RETROFIT-CQ Revisited: Leveraging Ontology Triples and Context in Few-Shot LLM Prompting

Meiqing Li[1,†], Reham Alharbi[1,†], Jacopo de Berardinis[1,†], Valentina Tamma[1,†] and Terry R. Payne[1,†]

[1]University of Liverpool, Liverpool L69 7ZX, UK

Abstract

This paper investigates the role of structured data—specifically ontology triples—in prompting large language models for the task of retrofitting competency questions from ontologies. Building on RETROFIT-CQ, our previous work that introduces a zero-shot method for generating competency questions from ontology triples, we explore how few-shot prompting can enhance the quality and contextual alignment of the generated questions. We incorporate examples of competency questions, ontology URIs, and textual descriptions into the prompts to evaluate how different combinations of structured and contextual data influence large language models performance. We empirically evaluate this few-shot approach on a selection of benchmark ontologies (Video Game, African Wildlife, and Vicinity Core) which were originally used to evaluate the previous zero-shot prompt. Our experiments demonstrate that few-shot prompting helps in reducing overgeneralisation and in improving semantic alignment.

Keywords

Competency Question, Prompting Techniques, Requirement Engineering

1. Introduction

Competency Questions (CQs) play a central role in ontology engineering. They serve multiple purposes across the ontology lifecycle, from supporting requirements elicitation in the early stages of development [1, 2, 3, 4, 5], to facilitating verification and validation during testing [6, 7], and even enabling ontology reuse by recommending suitable candidate ontologies [8, 9, 10]. Despite their importance, authoring CQs remains a non-trivial task for both ontology engineers and domain experts [11, 12, 13], and is often based on traditional knowledge elicitation approaches, i.e. card sorting or 20 questions [14].

Recent advances in Artificial Intelligence, particularly with the rise of Large Language Models (LLMs), have created new opportunities to support the authoring of CQs. For example, LLMs have been used to address challenges such as grammar correction and language variability, which often hinder CQ formulation [15]. Several recent efforts have leveraged LLMs for CQ generation: AgoCQs [15] explores generation from a corpus of text describing a domain; RevOnt [16] uses Wikidata as a source of background knowledge; OntoChat [17] facilitates dialogue-based CQ creation through interactive agents; and RETROFIT-CQ[18] generates questions for ontologies whose CQs have not been published with the ontology itself by leveraging ontology triples. A similar approach is also adopted in [19].

Although LLMs mitigate several traditional limitations in different AI tasks, their use introduces new research challenges and opportunities, particularly arising from variability across model architectures, parameter configurations, and prompt engineering strategies. Previous studies have explored the use of both open-source and proprietary LLMs, and examined parameters such as temperature, which can influence hallucination and creativity [20, 21]. However, one critical and underexplored aspect remains: prompting techniques, i.e. the strategies used to instruct LLMs.

RAGE-KG 2025: The Second International Workshop on Retrieval-Augmented Generation Enabled by Knowledge Graphs, co-located with ISWC 2025, November 2–6, 2025, Nara, Japan

*Corresponding author.

†These authors contributed equally.

sgmli30@liverpool.ac.uk (M. Li); reham.alharbi@liverpool.ac.uk (R. Alharbi); jacopo.deberardinis@liverpool.ac.uk (J. d. Berardinis); valentina.tamma@liverpool.ac.uk (V. Tamma); T.R.Payne@liverpool.ac.uk (T. R. Payne)

0000-0002-8332-3803 (R. Alharbi); 0000-0001-6770-1969 (J. d. Berardinis); 0000-0002-1320-610X (V. Tamma); 0000-0001-8419-6554 (T. R. Payne)

Prompting governs how humans interact with LLMs, typically through instructions or examples [22, 23, 24]. Common strategies include zero-shot prompting, where the model is asked to perform a task without any examples, few-shot prompting, where a small number of examples are provided, and more structured approaches like chain-of-thought prompting, which guide reasoning steps. While prior CQ generation studies have used zero-shot prompts extensively [18, 19, 16, 15, 25], and some have explored chain-of-thought prompting [17], few-shot prompting remains largely uninvestigated in this context.

In contrast, fields such as education have widely adopted few-shot prompting, demonstrating its effectiveness in generating high-quality, context-sensitive questions in reading comprehension or competency-based assessments [26, 27, 28]. These findings motivate our exploration of few-shot prompting for CQ generation.

In this study, we buid upon our previous method, RETROFIT-CQ, which generates CQs from structured RDF triples extracted from ontologies using tailored prompts for LLMs. RETROFIT-CQ has shown encouraging results in generating CQs that reflect the intent and content of manually authored CQs across a range of existing ontologies [21, 20, 18]. We extend this method by introducing few-shot prompting and compare its effectiveness with zero-shot prompting.

Our exploratory results indicate that few-shot prompting yields modest yet meaningful improvements in the quality of generated CQs, notably by reducing overgeneralisation and producing more concise, context-sensitive formulations. However, these improvements are not statistically significant at this stage. We therefore suggest that further investigation is required across a broader range of LLMs and ontologies to determine whether the benefits of few-shot prompting justify the associated computational and cost overhead—particularly in domains where such advantages may be more pronounced, such as education.

The remainder of this paper is structured as follows: Section 2 discusses related work, while Section 3 details the methodology used for this exploratory study, including the experimental design and evaluation metrics. We present the results in Section 4 and discuss key findings and limitations in Section 5. Finally, Section 6 concludes the paper and outlines directions for future work.

2. Related work

Several studies have investigated the generation of CQs, exploring a spectrum of approaches. Traditional approaches often rely on close interaction with domain experts and ontology engineers to manually craft CQs (e.g., [1, 14]). More recent approaches either continue to involve human input while incorporating automated techniques (LLM) [17], or aim to partially replace human involvement by using external knowledge resources along with LLM to support the CQ generation process [18, 19, 16, 15, 25].

Interaction with LLMs is typically carried out through prompting techniques. While various prompting strategies exist [22, 23, 24], most previous approaches for CQ generation have primarily used zero-shot prompting [18, 19, 16, 15, 25, 17]. To the best of our knowledge, few-shot prompting (i.e., providing illustrative examples within the prompt) has not yet been applied in methods for generating CQs in the context of ontology engineering. However, in related domains such as automatic question generation for educational applications [29], few-shot prompting has shown promising results, particularly in improving question relevance and quality. For example, in [26], the authors showed that few-shot prompting improves model performance in generating higher-order questions for comprehensive reading assessment compared to zero-shot prompting. Additionally, [27] explored a few-shot prompting strategy for controllable question generation in narrative comprehension. Their results demonstrate that questions generated with attribute-specific guidance closely match the corresponding ground-truth questions.

In [28], the authors proposed a method for extracting text from PDF documents, chunking it into 500-word segments, and preprocessing it to build a KG that captures contextual information. This is followed by generating competency questions for educational assessment using both zero and few shot prompts. However, the KG's role in competency questions generation is unclear: it is not fully specified whether the full KG or a subgraph is passed in the prompt. For example, in the few-shot prompt, the

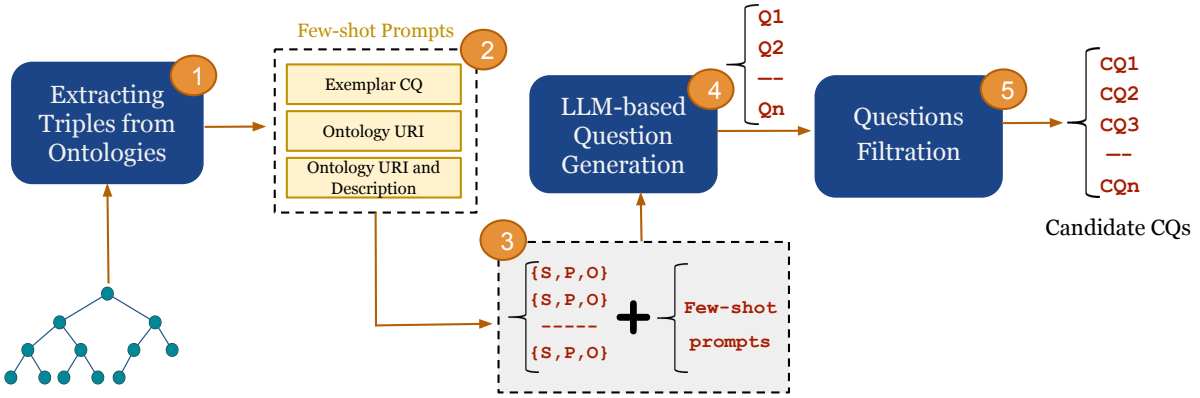

Figure 1: The diagram illustrates RETROFIT-CQ architecture. The arrows indicate the transition between each stage.

system is given a text describing operational waste management along with five example questions distilled from the text, but there is no explicit mention of the KG in the prompt. A key takeaway is that few-shot prompting improves the model's ability to generate contextually relevant questions aligned with the source content. This suggests a promising direction for investigating its applicability in generating CQs aimed at capturing ontology requirements.

We evaluate the CQs generated using few-shot prompting against the original CQs used in the ontology construction process for each of the ontologies used in our experiments. There is a lack of consensus within the ontology engineering community on what constitutes a *"good"* CQ [13, 12, 11] and on what are appropriate methods to evaluate their quality. We therefore compare the results of this extended approach using as baseline zero-shot prompting in [20, 21]. Through this comparative evaluation, we aim to identify strengths and limitations of each prompting approach, understand their impact on the CQs generated, and highlight promising directions for future research focused on improving competency question generation in ontology engineering.

3. Retrofit-CQ with Few-shot Prompting

The RETROFIT-CQ approach [18] addresses the absence of published CQs for a given ontology. Ideally, the statements within an ontology are derived from a set of CQs formulated by ontology engineers as part of the ontology construction process, following some of the most prominent ontology engineering methodologies [5, 1, 3]. However, often these CQs are not made available, either as part of the ontology documentation or in publications about the ontology. The goal of the RETROFIT-CQ pipeline is to reverse this process, reconstructing the CQs from the ontology's existing statements, which are represented as triples. The new pipeline, utilising few-shot prompting, is organised into five main phases as presented in Figure 1: (i) Triples are extracted from the ontology; (ii) Few-shot prompts are constructed by combining the CQ definition with various contextual elements (such as exemplar CQs, the ontology URI, and its description—discussed later in this section); (iii) These triples are embedded into the few-shot prompt to create input queries for the language model; (iv) The language model is then queried using these prompts to generate a diverse set of questions; and (v) The generated questions are filtered to remove duplicates and irrelevant entries, producing the final set of *candidate CQs*.

Previous studies present an empirical analysis of the RETROFIT-CQ approach conducted across various LLMs [18, 20, 21]. The results of these studies supports the claim that LLMs, when provided with structured input (in the form of ontology triples) and carefully designed prompts, can effectively generate valid CQs with high recall, measured by how well they matched the existing CQs for each ontology included in the experiment. In particular, [20, 21] assess the impact of both the creativity parameter (temperature) and the degree of contextual content included in zero-shot prompts used with both closed-source LLMs (e.g., *gpt-3.5-turbo* and *gpt-4*) and open-source models (e.g., *Flan-T5*, *Mistral*,

and *LLaMA*). A key insight from previous studies is that increasing the level of contextual information in zero-shot prompts can improve the precision of the generated CQs.

In this study, we therefore explore whether few-shot prompts with varying contextual elements affect the quality of the generated candidate CQs compared to zero-shot prompts. One of these contextual elements is the *role* and we distinguish between *system* and *user* roles, as shown in the following prompts:

```
messages = ["role": "system", "content": ("You are an ontology engineer working
on a project to develop a new ontology in the [DOMAIN]. Your task is to generate
competency questions (CQs) based on RDF triples from the ontology schema. A CQ is
a natural language question that can be answered using the information modelled in
the ontology. CQs help define the scope, purpose, and evaluation criteria for the
ontology.")]
```

The *system* role establishes that the LLM should act as if it was an ontology engineer tasked with generating CQs from RDF triples, ensuring responses remain focused on ontology design and CQ generation.

```
"role": "user", "content": (
"Here are some examples of ontology triples and the corresponding CQs:
Subject: [EXAMPLE_SUBJECT_1]
Predicate: [EXAMPLE_PREDICATE_1]
Object: [EXAMPLE_OBJECT_1]
Generated question: [EXAMPLE_CQ_1]
--------
--------
Subject: [EXAMPLE_SUBJECT_n]
Predicate: [EXAMPLE_PREDICATE_n]
Object: [EXAMPLE_OBJECT_n]
Generated question: [EXAMPLE_CQ_n]
Now, based on the RDF triple below, generate one or more relevant CQs:
Subject: {subject}
Predicate: {predicate}
Object: {object})"
```

The *user* role feeds the LLM with sample RDF triples with their corresponding CQs, then requests new CQs for a specific triple. This guides the model by providing the expected input-output format and framing the task. We also experimented with two variants of the *user* role, by including in the content 1) the ontology URI and 2) the ontology URI and the ontology description. The former simulates browsing or understanding of the knowledge model and helps evaluate how external context and schema patterns influence the quality and relevance of the generated CQs.

```
"role": "user", "content": (
"Here are some examples of ontology triples and the corresponding CQs:
Ontology URI: [ONTOLOGY_URI]
Subject: [EXAMPLE_SUBJECT_1]
Predicate: [EXAMPLE_PREDICATE_1]
Object: [EXAMPLE_OBJECT_1]
Generated question: [EXAMPLE_CQ_1]
--------
--------
Subject: [EXAMPLE_SUBJECT_n]
Predicate: [EXAMPLE_PREDICATE_n]
Object: [EXAMPLE_OBJECT_n]
Generated question: [EXAMPLE_CQ_n]
```

```
Now, based on the RDF triple below, generate one or more relevant CQs:
Subject: {subject}
Predicate: {predicate}
Object: {object})"
```

The latter includes the ontology description to provide additional context. This enhancement enables the language model to better understand the purpose, scope, and semantics of the ontology, leading to more accurate and meaningful CQs.

```
"role": "user", "content": (
"Below are some examples of ontology schema-based competency question generation,
derived from a vocabulary for describing [Ontology_DESCRIPTION].
The Ontology defines terms for describing [EXAMPLES_OF_CONTENT], including their
relationships and properties."
Ontology URI: [ONTOLOGY_URI]
Subject: [EXAMPLE_SUBJECT_1]
Predicate: [EXAMPLE_PREDICATE_1]
Object: [EXAMPLE_OBJECT_1]
Generated question: [EXAMPLE_CQ_1]
--------
--------
Subject: [EXAMPLE_SUBJECT_n]
Predicate: [EXAMPLE_PREDICATE_n]
Object: [EXAMPLE_OBJECT_n]
Generated question: [EXAMPLE_CQ_n]
Now, based on the RDF triple below, generate one or more relevant CQs:
Subject: {subject}
Predicate: {predicate}
Object: {object})"
```

We evaluate selected LLMs that were used in earlier work: for this exploratory study, we limit our selection to one closed-source model and one open-source model, i.e. *gpt-4* and *Flan-T5*. [1] The configurations of these LLMs use default settings to maintain comparability with zero-shot prompting in previous studies. For the dataset in this experiment, we selected three ontologies previously used in [18, 20, 21]. Two of these (Video Game, and VICINITY Core) were sourced from the CORAL repository [30], while the third (African Wildlife) was used in [31]. The ontologies were randomly selected from those that satisfy the following criteria: (i) the ontologies were produced by different developers (CQ style); (ii) they represent various domains (diversity); and (iii) each had a significant number of published CQs (significance).

To validate the candidate CQs generated by our approach, we compare them against the original baseline CQ for each ontology. This comparison is performed by embedding both the original and candidate CQs using SBERT [32], and then computing the cosine similarity between their embedding vectors. We report performance metrics similar to those used in previous studies, in particular; the *precision (Prec)* and *recall (Rec)* metrics are based on determining the number of CQs in the relevant original baseline CQ dataset - CQ^E, and the *candidate* CQ set - CQ^C, which correspond to the filtered CQs generated by the LLMs. The metrics used are given below, where $CQ^V \subseteq CQ^C$ is the set of candidate CQs that are assessed as having a similar meaning to those in the baseline CQ (CQ^E) according to *SBERT*, such that the cosine similarity is ≥ 0.7 (i.e. true positives); and the set of unmatched CQs (CQ^U) is the relative complement of the set CQ^V with respect to the set of baseline CQs, such that $CQ^U = CQ^E \setminus CQ^V$ (i.e. the CQs in the baseline set that are not in the set of true positives). The similarity threshold, 0.7,

[1] Comprehensive experimentation across a broader range of LLMs is left for future work.

was experimentally determined as the most discriminant, whilst allowing some variance between the questions.

Precision (*Prec*): This is the ratio of the number of *True Positives* ($|CQ^V|$) and the sum of both *True Positives* and *False Positives*, i.e. all of *Candidate CQs* ($|CQ^C|$).

$$Prec. = \frac{|CQ^V|}{|CQ^C|} \tag{1}$$

Recall (*Rec*): Also known as *sensitivity*, this is the ratio of the number of *True Positives* ($|CQ^V|$) and the sum of *True Positives* ($|CQ^V|$) and *False Negatives* ($|CQ^U|$) corresponding to the unmatched CQs in the baseline dataset.

$$Rec. = \frac{|CQ^V|}{|CQ^V| + |CQ^U|} \tag{2}$$

4. Results

This section presents a comparative evaluation of RETROFIT-CQ using both zero-shot and few-shot prompting strategies with two LLMs: *GPT-4* and *Flan-T5*. In the few-shot setting, we examine three types of contextual input, each incrementally building on the previous one during prompting: (i) CQ examples, (ii) CQ examples with an ontology URI, and (iii) CQ examples with an ontology URI and a description. The evaluation spans three ontologies— Video Game, African Wildlife, and Vicinity Core— and the generated CQs are compared against a gold-standard set of existing CQs for each ontology, measuring *Precision* (*Prec*) and *Recall* (*Rec*).

4.1. Few-Shot Prompting Results

Table 1 summarises the results of few-shot prompting. Overall, few-shot prompting improves performance across both models by enhancing *Prec* while maintaining or improving *Rec*. *Flan-T5* performs best when provided with CQ examples; its performance declines as the context becomes more abstract, particularly in complex domains. In contrast, *GPT-4* performs consistently well across all contexts, with CQ examples and CQ examples+URI+description yielding similarly strong results. This suggests *GPT-4's* robust ability to integrate both symbolic inputs (e.g., URIs) and natural language descriptions, despite not having access to external web data via an API.

Video Game Ontology *GPT-4* outperforms *Flan-T5* across all prompting contexts. Its highest score comes with CQ examples+URI+description input (*Prec* = 0.9921, *Rec* = 0.9766), indicating well-balanced performance. *Flan-T5* performs best with CQ examples (*Prec* = 0.9825, *Rec* = 0.9333), followed closely by CQ examples+URI+description (*Prec* = 0.9298, *Rec* = 0.9298). Its weakest performance occurs with the CQ examples+URI (*Prec* = 0.8596), highlighting the model's reliance on richer input. *GPT-4*, by contrast, consistently achieves high precision and recall across all contexts, demonstrating greater robustness to input variation.

African Wildlife Ontology Both models perform strongly on this ontology, likely due to its semantic simplicity. *GPT-4* achieves perfect *Rec* (1.0) across all settings and its best *Prec* (0.9691) with CQ examples. Differences across contexts are minimal, indicating *GPT-4's* ability to generalise with limited structured input. *Flan-T5* also achieves its highest *Prec* (0.9130) with CQ examples. While all settings yield perfect recall (1.0), precision varies—suggesting that the model sometimes generates more plausible but incorrect CQs when aiming for high coverage.

Vicinity Core Ontology This ontology proves most challenging, likely due to its complexity and specificity. *Flan-T5's* precision drops significantly—from (0.6603) with CQ examples to (0.6055) when CQs are combined with URIs. This indicates that adding URIs may introduce noise or irrelevant information, reducing the model's ability to focus on the core question pattern. *GPT-4* remains more stable, with *Prec* ranging from 0.8553 to 0.8274 across prompting strategies. Its highest score (*Prec* = 0.8553) occurs with the CQ examples+URI+description input, suggesting a strong ability to integrate structured inputs like URIs with accompanying natural language. This reflects *GPT-4's* capacity to generalize from prior exposure to symbolic and descriptive patterns, even without access to external web data or real-time resolution.

Table 2 shows example CQs generated for the triple *(Virtuosity subClassOf Achievement)* in the Video Game Ontology using *GPT-4*.[2] The quality of generated CQs improves with the richness of contextual input. With CQ examples alone, the output tends to be broad and generic. Adding the ontology URI leads to better structural alignment, while combining the URI with a textual description results in the most accurate and semantically grounded CQs. These findings emphasise the value of context in helping LLMs produce high-quality CQs for ontology engineering.

4.2. Few-Shot vs. Zero-Shot Performance Comparison

Table 3 compares zero-shot and few-shot prompting using each model's best-performing few-shot setting. The goal is to understand how few-shot guidance affects the quality of generated CQs, particularly in comparison to a zero-shot setup. For *Flan-T5*, few-shot prompting yields a clear improvement in precision while maintaining high recall— particularly beneficial in complex or ambiguous domains

[2]all results are available in https://github.com/Li563313/Question-generation

Table 1
Few-shot performance of RETROFIT-CQ using *Flan-T5* and *GPT-4* across three ontologies, under varying context settings (CQ example, CQ + URI, CQ + URI + Description, and Zero-shot). Metrics include counts of candidate CQs (CCQ) and validated CQs (VCQ), along with precision (Prec) and recall (Rec).

Ontology	LLMs	Context	CCQ	VCQ	Prec	Rec
Video Game	Flan-T5-l	CQ example	57	56	0.9825	0.9333
		CQ + URI	57	49	0.8596	0.9245
		CQ + URI + Desc	57	53	0.9298	0.9298
		Zero-shot	57	49	0.4622	0.8596
	GPT-4	CQ example	140	131	0.9357	0.9924
		CQ + URI	108	100	0.9259	0.9804
		CQ + URI + Desc	126	125	0.9921	0.9766
		Zero-shot	1178	839	0.7122	0.9976
African Wildlife	Flan-T5-l	CQ example	46	42	0.9130	1.0000
		CQ + URI	46	41	0.8913	1.0000
		CQ + URI + Desc	46	41	0.8913	1.0000
		Zero-shot	46	32	0.6956	1.0000
	GPT-4	CQ example	97	94	0.9691	1.0000
		CQ + URI	88	81	0.9205	1.0000
		CQ + URI + Desc	89	84	0.9438	1.0000
		Zero-shot	442	360	0.8145	1.0000
Vicinity Core	Flan-T5-l	CQ example	365	241	0.6603	0.9918
		CQ + URI	365	221	0.6055	0.9910
		CQ + URI + Desc	365	240	0.6575	0.9917
		Zero-shot	365	211	0.5780	0.9859
	GPT-4	CQ example	279	237	0.8495	1.0000
		CQ + URI	307	254	0.8274	1.0000
		CQ + URI + Desc	235	201	0.8553	0.9950
		Zero-shot	5088	3495	0.6869	1.0000

Table 2
Example CQs generated by RETROFIT-CQ under different context settings—CQ example only, CQ with URI, CQ with URI and description, and zero-shot—for the triple *(Virtuosity subClassOf Achievement)* in the Video Game Ontology using GPT-4.

Context	Generated CQs
CQ example	1. What kind of achievement is Virtuosity? 2. Is Virtuosity a type of Achievement in the game? 3. What are the sub-classes of Achievement in the game? 4. Under which category does Virtuosity fall in the game achievements?
CQ + URI	1. What type of achievement is Virtuosity?
CQ + URI + Desc	1. What type of achievement is 'Virtuosity' in the game? 2. Can you list all the subclasses of the 'Achievement' class? 3. Is 'Virtuosity' considered an achievement in the game?
Zero-shot	1. What are the characteristics of Virtuosity? 2. Can Virtuosity be considered as a sub-class of Achievement? 3. What are the different types of Achievements that one can attain? 4. Is there a hierarchy in the classification of Achievements? 5. Are there any other sub-classes under Achievement apart from Virtuosity? 6. Can Virtuosity exist without Achievement? 7. What are the prerequisites for Virtuosity? 8. Are there any levels or stages in Virtuosity? 9. What are the attributes associated with Virtuosity? 10. Can Virtuosity be considered as an Achievement in all contexts?

Table 3
Comparison of few-shot and zero-shot performance of RETROFIT-CQ using *Flan-T5* and *GPT-4* across three ontologies. Few-shot settings use either CQ examples or enriched contexts (CQ + URI + Description). The table reports precision, recall, and the corresponding gains over the zero-shot baseline.

Ontology	LLM	Setting	Prec	Rec	Prec Gain	Rec Gain
Video Game	Flan-T5-l	Few-shot (CQ example)	0.9825	0.9333	+0.1229	+0.0737
		Zero-shot	0.8596	0.8596	–	–
	GPT-4	Few-shot (CQ + URI + Desc)	0.9921	0.9766	+0.2799	-0.0210
		Zero-shot	0.7122	0.9976	–	–
African Wildlife	Flan-T5-l	Few-shot (CQ + URI + Desc)	0.9130	1.0000	+0.2173	+0.0000
		Zero-shot	0.6957	1.0000	–	–
	GPT-4	Few-shot (CQ example)	0.9691	1.0000	+0.1546	+0.0000
		Zero-shot	0.8145	1.0000	–	–
Vicinity Core	Flan-T5-l	Few-shot (CQ example)	0.6603	0.9918	+0.0822	+0.0058
		Zero-shot	0.5781	0.9860	–	–
	GPT-4	Few-shot (CQ + URI + Desc)	0.8553	0.9950	+0.1684	-0.0050
		Zero-shot	0.6869	1.0000	–	–

(e.g., vicinity core ontology). In the zero-shot setting, the model tends to overgenerate, often producing irrelevant or loosely related CQs. Few-shot examples help anchor the output into relevant patterns. For instance: (1) Video Game Ontology: *Prec* improves from 0.86 ⟶ 0.98. (2) African Wildlife Ontology: *Prec* increases from 0.70 ⟶ 0.91. (3) Vicinity Core Ontology: *Prec* improves from 0.59 ⟶ 0.66, despite the domain's complexity.

GPT-4 performs well even without examples, but benefits from a few-shot prompt in terms of reducing overgeneration. Recall remains high in the zero-shot setting, though precision may decline due to the inclusion of loosely relevant responses. Supplying a few examples helps the model respond more precisely. For example: (1) Video Game Ontology: Zero-shot *Rec* is nearly perfect 0.9976, but *Prec* is low 0.7122. With few-shot prompting, *Prec* rises to 0.9921. (2) Vicinity Core Ontology: *GPT-4* generates over 5,000 candidate CQs in zero-shot mode (CCQ in Table 1) with only 0.6869 *Prec.* Few-shot guidance

improves *Prec*to 0.8553.

While *GPT-4* exhibits strong baseline performance in both settings, both models benefit significantly from few-shot prompting. The improvements are especially pronounced in terms of precision and semantic relevance, demonstrating that structured examples and contextual enrichment are critical for generating high-quality CQs in knowledge engineering tasks.

5. Discussion

This study presents one of the first comparative analysis of few-shot and zero-shot prompting techniques applied to CQ generation for the construction of ontologies and KGs. While prior CQ generation efforts have predominantly relied on zero-shot prompting [15, 18, 17, 19, 16, 25], few-shot prompting has been shown to enhance question quality, particularly in educational and professional training settings [26, 28, 27]. Motivated by these findings, we investigated whether similar improvements hold for CQ generation in ontology engineering: specifically, whether the choice of prompting technique affects the quality of generated CQs, and to what extent.

Our analysis indicates that few-shot prompting enhances RETROFIT-CQ's performance compared to zero-shot by reducing the noise in generated CQs. That is, it results in more concise, relevant, and context-specific questions rather than overly general ones. This improvement is most clearly reflected in the increased precision scores shown in Table 3. Notably, while recall remains high (close to 1) in both prompting setups – indicating that both techniques are capable of capturing the majority of ground-truth CQs – few-shot prompting mitigates the problem of overgeneration, particularly in large or complex ontologies like the Vicinity Core ontology. This suggests that few-shot prompting not only improves precision but also controls verbosity and enhances semantic relevance.

The effectiveness of few-shot prompting also varies depending on the underlying LLMs and the type of contextual information provided. For instance, *Flan-T5*, an open-source model, performs inconsistently across different contexts. Its precision drops significantly when an ontology URI is provided—such as in the case of the Video Game ontology (Table 1), where precision is 0.8596. In contrast, performance improves when an ontology description is included (0.9298), and it is highest when only example CQs are used (0.9825). This discrepancy likely stems from *Flan-T5's* limitations: it is a static model without internet access, and thus cannot interpret URIs or resolve them to meaningful content.

A similar pattern is observed with *GPT-4*. Despite its superior capabilities, we did not enable internet access (e.g., through search APIs or web retrieval mechanisms).[3] This is because we wanted to maintain the same settings in order to compare the results against our previous paper [12]. Consequently, *GPT-4* also struggles when presented with ontology URIs. This underperformance highlights a broader issue: without contextual enrichment—such as ontology descriptions or structured example questions—even advanced LLMs are unable to infer the intended semantics of domain-specific identifiers. Therefore, the presence of well-crafted contextual information in few-shot prompts plays a critical role in generating high-quality CQs for knowledge engineering.

Despite the improvements seen with few-shot prompting, this technique raises one major concern: we must consider whether the gains in CQ quality justify the additional computational cost and time associated with few-shot setups, especially when using closed-source LLMs like *GPT-4*. The need for structured context (e.g., examples or descriptions) not only increases the complexity of the prompting pipeline but may also limit scalability in real-world ontology engineering tasks.

Whether the use of few-shot prompting is warranted for CQ generation, given the trade-offs in time, resources, and deployment complexity remains an open question. While high-quality, semantically rich questions are undoubtedly valuable in domains such as education and assessment, where understanding and precision are paramount, it remains unclear whether similar similar characteristics are is always necessary in the broader field of ontology engineering. More empirical studies are needed to assess whether the added precision translates to tangible benefits in ontology design, validation, or reuse.

[3] https://platform.openai.com/docs/guides/function-calling?api-mode=responses

Moreover, future work should include a broader range of LLMs (both open-source and closed—source) and ontologies of varying complexity. This would allow us to assess the generalisability of our findings and better understand how prompting strategies interact with model architecture, domain specificity, and the nature of the task.

6. Conclusions

This study provides the first comparative analysis of zero-shot and few-shot prompting for CQ generation in ontology engineering. Our results show that few-shot prompting improves the quality of generated CQs– particularly in terms of precision and relevance– by providing structured context that helps guide the model. However, this improvement comes with increased resource demands, raising important questions about cost-effectiveness in real-world applications. While few-shot prompting is valuable in domains requiring high precision, such as education or formal assessment, its broader utility in ontology engineering warrants further investigation. Future work should explore this trade-off across more models, domains, and ontology types to better understand when and where few-shot prompting truly adds value.

Declaration on Generative AI

The author used ChatGPT to improve the manuscript's readability and subsequently reviewed and edited the content, taking full responsibility for the final published article.

References

[1] N. F. Noy, D. L. McGuinness, Ontology development 101: A guide to creating your first ontology, Technical Report, Stanford knowledge systems laboratory technical report KSL-01-05, 2001.

[2] M. Poveda-Villalón, A. Fernández-Izquierdo, M. Fernández-López, R. García-Castro, LOT: An industrial oriented ontology engineering framework, Engineering Applications of Artificial Intelligence 111 (2022) 104755. doi:10.1016/j.engappai.2022.104755.

[3] V. Presutti, E. Daga, A. Gangemi, E. Blomqvist, Extreme design with content ontology design patterns, in: Proceedings of the 2009 International Conference on Ontology Patterns, volume 516 of *WOP'09*, CEUR-WS.org, 2009, p. 83–97.

[4] J. F. Sequeda, W. J. Briggs, D. P. Miranker, W. P. Heideman, A pay-as-you-go methodology to design and build enterprise knowledge graphs from relational databases, in: Proceedings of the 18th International Semantic Web Conference, ISWC 2019, Springer International Publishing, 2019, pp. 526–545. doi:10.1007/978-3-030-30796-7_32.

[5] M. C. Suárez-Figueroa, A. Gómez-Pérez, M. Fernández-López, The neon methodology framework: A scenario-based methodology for ontology development, Applied ontology 10 (2015) 107–145.

[6] C. Bezerra, F. Freitas, Verifying description logic ontologies based on competency questions and unit testing, in: Proceedings of the IX Seminar on Ontology Research and I Doctoral and Masters Consortium on Ontologies, volume 1908, 2017, pp. 159–164.

[7] C. M. Keet, A. Ławrynowicz, Test-driven development of ontologies, in: Proceedings of the 13th International Conference on The Semantic Web, ESWC 2016, Springer International Publishing, 2016, pp. 642–657.

[8] R. Alharbi, Assessing candidate ontologies for reuse, in: Proceedings of the Doctoral Consortium at ISWC 2021 (ISWC-DC), 2021, pp. 65–72. URL: https://api.semanticscholar.org/CorpusID:244895203.

[9] R. Alharbi, V. Tamma, F. Grasso, Requirement-based methodological steps to identify ontologies for reuse, in: Intelligent Information Systems, Springer Nature Switzerland, 2024, pp. 64–72.

[10] S. Azzi, A. Assi, S. Gagnon, Scoring ontologies for reuse: An approach for fitting semantic requirements, in: Proceedings of the Research Conference on Metadata and Semantic Research, MTSR 2022, Springer Nature, 2023, pp. 203–208.
[11] C. M. Keet, Z. C. Khan, On the roles of competency questions in ontology engineering, in: Proceedings of the 24th International Conference on Knowledge Engineering and Knowledge Management, EKAW2024, Springer Nature Switzerland, Cham, 2025, pp. 123–132.
[12] R. Alharbi, J. de Berardinis, F. Grasso, T. Payne, V. Tamma, Characteristics and desiderata for competency question benchmarks, in: Proceedings of the 23rd International Semantic Web Conference, ISWC, Lecture Notes in Computer Science, Springer, 2024. To appear.
[13] R. Alharbi, V. Tamma, F. Grasso, T. R. Payne, A review and comparison of competency question engineering approaches, in: Proceedings of the 24th International Conference on Knowledge Engineering and Knowledge Management, EKAW2024, Springer Nature Switzerland, Cham, 2025, pp. 271–290.
[14] L. Rao, H. Reichgelt, K. Osei-Bryson, Knowledge elicitation techniques for deriving competency questions for ontologies, in: Proceedings of the Tenth International Conference on Enterprise Information Systems (ICEIS 2008), volume ISAS-2, Barcelona, Spain, 2008, pp. 105–110.
[15] M.-J. Antia, C. M. Keet, Automating the generation of competency questions for ontologies with agocqs, in: Knowledge Graphs and Semantic Web, Springer Nature Switzerland, Cham, 2023, pp. 213–227.
[16] F. Ciroku, J. de Berardinis, J. Kim, A. Meroño-Peñuela, V. Presutti, E. Simperl, Revont: Reverse engineering of competency questions from knowledge graphs via language models, Journal of Web Semantics 82 (2024) 100822. doi:https://doi.org/10.1016/j.websem.2024.100822.
[17] B. Zhang, V. A. Carriero, K. Schreiberhuber, S. Tsaneva, L. S. González, J. Kim, J. de Berardinis, Ontochat: A framework for conversational ontology engineering using language models, in: Proceedings of the 21st Extended Semantic Web conference, ESWC, Springer Nature Switzerland, Cham, 2025, pp. 102–121.
[18] R. Alharbi, V. Tamma, F. Grasso, T. R. Payne, An experiment in retrofitting competency questions for existing ontologies, in: Proceedings of the 39th ACM/SIGAPP Symposium on Applied Computing, SAC '24, 2024, p. 1650–1658. doi:10.1145/3605098.3636053.
[19] Y. Rebboud, L. Tailhardat, P. Lisena, R. Troncy, Can LLMs generate competency questions?, in: Extended Semantic Web Conference, ESWC2024, Hersonissos, Greece, 2024.
[20] R. Alharbi, V. Tamma, F. Grasso, T. R. Payne, The role of generative ai in competency question retrofitting, in: Proceedings of the 21st Extended Semantic Web Conference, ESWC2024, Springer Nature Switzerland, Cham, 2025, pp. 3–13.
[21] R. Alharbi, V. Tamma, F. Grasso, T. R. Payne, Investigating open source llms to retrofit competency questions in ontology engineering, Proceedings of the AAAI Symposium Series 4 (2024) 188–198. URL: https://ojs.aaai.org/index.php/AAAI-SS/article/view/31793. doi:10.1609/aaaiss.v4i1.31793.
[22] P. Liu, W. Yuan, J. Fu, Z. Jiang, H. Hayashi, G. Neubig, Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing, ACM Computing Surveys 55 (2023). doi:10.1145/3560815.
[23] G. Marvin, N. Hellen, D. Jjingo, J. Nakatumba-Nabende, Prompt engineering in large language models, in: Proceedings of the Data Intelligence and Cognitive Informatics conference, Springer Nature Singapore, 2024, pp. 387–402.
[24] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, D. Amodei, Language models are few-shot learners, in: Proceedings of the 34th International Conference on Neural Information Processing Systems, NIPS '20, Curran Associates Inc., Red Hook, NY, USA, 2020.
[25] X. Pan, J. v. Ossenbruggen, V. de Boer, Z. Huang, A rag approach for generating competency questions in ontology engineering, in: Metadata and Semantic Research, Springer Nature Switzerland,

Cham, 2025, pp. 70–81.
[26] Y. Poon, J. S. Y. Lee, Y. Y. Lam, W. L. Suen, E. L. C. Ong, S. K. W. Chu, Few-shot question generation for reading comprehension, in: Proceedings of the 10th SIGHAN Workshop on Chinese Language Processing (SIGHAN-10), Association for Computational Linguistics, Bangkok, Thailand, 2024, pp. 21–27. URL: https://aclanthology.org/2024.sighan-1.3/.
[27] B. Leite, H. Cardoso, On few-shot prompting for controllable question-answer generation in narrative comprehension, in: Proceedings of the 16th International Conference on Computer Supported Education - Volume 2: CSEDU, INSTICC, SciTePress, 2024, pp. 63–74. doi:10.5220/0012623800003693.
[28] D. Di Nuzzo, E. Vakaj, H. Saadany, E. Grishti, N. Mihindukulasooriya, Automated generation of competency questions using large language models and knowledge graphs, in: Proceedings of the 3rd International Workshop on Natural Language Processing for Knowledge Graph Creation, NLP4KGC 2024, co-located with 20th International Conference on Semantic Systems (SEMANTiCS 2024), 2024, pp. 128–153.
[29] N. Mulla, P. Gharpure, Automatic question generation: a review of methodologies, datasets, evaluation metrics, and applications, Progress in Artificial Intelligence 12 (2023) 1–32.
[30] A. Fernández-Izquierdo, M. Poveda-Villalón, R. García-Castro, Coral: A corpus of ontological requirements annotated with lexico-syntactic patterns, in: Proceedings of the 16th International Conference on the Semantic Web, ESWC 2019, 2019, pp. 443–458.
[31] D. Wiśniewski, J. Potoniec, A. Ławrynowicz, C. M. Keet, Analysis of ontology competency questions and their formalizations in sparql-owl, Journal of Web Semantics 59 (2019) 100534.
[32] N. Reimers, I. Gurevych, Sentence-BERT: Sentence embeddings using Siamese BERT-networks, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, 2019, pp. 3982–3992.