineer's involvement at all [6]. Some tools were developed to support experts in these efforts [7][8], but this has not led to a widespread adoption of the approach. Although it seems unrealistic to completely remove the engineer from the equation, there is a clear trend towards a larger and essential role of experts in the creation of knowledge models. JIM aligns with this trend, and offers the additional advantage of supporting the entire knowledge acquisition process.

B. Knowledge Base Paradigm

Knowledge acquisition is important in the area of Artificial Intelligence (AI) focusing on knowledge representation and reasoning. In our work, we focus mainly on the creation of Knowledge Base Systems (KBS), i.e., AI systems that follow the Knowledge Base Paradigm (KBP) [9]. This paradigm emphasizes a strict separation between the description of domain knowledge (captured in a Knowledge Base (KB)), and how it's put to use by inference algorithms. The KB contains knowledge in a computer-readable format, often in a language that is based on formal logic. Importantly, this domain knowledge is declarative: it does not specify *how* certain tasks should be performed, but only *what* knowledge exists in the domain. This allows inference algorithms to be used independently of the domain, making the KB more maintainable and inferences more flexible across domains. Knowledge acquisition is at the same time indispensable, yet also the bottleneck, for the creation of the KB [0]. Therefore, the development of suitable knowledge acquisition method is an important factor for the development of KBSs.

C. Technology acceptance and usage

One common challenge in the introduction of new systems is the willingness of intended users to accept and use them. More than 80% of software projects are "challenged" or fail [10], which can be partially explained by the lack of change management and user acceptance of the system. Venkatesh et al. [11] propose a Unified Theory of Acceptance and Use of Technology (UTAUT), in which they quote four causal factors that determine the usage of applications: the user expectation on how the application will perform, the users expectation on the effort it will take them to use the application, social influences on the user and facilitation conditions. Turan [12] expands this model by placing it in an overarching theory. Relevant to understand the impact of JIM are the two factors that Turan recognises as preceding the UTAUT, namely personal innovativeness and user involvement. The JIM method puts the experts, who are typically also the users, central in the knowledge acquisition and application creation process. Moreover, the method aims explicitly to promote ownership of the knowledge base by experts.

III. Joint Interactive Modelling

JIM is a methodology for interactive KB creation. It replaces the typical three-step approach of knowledge acquisition with by single iterative process, in which an expert and an engineer jointly express, analyse and model the domain, all the while validating the resulting model to ensure correctness.

The KB contains domain knowledge on a given topic. The methodology does not focus on inference knowledge (how the domain knowledge can be used), or task knowledge (how inferences can be combined to execute a complex task) [13].

The KB that we envision exists of an ontology (which variables are manipulated), rules and constraints that determine the relation between these variables (e.g., [14][15]). This allows to represent knowledge in a broad area of domains: legislation, tax, investment profiles, adhesive selection, component design, planning and scheduling, game rules, electric circuits, $ldots$. As we focus solely on domain knowledge, the model should be epistemologically correct, allowing different ways of reasoning over it.

JIM is performed in workshops in which at least one expert and one engineer participate, although the inclusion of multiple experts has the advantage of covering a more complete view on the domain and aligning company practices [16]. Workshops follow a fixed pattern, as shown below, with step 4-5 repeating until the desired level of detail is reached.

- Step 1: Workshop introduction: explain goal application, system architecture and the modelling language
- Step 2: Scoping of the domain: determine the scope of the domain to be described in the knowledge base.
- Step 3 (optional): Composing high-level insight in the decision/constraint structure
- Step 4: Interactive discussion in which the engineer asks questions and steers the discussion. The expert shares their expertise. Together the engineer and expert add new knowledge to the model.
- Step 5: Validate the new knowledge and its integration
- Step 6: Releasing the model

By jointly creating the knowledge and validating frequently, JIM emphasizes actively including the expert in the knowledge formalization. The main idea is that **experts should keep ownership of the knowledge model** that is at the heart of an application, and should be able to maintain it, even after the engineer is gone. Therefore, a distinguishing feature of our method is the use of a **common modelling language shared between the expert and the engineer**, resulting in a single knowledge model that can be read, understood and maintained by both parties.In this way, using a common knowledge model decreases the cost and risks associated with the traditional creation of different types of models.

To support all this, we need a modelling language that supports users in their analysis of the domain, in order for the different employee roles to develop a common understanding of it. Moreover, it should be straightforward enough to be understood and used by everyone involved without extensive training effort. At the same time, the language cannot be too simple, as it should be sufficiently expressive to capture the complexity of the domain. Finally, in line with the differential access hypothesis, the model should allow for **interactive, evolutionary prototyping**. The prototype will support the users most in their knowledge acquisition if it is able to give real-time feedback, offers understandable and detailed explanations of outcomes and errors, allows high user-control of the workflow, has a clear and understandable interface, and can run simulations. In this way, it can not only validate the modelled knowledge, but also highlight gaps and support additional elicitation.

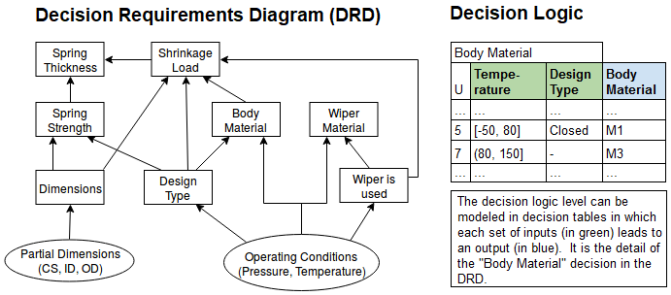


Figure 1. DRD and decision table extract for component design.

IV. Example Implementation

The previous section gave a theoretical description of JIM as a methodology and the requirements of the modelling language. In this section, we briefly elaborate on the practical approach. Given that steps 1-3 and 6 are quite straightforward, we choose to focus on step 4 and 5 by describing a concrete modelling language, and an interactive method for knowledge validation.

A. Constraint Decision Model and Notation

An example of a language that can be used for JIM, is the Decision Model and Notation (DMN) [18], published by the Object Management Group (OMG). In their words, DMN aims to "provide a common notation that is readily understandable by all business users, from the business analysts [...] and businesspeople to the technical developers [...]". In DMN, decisions are represented in straightforward decision tables. The relations between the different decision tables can be visualised in a decision requirements diagram (DRD), which visually shows the structure of the domain.

Figure 1 shows a DRD for the design of a component, and an extract of the decision logic for "Body Material". The hit policy "U" (left on the second row) means that all rows in this table are mutually exclusive, and at most one rule may apply. The green columns are inputs, the blue column is the output.

One limitation of DMN is that the tables only express rules: based on the inputs, an exact output is defined. For instance, it is not possible to exclude a value, or to leave a value open. This makes it difficult to capture more complex knowledge. To overcome this limitation, we have extended DMN with constraints, in a notation called Constraint Decision Model and Notation (cDMN) [19]. cDMN uses the same user-friendly, tabular format, but also adds the ability to express constraints and some other related concepts. For example, the MaxT constraint table in Figure 2 expresses that if a component is used, the environment temperature must be lower than the maximum temperature in which the component can operate. Here, the *E* (Every)* hit policy denotes a *constraint table*. This differs from a standard decision table in two main ways: a constraint table does not need to be complete, and does not need to specify an exact output value but can instead also specify ranges, negations, and more.

Max T constraints			
E*	Component	Component is used	max temp of Component
1	—	Yes	> environment temperature

Figure 2. cDMN table for Max T constraints

Besides constraint tables, cDMN also introduces other functionalities to make it easier to express complex knowledge, such as quantification, predicates, functions, and data tables. In summary, the goal of cDMN is to maintain DMN's user friendliness tabular format, while increasing its expressiveness in order to capture and represent more complex information. For more information on cDMN, including its semantics and some examples, we refer to [19].

B. Interactive Consultant

The second aspect of the JIM methodology is the ability to quickly and effortlessly generate prototypes. We use the IDP-Z3 reasoning engine [14] with its Interactive Consultant (IC) [20] interface for prototyping based on (c)DMN models. Behind the scenes, the (c)DMN model is automatically translated into a first-order logic based KB, after which IDP-Z3's generic inference algorithms allow reasoning over the KB. Among others, IDP-Z3 supports (1) verifying if a solution is possible, (2) generating solutions, (3) deriving consequences, (4) explaining why something is correct/false, and more. The IC is a generic interface for IDP-Z3: given any syntactically correct knowledge base, the IC will generate a view in which each symbol of the knowledge base is represented in a tile layout, as shown in Figure 3. Each of these tiles then allows a user to toggle on or off specific values for the symbols, which causes to system to automatically compute the consequences, and displays them. In this way, the IC offers a way of *interactively exploring* a problem domain: it gives users the opportunity to "play around" with the knowledge, and to see what effects some design choices might have.

V. Evaluation: optical lens embossing

So far, JIM has been successfully applied in two real-life case studies: (1) the selection and design of highly-specialized components [17], and (2) the selection of an appropriate adhesive for industrial applications [21][0]. Although this already demonstrates the practical usability of JIM, we now present for the first time a comparative case study to evaluate specific claims.

This case was conducted with the Photonics Lab at the Vrije Universiteit Brussel and concerns the embossing of lenses. The embossing process consists of five steps, going from pre-heating the material, to heating, embossing, cooling, and finally de-moulding the lens. During each of the steps, different parameters can be used with regards to temperature, time and pressure. After a lens is created, it is visually inspected by a highly-trained expert. Typically, the lens will show some deficiencies in the first trials, mostly scratches and shrinkage. The operator will go through a multiple trial tuning process, until the lens is visually perfect. After that, a scan of the lens is taken to measure if the dimensions of the lens are as required. Typically, two to three additional adjustments are necessary to achieve acceptable accuracy. The purpose of the workshop was described upfront as "...to create a knowledge base that helps users to identify the current quality grade of a lens and what to do to improve the current quality" based on the visual quality inspection (i.e., without scan measurement).

a) Methodology: Our aim is to compare JIM with the traditional knowledge elicitation method of structured interviewing, by applying both methods to the same task of creating a knowledge base for the lens embossing domain. We want to compare both the resulting KBs and the modelling effort needed to construct them. In particular, we have the following working Hypotheses (H) about the relation between JIM and structured interviewing:

- H1. The modelling effort using JIM is lower.
- H2. The knowledge base resulting from JIM is more correct.
- H3. Experts better understand the knowledge model that has been created with JIM.
- H4. Experts feel more involved with JIM.
- H5. Overall, experts are happier with the outcome and process of JIM.

To answer these questions, we organized two separate workshops on the same day, with the JIM workshop taking place in the morning, and the structured interview workshop in the afternoon. Both workshops were attended by the same two experts and the same observer. Both experts have an engineering background, with expert 1 being a computer scientist, and expert 2 a mechanical engineer. The JIM workshop was led by engineer KE1, and the structured interview by KE2. In both workshops, the same artifacts (lenses, reports, lab infrastructure) were used. The workshops both lasted 114 minutes, excluding the visit of the lab that was done with KE1 and KE2 together.

b) Results: The outputs from workshop 1 are a DRD and a set of decisions/constraint tables. The output from workshop 2 is a KB in the FO(·) format, which is the "main" input language for IDP-Z3 but is regarded as too difficult for people without a computer science background. Both KBs are syntactically sound, and can be loaded into the IC interface. However, for IP reasons, we are not allowed to completely share these outputs.

H1 Modelling effort. Our hypothesis is that the overall knowledge acquisition effort required to construct a given KB is lower when using JIM than when using traditional modelling methods. The setup aimed to produce comparable KBs from KE1 and KE2 to allow a direct comparison of knowledge acquisition time. However, when comparing the KBs, it became clear that despite efforts to clearly define the scope up front, KB2 is much more detailed than KB1. Hence, it is not possible to compare total knowledge acquisition times. Both workshops lasted 2 hours, but the traditional method required another 1.5 hours to finish KB2 afterwards. Since we cannot determine how much additional time JIM would have required to reach comparable detail, the results remain inconclusive with respect to this hypothesis. We can infer that

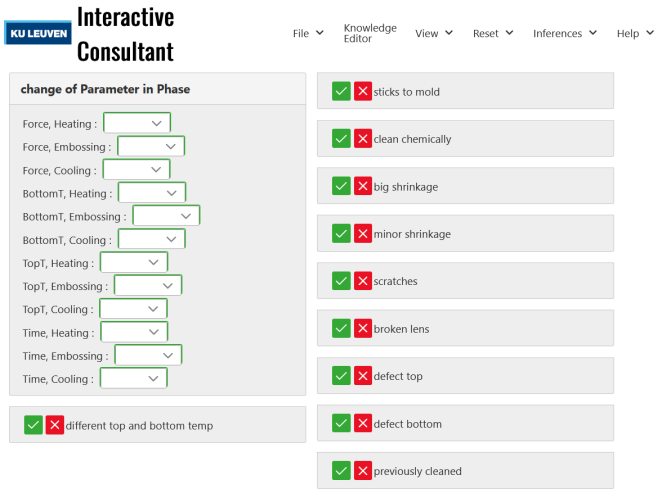


Figure 3. The IC interface for lens embossing, created with JIM

TABLE I. COMPARISON OF KNOWLEDGE BASES OBTAINED BY JIM AND BY SEMI-STRUCTURED INTERVIEWING

Component	JIM	Interviewing
types	3	10
possible values	10	42
predicates	11	11
formulas	9	27
process parameters	4	11
process steps	3	5

for a KB of similar detail, the traditional approach would have required less expert time than JIM in this specific case; however, this finding is trivial, as JIM is explicitly designed to involve experts throughout the modelling process.

H2 Knowledge base correctness. Deciding on the correctness of a knowledge base is not trivial: even when using the same knowledge acquisition technique, different modellers can end up with differently formalized knowledge, due to personal preferences for readability, performance, potential expandability, etc. Table I compares some characteristics of the KBs. The main difference is that the KB elicited by the interview method is both larger (in terms of the number of symbols and formulas) and broader in scope (in terms of the process parameters and steps). The validation with the experts did not reveal mistakes in either KB.

H3 Understanding the KB. A key element of JIM is that users should be able to understand and validate the KB. Ideally, they should be able to extend and adapt it to reflect new knowledge. To test this, we organized a validation workshop with the experts. This was a two hour meeting with the experts, the two engineers and the observer, in which both KBs were discussed. During the session, each engineer explained the structure of their KB (15 min.), and the correctness of the KB was discussed (25 min.). To test the expert's understanding, we asked them to add an additional (fictitious) phase to the model of the the embossing process ($\sim$20 min.). Afterwards, the experts filled questionnaires for both the JIM and the interview-based method ($\sim$20 min.). Our aim was to detect differences in attitude towards both modelling methods and the resulting applications. Surprisingly, both experts gave almost the same answers for both methods: only the statement "my interaction with the system would be clear and understandable" received from one respondent a score 4 (out of 5-best) for JIM method versus a score 5 for the interview method.

Both experts were able to adjust the model with minimal help. There was no difference observed between the modelling in cDMN and the modelling in FO($\cdot$). In the survey, one expert indicated a full understanding of the knowledge model, both in cDMN and in FO($\cdot$). The second expert indicated to understand the structure and some tables/sentences of the KBs, but not to the extent that they would feel comfortable explaining it to a colleague. In conclusion, our experiment does not show a difference in expert understanding between the cDMN KB versus a FO($\cdot$) KB.

H4 User involvement and H5 general appreciation. On top of our own survey, we used the UTAUT survey to probe for the application acceptance. Using scores from 1 to 5 and changing directions for negative questions (19, 27, 28, 29), the average score (on 5) is 4.025 for JIM and 4.041 for the traditional modelling approach. These high numbers confirm the positive expert feedback in the other survey, but differences are too small to draw further conclusions on differences between JIM and traditional modelling.

c) Discussion: The purpose of this study is to compare two knowledge acquisition methods on the same domain, in order to avoid distortion by different domain or task complexity. The case is big enough compared to real life use cases (e.g., [22] describes real-life investment profiles in 20 rules), yet small enough to be covered in a 2-hour workshop. Nevertheless, the described setup shows some shortcomings. Because a lack of experts with similar expertise, the same experts were used for both workshops, potentially creating a learning bias. Because the study involved only two experts, the results may not be generalizable to other contexts. As expected, less time is required to create a KB with JIM, because knowledge elicitation, analysis and formalization happen during the workshop. In the traditional modelling method, the time of the workshop was used for knowledge elicitation only, and modelling happened subsequently by the engineer. It is to be noted that an iterative approach using JIM may lead to an additional refactoring effort after the workshop, e.g., if it becomes clear halfway through the workshop that initial modelling choices were not optimal for the further detailing of the model. That would lead to additional modelling time, which may shed another perspective on the time difference.

In this use case, the JIM KB is smaller than the one created by traditional modelling. Further evaluation on the timing to create KBs of comparable scope is required to assess if the overall input-output effect of the two methodologies. The main difference between the methods seems to be the scope of the resulting model, which is more focussed in JIM than in the traditional modelling approach. The aim of the workshop was

to model the tuning process. Whereas KE1 modelled potential changes, KE2 also modelled the size of the change, and the quality level required to determine which size is relevant. The JIM KB gives the possibility to tune 4 process parameters in 3 process steps, whereas the interview KB shows 11 parameters in 5 process steps. In the latter, there is an additional relation that shows which actions have already been taken in the tuning process; this introduces a temporal element that reflects the iterative nature of tuning. This indicates that the tuning support itself is more fine-grained as result of the traditional method. Although this difference may be attributable to the difference in modelling time, another possibility is that the JIM method itself fosters a more focussed approach. Expert detours or overly detailed extensions are avoided by re-centring attention to the constraint tables. This is in line with earlier experiences ([17][21]), and with the differential access hypothesis, that states that different knowledge acquisition techniques result in different knowledge outputs. However, further experiments are needed to draw definitive conclusions. No difference was observed for user understanding of the KB. As cDMN was explicitly created to improve user-readability, this is a surprising outcome. A relevant follow-up question, therefore, is whether such a difference would emerge in the context of larger KBs, and if so, from which size on this is the case. Another question is whether the background of the experts, who are familiar with formal modelling, may explain the lack of difference: as cDMN was developed as a user-friendly modelling language for non-technical experts, its full advantages may only become apparent when evaluated within this target group. Alternatively, it is also plausible that the result is not linked to methodological shortcomings, but point to more fundamental issues in the current application of JIM. For instance, the assumed readability of cDMN representations may not hold in practice. Or JIM may be less effective when the engineer exerts a high degree of control over the elicitation process, suggesting that greater direct involvement of experts in model construction may be necessary.

VI. Conclusion and future work

JIM is a new method to formalize knowledge. It distinguishes itself from traditional knowledge acquisition methods by its user-centric approach and its emphasis on seamless prototyping. The knowledge acquisition process is traditionally seen to consist of 3 stages with distinct roles for the expert and the engineer. However, this approach ignores companies' need to continuously update their knowledge and carries an inherent risk of misunderstandings. In JIM, the expert and engineer together create a unique, executable knowledge model and prototype application. By evaluating this interactive prototype, new requirements or missing parts of knowledge can be added to the knowledge model according to the same principle.

As an example, we have introduced the user-friendly cDMN notation in conjunction with the Interactive Consultant and the IDP-Z3 reasoning engine for prototyping. We compared JIM with a traditional modelling approach in a use case on lens embossing. The experts found the cDMN model equally easy to read and use as the formal logic model. This prompts future work on readability of the cDMN notation across different expert profiles to test the hypothesis that the engineering background of the experts may be a mediating factor.

The main differences between the methodologies appeared in the time required for knowledge acquisition and in the scope of the model, which show an inverse relationship. The question is whether this relationship is causal: is a JIM model more focused simply because less time is spent on modelling, or does the method place less emphasis on intangible knowledge and therefore provide less access to the finer intricacies of the tuning process, thus allowing the work to be finalised more quickly? Consequently, further empirical investigation is required to disentangle these factors and to determine the precise causes underlying the observed outcomes.

To this end, we are currently in discussion with a board game club to engage its members as experts. The task would involve formalizing the rules of a given board game with which the engineers are unfamiliar. This setting offers several advantages: the rule set constitutes a well-scoped domain with real-world relevance, and the quality of the resulting knowledge base can be validated against the game's written rules. By collaborating with a board game club, we aim to involve approximately ten members with diverse professional backgrounds, who can be evenly split between the JIM approach and a traditional interviewing methodology.

VII. Acknowledgement

We sincerely thank the VUB photonics lab for their invaluable insights and support throughout this work and Christian Fleiner for the organisation of the experiment. This research was funded by the Flanders Make REXPEK project.

References

[1] G. Leu and H. Abbass, "A multi-disciplinary review of knowledge acquisition methods: From human to autonomous eliciting agents," *Knowledge-Based Systems*, vol. 105, pp. 1–22, Aug. 2016. DOI: 10.1016/j.knosys.2016.02.012.

[2] L. Hvam, K. Kristjansdottir, S. Shafiee, N. H. Mortensen, and Z. N. L. Herbert-Hansen, "The impact of applying product-modelling techniques in configurator projects," *International Journal of Production Research*, vol. 57, pp. 4435–4450, 14 2019. DOI: 10.1080/00207543.2018.1436783.

[3] R. R. Hoffman, N. R. Shadbolt, A. Burton, and G. Klein, "Eliciting knowledge from experts: A methodological analysis," *Organizational Behavior and Human Decision Processes*, vol. 62, no. 2, pp. 129–158, 1995. DOI: https://doi.org/10.1006/obhd.1995.1039.

[4] J. Fu, F. B. Bastani, and I.-L. Yen, "Model-driven prototyping based requirements elicitation," in *Innovations for Requirement Analysis. From Stakeholders' Needs to Formal Designs*, Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 43–61, ISBN: 978-3-540-89778-1.

[5] G. Lintern, B. Moon, G. Klein, and R. R. Hoffman, "Eliciting and representing the knowledge of experts," in *The Cambridge Handbook of Expertise and Expert Performance*. Cambridge University Press, 2018, pp. 165–191.

[6] I. M. Neale, "First generation expert systems: A review of knowledge acquisition methodologies," *The Knowledge Engineering Review*, vol. 3, no. 2, pp. 105–145, 1988.

[7] B. Gaines and M. Shaw, "Eliciting knowledge and transferring it effectively to a knowledge-based system," *IEEE Transactions on Knowledge and Data Engineering*, vol. 5, no. 1, pp. 4–14, 1993. DOI: 10.1109/69.204087.

[8] A. Kaba and C. K. Ramaiah, "Predicting knowledge creation through the use of knowledge acquisition tools and reading knowledge sources," *VINE Journal of Information and Knowledge Management Systems*, vol. 50, no. 3, pp. 531–551, 2020.

[9] M. Denecker and J. Vennekens, "Building a knowledge base system for an integration of logic programming and classical logic," in *International Conference on Logic Programming*, Springer, 2008, pp. 71–76.

[0] J. Cullen and A. Bryman, "The Knowledge Acquisition Bottleneck: Time for Reassessment?" *Expert Systems*, vol. 5, no. 3, pp. 216–225, Aug. 1988, ISSN: 0266-4720. DOI: 10.1111/j.1468-0394.1988.tb00065.x. Accessed: Feb. 27, 2025. [Online]. Available: https://doi.org/10.1111/j.1468-0394.1988.tb00065.x.

[0] J. P. Delgrande, B. Glimm, T. Meyer, M. Truszczynski, and F. Wolter, "Current and future challenges in knowledge representation and reasoning (dagstuhl perspectives workshop 22282)," *Dagstuhl Manifestos*, vol. 10, no. 1, J. P. Delgrande, B. Glimm, T. Meyer, M. Truszczynski, and F. Wolter, Eds., pp. 1–61, 2024, ISSN: 2193-2433. DOI: 10.4230/DagMan.10.1.1. [Online]. Available: https://drops.dagstuhl.de/entities/document/10.4230/DagMan.10.1.1.

[10] M. Haleem, M. Farooqui, and M. Faisal, "A critical analysis of software product failure: An indian and global perspective," *International Journal of Engineering and Advanced Technology*, vol. 8, pp. 106–113, Aug. 2019.

[11] V. Venkatesh, M. G. Morris, G. B. Davis, and F. D. Davis, "User acceptance of information technology: Toward a unified view," *MIS Quarterly*, vol. 27, pp. 425–478, 3 Dec. 2003. DOI: 10.2307/30036540.

[12] A. Turan, A. Ö. Tunç, and C. Zehir, "A theoretical model proposal: Personal innovativeness and user involvement as antecedents of unified theory of acceptance and use of technology," *Procedia - Social and Behavioral Sciences*, vol. 210, pp. 43–51, 2015, Proceedings of the 4th International Conference on Leadership, Technology, Innovation and Business Management (ICLTIBM-2014). DOI: https://doi.org/10.1016/j.sbspro.2015.11.327.

[13] J. Liebowitz and I. Megbolugbe, "A set of frameworks to aid the project manager in conceptualizing and implementing knowledge management initiatives," *International Journal of Project Management*, vol. 21, no. 3, pp. 189–198, 2003, ISSN: 0263-7863. DOI: https://doi.org/10.1016/S0263-7863(02)00093-5. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0263786302000935.

[14] P. Carbonnelle, S. Vandevelde, J. Vennekens, and M. Denecker, "IDP-Z3: A reasoning engine for FO(.)," 2022. DOI: 10.48550/arXiv.2202.00343.

[15] M. Younis and M. Abdel Wahab, "A capp expert system for rotational components," *Computers & Industrial Engineering*, vol. 33, no. 3, pp. 509–512, 1997, ISSN: 0360-8352. DOI: https://doi.org/10.1016/S0360-8352(97)00180-0. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0360835297001800.

[16] M. Grabowski, A. P. Massey, and W. A. Wallace, "Focus groups as a group knowledge acquisition technique," *Knowledge Acquisition*, vol. 4, no. 4, pp. 407–425, 1992, ISSN: 1042-8143. DOI: https://doi.org/10.1016/1042-8143(92)90003-J. [Online]. Available: https://www.sciencedirect.com/science/article/pii/104281439290003J.

[18] Object Management Group, *Decision model and notation v1.3*, 2021. Accessed: Apr. 7, 2026. [Online]. Available: http://www.omg.org/spec/DMN/.

[19] S. Vandevelde, B. Aerts, and J. Vennekens, "Tackling the DM challenges with cDMN: A tight integration of DMN and constraint reasoning," *Theory and Practice of Logic Programming*, pp. 1–24, 2021. DOI: 10.1017/S1471068421000491.

[20] P. Carbonnelle, M. Deryck, J. Vennekens, and M. Denecker, "An interactive consultant," in *BNAIC, Date: 2019/11/06-2019/11/08, Location: Bruxelles*, 2019.

[17] B. Aerts, M. Deryck, and J. Vennekens, "Knowledge-based decision support for machine component design: A case study," *Expert Systems with Applications*, vol. 187, p. 115 869, Jan. 2022. DOI: 10.1016/j.eswa.2021.115869.

[21] S. Vandevelde, J. Vennekens, J. Jordens, B. Van Doninck, and M. Witters, "Knowledge-Based Support for Adhesive Selection: Will it Stick?" *Theory and Practice of Logic Programming*, pp. 1–21, 2024. DOI: 10.1017/S1471068424000024.

[0] J. Jordens, S. Vandevelde, B. Van Doninck, M. Witters, and J. Vennekens, "Adhesive selection via an interactive, user-friendly system based on symbolic AI," ser. Procedia CIRP, Elsevier, 2022.

[22] M. Deryck, N. Comenda, B. Coppens, and J. Vennekens, "Combining logic and natural language processing to support investment management," in *Proceedings of the International Conference on Principles of Knowledge Representation and Reasoning*, vol. 18, 2021, pp. 666–670.

Proposal for a Process Governance Integrity Model

Building a Governance Integrity Model to Prevent Organizational Fraud and Departmental Conflicts

Shuichiro Yamamoto
Information Engineering, IPUT in Nagoya
Nagoya, Japan
e-mail: yamamoto.shu@n.iput.ac.jp

Abstract—**This paper elucidates the mechanisms by which recurring fraud, quality deviations, and departmental conflicts in organizations requiring high reliability, such as nuclear power generation, manufacturing, quality inspection, and the public sector, are generated structurally rather than accidentally, and proposes a new control theory, Process Governance Integrity (PGI), to address their root causes. PGI imposes Engineering/ Management/ Governance (EMG) invariant that must be satisfied by Engineering, Management, and Governance at process junctions, strictly guaranteeing the preconditions, execution trails, and postconditions before and after the process, resulting in a process in which deviations are structurally impossible. PGI is a unified process governance method that manages process governance completeness, a topic not addressed in existing theories such as Business Process Re-engineering, Theory of Constraints, Toyota Production System, and Functional Resonance Analysis Method.**

Keywords- Business Process Governance Integrity; Governance Integrity model; Business Process Conflict; EMG Invariant.

I. INTRODUCTION

Fraudulent practices, such as fraudulent inspections at nuclear power plants, fraudulent inspections in manufacturing, falsification of quality data, and cover-ups by administrative organizations, are most prevalent in organizations that require control. These are not the result of individual fraud but are a phenomenon that inevitably arises from flaws in the process structure. At the Hamaoka Nuclear Power Plant in particular, conflicts between the Nuclear Power Headquarters (management and management) and the Nuclear Civil Engineering Department (technology) deepened, causing the review process to malfunction, resulting in data fraud [1].

Behind this fraud lies a structural inconsistency in which technology, management, and administration are disconnected at the connection point, leading each department to pursue "local optimization."

The Swiss cheese model explains that many accidents and violations manifest at the interdepartmental connection points due to underlying organizational factors (role mismatch, procedural gaps, lack of responsibility transfer).

Recent empirical studies [2] have also confirmed that poor communication and weak procedural design can lead to a chain reaction of decision-making errors and violations.

This research aims to formalize process governance integrity, which has not been addressed in previous theories, as a process theory, and to present a theory that structurally prevents injustice and conflict.

The rest of this paper is organized as follows. Section II describes related research. Next, Section III proposes an EMG Invariant to ensure comprehensive completeness across all business processes. Section IV describes an application example of EMG Invariant. In Section V, we discuss our considerations, and in Section VI, we present a summary and future issues.

II. RELATED WORK

Below, we discuss related research on business process control.

A. Theory of Constraints (TOC)

Goldratt's Theory of Constraints [3] identifies and improves constraints but does not address control of process nodes.

The outline of TOC is as follows.

[Objective] Improve constraints to achieve overall optimization.

[Basic Philosophy] The weakest constraint determines overall performance.

[Process Perspective] View the entire flow as a single chain.

[Key Data] Constraint throughput/throughput

[Application Areas] Manufacturing, services, R&D, supply chain.

[Strengths] Overall optimization and bottleneck improvement are immediate results.

[Weaknesses] Misidentifying constraints can worsen problems.

B. Business Process Re-engineering (BPR)

BPR seeks to dramatically reform processes by eliminating waste in existing business processes. This makes it easy to destroy safety processes, and if misused, can destroy an organization. Hammer and Champy [4] emphasized that many failures in BPR stem not from technology or tool issues, but from "cultural resistance" and "failure to change people's mindsets." Therefore, BPR is both a technology-driven transformation and a cultural change model. The outline of BPR is as follows.

[Objective] Radical redesign of business processes.

[Basic Philosophy] Reconstruct business processes from scratch.

[Process Perspective] Destroy and reconstruct existing processes.

[Key Data] Structural data required for complete process redesign.

[Application Areas] Corporate reform, digital transformation, and structural transformation.

[Strengths] Large-scale reform and dramatic improvement in competitiveness.

[Weaknesses] Abstract and prone to misuse (easily disrupts safety processes).

C. Toyota Production System (TPS)

TPS [5] is strong in eliminating waste and stabilizing flow. However, there is a risk that important processes such as safety and quality may be treated as "waste" in the production process. The outline of TPS is as follows.

[Objective] Eliminate waste and achieve stable flow.

[Basic Philosophy] Just-in-time + automation + standardized work.

[Process Perspective] Stabilize and improve on-site flow.

[Key Data] Standardized work and waste analysis in the field.

[Application Areas] Manufacturing, quality control, logistics.

[Strengths] Strengthening quality, cost, and flow.

[Weaknesses] Misuse can lead to over-efficiency and lead to scandals.

D. Function Resonance Analysis Method (FRAM)

FRAM [6] can model the mechanism of accident occurrence to ensure resilience based on functional resonance. However, since the purpose is not to design a control system, it lacks preventive logic. The outline of FRAM is as follows.

[Purpose] Understanding fluctuations and resonances in complex systems and preventing accidents.

[Basic Concept] Emphasis on nonlinear inter-functional dependencies.

[Process View] Processes are fluctuating "collections of functions".

[Key Data] Inter-functional fluctuation and resonance patterns.

[Application Areas] Safety management, accident analysis, aviation, and medicine.

[Strengths] Analysis of accident mechanisms in complex systems.

[Weaknesses] Complex model requires skill for practical application.

E. Ji Koutei Kanketsu (JKK)

JKK [7] in Japanese is a word that translates to self (Ji), process (Koutei), and completion (Kanketsu). Self-process completion (JKK) is a method that optimizes the entire production process, not just a specific process.

JKK's requirements organization sheet describes acceptance criteria for each business process, as well as the criteria for determining whether the process output is good.

Defect Prevention Diagrams (DPDs) [8][9] make it possible to detect process deviations using exception conditions that were not available in JKK. Repetitive process control based on process exceptions enables reliable process operation that can respond to environmental changes. However, it does not address the design of control systems, such as not considering exception detection and responsibility for responding.

The outline of JKK is as follows.

[Objective] Maximize quality assurance and productivity throughout the entire process by preventing defects from being passed on to subsequent processes.

[Basic Philosophy] "Establishing conditions for good products." Establish a "scientific work process" that allows anyone to produce good products, rather than relying solely on the skills and awareness of individual workers.

[Process Perspective] "Causal chain." Manage the cause system rather than the result system, believing that a result (good product) always has a cause (procedure/condition).

[Important Data] "Conditions for good products (criteria and procedures)." Physical numerical values and procedural data for each task that guarantee a good product if followed.

[Area of Application] Originating in the manufacturing floor, now also applied to the work processes of staff departments (administrative and planning).

[Strengths] "Thorough prevention of recurrence." When a problem occurs, it is viewed as a "lack of conditions for good products" rather than as individual responsibility, and procedures are corrected, resulting in an extremely high organizational learning ability.

[Weaknesses] "Vulnerable to malice and deception." Because it is assumed that well-intentioned workers will "follow the correct procedures," there are weak logical constraints to detect and block intentional data rewriting and organizational concealment (the devil's room).

F. Assurance case

An Assurance Case is a practical technique for conducting evidence-based arguments regarding claims of the form "the system is in a certain state." Assurance case is called Safety case to assure safety. To claim that "a system is safe," evidence is required. When logically proving that a safety claim is correct based on this evidence, it is necessary to clearly state the prerequisites for the safety claim to be valid. In other words, it is necessary to prove that the safety claim is correct under the prerequisites.

Goal Structuring Notation (GSN), proposed by Kelly [10], is a notation for assurance cases that logically explain claims based on evidence. Safety cases are recommended in the functional safety standard ISO26262 [11].

However, Assurance Cases have problems in that they can be retroactively adjusted, they become well-written "stories," and they are difficult to detect if they are tampered with.

The existing research mentioned above has the limitation that none of them deal with the simultaneous three-tier control of engineering technology (E), management (M), and cooperate governance (G).

III. EMG INVARIANT

In the following, we propose an EMG invariant that serves as the basis for PGI. We then demonstrate that PGI generates a virtuous cycle of processes based on the EMG invariant.

A. PGI Prerequits

PGI is based on the EMG invariant, which states that the three elements of Engineering (technical validity), Management (procedural compliance and reproducibility), and Governance (accountability and legitimacy) must be satisfied simultaneously.

Fraud does not occur within a process, but at the process junctions between processes. PGI defines the EMG invariant at process junctions.

Preconditions, execution trails, and postconditions are defined for each process. PGI's three stages:

Precondition: The execution evidence of the preceding process and the EMG invariant must be satisfied at the junction.

Execution Trail: Full visibility of who did what and how.

Postcondition: The EMG invariant must be satisfied, with the three EMG parties approving the results created by the succeeding process based on the execution trail.

The EMG invariant requires simultaneous satisfaction of all three elements. This is because missing any one element creates a loophole.

Consider a business process where process P and process Q are connected at a connection point (P→Q). There are execution trails Trail P and Trail Q for process P and Q. The PGI invariant (PGII) for the connection point (P→Q) is PGII(P→Q). The record of EMG approval of this invariant is EMG(P→Q). Furthermore, the subsequent connection point of process Q can be expressed as (Q→), its PGI invariant as PGII(Q→), and its approval record as EMG(Q→). The PGI structure of processes P and Q is shown in Figure 1.

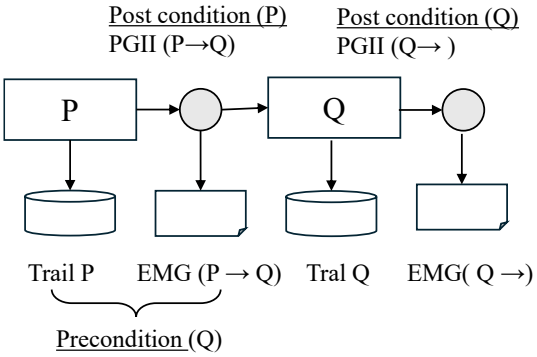


Figure 1. PGI structure.

B. PGI stops the vicious cycle and creates a virtuous cycle

Using Senge's [12] reinforcing loop and balancing loop, we explain how PGI can create a virtuous cycle in conflicts between the sales and production departments.

A typical vicious cycle (R-) between the sales and production departments is as follows: Sales pressures the production department to meet deadlines → weakened reviews → lack of evidence → on-site hesitation → fraudulent practices → further pressure from the sales department. This is a "self-reinforcing R- loop."

PGI's intervention in the B loop (balancing) enforces the following: Without evidence, subsequent processes cannot proceed → technical decisions cannot be overwritten by management → management accountability is recorded → the connection point is closed with three-party approval. This "forces a halt" to the negative vicious cycle (R-).

After implementing PGI, a positive virtuous cycle (R+) is naturally generated: improved integrity of evidence → increased transparency → increased trust → improved review quality → eliminated incentives for fraud.

IV. APPLICATION EXAMPLE

Below, we apply the proposed method to the case of the Chubu Electric Power Hamaoka Nuclear Power Plant.

A. The conflict in the Hamaoka Nuclear Power Plant

The fraud incident that occurred at Chubu Electric Power's Hamaoka Nuclear Power Plant [1] is outlined below.

The misconduct uncovered at Chubu Electric Power's Hamaoka Nuclear Power Plant revealed deep structural problems in the organization's nuclear division rather than a series of isolated technical errors. The core incident involved the inappropriate selection and manipulation of seismic ground-motion data used for regulatory safety reviews of Hamaoka Units 3 and 4. Japan's Nuclear Regulation Authority (NRA) judged the data to be unreliable and suspended the plant's safety review, prompting a broader investigation into the organization's internal processes.

At the heart of the issue was a structural conflict between two key divisions: the Civil Engineering Division, responsible for seismic modeling and ground-motion analysis, and the Nuclear Power Headquarters, which managed regulatory strategy and schedule commitments. Under pressure to accelerate safety review progress, the headquarters implicitly pushed for data that would avoid costly redesign requirements. Meanwhile, engineers in the Civil Engineering Division struggled to maintain scientific rigor under tightening deadlines. This misalignment created an environment in which "acceptable" outputs were prioritized over technically justified results.

The resulting misconduct—selecting seismic waveforms that favored regulatory approval, insufficient documentation of analytical reasoning, and inadequate internal review—was not simply a lapse in judgement but a manifestation of systemic weaknesses. These included poor process control at the interface between divisions, erosion of technical independence, and insufficient governance safeguards to ensure transparency and traceability.

The Hamaoka case illustrates how organizational pressure, fragmented authority, and inadequate process integrity can combine to compromise nuclear safety. It also demonstrates the need for a governance framework in which

engineering, management, and executive oversight align structurally to prevent the recurrence of such issues.

B. PGI introduction

The main processes in the Hamaoka Nuclear Power Plant incident were Civil Engineering Department analysis, review of analysis results, and preparation of submission documents. The connection points for these three processes are Civil Engineering Department analysis → PGI connection point 1 → review → PGI connection point 2 → preparation of submission documents → PGI connection point 3.

The EMG invariants for these three processes are shown below for connection point 1 (Civil Engineering Department analysis → review) and connection point 2 (review → preparation of submission documents).

[Connection point 1]

Precondition 1 is the first process, so there is no preceding evidence, but the following 3 EMG conditions must be met: E: Analysis basis and parameters are fully recorded and their validity confirmed. M: Analysis procedures are followed, and review inputs are complete. G: Transparency is ensured, and tampering is denied.

Execution evidence 1 consists of the analysis log, waveform selection reasons, parameter history, and technical supervisor approval log.

Postcondition 1 (EMG confirmation) is: E: The technical supervisor confirms that the analysis results are usable for review. M: The management supervisor confirms that the process transitions are as per procedure. G: The person responsible for governance confirms that the evidence is accountable to external parties. This allows the review process to officially begin.

[Connection Point 2]

Precondition 2 is the prerequisite: the execution evidence from Connection Point 1 is complete, and the following 3 EMG conditions are met: E: All technical issues have been addressed. M: Review and approval records are complete, and procedure compliance has been confirmed. G: The decision-making process is transparent and arbitrariness is eliminated.

Execution Evidence 2 consists of review minutes, a history of issues and responses, an approval log (E/M/G), and reference document records.

Postcondition 3 (EMG confirmation) is the following: E: The review results are technically finalized and cannot be overwritten. M: The format and content are confirmed to be correct for input to the next process. G: Accountability to regulatory authorities is confirmed.

This ensures complete evidence of the waveform selection in the Civil Engineering Department analysis. Furthermore, arbitrary changes by headquarters are no longer possible, automatically requiring three-party approval. Finally, the conflict at the connection point disappears. As a structural consequence of this, it is clear that the Hamaoka Nuclear Power Plant fraud would not have been possible under the PGI structure.

V. DISCUSSION

A. Novelty

The novelty of this proposal lies in its theorization of "control at process junctions," a concept that traditional safety engineering, quality assurance, and organizational control have been unable to fully address. PGI requires the existence of a unique execution trail A_p for each process and the EMG invariant EMG_p, which simultaneously establishes Engineering, Management, and Governance. This requirement is applied to all process junctions. This ensures an unbroken causal chain between processes, making it impossible for any process to proceed without referencing the authentic trail of the previous process. While traditional assurance cases are merely a documentation method for explaining safety and subject to retroactive additions and modifications, PGI structurally guarantees the generation of the trail itself, creating an assurance infrastructure that allows assurance cases to be generated whenever needed. Furthermore, by imposing invariants at the "natural joints"—the boundaries between technology, management, and business—PGI structurally prevents organizational conflicts and fraud, which are difficult to address with traditional business process design methods. PGI connection points consist of a trinity of "pre-conditions," "execution trail," and "post-conditions (EMG confirmation)," so if any one of these is missing, the connection point opens, preventing it from becoming an entry point for fraud.

B. Effectiveness

By applying this proposal to the Hamaoka incident, we demonstrated that the Hamaoka nuclear power plant fraud would not have occurred under the PGI structure. This result confirmed the effectiveness of this proposal. Because fraud and tampering are most likely to occur at the end of a process, the postcondition EMG serves as the final defense. Since the "validity of the result" cannot be determined from the execution trail alone, the result is confirmed through three-party approval of the EMG. Ensuring agreement among the three parties prevents the lack of responsibility and conflicts that occurred in the Hamaoka incident.

The existence of an execution trail Ap for every process p and the establishment of the EMG_p are conditions for the correctness of the PGII at all inter-process connection points.

The following two conditions must be met at a PGII connection point ($p \rightarrow q$):

(C1) The execution trail A_p of the predecessor process p exists.

(C2) The EMG invariant EMG_p is satisfied for that process p.

Unless these two conditions are met, the successor process q cannot start.

PGI ($\rightarrow q$) = $A_p \wedge EMG_p$

For a set of processes $P = \{p_1, p_2, p_3, \ldots\}$, PGI functions as a whole if the evidence and invariants hold for all processes.

$\forall p \in P$, $A_p \wedge EMG_p$) establishes PGII at all connection points ($p \rightarrow q$).

PGI ensures a "causal chain" because a subsequent process cannot begin without the evidence of the preceding process. PGI ensures process quality because a process cannot be completed without the necessary technology (E), management (M), and governance (G). Furthermore, A_p and EMG_p are required at each connection point. The absence of any one of these breaks the chain. Therefore, PGI can prevent fraud at the "connection points."

C. Assurance case Generation from PGI

PGI automatically records the following records for each process execution: the preceding process execution trail, invariant verification record, EMG three-party approval log, and process completion record (execution trail). If a deviation is detected under any of these conditions, the process cannot be completed. These records can be combined and formalized to form an Assurance Case. If the process cannot be completed, an Assurance Case cannot be created. Therefore, rather than creating Assurance Cases from business process diagrams, PGI allows Assurance Cases to emerge naturally. Assurance Cases are descriptions that explain process safety, and PGI is the generation mechanism that structurally guarantees process safety. Therefore, with PGI in place, Assurance Cases can be generated at any time, eliminating the need for separate preparation. Furthermore, this creates a new form of assurance: authentic, immutable, and tamper-proof, rather than retroactive. Previously, to demonstrate the safety of a system, the reasons for its safety had to be presented in written Assurance Cases. However, PGI inevitably leaves behind evidence that the process can only be safe, so the log naturally becomes an Assurance Case, providing structural assurance. Figure 2 shows the Assurance Case generated using PGI and EMG.

D. Comparison of PGI and BPR/TOC/BPR/TPS/FRAM

Because Business Process Modeling (BPM) is a representation model of business processes, it cannot handle the authenticity of process execution or the control of departmental boundaries, and its resistance to organizational fraud and inter-departmental conflicts is low.

TOC looks at constraints but does not address control structures. BPR is reform-oriented but has a high risk of destroying control processes. TPS is strong in on-site improvements but can sacrifice safety processes if misused. FRAM is strong in accident analysis but does not address system design.

Table I summarizes the fraud resistance and inter-organizational conflict resistance of conventional methods.

PGI is structurally incapable of fraud, making it highly resistant to fraud. Furthermore, PGI's resistance to organizational conflicts is high because the EMG invariant structurally seals off departmental conflicts.

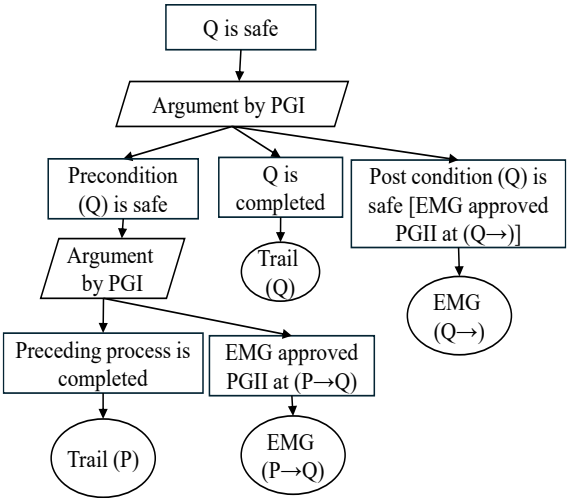


Figure 2. Assurance case generated from PGI structure

TABLE I. ISSUES OF CONBENTIONAL METHODS

Method	Resistance to fraud	Tolerance for organizational conflict
BPM	Low (cannot structurally prevent fraud)	Low (cannot handle the control of departmental boundaries)
TOC	Medium *(if constraint management is correct)*	Medium (depending on shared understanding of constraints)
BPR	Low (disruptive reforms tend to undermine safety processes)	Low (concentrated power, resistance, friction)
TPS	Medium to low (muda-elimination measures are prone to misuse)	Medium (emphasis on the field, potential for upstream/downstream conflict)
FRAM	Medium (potential for overemphasis on the field and upstream/downstream conflicts)	Medium (analysis-focused, weak control)
DPD	Medium (exceptions can be detected, but response is a challenge)	Medium (analysis-focused, weak control)

E. Achieving EMG Traceability

By recording Ep, Mp, and Gp for business process p, PGI's function as a "defect prevention measure" is traceable.

Forward tracing allows us to prove how E (technical truth) was accepted by M (management) and reflected in G (management) decision-making. Backward tracing, when fraud or an accident is detected, allows us to identify which "abnormal attribute value" in E was the cause, through the G layer's judgment and the M layer's verification process. Furthermore, traceability allows us to isolate the source of the inconsistency and initiate repairs by tracing back from the final state in which the connection points were normal.

F. The Limitations of PGI

The limitations of PGI lie not in theory, but in organizational resistance, as outlined below.

- Transparency makes responsibility visible
- Vested interests collapse
- Organizational psychology prioritizes short-term profits and rejects PGI
- Partial application of PGI creates loopholes and is counterproductive

This paper qualitatively clarifies the effectiveness of PGI using a single past case of misconduct. However, to confirm the effectiveness of PGI, it is necessary not only to apply it to other cases but also to quantitatively evaluate factors such as the effort required for process design.

VI. Conclusion And Future Work

PGI is the first control theory to integrate engineering, management, and governance, and it makes fraud and conflict "structurally impossible to occur." It also clarifies that fraud is not an "individual problem," but a structural problem resulting from a lack of control integrity. Furthermore, PGI can contribute to the design of controls in high-reliability organizations to realize process control that stops vicious cycles and creates virtuous cycles.

For more than three decades, business-process research—ranging from BPR and workflow engineering to BPM, Lean, Six Sigma, and safety-assurance methodologies—has attempted to improve organizational performance by optimizing process design, documentation, and continuous improvement. Despite these advances, a deep structural limitation remained largely unaddressed: none of the existing approaches systematically ensured the integrity of process execution across organizational boundaries. They improved how processes *should* operate but provided no mechanism to guarantee how they *actually* operate in real conditions marked by pressure, shortcuts, inconsistent governance, and interdepartmental conflicts.

PGI directly overcomes this foundational weakness. PGI introduces two structural concepts absent from prior research:

Execution Trace (A_p) for every process p, and a triadic governance invariant (EMG_p)—Engineering, Management, Governance—that must be simultaneously satisfied at every process interface.

By embedding these invariants at all junctions between processes, PGI ensures that no process can proceed unless the preceding one is both *traceable* and *governance-complete*. Prior business theories focused on internal tasks; PGI focuses on the process junctions, the very places where most organizational failures occur.

Traditional business-process research also assumed that compliance and quality could be verified through documentation, audits, or post-hoc assessments. PGI replaces this fragile assumption with a structural guarantee: the process itself produces tamper-proof execution traces that constitute genuine evidence of correct performance. This allows PGI to generate an Assurance Case for any process—solving a problem that BPR, TPS, and safety methodologies could only address retrospectively and incompletely.

Another long-standing limitation in business-process literature was the lack of an integrated view of technical accuracy, managerial discipline, and governance legitimacy. PGI unifies these into a single invariant (EMG), ensuring that engineering rigor cannot be overridden by managerial pressure, nor can governance requirements be satisfied through superficial documentation. This resolves the endemic misalignment between functional silos that prior research could describe but prevent.

In summary, PGI is not merely an extension of BPR or BPM but a conceptual leap: the first framework that guarantees process integrity by design, ensures cross-boundary coherence, auto-produces trustworthy assurance, and structurally prevents organizational drift and misconduct. It addresses precisely what decades of business-process research left unresolved.

References

[1] The Japan Daily, Data Manipulation at Hamaoka Nuclear Plant Sparks Calls for Policy Reform,7 Jan. 2026, [Online]. Available from: https://japandaily.jp/data-manipulation-at-hamaoka-nuclear-plant-sparks-calls-for-policy-reform/ [retrieved: Apr., 2026]

[2] G. D. Isbasoiu and D. Volosevici, "Organizational Determinants of Unsafe Acts: An Exploratory Study in Refinery Maintenance Operations", Safety 2025, vol.11, no.4, pp. 102, [Online]. Available from: https://doi.org/10.3390/safety11040102 [retrieved: Apr., 2026]

[3] H. Goldratt and J. Cox, "*The Goal: a process of ongoing improvement,"* (3rd rev.). Great Barrington, MA: North River Press, 2004.

[4] M. Hammer and J. Champy, "Reengineering the Corporation-A Manifesto for Business Revolution," Harper Business, 1993.

[5] T. Ohno, "Toyota production system: beyond large-scale production," Cambridge, Mass.: Productivity Press 1988.

[6] E. Hollnagel, "FRAM - the Functional Resonance Analysis Method: Modelling Complex Socio-Technical Systems." CRC Press, 2012.

[7] S. Sasaki, "Self-process completion - Quality is built in the process," JSQC selection, Japan Society for Quality Control, 2014. (in Japanese)

[8] S. Yamamoto, "Business Process Completeness," eKNOW, pp. 24-28, 2024, [Online]. Available from: https://www.proceedings.com/content/076/076892webtoc.pdf. [Online] [retrieved: Apr., 2026]

[9] S. Yamamoto, "Defect Prevention Review by Process Relationship Matrix," eKNOW, 2025, [Online]. Available from: https://www.iaria.org/conferences2025/fileseKNOW25/eKNOW_60014.pdf [retrieved: Apr., 2026]

[10] T. Kelly and J. McDermid, "Safety Case Construction and Reuse using Patterns," University of York, 1997.

[11] ISO: ISO26262 Functional Safety, ISO, 2011

[12] P. Senge, "The Fifth Discipline: The Art & Practice of The Learning Organization," Second edition, Random House Books, 2006.

Generative AI as a Good Questioner:A RAG-Based Question Transformation Approach for Eliciting Tacit Knowledge in Software Development Organizations

Soonnam Shin
Graduate School of System Design and Management
Keio University
Yokohama, Japan
e-mail: shinsoonnam@keio.jp

Takahiro Yakoh
Graduate School of System Design and Management
Keio University
Yokohama, Japan
e-mail: yakoh@keio.jp

***Abstract*—Knowledge in software development organizations often depends on tacit knowledge possessed by experts, leading to challenges, such as insufficient documentation and knowledge silos. Furthermore, changes in work environments driven by remote work and the widespread adoption of generative Artificial Intelligence (AI) have made it increasingly difficult to externalize such knowledge through conventional question–answer processes. Rule-based Knowledge Management Systems (KMS) also struggle to prevent knowledge obsolescence and sustain continuous knowledge update. This study proposes a question transformation approach for eliciting tacit knowledge through an interactive knowledge sharing system that leverages generative AI and Retrieval-Augmented Generation (RAG). By positioning generative AI as an effective "good questioner," the proposed method detects knowledge gaps in the knowledge base and reformulates user queries into exploratory questions designed to elicit tacit knowledge from domain experts. The elicited knowledge is then structured and formalized, enabling iterative updates and reinforcement of the knowledge base. A QA chatbot mediating between knowledge seekers and knowledge providers was implemented and evaluated using a real-world dataset from an operational system within a DevOps-oriented software development organization. The results demonstrate that the proposed approach effectively supports sustainable and efficient knowledge sharing in software development environments.**

Keywords-Knowledge Sharing; Tacit Knowledge; RAG (Retrieval-Augmented Generation); Generative AI; Question Transformation.

I. INTRODUCTION

This section outlines the background of the study and clarifies the research problem and objectives addressed in this paper.

A. Problem Definition

The rise of generative Artificial Intelligence (AI) has prompted a reconsideration of how knowledge is utilized and shared within organizations. While individual use of generative AI effectively supports personal problem-solving and idea generation, the insights and learning outcomes derived from such use tend to accumulate internally within individuals. As a result, they are less likely to be systematically shared at the organizational level, increasing the risk that they remain as tacit knowledge siloed to individuals. In software development, tacit knowledge, such as design rationales, experience-based precautions, and troubleshooting know-how—tends to concentrate among experts.

This study focuses on knowledge gaps and tacit knowledge in software development that are not explicitly documented in formalized artifacts, such as specifications and manuals. Conventional Knowledge Management Systems (KMS) rely heavily on rule-based structuring and manual maintenance, which require substantial operational costs and hinder sustainable management.

At the same time, research focusing specifically on the design of questions to elicit tacit knowledge from humans remains limited. Generative AI has primarily been treated as an "answering system," while its potential as a "skillful questioner" has not been sufficiently explored.

B. Research Objective and Contributions

To address these challenges, this study aims to design a RAG-based knowledge sharing system with a conversational interface powered by generative AI for software development organizations, and to verify the effectiveness of a question transformation method that actively elicits tacit knowledge for the continuous updating of the knowledge base.

This study makes three primary contributions:

- Theoretical Contribution: It reconceptualizes generative AI not merely as an answer generator but as a questioner and mediator, proposing a new role for AI in the organizational knowledge creation process.
- Methodological Contribution: It proposes, a question transformation method specifically designed for tacit knowledge elicitation.
- Empirical Contribution: It implements and evaluates a RAG-based knowledge sharing system within a DevOps-oriented software development organization, demonstrating its effectiveness.

Through these contributions, this research provides a novel perspective on the formalization of tacit knowledge from the viewpoint of question transformation in the design of generative AI–enabled knowledge sharing systems.

The remainder of this paper is organized as follows. Section II reviews related work. Section III presents the

 ISBN: 978-1-68558-387-3

proposed method and system implementation. Section IV describes the evaluation and results. Section V concludes the paper and discusses limitations and future work.

II. Related Work

This section reviews related research to position the present study within the existing body of knowledge. It first examines Knowledge Management approaches, followed by studies on tacit knowledge elicitation from experts and concludes with a review of research on RAG-based information augmentation and question transformation methods.

A. Knowledge Management

A prior study on ontology-based Knowledge Management tools for organizational knowledge sharing [1] provides a comprehensive review of case studies and implementation examples of ontology-based KMS. The study evaluates them using ten comparative criteria: motivation, domain, knowledge sources, types of knowledge, knowledge extraction processes, knowledge input processes, knowledge retrieval processes, knowledge sharing technologies, sources of ontology components, and ontology methodologies. However, challenges remain in addressing knowledge sharing problems that require person-to-person knowledge transfer, particularly in terms of technical approaches for extracting and retrieving knowledge from implicit sources across diverse knowledge domains. While ontology-based Knowledge Management tools are effective in organizing explicit knowledge, they face difficulties in extracting and updating tacit knowledge.

B. Tacit Knowledge Elicitation from Experts

To improve the collection of tacit knowledge in KMS, a prior study [3] proposes a storytelling-based approach for knowledge sharing. Compared to conventional interview-based or video-based methods, storytelling is suggested to reduce psychological resistance among knowledge holders and facilitate more natural knowledge sharing.

However, the study lacks large-scale experimental validation and quantitative evaluation, remaining issues regarding its practical effectiveness. Moreover, storytelling content tends to be subjective and unstructured, making knowledge standardization and systematic integration into knowledge systems difficult.

A case study on the use of Large Language Models (LLMs) in manufacturing environments [2] demonstrated their effectiveness for knowledge management and information retrieval support; however, mechanisms for eliciting tacit knowledge from domain experts were not sufficiently discussed.

Prior research has identified significant barriers to tacit knowledge sharing in software development teams, including team culture, trust, communication, and team dispersion [7]. These barriers limit access to tacit knowledge required for socio-technical tasks and contribute to project failures. However, existing studies primarily focus on identifying these barriers, with limited attention to mechanisms for actively eliciting tacit knowledge.

C. Retrieval-Augmented Generation (RAG)

A prior study analyzing the operational and validation challenges of systems based on RAG [4] systematically identifies seven Failure Points (FP) in RAG-based system design [5]: In particular, FP1 Missing Content highlights the problem of insufficient content caused by the absence of mechanisms for continuously maintaining and updating the knowledge base.

D. Question Transformation

A prior study on prompts for transforming ambiguous questions into more specific queries [6] proposes a novel prompting method called Ambiguity Type–Chain of Thought (AT-CoT). This approach enables LLMs to better understand user queries by identifying the type of ambiguity involved and generating clarification questions accordingly.

While existing research primarily focuses on query augmented techniques aimed at improving answer accuracy, relatively limited attention has been paid to question generation methods designed to elicit knowledge from humans (i.e., domain experts).

III. System Architecture and Implementation

The distinguishing feature of this study lies in proposing a knowledge sharing process centered on AI-mediated human interaction and question transformation for tacit knowledge elicitation. While conventional knowledge sharing and QA systems mainly focus on retrieving existing knowledge and generating answers, the proposed system extends this process by detecting knowledge gaps and transforming them into opportunities for tacit knowledge acquisition.

A. Knowledge Sharing Model

Figure 1 illustrates the Knowledge Sharing Model using the RAG-Based Question Transformation Method. The model consists of three interrelated flows: (A) Knowledge Sharing, (B) Knowledge Elicitation, and (C) Knowledge Acquisition.

Flow A, Knowledge Sharing, corresponds to a conventional QA process over a knowledge base. When a knowledge seeker submits a question, the system retrieves relevant information from the knowledge base through a RAG pipeline and generates a response in natural language.

Flow B, Knowledge Elicitation, is activated when the system detects that the knowledge base does not contain sufficient information to provide a reliable answer. In such cases, the AI agent GapNavigator generates exploratory questions through Exploratory Question Transformation (EQT). These questions are presented to a knowledge provider in order to elicit tacit knowledge that has not been explicitly documented.

Flow C, Knowledge Acquisition, structures and formalizes the elicited knowledge and incrementally integrates it into the knowledge base. Through this process, knowledge that was previously unavailable in routine QA interactions can be accumulated and reused in future knowledge sharing.

This model extends conventional QA-based knowledge sharing by introducing an explicit mechanism for identifying missing knowledge and converting it into opportunities for knowledge elicitation and knowledge base reinforcement.

B. Knowledge Gap Detection in the RAG Pipeline

A central component of the proposed system is the detection of knowledge gaps within the RAG pipeline. In this study, a knowledge gap is defined as a state in which the system cannot provide a sufficiently reliable or complete answer based on the current knowledge base.

Knowledge gaps are identified based on the following conditions:

1) Retrieved documents do not contain sufficient information to answer the query;

2) The generated response indicates uncertainty or the absence of relevant information;

When one or more of these conditions are satisfied, the query is classified as a knowledge-gap query, and GapNavigator is activated. In this way, the proposed system does not terminate at retrieval failure, but instead transforms failure into an opportunity for tacit knowledge elicitation.

C. The Concept of "Generative AI as a Good Questioner" and EQT Method

This study positions generative AI not merely as an answer generator but as a "good questioner" that facilitates the externalization of tacit knowledge. When a knowledge gap is detected, the proposed EQT method reformulates the original query into structured exploratory questions for knowledge providers. EQT is implemented through prompt engineering rather than model retraining or fine-tuning.

EQT is designed to elicit tacit knowledge in software development organizations from five predefined perspectives: (1) design philosophy and decision rationale, (2) knowledge provider heuristics and practical know-how, (3) dependencies and impact scope, (4) exceptional cases and failure knowledge, and (5) implicit rules and assumptions.

The prompt design strategy consists of four elements: role instruction, contextual augmentation, transformation constraints, and output constraints. Specifically, the LLM is instructed to act as an assistant for tacit knowledge elicitation, to use the original user query and retrieved context as input, to avoid simple paraphrasing, and to generate specific and answerable exploratory questions for knowledge provider.

The transformation procedure is defined in Algorithm 1. First, the system receives the original query and the retrieved context from the RAG pipeline. Second, when the available information is judged insufficient, the query is classified as a knowledge-gap query. Third, the relevance of the five tacit knowledge perspectives is evaluated based on the query and context. Fourth, the candidate perspectives are ranked, and up to the top three are selected. Finally, one exploratory question is generated for each selected perspective and presented to the knowledge provider.

The prioritization of perspectives is based on the semantic relevance to the original query, contextual relevance to the retrieved documents, the degree to which the missing information depends on knowledge provider judgment or experience, and the expected usefulness for knowledge base update. By explicitly constraining the transformation process in this way, EQT provides a transparent and reproducible mechanism for AI-mediated tacit knowledge elicitation while keeping the cognitive burden on knowledge providers manageable.

D. Implementation Overview

The system was implemented as a RAG-based chatbot using the generative AI development platform Dify. The knowledge base was constructed from documents related to an operational Intellectual Property Management System (IPMS) within a software development organization. The chatbot processes user questions through the RAG pipeline, while GapNavigator monitors the output and invokes EQT when a knowledge gap is detected.

Algorithm 1. Exploratory Question Transformation (EQT)

```
Input:  Original query Q, retrieved context C, tacit knowledge perspectives P
Output: Exploratory questions E
Receive Q and C from the RAG pipeline
if C is insufficient then
    Classify Q as a knowledge-gap query
    for each p ∈ P do evaluate r(p) based on Q and C
    Rank perspectives by r(p) and select up to the top three P′
    for each p ∈ P′ do generate one exploratory question e based on Q, C, and p
    Add e to E
    Present E to the knowledge provider
end if
```

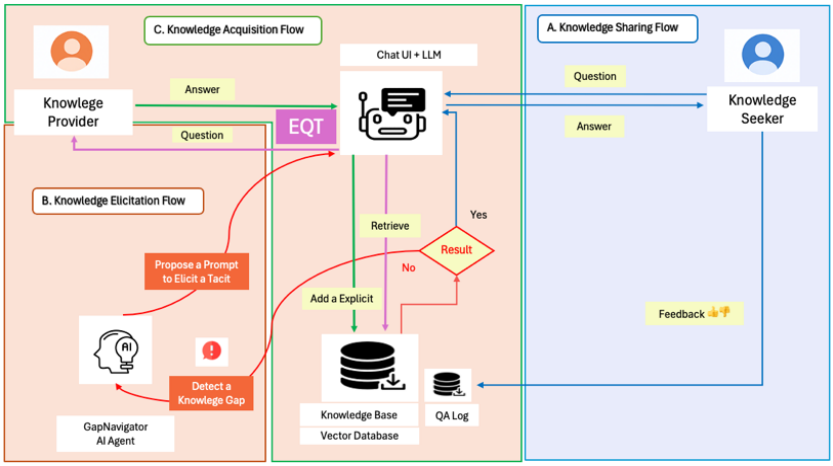


Figure 1. Knowledge Sharing Model with RAG-Based Question Transformation: from retrieval-based answering to knowledge gap detection, tacit knowledge elicitation, and iterative knowledge acquisition.

By integrating retrieval, gap detection, and tacit knowledge elicitation into a single framework, the proposed system supports not only access to existing knowledge but also the continuous acquisition and formalization of tacit knowledge for sustainable organizational knowledge sharing.

The current implementation is intended as a prototype to verify the feasibility of AI-mediated tacit knowledge elicitation in a software development context. Accordingly, the behavior of knowledge gap detection and EQT is influenced by the quality of the retrieved context and the prompt design. Therefore, the present implementation should be regarded as a design instantiation for feasibility validation rather than a universally optimized architecture.

IV. Evaluation and Experiment

This section evaluates the effectiveness of the proposed mechanism and knowledge sharing system in relation to the research objectives defined in Section I.

A. Experimental Setup

The evaluation was conducted using the proposed RAG-based chatbot implemented on the Dify platform. For the evaluation, the dataset was constructed from publicly available user documentation, including manuals, related to the IPMS, a real-world operational system in which the author is directly involved as a developer. These documents were converted into a structured knowledge base for the experiment

For the RAG configuration, the embedding model text-embedding-3-large was used, with the top-K retrieval parameter set to 2.

Using the IPMS dataset, two model conditions were compared: (1) a baseline model that generated general follow-up questions when a knowledge gap was detected, and (2) an EQT model that generated structured exploratory questions based on predefined tacit knowledge perspectives. Both models shared the same RAG pipeline and LLM configuration, while differing in the strategy used for question generation after knowledge gap detection. Both models employed the same LLM, gpt-5-chat-latest, with the temperature parameter set to 0.7 and the maximum token length set to 512. Table 1 summarizes the main differences in prompt design between the baseline and EQT models.

TABLE 1. COMPARISON OF PROMPT DESIGN BETWEEN THE BASELINE AND EQT MODELS.

Item	Baseline Model	EQT Model
Purpose	Generate general follow-up questions for knowledge-gap queries	Generate structured exploratory questions for eliciting tacit knowledge
Question generation style	General follow-up questioning	Exploratory questioning guided by tacit knowledge perspectives
Knowledge orientation	Focus on obtaining missing explicit information	Focus on eliciting tacit knowledge that is reusable and formalizable
Tacit knowledge perspectives	Not explicitly specified	Five predefined perspectives: design rationale, expert know-how, dependencies and impact, exceptions and failures, and implicit rules and assumptions
Multi-element query handling	Organize the content before generating questions	Break down the content step by step and generate exploratory questions

In the EQT model, the system prompt was configured to elicit tacit knowledge from five predefined perspectives commonly observed in software development organizations.

The evaluation was conducted within a DevOps-oriented software development organization. A total of 13 participants were recruited, representing diverse organizational roles, including system development, operations and support, sales, and management. The participants ranged from novice employees with less than two years of experience to senior professionals with more than 21 years of experience.

Participants entered the same original questions into both model conditions using both predefined and free-form queries and compared the generated outputs without being informed of the model names during the evaluation.

B. Metrics

The generated questions under the two model conditions were evaluated using a five-point Likert scale based on two criteria: (1) clarity in identifying and articulating knowledge gaps, and (2) effectiveness in eliciting tacit knowledge. In addition, participants conducted a comparative assessment to determine which model generated questions that were more effective in eliciting tacit knowledge.

1) Tacit Knowledge (TK) Elicitation Rate

The TK Elicitation Rate was used as an auxiliary metric to assess whether the exploratory questions generated by EQT were answerable by knowledge providers in practice. In this study, a generated question was regarded as valid if an answer could be provided by a participant acting as a knowledge provider. The metric was therefore based on actual response behavior rather than on a post hoc judgment of question quality.

The TK Elicitation Rate is defined as the proportion of answerable EQT-generated questions among all exploratory questions generated from knowledge-gap queries, as shown in (1). Because the EQT model generated up to three exploratory questions for each original knowledge-gap query, the denominator is defined as the number of original knowledge-gap queries multiplied by three.

The validity of generated questions was assessed through actual responses provided by participants according to their areas of expertise. When a participant was not able to answer a question because it fell outside their domain, the question was referred to another participant with relevant domain knowledge. This metric should therefore be interpreted as an operational indicator of the practical answerability of EQT-generated elicitation questions, rather than as a direct measurement of tacit knowledge itself. Therefore, no separate rubric table was used for this metric. In addition, no inter-rater reliability analysis was conducted, because validity was not determined by post hoc labeling across multiple evaluators, but by actual answerability in practice.

$$TK\ Elicitation\ Rate = \frac{Number\ of\ valid\ Questions}{3 \times (Number\ of\ Generated\ Questions)} \quad (1)$$

C. Results

Figure 2 presents the results of the model comparison using the Wilcoxon signed-rank test. For Knowledge Gap Clarifying Level, the EQT model showed slightly higher scores than the baseline model, although the difference was not statistically significant. For Tacit Knowledge Elicitation Level, the EQT model showed clearly higher scores, with a statistically significant difference ($p = 0.002$). In the direct comparative assessment, 93.3% of participants selected the EQT model as more effective for tacit knowledge elicitation ($p = 0.0034$).

These results suggest that the proposed EQT method more effectively generated questions that were perceived as useful for eliciting tacit knowledge from knowledge providers. The findings support the validity of transforming knowledge-gap queries into exploratory questions rather than treating retrieval failure as the end point of the interaction.

Using the EQT model, 102 exploratory questions were generated from 34 original knowledge-gap queries. Among these, 99 questions received actual responses from participants and were therefore regarded as valid, resulting in a TK Elicitation Rate of 97.1%. This result indicates that most EQT-generated questions were practically answerable by knowledge providers in the present experimental setting.

This metric should be interpreted with caution. It reflects the practical answerability of EQT-generated elicitation questions, rather than the completeness or quality of tacit knowledge eventually obtained. Taken together, the statistical comparison and the TK Elicitation Rate suggest that EQT was effective as a structured question transformation mechanism for tacit knowledge elicitation in the present study.

The results should also be interpreted in light of the limited experimental scope, as the evaluation was conducted in a single organization with a domain-specific dataset and a relatively small number of participants.

V. Conclusion and Future Work

This section summarizes the main findings of this study, discusses its limitations, and outlines directions for future research.

A. Conclusion

While existing studies primarily focus on knowledge retrieval and QA systems, this study introduces a question transformation approach that positions generative AI as a skillful questioner for eliciting and formalizing tacit knowledge.

Using a real-world dataset from an operational system, the experiments showed that knowledge providers could externalize tacit knowledge through EQT-generated exploratory questions, thereby strengthening the knowledge base and supporting organizational knowledge sharing.

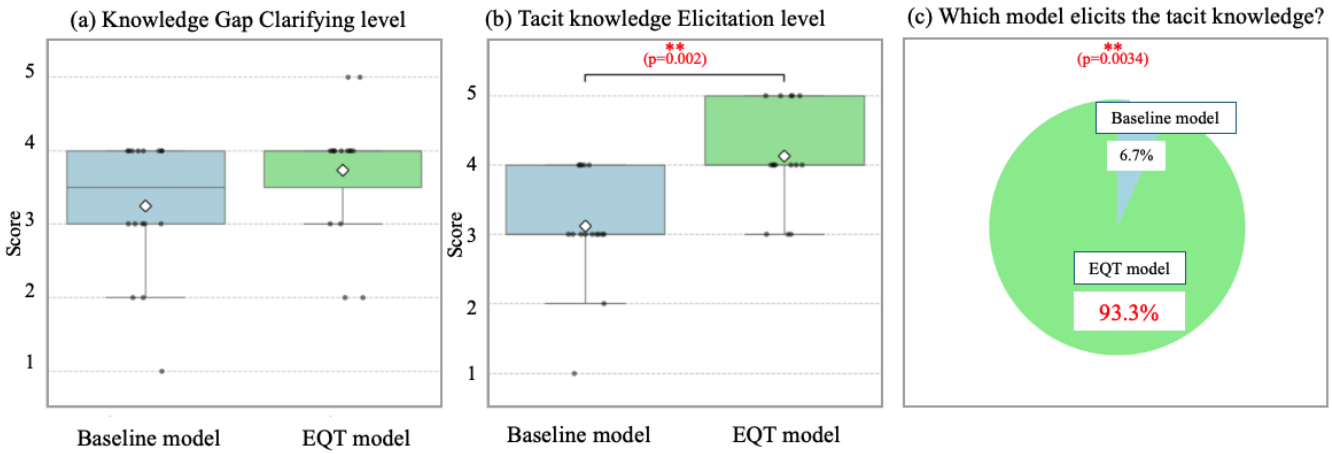


Figure 2. Model comparison: Wilcoxon signed-rank test.

The significance of this study lies in showing that tacit knowledge in software development can be externalized through ordinary Q&A interactions without relying solely on costly, non-routine methods such as interviews or workshops. Thus, the study contributes to knowledge sharing system design by advancing tacit knowledge formalization from the perspective of question transformation.

B. *Limitations*

This study does not aim to comprehensively examine all possibilities of generative AI–enabled knowledge sharing, nor does it evaluate the intrinsic performance of generative AI models or RAG architectures themselves. Accordingly, quantitative comparisons of retrieval accuracy, answer generation performance, and differences across LLM models were outside the scope of this research.

The proposed system focuses on tacit knowledge that is linguistically expressible but not spontaneously articulated; embodied skills and highly intuitive expertise were outside the scope of this study.

C. *Future Work*

Generative AI has the potential to provide a dynamic and interactive platform for managing knowledge sharing within organizations. However, careful consideration must be given to operational design and the potential burden placed on knowledge providers. Determining the appropriate degree of AI-agent intervention represents an important area for future investigation. In addition, organizational challenges remain, including how to evaluate and recognize the contributions of knowledge providers within AI-mediated knowledge sharing processes.

As a direction for future work, expanding the applicability of the proposed method is an important priority. By integrating knowledge sources of varying contents and formats, such as design review records, incident response logs, and chat histories—it may become possible to complementarily elicit a broader range of tacit knowledge.

REFERENCES

[1] M. A. Osman, S. A. M. Noah, and S. Saad, “Ontology-based knowledge management tools for knowledge sharing in organization—a review”, IEEE access, vol. 10, pp. 43267-43283,2022.

[2] S. K. Freire et al., “Knowledge sharing in manufacturing using LLM-powered tools: user study and model benchmarking.”, Frontiers in Artificial intelligence, vol. 7, 1293084, 2024.

[3] N. Shaw and P. Liu, “A knowledge management system (KMS) using a storytelling-based approach to collect tacit knowledge.”, IEEE, pp. 1-6, 2016.

[4] P. Lewis et al., “Retrieval- augmented generation for knowledge-intensive nlp tasks.”, Advances in neural information processing systems, vol. 33, pp. 9459-9474, 2020.

[5] S. Barnett, S. Kurniawan, S. Thudumu, Z. Brannelly, and M. Abdelrazek. “Seven failure points when engineering a retrieval augmented generation system.”, In Proceedings of the IEEE/ACM 3rd International Conference on AI Engineering-Software Engineering for AI, pp. 194- 199, 2024.

[6] A. Tang, I. Soulier, and V. Guigue, “Clarifying ambiguities: on the role of ambiguity types in prompting methods for clarification generation.”, In Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 20-30, 2025.

[7] E. Mtsweni and K. Gorejena, “Team barriers to tacit knowledge sharing in software development project teams.”, Electronic Journal of Knowledge Management, vol. 21(1), pp. 59-72, 2023.

Collaborative Ontology Development Approach for Data Spaces

Erik Paul Konietzko, Sonika Gogineni, Kai Lindow
Intelligent Integration
Fraunhofer Institute for Production Systems and Design Technology IPK
Berlin, Germany
e-mail: erik.paul.konietzko@ipk.fraunhofer.de
e-mail: sonika.gogineni@ipk.fraunhofer.de
e-mail: kai.lindow@ipk.fraunhofer.de

Abstract — **The increasing demand for reliable, secure, and sovereign cross-organizational data exchange has led to the emergence of data spaces. Effective collaboration hinges on shared semantic models, positioning collaborative ontology design as a critical research area. However, existing literature on approaches is limited and often overlooks the combination of essential factors for successful collaboration in engineering contexts, such as processes, stakeholders, and artifacts. This paper introduces the Collaborative Ontology Development approach for data Spaces (CODeS approach) considering these factors. The paper concludes with a discussion and proposes the next steps for evaluation.**

Keywords-Ontology alignment; data space; collaborative ontology development approach.

I. INTRODUCTION

Collaborative ontology development is increasingly relevant in the context of knowledge management and information sharing across diverse domains. Especially the conception of semantic web technologies was based on the advantages of sharing and reusing ontologies across various domains. The collaborative development facilitates the integration and alignment of heterogeneous data from multiple perspectives and stakeholders ensuring that ontologies are representative of a broader range of expertise and use cases [1]. The collaborative approach not only enhances the quality and usability of the ontologies but supports in lifecycle management of the ontologies and its applications, as it has verified inputs from various stakeholders working together as a team [2].

Over the last few years, the concept of data space has gained popularity and support to manage and integrate large, heterogeneous and distributed data sources over company boarders [3]. Data spaces are decentralized infrastructure which enable trustworthy, sovereign, and secure data exchange based on common principles and policies [4]. To manage heterogeneous data sources within such data spaces, the use of ontologies is becoming increasingly important, as they have proven to be beneficial for enhancing interoperability and integrating diverse data [4]. Consequently, they have already been incorporated into several initiatives, such as Catena-X or Gaia-X with more initiatives expected to follow [5].

Collaborative ontology development involves various processes, activities, artefacts, roles, and IT tools and technologies. Ontology engineering and management encompasses ontology requirements specification, implementation, evaluation, publication, evolution and maintenance [6][7]. These processes are made up of various activities, artefacts, roles, IT tools, and technologies. All these aspects need to be considered in the context of data space ontology development. There are very few publications which present approaches to deal with ontology engineering for data space involving various internal and external stakeholders [8][9]. These publications, however, do not focus on the aspects of harmonization of data models, of ontologies and of architectures, interactions between the stakeholders, change and dependency management and governance of the ontologies [5][8][10].

This publication aims to present the current state of the art for collaborative ontology development in data spaces, to derive the research challenges (Section II). This is followed by the presentation of the **C**ollaborative **O**ntology **De**velopment for Data **S**paces (CODeS) approach (Section III). The conclusion is documented in Section IV, and the outlook underscores the need for evaluation of the approach in practical projects. Furthermore, it highlights the importance of focused research and exploration of individual topics to strengthen the foundation of collaborative ontology design.

II. STATE OF THE ART: COLLABORATIVE ONTOLOGY DEVELOPMENT IN ENGINEERING DOMAINS

Semantically aligned ontologies play a pivotal role in the context of data spaces; however, significant research gaps persist across various related topics, both within and beyond data spaces. This section organizes and categorizes key topics essential for collaborative ontology development, emphasizing existing research while identifying areas requiring further investigation. The concluding subsection focuses on outlining research challenges to be addressed to support the progress of collaborative ontology development for data spaces.

A. Collaborative ontology development approaches

The authors of [2] present the evolution of ontology approaches over three generations. Early ontology engineering methodologies, such as [11], [12] emphasized

core activities including requirements analysis, conceptualization, implementation, evaluation, and maintenance. These methodologies assumed that formal domain knowledge specification precedes system development. In contrast, second-generation methodologies adopted a more iterative approach, integrating application-specific requirements into the requirements analysis phase and allowing for incremental releases of ontology versions to accommodate changing needs. A notable characteristic of these approaches is the clear division of responsibilities among domain experts, knowledge engineers, ontology engineers, and users, with engineers driving the process by gathering requirements, implementing them, testing ontologies, and managing their evolution. Examples include Methonotology [13] and OnToKnowledge [14][15]. The current third generation employs a participatory approach, emphasizing collaboration among a diverse group of contributors and providing technological support to enable non-experts to engage in ontology development activities beyond requirements of engineering. Several methodologies detail the collaborative engineering process for developing and maintaining ontologies in decentralized scenarios, with DILIGENT [16] and HCOME [17] being the most notable examples. However, these methodologies are limited in terms of concrete case study descriptions and associated technological support. In this context, approaches, such as Ontology Maturing [18], suggest and evaluate tool support based on degree of maturity and phase of ontology usage. Additionally, RapidOWL [19] presents a valuable set of guidelines that can inform the design of collaborative ontology engineering methodologies and help align existing methodologies with broader principles of agile engineering and rapid prototyping. In the Hozo approach an environment for distributed ontology development is described based on types of dependencies between ontologies and resulting change patterns, as well as collaborative implications [20]. This evolution in methodology underscores the ongoing need for adaptable frameworks to support effective ontology development in diverse and dynamic environments. However, these methodologies do not focus on the dynamics and challenges of collaborative ontology engineering specific to data spaces.

The authors of [9] present the AIME methodology for collaborative ontology development in data spaces. They focus on the data space challenges, integration of FAIR principles and various stakeholders. However, the identification and management of change and dependency management are not explicitly detailed. The authors of [21] focus on ontology-based data access for data spaces and do not provide a methodology for collaboratively engineering the ontology. In [22], the authors present agile-based collaborative steps for developing the information model for the international data space initiative. The model includes the following conceptual areas: digital resources, participants, roles, identities, usage of contracts / policies, metadata, and infrastructure processes. However, no explicit approach is defined as a detailed collaborative approach to develop ontologies in data spaces.

B. Identified research challenges

As identified in the preceding subsections, there are several research gaps in literature, which need to be addressed or further explored. The gaps have been grouped into five challenges based on the type of gap:

Challenge 1 (C1) – Aligned knowledge management: Managing knowledge across varied domains poses substantial challenges due to differences in conceptual understanding, workflows, data models, and contextual data interpretation. Effective knowledge integration requires harmonizing disparate perspectives, aligning processes, and coordinating multiple stakeholders [23]. A critical aspect of this effort involves developing unified ontology for the shared data space or establishing effective mechanisms to integrate existing ontologies with newly defined ones.

C2 - Access and governance in data spaces: Data spaces face challenges in access and governance, stemming from ambiguity around who holds the authority to define key elements, their roles within the ecosystem, and the rights they possess [8][9]. Participation is constrained by sector-based eligibility and closed governance, limiting cross-domain expertise and innovation. Opaque decision rights and unclear ownership of ontology elements deter external contributors and create bottlenecks.

C3 - Collaborative models in data spaces: The key challenge is to identify and implement collaborative structures that actively engage diverse participants with varying motivations. This requires a framework that aligns incentives, appreciates the complexities of collaboration, and clarifies the nature of shared responsibilities in defining data spaces and their semantics. There is a need to understand different types of open-source collaboration, as these models significantly impact participation incentives [24]. Commonly, a core team manages development, while other contributors suggest or implement changes — a structure that may limit broader engagement.

C4 - Harmonization of data models, ontologies and architectures: This challenge concerns the establishment of harmonized semantic annotations within the data space and with particular use cases it supports. Hence, there is a need for collaboration formats, mapping standard and operating procedures to integrate and improve interoperability [8].

C5 - Data value: Despite advancements in transaction metadata and data format specifications, data spaces face a critical challenge in the absence of shared, machine-readable semantics for exchanged data. Distribution mechanisms remain fragmented and proprietary [1].

The solution lies in developing methods, processes, and governance frameworks to collaboratively create and maintain shared, extensible ontology and semantic profiles. These must facilitate consistent interpretation and integration of heterogeneous data across data spaces and support incremental adoption.

III. CODeS Approach: Collaborative Ontology Development for Data Spaces approach

To address the previously identified challenges, this section outlines a structured approach built upon best practices identified in the literature and insights gained from practical experience in developing data spaces [25][26][27]. This methodology integrates established theoretical frameworks with lessons learned from real-world projects, for the development of a novel approach tailored to the complexities of data spaces. Figure 1 outlines the ten steps of CODeS for collaborative ontology development in data spaces. The steps are outlined in detail below:

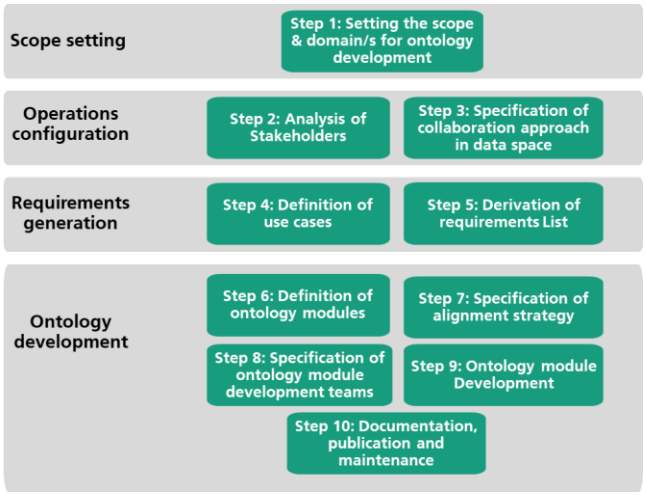


Figure 1. Visualization of CODeS approach steps.

A. Step 1: Setting the scope of ontology to be developed in data space

There are two main parts in this step. The first is to determine the domain/s the data space aims to address. The second is to identify the existing ontologies, semantic artefacts, governance frameworks and standards in the identified domain/s [5][9]. Selecting relevant state of the art for the data space with the existing stakeholders. This includes commonly used data, models, communication and certification mechanisms, infrastructure (protocols, interfaces, etc.) and tooling in the domains and their development environments.

B. Step 2: Stakeholder Analysis

Stakeholder analysis defines roles, responsibilities, and collaboration approaches through two sub-steps: (1) identifying and verifying stakeholder coverage across development areas, and (2) documenting assignments and commitments while acknowledging their evolution over time.

Figure 2 illustrates role distribution in ontology engineering within data spaces. Indirect roles (data space participants, domain experts) contribute content with varying involvement levels. Direct roles focus on ontology development across three levels.

Roles in data space ontology development define general semantics for data space basis or federation services at the most abstract level, with largely decoupled collaboration to individual dataspace communities.

Federation ontology development roles specify semantics for core services and standards (versioning, interfaces), requiring close collaboration with data space design teams.

Roles for domain ontology development collaborate with domain experts and participants, adapting to domain-specific engagement forms.

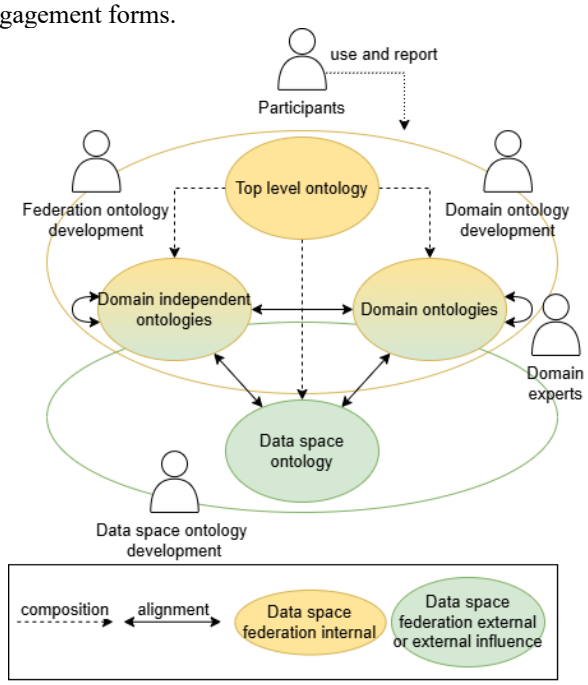


Figure 2. Semantic distribution and roles in data space setting.

Examples of stakeholders can be found in [2][9][28]. The key stakeholders are summarized below, although this list is not exhaustive: Participant/Technology user roles (end-users, Original Equipment Manufacturers (OEMs), suppliers); Domain roles (domain experts, engineers, data stewards, legal/business representatives); Ontology roles (engineers, stewards, maintainers); Collaboration roles (project managers, team leaders, contributors); Federation roles (federators, governance/compliance managers); and external roles (standardization bodies, prospective participants) ensuring continuity, alignment, and adoption.

C. Step 3: Specification of collaboration approach in the data space

While not directly a development step for collaborative ontology development, the selected collaboration model (e.g., open/closed source or consortium-based) plays a critical role in shaping stakeholder participation in ontology development within the data space [24][29]. This step involves the following key aspects:

1) Specification of collaboration scope

Building on the stakeholder documentation from step 2, this involves defining the roles, activities, and responsibilities required for collaboration, along with their timeframes and allocation.

2) Specification of collaboration roles and cruciality

Collaboration roles depend on the chosen collaboration model and the data space structure. Key roles may include the core team (comprised of associations, standardization bodies, research institutions, and industry representatives), contributors (data space participants), driving forces, and federators.

3) Specification of collaboration processes

Collaboration processes define how and when stakeholders interact. For instance, contributors might propose ideas, while the core team evaluates, develops, integrates, and communicates these contributions. Additionally, the mechanisms for redefining or reassigning roles and processes over time need to be outlined in this step.

4) Specification of collaboration technology stack

Identify the tools, platforms, and documentation formats required to support collaboration. This includes tools for ontology specification and processes, such as annotating ontologies with custom metamodel attributes during collaboration.

5) Specification of communication artifacts and channels

Define what information will be visible to external parties, how contributors will be informed, and strategies to foster engagement. For example, regular releases of models and specifications can serve as communication artifacts to maintain transparency and encourage participation.

D. Step 4: Definition of use cases

This step focuses on identifying and specifying the use cases that will guide activities within the data space. It is a critical step for determining the actions to be performed and the information required for successful implementation [9], [30]. The process consists of the following activities:

1) Define Use Cases

Stakeholders should collaboratively define the use cases for the data space. This includes outlining specific scenarios where the data space will be used for offering and consuming various services, ensuring a comprehensive understanding of user needs and operational requirements.

2) Cluster and prioritize Use Cases

Domain experts and participants play a pivotal role in clustering the defined use cases based on relevance and interdependence. Prioritization should follow to ensure that the most critical use cases are addressed first, aligning with the overall goals of the data space.

3) Document Use Cases (Backlog)

Finally, all defined and prioritized use cases should be thoroughly documented in a backlog. This documentation serves as a reference point for the next sprint of collaborative development process, ensuring alignment and clarity as the project progresses.

4) Map Roles to Use Cases

It is essential to map the roles identified in Step 2 to the defined use cases. This mapping should focus on specifying responsibilities in the collaborative process and the expected outcomes for each role, ensuring that everyone understands their contributions to the use cases.

5) Match Stakeholders with Roles

Once roles are mapped, individual stakeholders should be matched to these roles based on their expertise and capacity. Open and transparent communication during this phase is vital, allowing stakeholders to express their capabilities and constraints, such as their availability.

E. Step 5: Derivation of requirements list for ontology development

In collaborative developments involving diverse expertise, it is essential to translate stakeholder needs into clear requirements. Since data spaces combine content-related and technical components, multiple requirement categories must be considered when designing data space semantics.

Content-related requirements are derived from domain vocabularies, standards, specifications, and data models. Competency questions validate the ontologies by comparing expected and actual query results.

Functional and technological requirements define semantic functionalities and technologies, such as supported languages (e.g., Resource Description Framework (RDF), Web Ontology Language (OWL), JavaScript Object Notation for Linked Data (JSON-LD)), software tools (e.g., for modeling or validation), inference depth, rule usage, validation mechanisms (e.g., Shapes Constraint Language (SHACL)), ontology topologies (e.g., single vs. multiple ontologies), modularization, versioning, alignment strategies, and lifecycle management.

Quality requirements encompass structural, syntactic, and content checks for ontology instances, as well as maintainability, maturity, and usability. Best practices for modeling and overarching quality criteria should be collaboratively defined and applied.

Performance requirements focus on system and semantic performance (e.g., query and reasoning efficiency) and influence decisions on technologies and semantic networks, such as virtualization vs. materialization, distributed reasoning, query batching, networked repositories, and database partitioning.

However, the requirement categories depend on the data spaces and the use cases and are not exhaustive.

Nachabe et al. [9] present interaction modules for requirement identification. The method for identifying and documenting requirements must be defined among stakeholders according to the agreed collaboration form. Extensive approaches, such as data flow analysis [31], offer deeper insights for specifying infrastructure, performance, and functional requirements of the ontology topology and core data space services within holistic development environments [25].

F. Step 6: Definition of ontology modules

In data spaces, multiple domains converge with overlapping information requirements, necessitating a modular ontology structure. Decomposing complex domain semantics into manageable subgraphs — referred to as ontology modules [9] — improves maintainability, reusability, access control, and interdisciplinary collaboration, while enabling standardized linkage across heterogeneous

environments [32]. The resulting ontological topology is derived from the requirements established in the preceding step. Three module types are proposed:

Ontology domain modules represent domain-specific subgraphs developed collaboratively by domain experts. They define and refine concepts from existing sources, adapted to the data space context, while managing internal dependencies and alignment with shared vocabularies.

Ontology alignment modules act as bridges between domain ontologies, defining cross-domain dependencies and relationships, such as equivalence or similarity (e.g., via Simple Knowledge Organization System (SKOS) ontology). They support structured queries, interoperability, and governance functions including versioning and dependency management [20].

Ontology metamodules provide generic, administrative models applicable across all modules, defining lifecycle management guidelines, versioning, quality assurance, and modeling standards (e.g., via SHACL shapes). They enforce governance and compatibility across the data space and can elevate domain models to meta-level hierarchies where required (e.g., via Data Catalog Vocabulary (DCAT)).

Together, these three module types simplify ontology development, enhance cross-domain collaboration, and ensure long-term maintainability within the data space.

G. Step 7: Specifation of alignment startegy

In this step, alignment strategies and alignment modules are individually defined. An alignment module specifies the links or mismatches between ontologies that need to be addressed. The specification should include documentation of conditions under which alignments were established, the use cases and requirements covered, and the stakeholders involved. Ideally, this information should be incorporated directly into the model itself.

There are several ways to detect and express ontology alignments [33]. The authors of [20] propose strategies for handling dependencies in distributed environments, defining a variety of relationship types. The authors of [32] introduce the concept of "linkset," describing dependencies and alignments between ontologies. As a lightweight approach, alignments are formulated as a triple structure between two ontologies, each as a subject and object. Thus, posing as simple way to specify alignments in a model-wise manner and as easy way to communicate between model domains. These approaches offer additional advantages when embedded in the data space context, such as reduced complexity, enhanced manageability, and extensibility. Expressing links between ontologies as independent, customizable concepts allows for further detailing or restriction, such as using metamodules for dependency management if required.

For each alignment module, the following elements must be specified, mapped, and documented: the dependencies between interfacing concepts and their relating domain ontologies; the involved stakeholders and notification requirements; the targeted ontology requirements; and the applicable circulation criteria, including temporal or event-based triggers and communication channels. Additionally, the collaboration type between domain developments must be defined (e.g., separate, joint, or partially integrated workflows), alongside the alignment type (e.g., ontology merging, OWL axiom or SKOS mapping, custom translators, or shared vocabularies). Finally, the technical and IT requirements necessary to implement the specified alignment must be established, and detailed alignment specifications documented accordingly.

H. Step 8: Specification of ontology module development teams

Building on the previous steps, it is essential to establish cross-domain teams responsible for developing and managing the different ontology modules. This step should be conducted in parallel with defining the ontology modules and specifying alignments between domains. The collaborative principles guiding each module development team should be outlined to ensure integration into the overall data space collaboration framework while allowing flexibility for team-specific workflows. The following key aspects should be addressed:

1) Assignment of ontology modules

Map ontology modules to their respective requirements and assign them to the appropriate stakeholders or development teams.

2) Specification of requirements

Clearly define the requirements for each ontology module and document them for the development process.

3) Collaboration format

Allow each team to establish its own collaboration format, provided it aligns with the overarching collaboration framework for the data space (Step 3). Teams may choose formats, such as agile workflows, workshops, or distributed version control (e.g., Git-based workflows).

4) Circulation criteria for dependencies and alignments

Define how identified dependencies and alignments will be communicated and managed. Ensure these criteria are consistent with the agreed collaboration approach (Step 3) to facilitate smooth coordination across teams.

I. Step 9: Ontology module development

Once individual ontology modules and their collaboration frameworks have been defined, the modules are developed in parallel. Various procedural models for ontology development are available in the literature, such as the NeOn Methodology [7] and the Methontology Framework [13]. These models provide structured approaches for creating, managing, and iterating ontologies. The choice of method should align with the specific requirements of the domain and the overarching data space framework.

During development, the specified circulation criteria for interactions with alignment and metamodules must be adhered to. These criteria ensure that dependencies and alignments between modules are effectively managed. Alignment modules and linksets can also be used to monitor and control iterative progress across modules, ensuring consistency and interoperability.

New requirements arising during ontology module development should be carefully evaluated. Decisions must be made on whether to include these requirements in the use case backlog for future consideration or to address them

immediately through iterative alignment and updates to the module.

It is important to note that ontology module development is a highly domain-specific and individualized process. The complexity of the domain, the roles of the stakeholders, and the specific use cases will significantly influence the approach taken. Collaborative tools, versioning systems, and regular coordination meetings are often critical to ensuring progress and alignment across parallel developments.

J. Step 10: Documentation, publication and maintenance

The final step in the approach involves systematically documenting, publishing, and maintaining all developments. This encompasses every aspect of the process, from domain and stakeholder analysis to use case definitions, requirement derivations, and the specification and development of individual ontology modules. Documentation should be structured to provide a clear, traceable history of decisions, assumptions, and development artifacts. It should also outline next steps and anticipated challenges, referencing the use case backlog or requirements that have already been addressed and iterated upon. The documentation should answer key questions, such as:

- What has been defined? This includes key use cases, requirements, and assumptions.
- Who has participated? Identifying stakeholders, roles, and contributions.
- How has the work been conducted? Detailing collaboration formats, workflows, and processes.
- What are the scope, development horizon, and timeline of the ontology modules?

Publishing ensures that ontology modules and related artifacts are accessible, reusable, and transparent. Metadata must accompany the ontology, detailing its purpose, version, authorship, licensing, and usage instructions. To enhance credibility, validation results from tools, such as SHACL or OWL reasoners should be included to demonstrate correctness. Additionally, access permissions should be clearly defined, whether open, restricted, or tiered, depending on the collaboration model. A clear process for versioning and publishing updates is also essential to maintain consistency and compatibility over time.

Maintaining ontology modules is critical for ensuring their long-term functionality and relevance. This involves regularly reviewing the use case backlog to identify new requirements and evaluating change requests to decide whether they should be implemented. Alignment modules play a key role in managing the cascading effects of changes across modules, ensuring that dependencies are addressed effectively. Version control systems, such as Git, should be used to maintain clear versioning and compatibility tracking. Updates must be validated using semantic tools like Pellet, HermiT, or SHACL to ensure consistency and correctness, and all changes should be thoroughly documented to keep records of updated requirements and refinements. Lifecycle management processes should be established to retire outdated modules, introduce new ones, and manage transitions. Engaging stakeholders through regular feedback ensures continuous improvement and alignment with the evolving needs of the data space.

IV. Conclusion and Future work

This paper identified key elements for collaborative ontology development in data spaces, analyzed semantic interoperability gaps, and derived challenges inadequately addressed by current methodologies. Building on existing approaches and practical experience, we proposed CODeS — a structured ten-step methodology guiding stakeholders from domain analysis through collaborative ontology development via a modular, stakeholder-driven process applicable across heterogeneous organizational contexts.

CODeS addresses the identified challenges through its modular approach: ontology domain and alignment modules decompose complex semantic graphs into negotiable subgraphs (C1); meta modules enforce consistent modeling guidelines and clarify roles and decision rights (C2); pre-defined, reusable modules lower participation barriers across expertise levels (C3); structured collaboration formats and standardized mapping procedures enable systematic integration while preserving domain autonomy (C4); and metamodules provide a foundation for machine-readable semantics and incremental adoption of shared data exchange mechanisms (C5).

Several aspects require further attention. Versioning, change management, and comprehensive documentation remain indispensable for long-term sustainability, and appropriate tooling for co-creation, alignment validation, and deployment must be developed. Future work will therefore focus on: empirical validation through real-world use cases with measurable metrics; development of a supporting tooling ecosystem; governance frameworks for managing modules across organizational boundaries; incentive mechanisms encouraging broader participation; and practitioner guidance in the form of a comprehensive handbook. These efforts aim to transform CODeS into a proven, tool-supported methodology for data space initiatives.

References

[1] G. Solmaz *et al.,* "Enabling data spaces," in *Proceedings of the 1st International Workshop on Data Economy*, Rome, Italy, 2022, pp. 42–48.

[2] E. Simperl and M. Luczak-Rösch, "Collaborative ontology engineering: a survey," *The Knowledge Engineering Review*, vol. 29, no. 1, pp. 101–131, 2014, doi: 10.1017/S0269888913000192.

[3] A. Hutterer and B. Krumay, "The adoption of data spaces: Drivers toward federated data sharing," in *Proceedings of the 57th Hawaii International Conference on System Sciences (HICSS)*, 2024.

[4] E. Curry, S. Scerri, and T. Tuikka, *Data Spaces: Design, Deployment and Future Directions,* 1st ed. Cham: Springer International Publishing, 2022.

[5] L. Sánchez-González, A. Iglesias-Molina, O. Corcho, and M. Poveda-Villalón, "On the Governance of Semantic Artefacts in Dataspaces," *CEUR Workshop Proceedings*, vol. 3705, 2024.

[6] A. Gómez-Pérez, M. Fernández-López, and O. Corcho, *Ontological Engineering: With examples from the areas of Knowledge Management, e-Commerce and the Semantic Web,* 1st ed. London: Imprint Springer; Springer London, 2004.

[7] M. C. Suárez-Figueroa, A. Gómez-Pérez, and M. Fernández-López, "The NeOn Methodology for Ontology Engineering," in *Ontology Engineering in a Networked World*, M. C. Suárez-Figueroa, A. Gómez-Pérez, E. Motta, and A. Gangemi, Eds., Berlin, Heidelberg: Springer Berlin, Heidelberg, 2012, pp. 9–34.

[8] S. Copei and L. Rülicke, "Best Practices to overcome challenges and barriers during the implementation of a data space for the energy domain: An experience report," *Data in brief*, vol. 61, p. 111838, 2025, doi: 10.1016/j.dib.2025.111838.

[9] L. Nachabe, F.-Z. Hannou, M. Lefrançois, and M. Jubault, "Toward Agile Interaction Model based ontology development Methodology (AIME) for FAIR European data spaces," *FAIR Principles for Ontologies and Metadata in Knowledge Management (FOAM)*, 2024.

[10] S. Steinbuss, A. H. Almeida, B. van den Wouter, and B. Sthefan Gabriel, "Semantic Interoperability in Data Spaces," Zenodo, Nov. 2025. doi:10.5281/zenodo.17630664.

[11] M. S. Fox and M. Grüninger, "Ontologies for Enterprise Modelling," in *Research Report Esprit, Enterprise Engineering and Integration: Building International Consensus, Proceedings of the International Conference on Enterprise Integration and Modeling Technology, ICEIMT 1997, Torino, Italy, October 28-30, 1997*, K. Kosanke and J. G. Nell, Eds., Berlin, Heidelberg: Springer, 1997, pp. 190–200.

[12] M. Uschold and M. King, "Towards a Methodology for Building Ontologies," *Workshop on Basic Ontological Issues in Knowledge Sharing, held in conjunction with IJCAI-95*, 1995.

[13] M. Fernández López, A. Gómez-Pérez, and N. Juristo Juzgado, "METHONTOLOGY: From Ontological Art Towards Ontological Engineering," in *Proceedings of the Ontological Engineering AAAI-97 Spring Symposium Series*, 1997.

[14] Y. Sure, S. Staab, and R. Studer, "Methodology for development and employment of ontology based knowledge management applications," *SIGMOD Rec.*, vol. 31, no. 4, pp. 18–23, 2002, doi: 10.1145/637411.637414.

[15] Y. Sure, S. Staab, and R. Studer, "On-To-Knowledge Methodology (OTKM)," in *International Handbooks on Information Systems, Handbook on Ontologies.*, S. Staab and R. Studer, Eds.: Springer Berlin, Heidelberg, 2004, pp. 117–132.

[16] H. S. Pinto, C. Tempich, and S. Staab, "Ontology Engineering and Evolution in a Distributed World Using DILIGENT," in *International Handbooks on Information Systems, Handbook on Ontologies*, S. Staab and R. Studer, Eds.: Springer, Berlin, Heidelberg, 2009, pp. 153–176.

[17] K. Kotis and G. A. Vouros, "Human-centered ontology engineering: The HCOME methodology," *Knowl Inf Syst*, vol. 10, no. 1, pp. 109–131, 2006, doi: 10.1007/s10115-005-0227-4.

[18] S. Braun, A. Schmidt, A. Walter, and V. Zacharias, "The ontology maturing approach for collaborative and work integrated ontology development: evaluation results and future directions," in *ESOE'07: Proceedings of the First Interational Conference on Emergent Semantics and Ontology Evolution*, L. L. Chen, p. Cudré-Mauroux, P. Haase, A. Hotho, and E. Ong, Eds., Aachen, Germany: CEUR-WS.org, 2007, pp. 5–18.

[19] S. Auer and H. Herre, "RapidOWL — An Agile Knowledge Engineering Methodology," in *Lecture Notes in Computer Science*, vol. 4378, *Perspectives of Systems Informatics. PSI 2006.*, I. Virbitskaite and A. Voronkov, Eds., Berlin, Heidelberg: Springer, 2007, pp. 424–430.

[20] E. Sunagawa, K. Kozaki, Y. Kitamura, and R. Mizoguchi, "An Environment for Distributed Ontology Development Based on Dependency Management," in *Lecture Notes in Computer Science*, vol. 2870, *The Semantic Web - ISWC 2003*, D. Fensel, K. Sycara, and J. Mylopoulos, Eds., Berlin, Heidelberg: Springer, 2003, pp. 453–468.

[21] M. Andresel, V. Siska, R. David, S. Schlarb, and A. Weißenfeld, "Adapting Ontology-based Data Access for Data Spaces," in *CEUR Workshop Proceedings, SDS@ESWC*, J. Theissen-Lipp, P. Colpaert, S. K. Sowe, E. Curry, and S. Decker, Eds.: CEUR-WS.org.

[22] S. Bader *et al.,* "The International Data Spaces Information Model – An Ontology for Sovereign Exchange of Digital Content," in *Lecture Notes in Computer Science*, vol. 12507, *The Semantic Web – ISWC 2020*, J. Z. Pan et al., Eds., Cham: Springer, 2020, pp. 176–192.

[23] S. Auer, "Semantic Integration and Interoperabililty," in *Designing Data Spaces*, B. Otto, M. ten Hompel, and S. Wrobel, Eds., Cham: Springer, 2022, pp. 195–210.

[24] R. Mies *et al., Open source hardware development: A handbook for collaborative product creation*. Berlin: Berlin Universities Publishing, 2024.

[25] E. Konietzko, C. Tanrikulu, F. Schwarz, and K. Lindow, "How to Gaia-X?," *Industrie 4.0 Management*, vol. 38, no. 6, pp. 54–58, 2022, doi: 10.30844/IM_22-6_54-58.

[26] C. Tanrikulu, S. Gogineni, and K. Lindow, "Decentralized Data Spaces with Gaia-X," in *ProduktDaten Journal 1 2024.* Accessed: Apr. 9 2026. [Online]. Available: https://prostep.epaper-pro.org/pdj1-2024_english/#84

[27] H. Berg, S. Gogineni, C. Tanrikulu, E. Konietzko, and K. Lindow, "Solution Approach for Asset Integration in Federated Ecosystems," in *Lecture Notes in Business Information Processing*, vol. 536, *Information Systems. EMCIS 2024*, M. Themistocleous, N. Bakas, G. Kokosalakis, and M. Papadaki, Eds., Cham: Springer, 2025, pp. 208–217.

[28] J. Gessler, M. R. Cencic, C. Metzner, H. Wieker, K. Lindow, and W. H. Schulz, "Business models and organizational roles of data spaces: A framework for value creation in data ecosystems," *Data in brief*, vol. 61, p. 111795, 2025, doi: 10.1016/j.dib.2025.111795.

[29] E. P. Konietzko and S. Gogineni, "Ontology Based Skill Matchmaking Between Contributors and Projects in Open Source Hardware," in *Communications in Computer and Information Science*, vol. 1789, *Metadata and Semantic Research. MTSR 2022*, E. Garoufallou and A. Vlachidis, Eds., Cham: Springer, 2023, pp. 14–25.

[30] A. Cockburn, *Writing effective use cases,* 24th ed. Boston: Addison-Wesley, 2012.

[31] K. Lindow, T. Riedelsheimer, P. Lünnemann, and R. Stark, "Betrachtung des Entwicklungsumfeldes durch die methodische Datenflussanalyse," in *ProduktDaten Journal 2 2017.* [Online]. Available: http://prostep.epaper.pro/journal-2017-02/de/#52

[32] R. Rik, S. Stol, and T. Mollema, "Ontology Matching trough alignment and extension: a Best Practice," Jun. 2025. Accessed: Apr. 9 2026. [Online]. Available: https://docs.crow.nl/ontology-alignment/whitepaper/

[33] I. Osman, S. F. Pileggi, S. Ben Yahia, and G. Diallo, "An Alignment-Based Implementation of a Holistic Ontology Integration Method," *MethodsX*, vol. 8, p. 101460, 2021, doi: 10.1016/j.mex.2021.101460.

Genre Prediction Using RNNs and LLM-Enhanced Video Game Review Data

Gabriel Young, Susan Gauch
Department of Electrical Engineering & Computer Science
University of Arkansas
Fayetteville, AR 72701, USA
Emails: gpy001@uark.edu, sgauch@uark.edu

***Abstract* — Large Language Models (LLMs) are powerful tools for engaging with textual data, carrying many advantages over classical Natural Language Processing (NLP) and Machine Learning (ML) approaches. However, a classical ML model can still be faster, more efficient to run, and accessible than an LLM. We seek to capture the benefits of LLM-based text comprehension and preserve them within a classical ML model through a hybrid approach. The LLM operates on text to identify relevant information and associations within our problem space, then the ML model trains on the LLM output. The model may learn from the LLM and provide a more efficient alternative to querying the LLM directly for future data. Using review data from the video game marketplace Steam, we conduct a series of experiments toward this end. We prompt the LLM to surface various information from the raw data and train Recurrent Neural Networks (RNNs) to predict a single genre of the games, "Role-Playing Game" ("RPG"). We then evaluate the performance of the trained RNN models on the raw data, checking for generalizability and performance loss/improvement. Results are promising. At baseline, using raw review data, a balanced (50% RPG, 50% non-RPG) dataset, and no LLM assistance, a shallow RNN can predict the genre under test with an average accuracy of 64.1%. The maximum accuracy of the LLM on this same dataset is 84.1%. Our other models under test lie between these two bounds and demonstrate merits from engaging the LLM during their training.**

Keywords - Classic ML; AI; LLM; NLP; Video Games; Reviews

I. INTRODUCTION

Large Language Models (LLMs) are able to solve or circumvent many of the greatest challenges in language comprehension. They are particularly useful for surfacing subtle, subtextual information from spoken and written language data. They are also user-friendly, requiring low effort on the part of a developer, data scientist, or layperson to leverage. They are now dominating the field of Natural Language Processing (NLP) and predictive modeling from language in popular use, enterprise applications, and, increasingly, research [7]. LLMs are not the most appropriate solution for every use case. An LLM is, by its nature, a generalist. It accepts many formats of input and answers a wide range of requests. This is by virtue of its having trained on large amounts of cross-functional data and a complex, computationally expensive network architecture [5]. For large-scale applications with hundreds of thousands, or millions, of users and a well-defined, domain-specific use case it can be an inefficient, or even infeasible, option [13]. Accessibility can also be an issue, since 3rd-party LLMs charge for services.

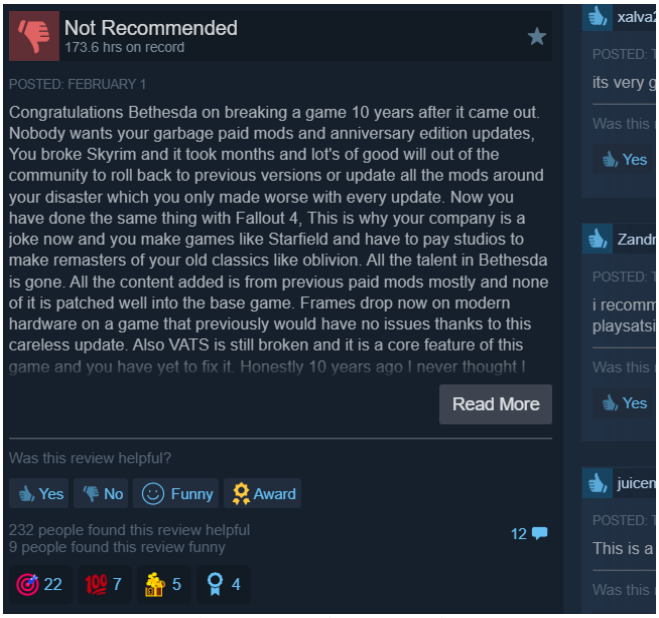


Figure 1. Example Steam Review.

One of the traditional approaches to predictive modeling is the bidirectional Recurrent Neural Network (RNN). An RNN must be more tailored to its problem space than an LLM and is generally less complex and less powerful, underperforming the LLM without a great deal of training and fine-tuning [7]. However, this means it is simpler to build and own, and much less resource intensive to run repeatedly [13].

The question to answer is this: is it possible to build a simple RNN, train it on LLM-modified data, and retain some of the benefits of the LLM's more capable text processing? We set out to explore this approach, using an LLM to generate summary data and surface key information from text, training RNNs on the LLMs output, and then evaluating the RNNs' performance.

For this, we need a difficult problem in written language comprehension on which both an LLM and RNN might perform and be compared. We chose to predict genre classifications for video games on the Steam service, based on user reviews (Figure 1). Steam is a popular gaming marketplace, the largest for online digital downloads. It makes sales, usage, site page data and review data for its games publicly available through a research API [2]. Among other information, each game listing on Steam includes a set of tags. The tags may be placed on a game's entry by developers and users and the top 20 tags are surfaced for viewing on the game's page [3]. Genres for each game are a subset of these tags belonging to Steam's list of recognized genres.

Because the tags, and by extension genres, are user-generated, there should be consistency between users' choices of genres for the game and references to a game's genres in the reviews. It should be possible, with sufficient informative review data, to predict genre for these games with some degree of accuracy.

The level of difficulty in this problem varies depending on the genre being classified. Some genres are more prevalent than others among games overall or are mentioned in reviews more often and more directly. Classifying genre is difficult in the case of the "Indie" genre. The classification pertains to the size of the development studio, a factor separate from the direct experience gained by playing the game. Consequently, it is mentioned infrequently by reviewers. By contrast, the "Sexual Content" genre is trivial to identify from user reviews, with frequent direct mentions.

In the best functional case, the interpretive advantages of the LLM on text data would be gained and then passed to the model, highlighting informative features of the text and cutting away uninformative or misleading features. In the worst case, a model trained on the LLM data will no longer generalize properly when exposed to raw text. A model that can only operate on the LLM output realizes no improvement over using the LLM by itself. However, if there is only acceptable (user and use-case determined) performance loss relative to using the LLM alone, the result is a lighter-weight, more efficient solution to the problem.

We are using simple RNNs for this experiment. It is very possible that more tuned models will achieve higher initial performance on the dataset and less performance gain from incorporating the LLM data. We are also predicting a single genre for games, and using data from a single source, so the scope of our conclusions will, necessarily, be limited.

The remainder of this article is organized as follows:

- background for RNNs, LLMs, and genre prediction with a discussion of related works
- a summary and analysis of our dataset

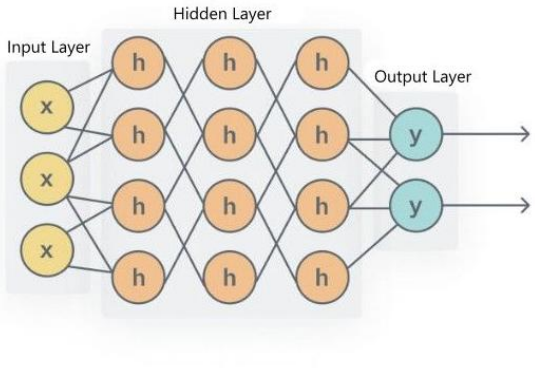


Figure 2. High-Level RNN Diagram.

- the design of our experiments and our different experiment cases
- an explanation of our experiment results
- an interpretation, the conclusions of our experiment results, and future works

II. RELATED WORK

A. Classic RNN

RNNs are a form of deep learning algorithm. "Recurrent" refers to the feedback pattern of information as it passes through the layers of the network, with the output of previous inputs being maintained as memory or "hidden state" at each node, then passed in as input to the node again alongside the next round of regular input. They have proven effective at discerning subtle connections between input and output in problems where order and context of the input matter. They process data sequentially across their layers (Figure 2) retaining information from previous steps by passing layer output back in as input during the next iteration [4].

Long Short-Term Memory (LSTM) is a specialized type of RNN capable of maintaining useful context in its hidden state for long periods, but also discarding used context from further calculations [10]. This capability is due to the structure of its nodes, which are modified with three "gates" to the node's hidden state: an input gate, output gate, and forget gate. The node can receive information, persist it where applicable, but also reject it after use. This innovation was designed to address the vanishing gradient problem of RNNs, wherein earlier layers of the network see much more intensive calculations during backpropagation than later layers. The ability to reject and forget information reduces the calculation complexity [10]. We use LSTM in these experiments because it has been shown to carry advantages in text processing, where context is often critical to correct interpretation and needs to be persisted until it is applied [10].

B. LLMs

An LLM is an extremely large deep learning model, trained on a vast amount of text data and a diverse array of subject matter. The data an LLM is trained on varies, sometimes according to its intended function but often based on what is accessible. Many are trained on sections of the Internet, with information that is free and publicly available. Their transformer architecture (Figure 3) allows for the processing of input data in parallel rather than sequentially, which is thought to enable their enhanced interpretive abilities [11].

LLMs have proven extremely capable at deriving meaning from language. They have been shown to navigate very complex language, with subtextual and contextual references, sarcasm, idiomatic speech, and linguistic artifacts [11]. Zhu et al. [7] examine several high-profile LLMs currently on the market and assess their capabilities for text comprehension and use in information retrieval (classically the domain of search engines). One of the areas of information retrieval cited as showing improvement under LLMs is query rewriting: refining, expanding, or otherwise modifying users' search queries to better surface relevant information [7]. This method of using LLMs to improve user text clarity is invoked similarly during these experiments.

C. Genre Prediction

In the recent work of Raj et al. [5], LLMs were leveraged for the multilabel prediction (overlapping, non-exclusive categories) of movie genres and compared against classical Machine Learning (ML) approaches [5]. Their results demonstrated that ChatGPT 3.5 consistently outperformed traditional classifiers (logistic regression, K-Nearest Neighbors (KNNs), Support Vector Machines (SVMs)), working off subtitle data for movies to predict genre [5].

Ströbel et al. [6] investigate the problem of genre prediction for literature from text, using a Gated Recurrent Unit (GRU) RNN. GRU is a variant of RNN that tends to see performance increases in smaller datasets [6]. They can achieve high rates of prediction accuracy (0.90) for the five disparate genres under test using a relatively "simple" RNN model. This work helps highlight the efficacy of RNNs in working with text for genre prediction and also supports the choice of simpler models with minimal text pre-processing as appropriate benchmarks in our own experiments [6].

D. Steam Data

Olmedilla et al. [8] undertake an exploration of Steam reviews and compare a few predictive modeling approaches. They train regression models, and a Bidirectional Encoder Representations from Transformers (BERT) model on the review data in order to predict a helpfulness score of the review. Helpfulness is a weighted vote score representing whether users who read the review found it informative. It is a user-defined and nebulous metric similar to the genres we are trying to classify in our experiments. The authors note only modest results from their model, highlighting the difficulties posed in learning from review text using a classical approach [8].

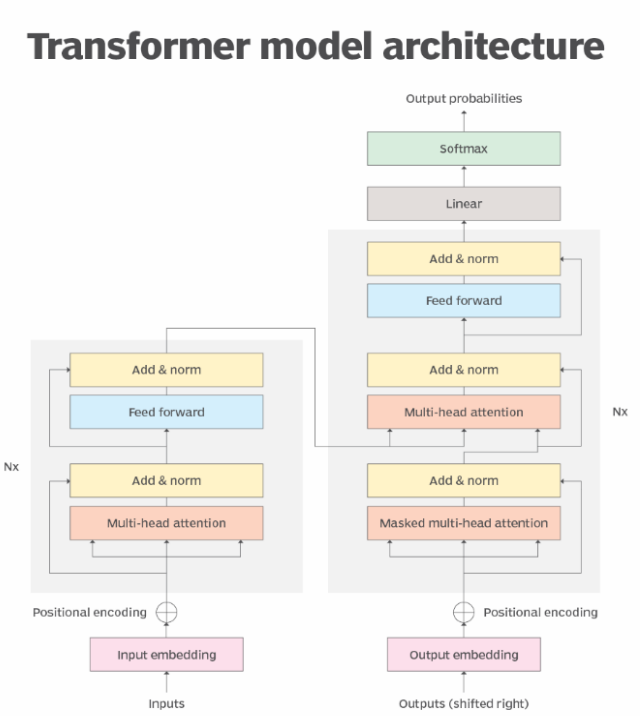


Figure 3. High-Level LLM Transformer Architecture.

E. LLM and RNN Hybrid Approaches

Chen et al. [9] engineered a hybrid LLM-Convolutional Neural Net (CNN) approach to financial modeling. Similarly to the approach we've taken for our experiments, they use LLMs to "first processes textual inputs to extract structured features" and subsequently "feed these features into a predictive model alongside numerical data." They realize consistent performance gains for the integrated solutions over either solution individually. They also note the criteria that must be considered during the design and approach to hybridized systems. "These architectures must balance competing objectives including predictive accuracy, computational efficiency, interpretability [...]" [9]. Our work extends on this consideration, seeking gains of efficiency with an acceptable cost to accuracy between the RNN-LLM hybrids and the LLM by itself.

III. DATASET

We began with the Steam Games Dataset, compiled in 2022 for all games on Steam at the time [1]. Each entry in the dataset represents a game, and each column contains information scraped from the game's page. The dataset contains information on both the genres and tags shown on the game's page at the time of compilation. Genres and tags for a game might have changed on the site since the data was gathered.

We created a dataset of 5000 Role-Playing Games (RPGs) and 5000 non-RPG games by taking random samples of gameIDs from the Steam Games dataset with "RPG" present or absent in their genre lists, respectively. We then queried the Steam API for the review data of these games, up

to 500 of their most recent reviews. Games for which we were unable to retrieve at least ten reviews were discarded from both samples. This resulted in a 7766-entry set.

The raw reviews were cleaned for processing by our RNN as well as the LLM. Common stop words ("a", "an", "for", etc.) were removed, and a minimum and maximum word length were enforced. We decided to consolidate our dataset to only those reviews with information relevant to the task, i.e. reviews with strong indicators about the presence or absence of the game's RPG classification. To assess this, we ran the full set of entries through the LLM with the following prompt:

- *"RPG stands for "Role-Playing Game". The genre often features character customization and rich narrative elements. Read the following reviews for this game and determine whether or not it is an RPG. If there is not enough information in the reviews to determine this, return an output of '[Insufficient Information]'. Reviews: {reviews}"*

Allowing the LLM to curate the dataset to only those reviews it finds informative might seem to introduce bias, advantaging the LLM. However, it has no knowledge of the true classifications for each game when it reads the reviews. The reduced dataset only includes entries for which the LLM had high confidence in its predictions. It can still, very confidently, misclassify.

After filtering, the dataset was drawn down to 2340 entries. The distribution of genre among those entries was 1122 RPG and 1218 non-RPG. We undersampled the majority class to balance the dataset, resulting in 2244 total entries.

This dataset was split 80-20 into testing and validation sets, both of which maintained the 50% class distribution.

IV. EXPERIMENTS

A. RNN Setup

The RNN was built using Python's tensorflow keras libraries [14]. For RNN classification, we need a dictionary and encoder to convert the words of our text into tokens. We use keras's TextVectorization layer object to accomplish this. Using its adapt function, we have it create a dictionary of 10,000 distinct words from our review data. When reviews or the LLM summaries are passed into this layer, they will be converted to numeric vectors based on the words present, their incidence, and their order [12].

We define our RNN as a sequential, single output, binary prediction model (Figure 4). The first layer is the encoder, and the second is a single bidirectional connected layer of 32 nodes. Bidirectional layers propagate the input forward and backwards through the layer, preserving sequential information about the tokens [4]. There is a subsequent activation layer, a dropout layer with a 50% drop rate to forestall overfitting, and finally a sigmoid activation layer. The model uses binary cross-entropy for its loss function and an Adam optimizer. Overall, the design is a minimally complex model that can still perform the desired function.

```
tf.keras.Sequential([
  tf.keras.layers.Embedding(
    input_dim=len(encoder.get_vocabulary()),
    output_dim=32),
  tf.keras.layers.Bidirectional(tf.keras.layers.LSTM(32)),
  tf.keras.layers.Dense(32, activation='relu'),
  tf.keras.layers.Dropout(0.2),
  tf.keras.layers.Dense(1, activation="sigmoid")
])
```

Figure 4. Model Definition.

B. LLM Setup

The LLM used in these experiments is ChatGPT v4.1-mini, accessed through its batched input API. This version was chosen for this experiment for reasons of availability, reliability, popularity, and cost. The data is split into 500 entry batches, sent to the LLM service, and for each experiment prompt the entries are collected and processed back into the dataset as new columns.

C. Experiment Prompts

The RNN is trained on the LLM responses (except in the control experiments) for 10 epochs. The final model is evaluated over a verification set of the response data five times to produce an average accuracy. Then, it is evaluated in the same manner over the original raw data 10 times to test generalizability. Here, we also collect average precision, recall, and F1 scores.

The experiments and their prompts to the LLM are as follows:

- Experiment 1: RNN Control
  - No LLM used.
- Experiment 2: General Summary
  - "Write a summary of the following reviews for a video game. Reviews: {text}"
- Experiment 3: Surface Genre Information
  - "Write a summary of the following reviews for a video game. If possible, surface the name of the game and a few probable genres in which it might be classified. Reviews {text}"
- Experiment 4: Surface RPG Information
  - "Write a summary of the following reviews for a video game. If possible, surface the name of the game and whether or not it is of the RPG genre. RPG stands for 'Role-Playing Game'. The genre often features character customization and rich narrative elements. Reviews: {text}"
- Experiment 5: LLM Control
  - "RPG stands for 'Role-Playing game'. The genre often features character customization and rich narrative elements. Read the following reviews for this game and return an output of '[RPG]' if it is of the RPG genre, or '[Not An RPG]' if it is not. Reviews: {text}"

V. RESULTS

Table 1 shows the final binary validation accuracy of the models during training as well as the accuracy, precision, recall, and F1 scores when they are reintroduced to the raw text. 'Accuracy' here serves as our primary basis of comparison on model performance, with the precision and recall metrics serving supplemental information about the false negative and false positive rates, respectively. The F1 score, computed as the harmonic mean of precision and recall, conveys this same information as a single metric. We also show the training and validation curves of the models (Figure 5) for Experiment 4, the highest performing, compared against the training curve and model of Experiment 1, our control.

The more specific about RPG genre classification we were with our prompting, the more the models' performance approached the LLM's. There was better performance by the LLM-informed models on the raw data with the exception of Experiment 3. While that experiment saw the highest gains in false negative identification, it suffered on true positive identification, evidenced in the lower recall score and F1 scores.

VI. CONCLUSION AND FUTURE WORK

We compared several models predicting the RPG genre of games based solely on text: an RNN trained on generic summary data, an RNN trained on surfaced genre data, and an RNN trained on surfaced RPG genre data. The LLM's accuracy on the data was 84.1%, the upper bound for accuracy in our experiments. The models were capable of performing within 1-2% of accuracy of the LLM itself while validating with LLM summary text. While losses in accuracy were observed for all models when exposed to the raw review data, the models still realized significant gains of between 5% and 10% accuracy over the baseline RNN, our lower bound at 64.1%. This suggests that benefits of the LLM processing were preserved in the training and utilized in model prediction.

We conclude that there are significant advantages in using an LLM to process and predict from text over the simpler RNN models. We can also tentatively conclude that LLMs can successfully inform RNNs during training, with gains of accuracy on real data without fine-tuning or increasing the complexity of the RNN.

We would like to apply this same series of tests to the prediction of other, single genres in our dataset such as the "Adventure", "Puzzle", and "First Person Shooter (FPS)" genres of games and compare the results to establish generalizability across genres. These genres are non-exclusive, and a single game may belong to any or all of them. Expanding the modeling problem for multi-label classification makes the experiment more applicable to real-world use cases. If the results of these extensions support continued investigation, testing prominent available LLMs and comparing their utility in surfacing text information during model training will be another series of experiments.

We are also working to increase the size and representation of our dataset, which will enable further experimentation and grant advantages in reliability for our results.

TABLE I. EXPERIMENT RESULTS

Experiment	Final Model Avg Value Acc.	Final Model Raw Text Avg Value Acc.	Final Model Raw Text Avg Prec.	Final Model Raw Text Avg Recall	Final Model Raw Text Avg F1
1. No LLM	0.641	0.646	0.599	0.690	0.641
2. General Summary	0.794	0.719	0.701	0.763	0.729
3. Surface Genres	0.831	0.682	0.746	0.536	0.623
4. Surface RPG Genre	0.822	0.739	0.724	0.757	0.740
5. LLM Only	0.841	0.841	NA	NA	NA

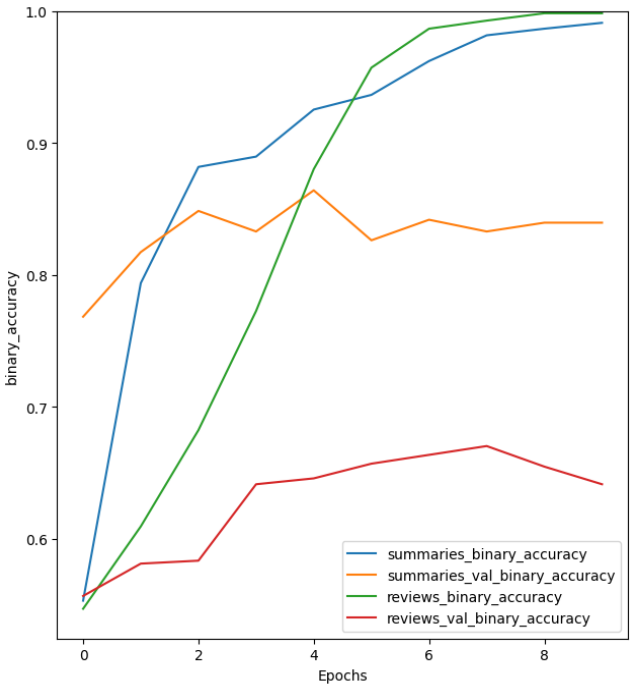


Figure 5. Experiment 4 Accuracy Curves vs. Experiment 1.

REFERENCES

[1] M. Bustos, "Steam Games Dataset", Kaggle, 2022, retrieved April, 2025 https://doi.org/10.34740/KAGGLE/DS/2109585

[2] Steamworks API Reference (Steamworks Documentation). 2026, retrieved May 2025 https://partner.steamgames.com/doc/api

[3] Steam Database Team, "SteamDB", retrieved March, 2026, https://steamdb.info

[4] C. Stryker, “Recurrent Neural Network (RNN)”. October 24, 2021, retrieved January, 2026, https://www.ibm.com/think/topics/recurrent-neural-networks

[5] S. Raj, S. Saha, B. Singh, and N. Pedanekar. “Demystifying chatgpt: How it masters genre recognition.” Natural Language Processing Journal, vol 14, 100198. 2026. https://doi.org/10.1016/j.nlp.2026.100198

[6] M. Ströbel, E. Kerz, D. Wiechmann, and Y. Qiao. “Text Genre Classification Based on Linguistic Complexity Contours Using A Recurrent Neural Network.” MRC@ IJCAI. July, 2018, pp. 56-63.

[7] Y. Zhu et al, “Large language models for information retrieval, a survey.” 2023, retrieved February, 2026, https://arxiv.org/abs/2308.07107

[8] M. Olmedilla, L. Espinosa-Leal, J. C. Romero-Moreno, and Z. Li, “Unveiling the Value of User Reviews On Steam: A Predictive Modeling Of User Engagement Approach Using Machine Learning”. In A. Abraham, G. C. Peng, P. Isaias, & P. Isaias (Eds.). Proceedings of the International Conferences on Big Data Analytics, Data Mining and Computational Intelligence 2024, BigDaCI 2024; Connected Smart Cities 2024, CSC 2024; and e-Health 2024, EH 2024. pp. 43-49

[9] S. Chen, S. Ren, and Q. Zhang, “Hybrid Architectures that Combine LLMs and Predictive Analytics for Next-Generation Financial Modeling. Mathematical Modeling and Algorithm Application”, 2025, pp 31-43.

[10] M. E. Peters et al. “Deep contextualized word representations,” in Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers), M. A. Walker, H. Ji, and A. Stent, Eds. Association for Computational Linguistics, 2018, pp. 2227–2237.

[11] L. Xu et al. “Prompting large language models for recommender systems: A comprehensive framework and empirical analysis." 2024, arXiv:2401.04997.

[12] Keras Team, “Keras documentation: Multi-GPU distributed training with TensorFlow.” Keras.io, 2020, retrieved January, 2026, https://keras.io/guides/distributed_training_with_tensorflow/

[13] A. P. Behera, J. P. Champati, R. Morabito, S. Tarkoma, and J. Gross, “Towards efficient multi-llm inference: Characterization and analysis of llm routing and hierarchical techniques.” 2025, arXiv preprint arXiv:2506.06579.

[14] Keras, Home - Keras Documentation, Keras.io; Keras, 2019, retrieved May, 2025, https://keras.io/

Improving Multi-Hop Retrieval for Question Answering via Bipartite Question-Oriented Graphs

Micah McCollum
Department of Electrical Engineering and Computer Science
University of Arkansas
Fayetteville, Arkansas, United States of America
email: mlm132@uark.edu

Susan Gauch
Department of Electrical Engineering and Computer Science
University of Arkansas
Fayetteville, Arkansas, United States of America
email: sgauch@uark.edu

***Abstract*—Accurately answering multi-hop questions requires full retrieval of multiple, interdependent passages and is a long-standing problem in the area of natural language question answering. While retrieval-augmented generation helps address single-hop questions, many retrievers presently focus on semantic similarity in a dense vector space, which is insufficient for handling multi-hop questions specifically. To ameliorate this, we propose constructing a bipartite question graph composed of hypothetically generated questions connected to passage chunks at index time. The construction of the graph is guided by a large language model to prioritize the formation of edges that signal whether a question can be answered by a text passage. During retrieval, the graph is traversed starting from semantically similar seed questions and accrues relevant connected passage chunks after a set number of hops. Results from preliminary experiments on a challenging multi-hop dataset show promise in this approach. Full context retrieval accuracy was 9% for $k = 5$ and 32% for $k = 20$ compared to 5% and 21%, respectively, for the naive vector-only baseline. These results highlight the potential of graph-based retrievers in the area of multi-hop question answering, leading to improvements in downstream applications such as chat bots, search engines, web browsers, and other applications involving natural language interaction, knowledge discovery, and information retrieval.**

***Keywords*—information retrieval; question answering; retrieval-augmented generation; large language models; graphs.**

I. INTRODUCTION

Large Language Models (LLMs) have transformed the ways with which software is interacted. One key area that has been affected is information retrieval. Companies such as Google are supplementing their search engine results with Artificial Intelligence (AI) summarizations powered by LLMs, a synthesis of traditional information retrieval and knowledge acquisition techniques with recent developments in AI. However, despite rapid adoption, LLMs alone are not equipped to comprehensively handle all types of queries. Many complex natural language questions are multi-hop in nature, requiring the combination of multiple, interdependent pieces of information to produce a satisfactory answer. Techniques like Retrieval-Augmented Generation (RAG) aim to improve general question answering by integrating a retrieval component with the LLM but often remain insufficient for reliably answering multi-hop questions [1].

To address this limitation, we propose a novel graph approach that: (1) generates hypothetical questions via an LLM that can be answered by text chunks from the corpus; (2) connects these text chunks with the generated questions in a bipartite graph, according to whether the question can be answered by a chunk, as judged by the LLM; and (3) enables efficient query-time traversal over the graph to select the most relevant chunks. By prompting an LLM to form edges up front, the retriever maintains efficiency at query time while amortizing compute costs.

The rest of the paper is structured as follows. We examine related work in Section II and describe our approach in Section III. In Section IV, we present preliminary experimental results, finally concluding with a brief discussion thereof in Section V.

II. RELATED WORK

Information retrieval is an area concerned with the systematic storage and retrieval of unstructured data, typically from the web [2]. The extracted data is then preprocessed and tokenized to prepare for indexing. Traditionally, this involves building an inverted index mapping unique terms to the documents in which they are contained. Queries and documents are represented as sparse vectors and weighted according to schemes like Term Frequency-Inverse Document Frequency (TF-IDF) [2]. Similarity measures, such as cosine similarity, may be used to compare the query to documents for the final results [2].

While effective, traditional lexical searches use exact term matching and often cannot disambiguate matches that have no semantic relevance to each other, as in the case of "rock music" and "rock salt." Later approaches incorporate dense vector embeddings or learned representations to improve semantic understanding [2].

In 2020, Facebook AI Research (now Meta) introduced RAG to leverage the generative capabilities of LLMs in producing natural language answers from retrieved results ("non-parametric memory") [1]. RAG showed improvements in the final outputs and reducing hallucinations due to an incomplete, inaccurate, or outdated knowledge base from pretraining parameters. This underscores the importance of dynamic retrieval in capturing the context needed to accurately answer arbitrary questions, especially those that are domain specific or are not sufficiently accounted for in the training data.

In 2024, Microsoft Research followed up with GraphRAG, introducing community hierarchies and summarizations from a knowledge graph to perform global reasoning [3]. By leveraging the structured connectivity between contexts instead of only relying on semantics, GraphRAG demonstrated improvements in comprehensive and diverse question answering.

One pertinent method of improving retrieval performance is augmenting embeddings of real textual signals with synthetically generated text. Hypothetical Document Embeddings (HyDE) is one such approach, which uses a language model to create hypothetical documents based on the query and embeds them alongside real documents [4]. While HyDE is a query-time approach, Question-Oriented Text Embeddings (QuOTE) is an index-time approach which instead generates and embeds hypothetical questions for each chunked text passage in the document corpus, where the generated question can be answered by the passage on which it is conditioned [5].

Past studies have shown query decomposition as a viable technique for improving multi-hop reasoning, since multi-hop queries can be viewed as a sequence of multiple single-hop queries that are easier to answer individually [5][6][7]. Taken together, these works inform our methodology and touch on several key facets of it.

III. Approach

Our approach features a key contribution from QuOTE in the form of hypothetical question generation and takes it one step further by constructing a *Bipartite Question-Oriented Graph* (BiQOG). The combination of these two concepts reflects intuition from query decomposition; given a collection A of generated hypothetical questions, it is possible that an arbitrary multi-hop question can be decomposed into a set B of multiple single-hop questions such that there is overlap between A and B [6][7]. Through this lens, it is primarily a matter of mapping the multi-hop question to an initial set of seed questions after graph construction.

The graph construction process is facilitated by an LLM which is instructed to provide a binary answer for whether or not a given chunk answers a hypothetical question. If that binary answer is yes, then an undirected edge is formed. By forming explicit edge connections between these two types of data entities, latent relationships between different questions arise, thereby aiding multi-hop reasoning. In this way, query decomposition is moved from query time to index time. Note that while there is an extra graph construction step, parallel processing can significantly reduce runtime.

During retrieval, the retriever first embeds the query and matches it to semantically similar seed questions, from which the graph traversal starts. Over a predefined number of hops, the text chunks neighboring the seed questions are collected and reranked with a cross-encoder according to a width, which is the number of neighbors used to traverse the graph further. After the hops are finished, the list of seen text chunks is reranked a final time before the top-k are passed into the LLM's context window for downstream answer generation.

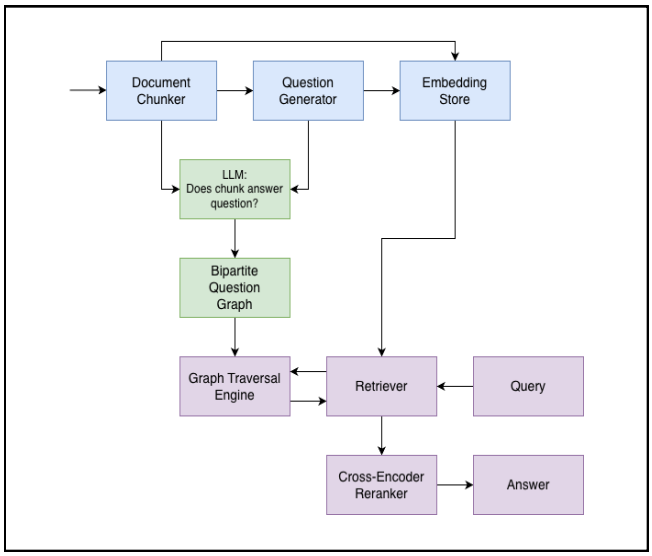


Figure 1. BiQOG High-Level Architecture.

Figure 1 illustrates the high-level architecture of our method and is comprised of text indexing, graph construction, and retrieval. Once the graph is constructed, it is used during the graph traversal stage of retrieval. Importantly, the graph's construction process is amenable to incremental updates and is left as an implementation detail.

IV. Experiments

The dataset used to evaluate BiQOG and the baselines is MultiHop-RAG, a challenging dataset specifically made for evaluating question answering and retrieval tasks in a multi-hop RAG setting [8]. It features four types of queries: *Inference, Comparison, Temporal,* and *Null.* For the purpose of scope, we exclude *Null* queries from the test set since they are unanswerable. Although we are principally motivated by RAG as a downstream application, the quality of RAG depends upon the quality of retrieval. Furthermore, RAG is difficult to accurately assess for end-to-end question answering. Thus, our evaluation focuses exclusively on retrieval quality to maintain a well-defined scope for experiments as well as generalizability. Our baselines include naive dense vector retrieval and QuOTE.

We follow the choice of metrics found in QuOTE for multi-hop evaluation, which is Full@k and Recall@k [5]. Recall@k measures the average fraction of gold evidence found within the top-k over all queries. Full@k indicates that all gold evidence is found within the top-k retrieved results; in other words, it measures how many queries on average retrieve 100% recall. Because the accuracy of an answer to a multi-hop question requires having all relevant passages, Full@k is the most important metric in determining the effectiveness of multi-hop retrieval. We conduct multiple experiments with different values of k for a comprehensive evaluation. Table 1 shows our preliminary results.

BiQOG demonstrates better performance in retrieving all necessary gold context compared to the baselines and is competitive with or better than baselines for queries where only part of the gold is retrieved. Notably, at higher values of *k*, BiQOG breaks away from the baselines in both metrics, showing that it is more adept at retrieving gold within the top-*k* altogether and achieving coverage, irrespective of exact ranking. QuOTE outperforms BiQOG in Recall@5, suggesting that BiQOG experiences a gap where it retrieves either all evidence (in the case of Full@5) or it retrieves a slightly lower fraction of gold compared to QuOTE. Despite this, the strong results from BiQOG illustrate the effectiveness of a graph structure for multi-hop retrieval compared to baseline approaches which do not leverage any graph techniques.

TABLE I. COMPARISON OF NAIVE, QUOTE, AND BIQOG BASELINES ACROSS RETRIEVAL TASKS IN MULTIHOP-RAG. BEST ENTRIES ARE IN **BOLD**.

Approach	**MultiHop-RAG**			
	Full@5	***Full@20***	***Recall@5***	***Recall@20***
Naive	5.00	21.00	27.08	51.50
QuOTE	8.00	29.00	**32.42**	59.00
BiQOG	**9.00**	**32.00**	31.92	**60.33**

For all LLM tasks, including question generation and edge formation, we use gpt-4o-mini due to its cost-effectiveness. Similarly, the embedding model and reranking model used in the experiments are all-MiniLM-L6-v2 and ms-marco-MiniLM-L6-v2, respectively. For this study, we elected not to use state-of-the-art language and embedding models due to cost concerns, but model choice should be agnostic [5].

A limitation of the approach is the heavy use of an LLM, resulting in higher token costs and inference latency. However, this usage is exclusively offline, so the impact is relegated to a one-time cost. Additionally, the graph is loaded into memory for these experiments. Finally, from failed cases in which recall was zero for a query, *Temporal* appeared as the most problematic query type, suggesting our approach alone may not be sufficient in handling all types of queries, and so other techniques may have to be supplemented to compensate.

Overall, the results show promising improvements in full context retrieval over baselines, demonstrating the importance of the graph structure in multi-hop settings.

V. CONCLUSION AND FUTURE WORK

In this paper, we propose the use of a novel graph representation for multi-hop retrieval in RAG applications by encoding hypothetical questions conditioned on passage chunks into a bipartite graph structure at index time. We employed LLMs to generate the hypothetical questions and perform edge formations during offline graph construction and utilized simple but effective retrieval at query time. Experimental results demonstrate that in full context retrieval, a necessary requisite for accurate multi-hop question answering, our approach reports a Full@20 of 32% over the naive vector-only baseline 21% and a Full@5 of 9% over the baseline 5%. We plan to do further testing on a large dataset to get a fuller evaluation. In the future, we anticipate that incorporating query decomposition may yield improvements in retrieval accuracy, since multi-hop questions can be seen as a composition of multiple single-hop questions [6][7]. Thus, further investigation into the effective mapping between queries and single-hop seed question nodes in the bipartite graph could prove fruitful.

REFERENCES

[1] P. Lewis et al., "Retrieval-augmented generation for knowledge-intensive nlp tasks," in Advances in Neural Information Processing Systems, vol. 33, pp. 9459-9474. 2020.

[2] C. D. Manning, P. Raghavan, and H. Schütze, An Introduction to Information Retrieval. Cambridge, England: Cambridge University Press, 2009.

[3] D. Edge et al., "From local to global: a graphrag approach to query-focused summarization," arXiv preprint arXiv:2404.16130, 2024. [Online]. Available from: https://arxiv.org/pdf/2404.16130. [Retrieved: March, 2026]

[4] L. Gao, X. Ma, J. Lin, and J. Callan, "Precise zero-shot dense retrieval without relevance labels," Proc. 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Jul. 2023, pp. 1762-1777, doi:10.18653/v1/2023.acl-long.99.

[5] A. Neeser, K. Latimer, A. Khatri, C. Latimer, and N. Ramakrishnan, "Quote: question-oriented text embeddings," arXiv preprint arXiv:2502.10976, 2025. [Online]. Available from: https://arxiv.org/pdf/2502.10976. [Retrieved: March, 2026]

[6] S. Min, V. Zhong, L. Zettlemoyer, and H. Hajishirzi, "Multi-hop reading comprehension through question decomposition and rescoring," Proc. 57th Annual Meeting of the Association for Computational Linguistics, Jul. 2019, pp. 6097-6109, doi: 10.18653/v1/P19-1613.

[7] R. Fu, H. Wang, X. Zhang, J. Zhou, and Y. Yan, "Decomposing complex questions makes multi-hop qa easier and more interpretable," Findings of the Association for Computational Linguistics: EMNLP 2021, Nov. 2021, pp. 169-180, doi:10.18653/v1/2021.findings-emnlp.17.

[8] Y. Tang and Y. Yang, "Multihop-rag: benchmarking retrieval-augmented generation for multi-hop queries," arXiv preprint arXiv:2401.15391, 2024. [Online]. Available from: https://arxiv.org/pdf/2401.15391. [Retrieved: March, 2026]

Persona-Conditioned Emotion Classification of Conversation Using LLMs

Israel Cuevas, Andrew Mackey, Susan Gauch
Department of Electrical Engineering and Computer Science
University of Arkansas – Fayetteville
Fayetteville, Arkansas, USA
e-mail: ibcuevas@uark.edu, almackey@uark.edu, sgauch@uark.edu

***Abstract*—Large Language Models (LLM) have demonstrated success across a wide range of tasks in the field of natural language processing, including within the emotion classification task of language. With the recent advancements of agentic workflows and conversational chatbots in the field of artificial intelligence, it is fairly common to employ the use of personas to bias LLM interactions toward domain-specific applications. In this study, we investigate the impact of persona-conditioned models for the task of emotion classification along with model confidence of performance under these persona-conditioned settings. Our statistically-significant results ($p < 0.001$) demonstrate that persona-conditioned models affect model performance while also demonstrating the performance differences between each of the personas. Furthermore, through our experiments we observed variations in model confidence between both open and closed LLMs for the Emotion Recognition in Conversation (ERC) task.**

Keywords-natural language processing; emotion analysis; large language models.

I. INTRODUCTION

Emotion Recognition in Conversation (ERC) is a task in which the affective state of language in conversation is identified for a wide range of applications [1][2]. Recent advancements in the field have led to major gains in performance by leveraging contextualized representations of utterances and surrounding conversation.

Persona conditioning is a technique where speaker-specific variations are captured in domain-specific settings [3]. This can be observed in conversational settings where speakers often differ in their communicative style, interpersonal roles, affective tendencies and expressiveness, and habitual responses to events. One example of this can be found in sarcastic language as it may indicate amusement for one speaker, while at the same time, the same words or expressions may be indicative of frustration for another. As a consequence, emotionally ambiguous utterances may necessitate the use of context-specific features, such as speaker or prior conversation, to accurately interpret and categorize the meaning. Prior literature for emotion classification has focused primarily on utterance-level semantics, local conversational context, or multimodal cues [3][4][5][6]. Work has included the use of speaker identity or persona, but this has often be leveraged as a shallow feature, or it was omitted during the ERC classification task. As a result, current models may fail to distinguish between emotion signals that are linguistically similar but different across speakers.

The goal of this paper is to investigate persona conditioning within the emotion classification task. The work considers whether explicit speaker representations can improve the recognition of emotions beyond text-only and context-only approaches. We define persona broadly to include information associated with the speaker, such as stable profile attributes, speaker-specific embeddings, and historical interaction patterns that characterize how emotion is typically expressed. The incorporation of personas into the emotion classification task can alter the performance of classification by reducing ambiguity, personalizing contextual interpretation, and enabling models to learn systematic differences in affective expression across roles. It is also important to observe that persona conditioning raises important questions regarding how persona should be represented, how it interacts with discourse context, and under what conditions it contributes meaningful gains across models.

To address these questions, this work investigates both open and closed LLM models that integrate persona signals via prompts into emotion classification and compare them against strong non-persona, or neutrally-conditioned persona, baselines. Our analysis focuses not only on overall predictive performance, but also on robustness across personas, flip rates between personas, and model confidence of accuracy. Through this investigation, we aim to clarify the role of persona in affective language understanding and to show that emotion classification can benefit from moving beyond generic contextual modeling toward more speaker-aware representations. In doing so, this work contributes to a broader view of Natural Language Process (NLP) systems as interpreters of language that is socially situated, personalized, and shaped by recurring human identities rather than by text alone.

The remainder of the paper is organized as follows: Section II provides background information relevant to the task of emotion analysis in conversations. Section III outlines the persona-conditioned emotion classification in conversation task. Section IV outlines the datasets that were used in our experiments. Section V provides information regarding the design of our experiments. Section VI provides the results from our experiments. Section VII summarizes our findings and details future directions for this work.

II. RELATED WORK

Persona conditioning is an approach to using Large Language Models (LLMs) to model subjectivity in a wide range of tasks. Personas were introduced in [3] by implementing persona embeddings and speaker-specific conditioning to generate more

 ISBN: 978-1-68558-387-3

consistent, personalized neural dialogue responses. The authors constructed two persona-based models: a Speaker Model to model the respondent's personality, and a Speaker-Addressee model to parallel the respondent adaptation to a given addressee. The work demonstrated that personal characteristics could be captured through distributed representations, such as speaking style. Other work was done to demonstrate that grounding dialogue in explicit persona sentences improves consistency and engagement while also demonstrating that dialogue can be used to predict profile information [4].

As language models continued to make advancements, *prompt-based learning* strategies demonstrated promising results across a wide range of tasks in the field of NLP [7]. Instead of supervised machine learning tasks where by the goal is to predict $\mathbf{y}$ based on input $\mathbf{x}$, conditioned as $\Pr(\mathbf{y} \mid \mathbf{x}; \boldsymbol{\Theta})$, language models were leveraged to formulate a new input $\hat{\mathbf{x}}$ from $\mathbf{x}$ to be used to obtain the target $\mathbf{y}$. Prior work has demonstrated that as language models continued to scale in size, they are capable of performing in-context learning where training examples can be provided to facilitate to performance of a task without the need for fine-tuning models [8]. This provided the ability to build architectures that are task-agnostic while also achieving competitive results. The work presented in [9] expanded on this idea by proposing a prompt-based fine-tuning method along with automatic prompt generation and better demonstration selection for strong few-shot text classification.

Other work has been done to leverage prompt-based learning-templates, verbalizers, tuning strategies, and evaluation to provide a unified vocabulary and taxonomy [7]. In [10], the authors quantified the variance persona variables explain in subjective NLP dataset labels and find that persona prompting yields modest, but significant gains, mainly when persona truly predicts disagreement patterns. The work demonstrated that the gains are realized in situations where the entropy in annotation is high with a lower standard deviation, and that persona variables explain less than 10% of the variation in the human annotations. The work also demonstrated a clear association between predictive persona variables and human labels, with a zero-shot 70B model reaching 81% of the annotation variance achieved by a linear regression model trained on ground-truth annotations.

The authors in [11] demonstrated that injecting persona descriptions into LLM prompts can produce more diverse, controllable annotations that align with the subjective differences as seen in human annotations. To introduce personality-affected emotion transition modeling for dialogue systems, one study framed response emotion selection as a personality-affected state transition in Valence-Arousal-Dominance (VAD) space where the emotion for response is obtained through the sum of preceding emotion and variation [12].

Work has also been performed to investigate the performance of closed LLM models on the ERC task. ChatGPT was evaluated on its emotional dialogue understanding and capabilities, including ERC, under zero-shot and few-shot prompting and analyzes misalignment with dataset annotation standards [13]. To evaluate open LLM models, [14] fine-tuned LLaMA-family models with instructions to improve ERC performance through two-stage learning that includes speaker characteristics and emotion recognition. Another study built instruction-tuned emotional LLMs while constructing a large Affective Analysis Instruction Dataset (AAID) and an Affective Evaluation Benchmark (AEB) covering multiple affective tasks [15]. To improve emotion classification performance, a long-context emotional intelligence benchmark spanning tasks including emotion classification was introduced while also proposing Retrieval Augmented Generation (RAG) and Collaborative Emotional Modeling (CoEM) strategies to improve performance [16].

III. Emotion Classification

Emotion classification is a subfield of NLP that involves the identification and classification of emotional content expressed in language. The problem often is formulated as a problem in which the model maps input, such as sentences, posts, or dialogue turns, to one or more emotion labels. Emotion labels may be annotated into discrete categories, such as anger, disgust, fear, joy, sadness, or surprise, or to various affective states as defined by various psychological frameworks. These emotion labels and affective states allow for granular analysis of language by providing a more detailed account for the subjective meaning behind the input.

There are many challenges that exist with the task of emotion classification. One input may convey many meanings where a deeper understanding may be required. For example, it is possible for a given input document to convey emotions by employing the use of sarcasm. Similarly, emotion classification in text must also account for emotional expression by way of emojis, code-switching, domain-specific vocabulary, and variation across cultures.

For the purpose of the work presented in this paper, we will approach the emotion classification task by considering the classification of an utterance by a speaker from a given conversation. Each utterance will be assigned a single emotion e_i from a discrete set of dataset-dependent target classes $e_i \in \{e_1, e_2, ..., e_k\}$.

IV. Datasets

Two datasets are used throughout the analyses performed in this work. The Multimodal EmotionLines Dataset (MELD) is a large-scale multimodal, multi-party emotional conversational dataset that was constructed from the TV-series *Friends*. The dataset includes both conversations and utterances with each utterance being assigned an emotion label: {surprise, anger, neutral, sadness, disgusting, joy, and fear} [17]. The interactive emotional dyadic motion capture database (IEMOCAP) dataset includes data from ten actors in dyadic sessions that included emotional scripts in hypothetical scenarios to elicit the following emotions: {excited, frustrated, neutral, sad, happy, and angry }. There were 151 recorded conversation videos where clips were spread across five sessions per actor. The frequency

of utterances for each class can be seen in Figure 1 for the MELD dataset and Figure 2 for the IEMOCAP dataset.

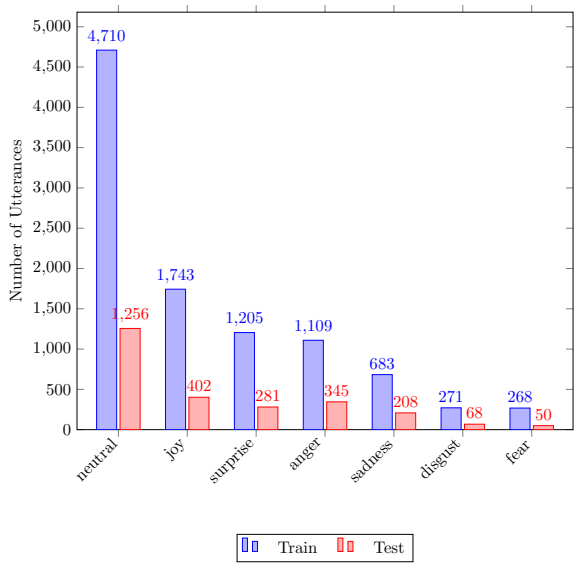


Figure 1. Emotion frequency for the labels in the MELD dataset.

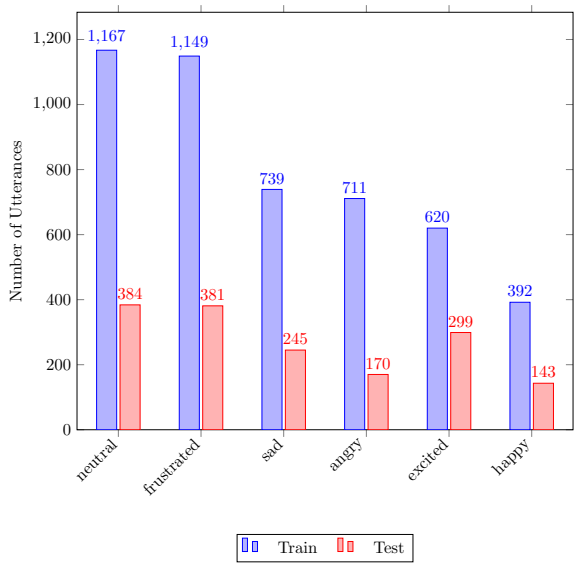


Figure 2. Emotion frequency for the labels in the IEMOCAP dataset.

V. Methodology

We evaluate the effects and impact of persona-conditioned emotion classification using LLMs through a comprehensive analysis using quantitative metrics. The goal of these experiments is to address the following Research Questions (RQs):

1) **RQ1**: Do measurable differences exist between persona-conditioned inputs in comparison to unconditioned, or neutrally-conditioned, inputs?
2) **RQ2**: How does predictive confidence compare to actual performance across models in persona-conditioned emotion classification?
3) **RQ3**: What variations can be observed across personas in the persona-conditioned emotion classification task?

Five personas were used throughout the experiments that follow. One baseline, or default, persona was used along with four personas were synthetically generated following other synthetic persona datasets to evaluate the effects of the models: *neutral*, *skeptical*, *empathic*, *social*, and *knowledgeable*. The *neutral* persona is defined as being a neutrally-conditioned, or unconditioned, persona where no additional context is provided to bias the model. The *skeptical* persona instructs the model to be skeptical and only perform a task when there exists strong emotions, while defaulting to neutral classifications otherwise. The *empathic* persona instructs the model to infer the primary emotion of the speaker from the text or context provided, even when it is subtle. The *social* persona instructs the model to assume expertise in conversational pragmatics and sarcasm while utilizing context in the conversation to detect implied emotion. The *knowledgeable* persona instructs the model to assume expertise in the given dataset topic domain while using the given context to decide the most probable emotion.

For each dataset, we generated $k = 5$ different sample sets comprised of $n = 500$ randomly selected samples each. The experimentation was conducted on four LLMs: GPT-4o, GPT-4.1, Llama 3.1 8B, and Gemma 3 12B. Llama 3.1 8B has demonstrated capability-to-efficiency trade-off with a 128K context window. Gemma 3 12B is beneficial for multimodal and multilingual capabilities with a 128K context window. GPT-4o supports text and image input with text output and a 128K context window to provide versatility in tasks. GPT-4.1 is a non-reasoning GPT model for instruction following and tool use.

To evaluate whether there exists statistical significance between personas and our baseline model, we used McNemar's test, which is a non-parametric significance test that is appropriate for paired nominal data by evaluating both systems on the same instances. Under the null hypothesis, the two models have the same misclassification, or error, rate from which the evaluation set is drawn. A statistically significant result would indicate that a model is more likely than the other to classify the same instances correctly.

Flip Rate (FR) is a measurement of the prediction instability as a result of perturbation. FR reflects the percentage of examples with label changes in a classification task due to controlled experimentation settings or levels. Given an input x_i and a perturbed version x_i', the flip rate reflects the proportion of instances when $f(x_i) \neq f(x_i')$. The label flip rate is defined for our purposes as being

$$\text{FR} = \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}\left[f(x_i) \neq f(x_i')\right] \tag{1}$$

Lower flip rates indicate greater prediction consistency, whereas higher flip rates indicate less stability under perturbation and greater sensitivity to input modifications. For our purposes, higher rates would serve as an indication for stronger persona-induced shifts.

 ISBN: 978-1-68558-387-3

Expected Calibration Error (ECE) is a metric for the evaluation of probabilistic calibration in models where it reflects the discrepancy between a model's predicted confidence and its empirical accuracy. ECE is defined as being

$$\text{ECE} = \sum_{m=1}^{M} \frac{|B_m|}{n} |\text{acc}(B_m) - \text{conf}(B_m)| \quad (2)$$

where B_m is the confidence of bin m, $\text{acc}(B_m)$ is the empirical accuracy of the given bin, $\text{conf}(B_m)$ is the mean confidence in that bin, and n is the total number of predictions. Lower ECE values serve as an indication that model's confidence scores are more reflective of the true probability of correctness, whereas higher values suggest a greater discrepancy or misalignment between confidence and actual predictive performance.

VI. Results

In this section, we present an analysis of the results from the experiments that were conducted.

A. Model Analysis

Our results demonstrate that persona conditioning affects both predictive performance and calibration. In Table II, we observe in GPT-4o and GPT-4.1 that the Empathic persona yields the best accuracy and macro-F1, improving the results substantially over the Neutral (baseline) persona in terms of accuracy (0.436 vs. 0.382) and macro-F1 (0.431 vs. 0.378) for GPT-4o, and in terms of accuracy (0.426 vs. 0.382) and macro-F1 (0.417 vs. 0.378) for GPT-4.1. For Gemma 3, the Knowledgeable persona achieves the best accuracy, macro-F1, and ECE scores of 0.483, 0.472, and 0.311, respectively. Llama produces the best results with the Social persona in terms of accuracy (0.515) and ECE (0.294), while the Knowledgeable persona achieves the best macro-F1 score (0.413).

As demonstrated in Table II, a persona-conditioned model generally outperformed the baseline model on the shared evaluation set for each LLM used in the experiments. McNemar's test on paired instance-level correctness found the difference to be significant as demonstrated in Table I at the significance level of $\alpha = 0.001$ using the exact p-value. This answers **RQ1** regarding the existence of measurable differences by infusing personas into the instructions for the emotion classification task using LLMs.

Model predictions are shown to be affected by persona conditioning in Figure 3. The effect is shown to differ between both models and persona pairings. The two closed models are shown to have more robust results with lower pairwise flip rates in comparison to the open source models, Gemma 3 and Llama, across both datasets. The *skeptical* persona demonstrates high prediction instability when compared to the knowledgeable and empathic personas, while moving from skeptical to neutral induces the smallest change. The effect is also observed as having a dataset-dependency given that Gemma 3 is sensitive to the IEMOCAP dataset whereas GPT-4.1 remains relatively stable in comparison. Neutral appears the most stable overall. We can also see the Empathic-Skeptical pairs often produce the largest disagreement.

B. Confidence Score Analysis

In this section, we investigate the impact of model confidence in persona-conditioned models for **RQ2** and **RQ3**. Figure 4 reflects the model's confidence with the given classification of the conversation utterance. With Gemma 3, we observe in the MELD dataset that the InterQuartile Range (IQR) was low for both the empathic and neutral personas of $IQR = 0$ and $IQR = 0.10$, $\bar{x} = 0.802$ and $0.\bar{7}89$, and $m = 0.800$ and $m = 0.800$ for correct classification, respectively. The confidence scores had the greatest range for the social persona in both correct and incorrect classifications where $IQR = 0.186$, $\bar{x} = 0.789$, $m = 0.800$ for correct classifications and $IQR = 0.200$, and $0.\bar{7}85$, and $m = 0.800$ for incorrect classifications. We observe a consistent range for the empathic, neutral, and skeptical personas, while the social persona has the greatest range and the knowledgeable persona has the smallest range. For the IEMOCAP dataset, we observed consistent outputs for Q1 and the median with the correct classification, whereas there were consistent outputs with the median and Q3 values with an incorrect classification. The social persona had had the largest range with more consistent and lower confidence scores with incorrect classifications.

Llama produced more consistent confidence scores with the IEMOCAP dataset with values consistently around the median score $m = 0.800$ and $IQR = 0$ for both correct and incorrect classifications with a mean score of $\bar{x} = 0.807$ and standard error of $SE = 0.004$. In the MELD dataset, the confidence scores for incorrect classifications were consistently about the median $m = 0.800$ for all personas with knowledgeable and neutral personas having a median score of $m = 0.900$ for correct classifications. The confidence scores for correct classifications had an increased IQR score when compared to the incorrect classifications where the incorrect classifications had an $IQR = 0$ for all personas. For correct classifications, we observed $IQR = 0$ for the skeptical persona, $IQR = 0.100$ for the empathic and social personas, and $IQR = 0.200$ for the knowledgeable and neutral personas, which also had an increase in median value compared to other classes.

The ranges and IQR metrics for confidence scores within the GPT models demonstrated greater consistency across both correct and incorrect classifications for the datasets. For the GPT-4o model, using the persona-by-correctness means, the mean confidence score was $\bar{x} = 0.739$ with $SE = 0.010$ in the IEMOCAP dataset. For correct classifications, the empathic, neutral, skeptical, and social personas each had $IQR = 0.100$; the empathic persona had a median of $m = 0.700$ and mean of $\bar{x} = 0.751$, the neutral persona had a median of $m = 0.700$ and mean of $\bar{x} = 0.761$, the skeptical persona had a median of $m = 0.800$ and mean of $\bar{x} = 0.778$, and the social persona had a median of $m = 0.800$ and mean of $\bar{x} = 0.771$. The knowledgeable persona had the largest spread for the correct classifications, with $IQR = 0.150$, a median of $m = 0.800$, and mean of $\bar{x} = 0.775$. For incorrect classifications, the empathic, knowledgeable, skeptical, and social personas each had $IQR = 0.100$ with the correct classification of emotions,

TABLE I. EXACT POOLED MCNEMAR p-VALUES COMPARING EACH PERSONA-CONDITIONED MODEL AGAINST THE NEUTRAL BASELINE. * INDICATES SIGNIFICANCE AT THE LEVEL OF $\alpha = 0.001$.

	IEMOCAP				MELD			
Persona	**GPT-4o**	**GPT-4.1**	**Llama**	**Gemma3**	**GPT-4o**	**GPT-4.1**	**Llama**	**Gemma3**
Empathic	3.96e-12*	4.73e-11*	0.195	0.00202*	8.15e-05*	3.71e-06*	1.81e-16*	3.17e-41*
Knowledgeable	3.83e-10*	4.03e-06*	1.41e-06*	1.38e-05*	1.73e-05*	0.00181*	4.78e-10*	1.58e-25*
Skeptical	4.27e-25*	1.73e-15*	0.283	6.21e-13*	2.99e-04*	0.163	2.69e-13*	0.873
Social	8.75e-05*	0.0535	8.01e-10*	0.349	1.87e-17*	1.57e-13*	1.10e-06*	1.63e-06*

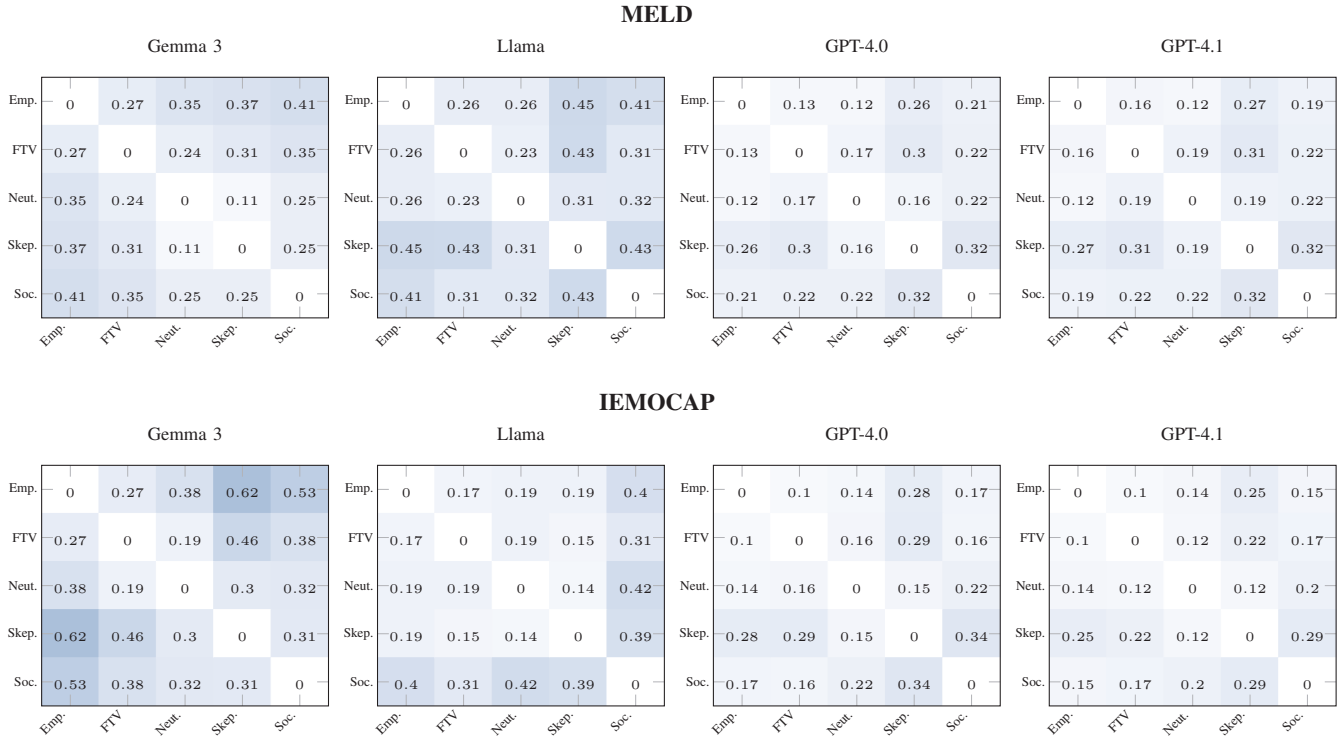


Figure 3. Pairwise persona flip-rate heatmaps by models.

while the neutral persona had the smallest spread with $IQR = 0$. For the MELD dataset, the mean confidence score was $\bar{x} = 0.758$ with $SE = 0.007$ in the dataset. The empathic, knowledgeable, neutral, and social personas each had $IQR = 0.150$; the empathic persona had a median of $m = 0.700$ and mean of $\bar{x} = 0.754$, the knowledgeable persona had a median of $m = 0.800$ and mean of $\bar{x} = 0.771$, the neutral persona had a median of $m = 0.700$ and mean of $\bar{x} = 0.756$, and the social persona had a median of $m = 0.800$ and mean of $\bar{x} = 0.768$. The skeptical persona had the largest spread for the correct classifications, with $IQR = 0.200$, a median of $m = 0.800$, and mean of $\bar{x} = 0.798$. For incorrect classifications, the knowledgeable and social personas each had $IQR = 0.150$, while the empathic, neutral, and skeptical personas each had $IQR = 0.100$.

For the GPT-4.1 model, the model produced a mean confidence score across classes and correctness of $\bar{x} = 0.802$ with $SE = 0.008$ in the IEMOCAP dataset. For correct classifications, $IQR = 0.150$ for the empathic persona with $m = 0.800$ and $\bar{x} = 0.812$, $IQR = 0.100$ for the knowledgeable and neutral personas with the knowledgeable persona having a median of $m = 0.85$ and mean $\bar{x} = 0.832$ while the neutral persona had a median of $m = 0.85$ and mean of $\bar{x} = 0.830$. The skeptical persona had an IQR metric of $IQR = 0.250$, which is the largest of the personas, with a median of $m = 0.85$ and mean of $\bar{x} = 0.833$. The social persona had an IQR score of $IQR = 0.150$, a median of $m = 0.800$ and mean of $\bar{x} = 0.807$. For the incorrect classes, the IQR scores for each persona were either the same (skeptical and social) or within a ± 0.050 difference (empathic, knowledgeable, and neutral). For the MELD dataset, the mean confidence score was $\bar{x} = 0.790$ with $SE = 0.012$ in the dataset. The empathic persona had $IQR = 0.200$ with a median of $m = 0.850$ and mean of $\bar{x} = 0.823$ for correct classifications. The knowledgeable persona had the smallest spread, with $IQR = 0.050$, a median of $m = 0.850$, and mean of $\bar{x} = 0.828$.

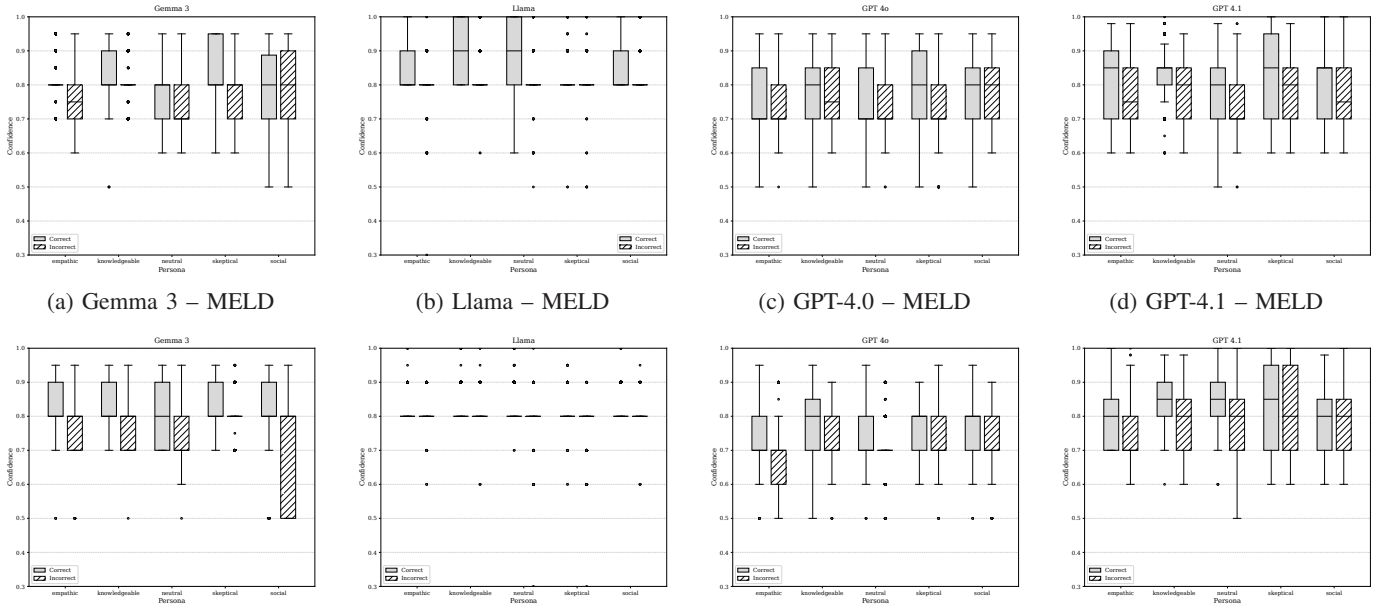
TABLE II. MEAN CLASSIFICATION ACCURACY, MACRO-F1, CONFIDENCE, AND EXPECTED CALIBRATION ERROR (ECE) ACROSS PERSONAS. † INDICATES THE BASELINE PERSONA.

Dataset	Model	Persona	Accuracy ↑	Macro-F1 ↑	Confidence ↑	ECE ↓
IEMOCAP	GPT-4o	Empathic	**0.4360 ± 0.0016**	**0.4306 ± 0.0046**	0.7166 ± 0.0037	**0.2806 ± 0.0041**
		Knowledgeable	0.4327 ± 0.0066	0.4273 ± 0.0026	0.7433 ± 0.0031	0.3106 ± 0.0107
		Neutral†	0.3820 ± 0.0130	0.3781 ± 0.0151	0.7207 ± 0.0009	0.3387 ± 0.0165
		Skeptical	0.2993 ± 0.0109	0.2861 ± 0.0206	**0.7463 ± 0.0037**	0.4469 ± 0.0097
		Social	0.4213 ± 0.0084	0.3993 ± 0.0021	0.7415 ± 0.0026	0.3201 ± 0.0125
		Overall	**0.3943 ± 0.0512**	**0.3843 ± 0.0527**	**0.7337 ± 0.0014**	**0.3394 ± 0.0100**
	GPT-4.1	Empathic	**0.4260 ± 0.0102**	**0.4174 ± 0.0057**	0.7853 ± 0.0010	**0.3593 ± 0.0132**
		Knowledgeable	0.4087 ± 0.0137	0.4069 ± 0.0115	0.8017 ± 0.0026	0.3984 ± 0.0201
		Neutral†	0.3773 ± 0.0115	0.3781 ± 0.0148	0.7993 ± 0.0049	0.4233 ± 0.0101
		Skeptical	0.3220 ± 0.0075	0.3173 ± 0.0094	**0.8183 ± 0.0065**	0.4963 ± 0.0080
		Social	0.3960 ± 0.0142	0.3751 ± 0.0143	0.7820 ± 0.0039	0.3900 ± 0.0131
		Overall	**0.3860 ± 0.0357**	**0.3790 ± 0.0348**	**0.7973 ± 0.0022**	**0.4135 ± 0.0113**
	Llama	Empathic	0.4187 ± 0.0213	0.3869 ± 0.0152	0.8078 ± 0.0041	0.3891 ± 0.0266
		Knowledgeable	0.4753 ± 0.0255	**0.4133 ± 0.0195**	0.8133 ± 0.0030	0.3379 ± 0.0304
		Neutral†	0.4307 ± 0.0165	0.4129 ± 0.0172	0.8074 ± 0.0024	0.3767 ± 0.0185
		Skeptical	0.4393 ± 0.0217	0.4039 ± 0.0250	0.7940 ± 0.0010	0.3550 ± 0.0260
		Social	**0.5147 ± 0.0146**	0.3946 ± 0.0102	**0.8088 ± 0.0009**	**0.2942 ± 0.0188**
		Overall	**0.4557 ± 0.0350**	**0.4023 ± 0.0103**	**0.8063 ± 0.0021**	**0.3505 ± 0.0219**
	Gemma 3	Empathic	0.4100 ± 0.0255	0.4055 ± 0.0250	0.7892 ± 0.0047	0.3792 ± 0.0272
		Knowledgeable	**0.4833 ± 0.0157**	**0.4719 ± 0.0101**	0.7938 ± 0.0050	**0.3105 ± 0.0143**
		Neutral†	0.4453 ± 0.0077	0.4432 ± 0.0035	0.7715 ± 0.0041	0.3262 ± 0.0081
		Skeptical	0.3680 ± 0.0071	0.3647 ± 0.0094	**0.8100 ± 0.0048**	0.4420 ± 0.0131
		Social	0.4333 ± 0.0159	0.3908 ± 0.0118	0.7637 ± 0.0057	0.3304 ± 0.0149
		Overall	**0.4280 ± 0.0382**	**0.4152 ± 0.0380**	**0.7856 ± 0.0043**	**0.3576 ± 0.0093**
MELD	GPT-4o	Empathic	0.6240 ± 0.0240	0.5075 ± 0.0264	0.7488 ± 0.0444	0.1376 ± 0.0600
		Knowledgeable	0.6160 ± 0.0166	0.5073 ± 0.0194	0.7675 ± 0.0386	0.1571 ± 0.0533
		Neutral†	**0.6500 ± 0.0213**	**0.5108 ± 0.0259**	0.7470 ± 0.0462	**0.1038 ± 0.0580**
		Skeptical	0.6225 ± 0.0216	0.4373 ± 0.0258	**0.7705 ± 0.0250**	0.1564 ± 0.0398
		Social	0.5765 ± 0.0222	0.4806 ± 0.0186	0.7702 ± 0.0368	0.2104 ± 0.0489
		Overall	**0.6178 ± 0.0237**	**0.4887 ± 0.0279**	**0.7608 ± 0.0382**	**0.1446 ± 0.0553**
	GPT-4.1	Empathic	0.6070 ± 0.0215	0.5049 ± 0.0289	0.8012 ± 0.0165	0.1963 ± 0.0330
		Knowledgeable	0.6115 ± 0.0140	**0.5237 ± 0.0257**	0.8074 ± 0.0163	0.2018 ± 0.0116
		Neutral†	0.6385 ± 0.0225	0.5150 ± 0.0243	0.7845 ± 0.0107	**0.1490 ± 0.0316**
		Skeptical	**0.6500 ± 0.0273**	0.5133 ± 0.0296	**0.8290 ± 0.0053**	0.1859 ± 0.0271
		Social	0.5725 ± 0.0170	0.4874 ± 0.0201	0.7623 ± 0.0309	0.2371 ± 0.0123
		Overall	**0.6159 ± 0.0270**	**0.5089 ± 0.0123**	**0.7969 ± 0.0149**	**0.1927 ± 0.0222**
	Llama	Empathic	0.4307 ± 0.0098	0.4002 ± 0.0181	0.8128 ± 0.0036	0.3848 ± 0.0089
		Knowledgeable	0.4553 ± 0.0096	0.4171 ± 0.0028	**0.8521 ± 0.0044**	0.4021 ± 0.0107
		Neutral†	0.5167 ± 0.0172	0.4213 ± 0.0084	0.8459 ± 0.0033	0.3439 ± 0.0218
		Skeptical	**0.6053 ± 0.0282**	**0.4327 ± 0.0178**	0.8149 ± 0.0044	**0.2163 ± 0.0351**
		Social	0.4613 ± 0.0066	0.3822 ± 0.0038	0.8299 ± 0.0025	0.3752 ± 0.0071
		Overall	**0.4939 ± 0.0624**	**0.4107 ± 0.0177**	**0.8311 ± 0.0032**	**0.3444 ± 0.0167**
	Gemma 3	Empathic	0.4690 ± 0.0052	0.4205 ± 0.0174	0.7771 ± 0.0010	0.3081 ± 0.0067
		Knowledgeable	0.5220 ± 0.0081	0.4537 ± 0.0129	0.8151 ± 0.0015	0.2946 ± 0.0113
		Neutral†	0.6170 ± 0.0184	0.4770 ± 0.0240	0.7698 ± 0.0021	**0.1528 ± 0.0214**
		Skeptical	**0.6185 ± 0.0221**	**0.4790 ± 0.0199**	**0.8245 ± 0.0028**	0.2060 ± 0.0236
		Social	0.5710 ± 0.0267	0.4468 ± 0.0160	0.7873 ± 0.0034	0.2173 ± 0.0332
		Overall	**0.5595 ± 0.0575**	**0.4554 ± 0.0215**	**0.7948 ± 0.0015**	**0.2356 ± 0.0177**

The neutral and social personas each had $IQR = 0.150$; the neutral persona had a median of $m = 0.800$ and mean of $\bar{x} = 0.807$, while the social persona had a median of $m = 0.850$ and mean of $\bar{x} = 0.783$. The skeptical persona had the largest spread for the correct classifications, with $IQR = 0.250$, a median of $m = 0.850$, and mean of $\bar{x} = 0.853$. For incorrect classifications, the empathic, knowledgeable, skeptical, and social personas each had $IQR = 0.150$, while the neutral persona had the smallest spread with $IQR = 0.100$.

VII. CONCLUSION AND FUTURE WORK

The work presented in this paper demonstrates that persona conditioning of models is a significant control variable for emotion classification as opposed to a superficial variation of prompts. While the results produced statistically significant results to demonstrate that persona-conditioning can bias the model that alters its performance with the emotion classification task, it is important to observe that there does not exist uniformity across settings as the best persona depended on both



(a) Gemma 3 – MELD (b) Llama – MELD (c) GPT-4.0 – MELD (d) GPT-4.1 – MELD

(e) Gemma 3 – IEMOCAP (f) Llama – IEMOCAP (g) GPT-4.0 – IEMOCAP (h) GPT-4.1 – IEMOCAP

Figure 4. Boxplots for each model on MELD (top) and IEMOCAP (bottom).

the model family and dataset. The results also indicate that some personas can increase confidence without improving correctness, and sometimes this can lead to severe miscalibration. Our results indicate that persona conditioning and selection should be utilized as a tunable modeling choice with evaluation including metrics to evaluate calibration in addition to accuracy and macro-F1.

Future work in this space could investigate the biases that associated with personas and how they shift the classification task as a result. The task could also be applied to natural dialogues as opposed to scripted language from TV sources to better understand the impact of personas on real data. In addition, given that persona declarations in prompts often contain emotion words or context that leak signal, these factors which contribute to emotion classification task should be considered to understand how the contribute to certain emotions, such as through the deployment of hold-out paraphrases, shuffling personas, etc. Other directions may consider speaker-receptor relations, cultural specifics, or environmental sets.

REFERENCES

[1] Y. Liu, J. Zhao, J. Hu, R. Li, and Q. Jin, "DialogueEIN: Emotion interaction network for dialogue affective analysis", in *Proceedings of the 29th International Conference on Computational Linguistics*, N. Calzolari et al., Eds., Gyeongju, Republic of Korea: International Committee on Computational Linguistics, Oct. 2022, pp. 684–693.

[2] S. Poria, N. Majumder, R. Mihalcea, and E. Hovy, *Emotion recognition in conversation: Research challenges, datasets, and recent advances*, 2019. arXiv: 1905.02947 [cs.CL].

[3] J. Li et al., "A persona-based neural conversation model", in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, K. Erk and N. A. Smith, Eds., Berlin, Germany: Association for Computational Linguistics, Aug. 2016, pp. 994–1003. DOI: 10.18653/v1/P16-1094.

[4] S. Zhang et al., "Personalizing dialogue agents: I have a dog, do you have pets too?", in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, I. Gurevych and Y. Miyao, Eds., Melbourne, Australia: Association for Computational Linguistics, Jul. 2018, pp. 2204–2213. DOI: 10.18653/v1/P18-1205.

[5] A. Mackey, S. Gauch, and I. Cuevas, "Prompt distillation for emotion analysis", in *Proceedings of the 16th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management - Volume 1: KDIR*, INSTICC, SciTePress, 2024, pp. 328–334, ISBN: 978-989-758-716-0. DOI: 10.5220/0012951200003838.

[6] H. Zhou, M. Huang, T. Zhang, X. Zhu, and B. Liu, *Emotional chatting machine: Emotional conversation generation with internal and external memory*, 2018. arXiv: 1704.01074 [cs.CL].

[7] P. Liu et al., "Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing", *ACM Comput. Surv.*, vol. 55, no. 9, Jan. 2023, ISSN: 0360-0300. DOI: 10.1145/3560815.

[8] T. Brown et al., "Language models are few-shot learners", in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., vol. 33, Curran Associates, Inc., 2020, pp. 1877–1901.

[9] T. Gao, A. Fisch, and D. Chen, *Making pre-trained language models better few-shot learners*, 2021. arXiv: 2012.15723 [cs.CL].

[10] T. Hu and N. Collier, "Quantifying the persona effect in LLM simulations", in *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, L.-W. Ku, A. Martins, and V. Srikumar, Eds., Bangkok, Thailand: Association for Computational Linguistics, Aug. 2024, pp. 10 289–10 307. DOI: 10.18653/v1/2024.acl-long.554.

[11] L. Fröhling, G. Demartini, and D. Assenmacher, *Personas with attitudes: Controlling llms for diverse data annotation*, 2024. arXiv: 2410.11745 [cs.CL].

[12] Z. Wen, J. Cao, R. Yang, S. Liu, and J. Shen, "Automatically select emotion for response via personality-affected emotion transition", in *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, C. Zong, F. Xia, W. Li, and R. Navigli, Eds., Online: Association for Computational Linguistics, Aug. 2021, pp. 5010–5020. DOI: 10.18653/v1/2021.findings-acl.444.

[13] W. Zhao et al., *Is chatgpt equipped with emotional dialogue capabilities?*, 2023. arXiv: 2304.09582 [cs.CL].

[14] Y. Fu et al., *Laerc-s: Improving llm-based emotion recognition in conversation with speaker characteristics*, 2025. arXiv: 2403.07260 [cs.CL].

[15] Z. Liu, K. Yang, Q. Xie, T. Zhang, and S. Ananiadou, "Emollms: A series of emotional large language models and annotation tools for comprehensive affective analysis", in *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, ser. KDD '24, ACM, Aug. 2024, pp. 5487–5496. DOI: 10.1145/3637528.3671552.

[16] W. Liu et al., *Longemotion: Measuring emotional intelligence of large language models in long-context interaction*, 2025. arXiv: 2509.07403 [cs.CL].

[17] S. Poria et al., "MELD: A multimodal multi-party dataset for emotion recognition in conversations", in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, A. Korhonen, D. Traum, and L. Màrquez, Eds., Florence, Italy: Association for Computational Linguistics, Jul. 2019, pp. 527–536. DOI: 10.18653/v1/P19-1050.