Ontology Learning from Text: an Analysis on LLM Performance

Roos M. Bakker[1,2,*], Daan L. Di Scala[1] and Maaike H. T. de Boer[1]

[1]TNO Netherlands Organisation for Applied Scientific Research, Department Data Science, Kampweg 55, 3769 ZG Soesterberg, The Netherlands.
[2]Leiden University Centre for Linguistics, Leiden University, 2311 BE Leiden, Reuvensplaats 3-4, the Netherlands

Abstract

Ontologies provide a structured framework to represent and integrate domain knowledge. Developing them is a complex and time-consuming task, requiring domain expertise to ensure accuracy and consistency. Ontology learning aims to automate this process by learning full ontologies, or parts of them, from sources such as textual data. In this paper, we research the potential of Large Language Models (LLMs), specifically GPT-4o, in ontology learning, using a real-world use case. We introduce a manually constructed ontology based on knowledge in a news article, and compare it to ontologies extracted using three different prompting approaches over multiple runs. The resulting ontologies are evaluated both quantitatively and qualitatively, to ensure that differences in performance due to modelling choices are also considered. The results show that, while the LLM effectively identifies important classes and individuals, it often does not include properties between classes, and adds inconsistent and incorrect properties between individuals. Prompting on a sentence level leads to more correct individuals and properties, however, quantitative evaluation shows more hallucinations and incorrect triples. Despite these issues, LLMs advance previous ontology learning methods by considering classes, individuals, and properties as a whole, creating a more complete ontology rather than isolated elements. This provides a new perspective on ontology learning and highlights the potential of LLMs to offer a first version of an ontology or an extension to an existing one based on new information.

Keywords

ontology learning, knowledge graph extraction, large language models, information extraction

1. Introduction

Ontologies play an important role in structuring and integrating domain knowledge. They support data interoperability, standardisation, and increasingly more, explainability and reasoning for statistical models [1, 2]. Developing them is time-consuming, costly, and prone to errors [3]. Ontology Learning aims to tackle these challenges by automatically learning ontologies or parts of them. Previous work has shown the potential of leveraging Natural Language Processing techniques for ontology learning or knowledge graph extraction [4, 5, 6]. Term extraction is relatively successful with techniques such as Named Entity Recognition, and some degree of success can be seen on the relation extraction task on a sentence level [7, 8]. However,

NLP4KGC: 3rd International Workshop on Natural Language Processing for Knowledge Graph Creation in conjunction with SEMANTiCS 2024 Conference. , September 17–19, 2024, Amsterdam, The Netherlands
*Corresponding author.
roos.bakker@tno.nl (R. M. Bakker); daan.discala@tno.nl (D. L. Di Scala); maaike.deboer@tno.nl (M. H. T. d. Boer)
0000-0002-1760-2740 (R. M. Bakker); 0000-0003-1548-6675 (D. L. Di Scala); 0000-0002-2775-8351 (M. H. T. d. Boer)

such extractions lead to a flat graph, often only containing individuals and simple relations. The step towards ontologies with classes, taxonomical structures, and other abstractions is an open challenge. Recent work by Bakker and Di Scala [6] and Babaei Giglou et al. [3] shows the potential of Large Language Models (LLMs) for knowledge graph extraction and ontology learning. These works demonstrate first steps towards using LLMs for ontology learning in the form of extracting knowledge graphs [6], and extracting parts of ontologies with a step-by-step approach [3]. While these advancements mark significant progress, these works and previous efforts have focused on steps or parts of an ontology. Achieving a complete, fully integrated model from domain-specific documents remains a challenge.

In this work, we research the potential of LLMs for ontology learning by extracting an ontology from a news message. Specifically, we utilise GPT-4o to generate ontologies based on a news message about the Nord Stream pipeline leakage incident. To evaluate this model on the task of ontology learning, we introduce annotations for the selected news message and a manually created ontology by an ontology developer. We prompt the LLM with three different approaches: 1) a direct approach, in which one query is sent to extract a full ontology from the complete document, 2) a sequential approach, in which three queries are sent to extract classes, individuals, and properties sequentially, and 3) a sentence based approach, in which we extract the ontology elements for each sentence. This comparison is conducted across multiple runs to analyse the consistency and reliability of the LLM-generated ontologies. We evaluate the generated ontologies against the manually created ground truth.

The paper is structured as follows: Section 2 provides an overview of related work in ontology learning and knowledge graph extraction using LLMs. Section 3 describes the news message, the annotations and manually created ontology, and the different prompting approaches. Section 4 presents the results, offering a quantitative evaluation of the extracted ontologies against the manually created one, as well as a qualitative evaluation which highlights patterns and differences. Section 5 discusses the results, and finally, Section 6 concludes this work with a summary and suggestions for future work.

2. Related Work

In this section, we will describe the fields of ontology learning, including evaluation of ontologies, and knowledge graph extraction.

2.1. Ontology Learning

An ontology is a formal specification of concepts in the world [9]. Ontologies often describe a part of the world. Two examples often used in the field are the pizza ontology [10], which describes different type of pizzas, toppings and for example types of dough, and the wine ontology [11], which describes types of wine, colours, grapes and flavours. Current ontologies are mostly created manually. This is an extensive task, which involves the time of at least one domain expert and a modeller. Ontologies also require maintenance, which also is costly and prone to errors. Therefore, the field of ontology learning has focused on (semi-)automatically creating (part of) ontologies. Many overview or survey papers on this topic have been published,

such as by Buitelaar et al. [12], Zhou [13], Wong et al. [14], Asim et al. [15], Khadir et al. [16] and Du et al. [17].

Buitelaar et al. [12] give an overview of the field until 2005, in which they divide the task into complexity levels, based on the ontology learning layer cake: Starting in complexity with terms, going up to synonyms, then concepts and concept hierarchies, followed by relations and finally rules and axioms. The majority of the techniques at this time were rule-based and focused on lexico-syntactical patterns. Performance, especially recall, was low, as such patterns could not consistently be identified [12]. Relation extraction was often done by combining linguistic patterns, such as dependencies, with statistical analysis [18]. For all approaches on all levels, manual work was necessary to produce a coherent ontology.

In the work of Wong et al. [14] on ontology learning in 2012, they state that the newer techniques were able to extract a lightweight ontology – or in later terms a simple knowledge graph – with classes and relations but without axioms [19], but not yet heavyweight ontologies that have extensive axioms [20].

In the years to follow, statistical approaches gained in popularity due to the increase of computing power and the uprise of machine learning applications. Some of these statistical approaches include co-occurrences, hierarchical clustering, and the start of vector-based methods [15]. In the overview paper of Asim et al. [15], linguistic and statistical approaches are distinguished, as well as the different terms of the layer cake, mainly term extraction and relation extraction.

From 2021 onwards, Natural Language Processing (NLP) techniques have provided for more efficient and scalable ontology learning [21, 16]. Another trend is that the term ontology is used less in the literature, as the term knowledge graph is used more often. The consensus seems to be that ontologies are stricter than knowledge graphs and that ontologies could be considered a subclass of knowledge graphs [22]. In the overview of Khadir et al. [16], the distinction between linguistic and statistical / machine learning approaches is still there, but they are combined more often, such as in Tian et al. [23], in which a graph convolutional network is trained using dependency trees and in Jaradeh et al. [24] in which the multi-tool Plumber is introduced. For different texts, different approaches are suggested. Its dynamic pipelines outperform static pipelines, but scores for the knowledge graph completion task remain low. This has to do with performance of the individual components, where results can still be improved [24].

Currently, decoder-only models such as GPT [25] have gained prominence [17]. They are known as generative LLMs, due to the vast amounts of texts and parameters with which they are trained, and can be applied to advance the field of ontology learning. For example, Babaei Giglou et al. [3] test different LLM model families on several ontology learning tasks such as term typing, taxonomy discovery, and extraction of non-taxonomic relations. Hao et al. [26] use prompting to extract (implicit) information from Language Models such as BERT to construct an ontology / knowledge graph. Most recent papers focus on subtasks of ontology learning, but not the creation of a full ontology. However, they show that LLMs could, at least on these tasks, introduce new opportunities for ontology learning. Du et al. [17] mention that future directions using LLMs in ontology learning could be to develop benchmarks, work on non-taxonomic relation extraction and axiom discovery, the collaboration between domain expert and prompt engineering and leveraging LLMs for dynamic ontology updating. Therefore, our main focus in this paper is purely on Generative LLM capabilities on Ontology Learning.

2.1.1. Evaluation of Ontologies

An important aspect in ontology learning is evaluation [14, 27]. As soon as ontologies increase in complexity, for instance by adding a taxonomical structure, evaluation techniques come short. Several metrics and tools have been created in the past years, of which for example Wilson et al. [27] explain the characteristics consistency, coverage, conciseness, correctness and modularity, while McDaniel and Storey [28] introduce DOORS, which is a ranking framework for ontologies by using syntactic, pragmatic, semantic and social quality metrics. Recent work by Bakker and de Boer [29] extend previous metrics and test them in an experimental setting on various ontologies and taxonomies. They show that such metrics can indicate the quality of changes, but evaluation of a first version of a knowledge graph or an ontology still requires manual steps.

2.2. Knowledge Graph Extraction

A task related to ontology learning is Knowledge Graph Extraction (KGE), in which the aim is to extract knowledge graphs from different sources, using a variety of techniques. KGE is has similar subtasks that overlap with ontology learning, especially those related to the lower levels of the layer cake, in which terms and relations are extracted.

KGE is also often linked to the field of Open Information Extraction (OpenIE) [30]. One of the tasks in OpenIE is to generate triples, containing a subject, relation, and object, based on textual data. The field of OpenIE evolved in a similar way as ontology learning, coming from statistical approaches and linguistic approaches, to word embeddings [31], to BERT models [32] and multiple techniques are often combined, similar to Jaradeh et al. [24] as described above.

Similar to ontology learning, research efforts currently focus on LLMs. LLMs provide new possibilities in a variety of NLP tasks, including triple generation and KGE. Some approaches include using prompting schemes, like in-context learning [33, 34] and manual prompting for completing subject-predicate-object triples [35]. In this way LLMs are often used in the inference or completion of knowledge graphs rather than their direct production [36]. Knowledge graph extraction is, however, also done end-to-end [37]. A recent solution for knowledge graph extraction, or any related NLP task such as Named Entity Recognition and relation extraction, is OntoGPT[38]. It integrates current LLMs such as GPT-3.5 and GPT-4 and has instruction prompts and ontology-based grounding, especially for the biomedical domain. Most related to this work, in Bakker and Di Scala [6] relation extraction methods, including REBEL [7], KnowGL [39] and GPT-3.5 and GPT-4, are used on a real-world textual dataset similar to the one in this paper to explore the performance of the different models on this task. Results show that LLMs show potential in this task, but do not yet compare to manual annotation.

2.2.1. Evaluation of Knowledge Graphs

Knowledge graphs are often evaluated using their performance on downstream tasks, such as Named Entity Recognition or relation extraction [40]. Evaluation metrics often include the standard machine learning metrics such as accuracy, precision and recall. Chen et al. [41] introduced requirements for knowledge graphs on the dimensions of accessibility, intrinsic, contextual and representational, which are similar to metrics introduced for ontology learning.

Mihindukulasooriya et al. [42] introduced Text2KGBench to evaluate knowledge graphs generated by language models from natural text guided by an ontology. The evaluation metrics include precision, recall and F1 score, but also ontology conformance and subject, relation and object hallucinations.

3. Method

3.1. Data

In this paper, we aim to extract an ontology from text using Large Language Models. As a use case, we selected a news message on an incident in the safety domain. In this domain, multiple sources of information have to be considered to gather all information and make good decisions. One important source of information is the news; news messages provide up-to-date information albeit at a non-confidential level. With the use of an ontology, safety organisations can keep track of different sources of information in a structured manner. Therefore, we created a ground truth ontology for a news message on the Nord Stream Pipeline incident by Reuters[1]. The news message describes the incident on September 26 in 2022 when a gas leak occurred in the two Nord Stream gas pipelines. The leaks in international waters lead to separate investigations by Denmark, Germany, and Sweden to find the cause. Due to the Russian invasion of Ukraine, the pipelines were inactive. As of the writing of this paper, investigations continue.

We annotated the news message according to the ontology learning layer cake described in Section 2. First, terms were annotated, after which the synonyms, and thirdly the concepts. In the fourth step, the concept hierarchy was annotated for taxonomic relations. As a final step, properties were annotated. The top steps of the ontology learning layer cake – rules and axioms in Buitelaar et al. [12] and the relation hierarchy, axioms schemata, and general axioms in Asim et al. [15] – were left out of the annotation due to their complexity and the scope of the text. From the annotations, an RDF ontology was built in ttl (turtle) syntax including classes (concepts), individuals (terms), and object properties and data properties (relations). The ontology was created based on the first 12 sentences of the news text, and resulted in a Ground Truth ontology, which can be found in the repository of this paper[2]. The process of annotating the text and modelling the annotations in an RDF ontology took 16 hours, including 2 hours of discussion of modelling choices with experienced ontology developers. The Ground Truth ontology consists of 33 classes, 38 individuals, 20 object properties and 15 datatype properties. Visualisations of the classes and part of the individuals are shown in Figures 1 and 2, respectively. The full visualisation of all individuals can also be found in the repository[2].

[1] https://www.reuters.com/business/energy/pressure-defunct-nord-stream-2-pipeline-plunged-overnight-operator-2022-09-26
[2] https://gitlab.com/knowledge-graphs/text2onto

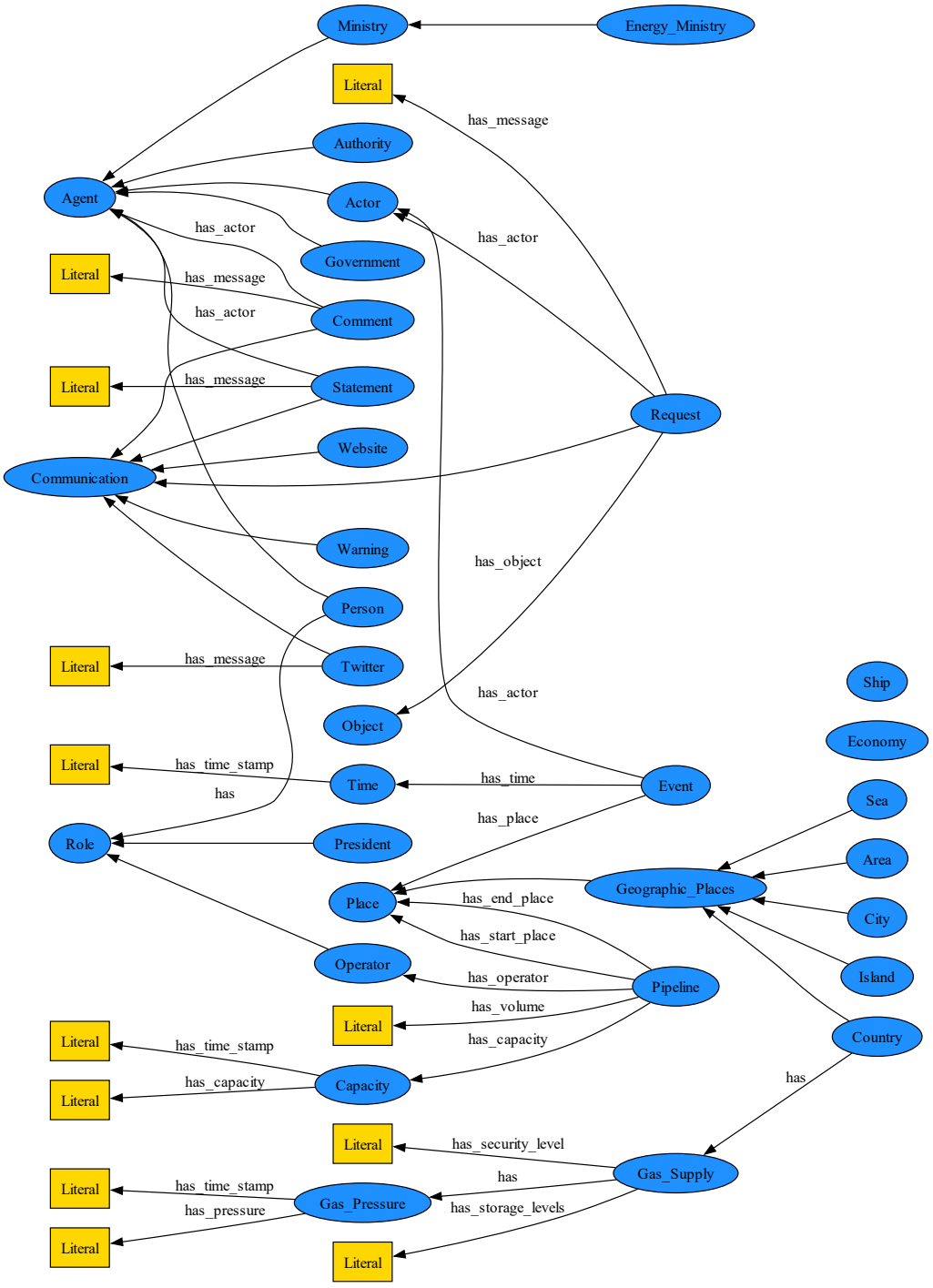

Figure 1: ground truth ontology's classes (blue circles) and literals (yellow boxes). RDFS:SUBCLASSOF relations are denoted by non-labeled arrows.

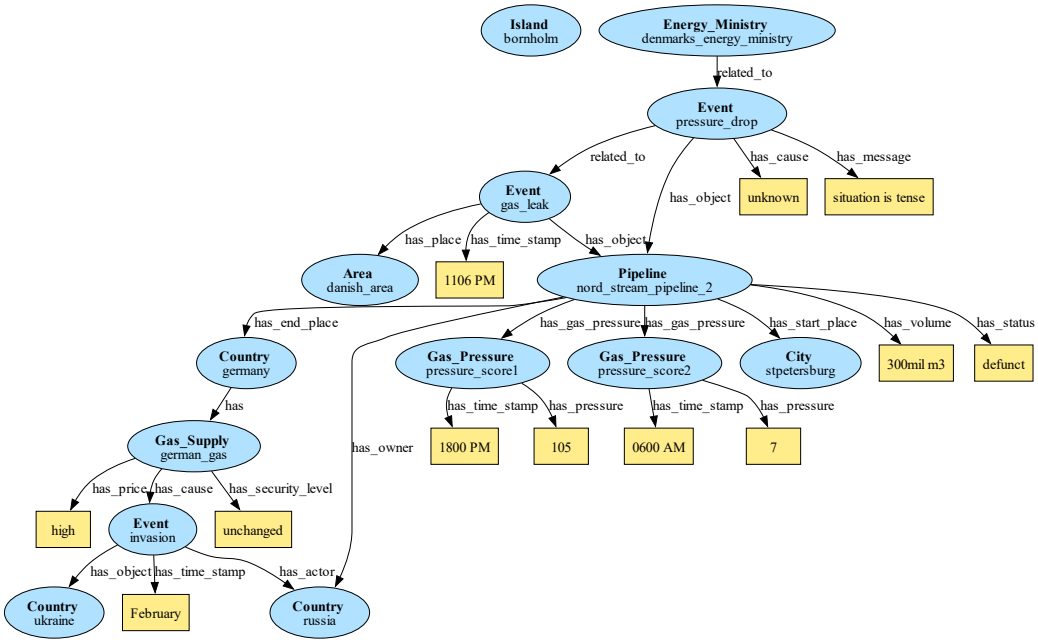

Figure 2: Part of the ground truth ontology's individuals (blue circles) and literals (yellow boxes).

3.2. Experimental Setup

We implemented three approaches for prompting GPT-4o for ontology extraction, as shown in Figure 3. GPT-4o is chosen because GPT-4 has shown superior performance over GPT-3.5 and other models in earlier related research Bakker and Di Scala [6], and GPT-4o has been claimed to match GPT-4 Turbo performance on text in English https://openai.com/index/hello-gpt-4o/, while being less expensive to use. We assessed the consistency of the output by running each approach three times. The temperature of the model is set at a low score of 0.3 to minimise hallucinations and ensure greater consistency.

A. Direct. The first approach involves prompting GPT4-o directly to extract the ontology from given text. This was done with the following prompt: *"Extract an ontology in ttl format from the following text, only return the created ttl code. Make sure to label all classes as rdfs:Class, all individuals as owl:NamedIndividual, and all properties as owl:ObjectProperty, owl:DatatypeProperty, or owl:AnnotationProperty."*

B. Sequential. Secondly, we followed a sequential approach, in which we split up the task of ontology extraction into three extraction parts: class extraction, individual extraction and relation extraction. For this, we prompted GPT-4o three separate times and included the output of the previous prompts. First, GPT-4o was prompted to extract all classes based on the given text, with a similar prompt as used in the direct approach. Then, GPT-4o was prompted to

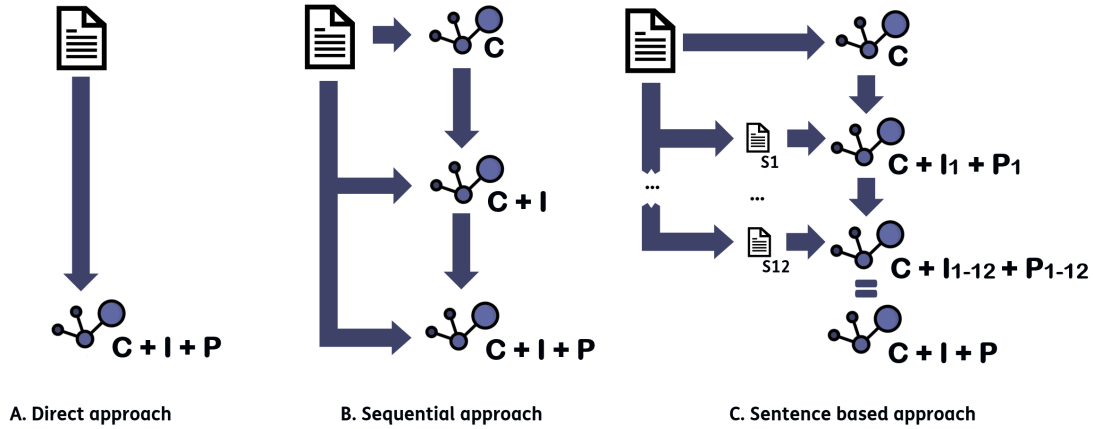

Figure 3: Overview of the three implemented ontology extraction approaches: A) a direct approach, B) a sequential approach splitting the extraction process up in classes, individuals, and relations and C) a sentence based step-by-step approach dividing the extraction process over each sentence.

extract all individuals, considering both the provided text and the extracted classes. Finally, considering the text, the extracted classes and the extracted individuals, GPT-4o was prompted to extract all relations. The three resulting ttl files were then merged to yield the full ontology.

C. Sentence. Our third extraction approach entails a sentence based step-by-step method, in which we first extracted the classes from the full text, same as the sequential approach. Then, the text was split into sentences, and for each of the sentences GPT-4o was prompted to extract the individuals and relations. For each new sentence, GPT-4o was prompted to take into account the extracted classes and the previously extracted individuals and relations. Finally, all results were merged in a resulting ontology.

3.3. Evaluation

The resulting ontologies were evaluated quantitatively and qualitatively. For the quantitative evaluation, the precision, recall, and F1 score were calculated for the classes, individuals, relations, and overall performance. To measure the interconnectedness of the resulting ontology, we calculated the average degree score D_{avg} for each of the approaches [43]. This degree score is calculated by the following formula: $D_{avg} = \big(|I|+|C|\big)/\big(|P_{II}|+|P_{IC}|+|P_{IL}|\big)$, with $|I|$ and $|C|$ being the amount of individuals and classes and $|P_{II}|$, $|P_{IC}|$ and $|P_{IL}|$ being the amount of properties between individuals and individuals, between individuals and classes and between individuals and literals. The degree score gives an indication of the interconnectedness of the resulting ontology. As discussed in Guéret et al. [43], the goal of ontologies is not be to be fully complete, as it indicates a high amount of properties that lack any meaning. Therefore, the aim of this degree metric for the resulting ontologies is not to be as high as possible, but to match closely with the ground truth's degree score, which indicates a similar level of interconnectedness.

In addition to the quantitative evaluation, a qualitative evaluation is performed. Observations

Approach	Run	# Classes	# Individuals
Ground Truth		**33**	**38**
Direct	**1**	24	9
Direct	**2**	15	8
Direct	**3**	10	14
Direct	**Average**	16.33	10.33
Sequential	**1**	**31**	13
Sequential	**2**	36	12
Sequential	**3**	**31**	14
Sequential	**Average**	32.67	13
Sentence	**1**	**35**	64
Sentence	**2**	38	60
Sentence	**3**	**31**	**57**
Sentence	**Average**	34.67	60.33

Table 1
Amount of extracted classes and individuals for each approach and run, including ground truth. Bold indicates closest amounts to the ground truth, whereas underlining indicates the closest average.

were made on various levels, including the form of the output, the quality and correctness of full triples, and the level of detail. Development choices made during annotation were also considered.

4. Results

4.1. Quantitative Results

Table 1 shows the amount of classes and individuals. The results show that the sequential approach is closest to the ground truth in terms of amount of individuals. As expected, the direct approach produces the least amount of classes and individuals, and the sentence based approach the most. Note that there is only a slight difference in amount of classes between the sequential and sentence based approach, as these use the same prompt for class extraction.

Table 2 shows the F1 scores for all approaches and all three runs for each approach. The results show that the sentence based approach, in which classes are extracted for the full document and individuals and properties per sentence, yields the highest overall F1 score on average and absolute on the second run. This run for the sentence based approach also lead to the highest score for classes and properties, whereas the highest average score for classes was achieved by the sequential approach. The highest score on individuals is achieved by the sentence based approach in the third run. Comparing performance of the different types of extraction (classes, individuals, properties), the properties have a relatively low score, especially for the direct and sequential approach. Individuals consistently have a higher score compared to the other two over all runs and types.

Figure 4 shows the precision, recall and F1 scores for all approaches in a bar plot, including

Approach	Run	Classes	Individuals	Properties	Overall
Direct	**1**	0.53	0.34	0.29	0.40
Direct	**2**	0.33	0.41	0.22	0.33
Direct	**3**	0.28	0.48	0.23	0.34
Direct	**Average**	0.38	0.41	0.25	0.36
Sequential	**1**	0.44	0.52	0.17	0.39
Sequential	**2**	0.52	0.51	0.17	0.42
Sequential	**3**	0.50	0.51	0.18	0,42
Sequential	**Average**	0.49	0.51	0.17	0.41
Sentence	**1**	0.30	0.56	0.38	0.41
Sentence	**2**	**0.56**	0.46	**0.77**	**0.60**
Sentence	**3**	0.36	**0.65**	0.36	0.47
Sentence	**Average**	0.41	0.56	0.50	0.49

Table 2
F1 scores for the extracted classes, individuals, properties and overall, for each approach and run. Bold indicates highest score per column for the individual runs, whereas underlining indicates the highest score on the average per column.

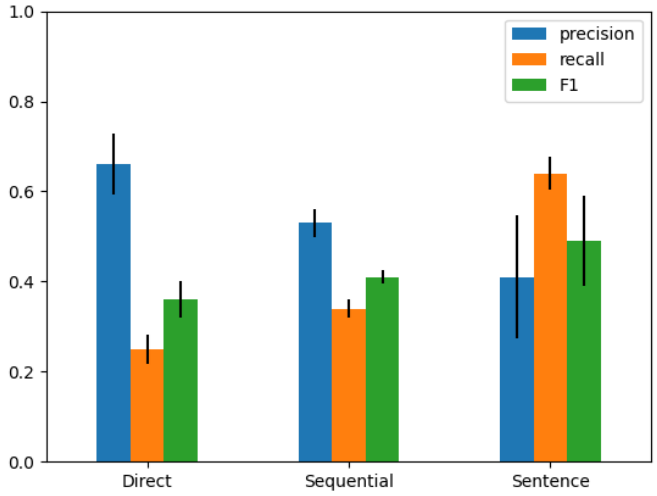

Figure 4: Average precision, recall and F1 scores for the three runs of all approaches, including standard deviation.

the standard deviation. The figure shows that for the direct and sequential approaches precision is higher than recall, whereas in the sentence based approach recall is higher than precision. The standard deviation is lower for the sequential approach compared to the other two approaches. Especially the precision of the direct and sentence based approach have a relatively high standard deviation.

Figure 5: Average degree scores of all approaches and ground truth.

Figure 5 shows the degree scores of all approaches and ground truth in a bar plot. The figure shows that the sentence based approach is closest to the ground truth degree. The direct approach is furthest away from the ground truth. One outlier is the third run of the direct approach, which is due to its low amount of individuals and classes, while still outputting correct properties between individuals and classes, as well as the inclusion of a few datatype properties.

4.2. Qualitative Results

In this section, we analyse qualitative findings of ontology extraction split over class extraction and individuals extraction. All visualisations of the resulting ontologies can be found in the repository[2].

4.2.1. Class Level

Overall, the connectivity of the output is very low, as GPT-4o produces barely any relations between classes. None of the three approaches yield any properties from classes to literals, whereas the ground truth includes 15 datatype properties. The only exception is the third run from the sentence based approach, in which the triple :Ship :radius "five nautical miles"^^xsd:string is found, with :radius correctly being classified as datatype property. The ground truth also includes a lot of taxonomic (rdfs:subClassOf) relations, which are not produced by the LLM.

Furthermore, on many accounts the results do not adhere to ontology creation standards and conventions. The prefixes are incorrectly given or missing (e.g. xsd: being used as prefix but

prefix xsd: <http://www.w3.org/2001/XMLSchema#> not being provided as prefix definition at the top of the document), and no rdfs:domain, rdfs:range or rdfs:label descriptions are given.

In general, while some terms from the text are correctly extracted, correctly classifying them remains a challenge. Some terms that should be identified as individuals, are classified as classes (e.g., :EnergyWar). Or supposed properties are classified as classes (e.g., :Bar or :Volume). This happens most frequently in the direct and sequential approach.

One notable property produced by the third run of the sentence based approach, is that the concepts of Twitter and Website are linked with each other (:Twitter rdf:type :Website), which is arguably correct. However, this requires some world knowledge which is not directly present in the text.

Another frequently occurring error are reflexive rdf:type properties, which happens because concepts in the text are designated as both an individual and a class. For example, GasStorageLevel rdf:type Class and GasStorageLevel rdf:type GasStorageLevel; rdf:type NamedIvindivual. This is another issue that occurs due to poor naming conventions, in this case the inconsistent use of lowercase letters for individuals.

4.2.2. Individual Level

Overall, GPT-4o is consistent in its format (e.g., UpperCamelCase for classes, lowerCamelCase or snake_case for properties) per call. However, in the sequential and sentence based approach, results are aggregated from different calls. This leads to these conventions being mixed and therefore the results to be inconsistent.

More specifically, notable errors by the sequential approach are the creation of tuples instead of triples (that is, it creates a triple with an empty object), and connecting terms that have no logical connection and are incorrect (e.g., :GermanNetworkRegulator :hasStatement :Pressure or :NordStream1 :disclosedPressureDrop :Pressure).

The sentence based approach produces a high amount of different relations between individuals, but does not make relevant connections between sentences. For example, :NordStream2Pipeline rdf:type :Pipeline and NordStream2 rdf:type :Pipeline are both extracted, but while it is clear that these individuals pertain to the same concept, they are not connected by the LLM.

Most notably, on close inspection the sentence based approach hallucinates information, such as dates of the pipelines' completion and cancellation. E.g., :Pipeline1 :completedOn "2022-02-20" and :Germany :cancelledOn "2022-02-22". These dates are not given in the text, and also do not correspond to the actual dates so their extraction is not grounded in additional world knowledge either.

Finally, in the second run of the direct approach rdfs:comment is used to label a part of the corresponding text. This positively adheres to ontology creation standards, as the usage of comments in this manner contribute to the explainability of the extraction process. This run is highlighted in Figure 6.

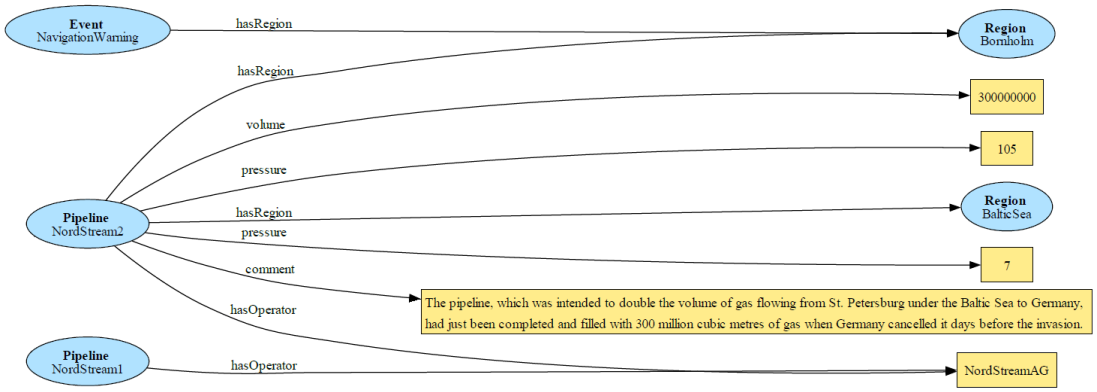

Figure 6: Snippet of the result of the third run of the direct approach. It shows part of the extracted classes and individuals. Most notably, a RDFS:COMMENT property is used to refer back to the text.

5. Discussion

When manually developing an ontology, different design choices can be made depending on the use case and goal of the ontology. In this work, the goal of the ontology was to describe the information in the news message. Therefore, the manually created ontology was highly detailed. As described in the previous section, the sentence based approach, where we extracted classes on a document level but individuals and properties per sentence, led to the best results overall, which might be due to the fact that it is most detailed. When a lower level of detail is desired, for instance when someone is only interested in the cause of the gas leak, the manually created ontology would be less detailed, and one of the other approaches might lead to better results.

Another aspect in which ontology developer choices play a role is the way abstractions are made when implementing classes. For instance, in the manually created ontology, abstractions are made in the form of events and the roles involved in an event. While for this singular use case such abstractions are not required, they become necessary when new information is added over time to ensure consistency. With a bigger use case, abstractions are also important to establish patterns between similar concepts in data. Finally, abstractions enable the inclusion of rules and axioms in an ontology.

Regarding the approaches, in preliminary trials, it was notable that not all extracted classes, individuals and relations were correctly identified as RDFS:CLASS, OWL:NAMEDINDIVIDUAL, or OWL:PROPERTY by the LLM when it is not specifically instructed to do so. To increase overall completeness of the output, we included an instruction for this in the prompts to steer GPT to explicitly include these labels in its output, for the final three approaches.

While the three approaches we compare in this work yield interesting results, it is possible to further extend the approaches or opt for additional alternatives. Prompting LLMs can be done in many different ways, leading to different results. Our quantitative analysis showed that the different approaches of prompting lead to big differences in F1 scores. The direct approach produces a low amount of terms when compared to the other approaches, which is a result from the single one-shot prompt. This approach could benefit from few-shot learning, chain of

thought prompting or targeted instructions to include a certain given minimum of concepts and relations, to ensure completeness. Other instructions could also be included to improve the results depending on the use case. For instance, while we focused on classes, individuals and their respective properties, the prompt can further include instructions to focus on including domain and range in the properties. However, guiding the LLM what parts of ontology creation to focus on requires the suitable knowledge of ontology modelling. It would be interesting to explore whether an LLM can be prompted for this information, and instructed to adhere to these conventions in next steps.

It would be beneficial to further expand on this work by testing these approaches in a setting with more texts, larger texts or more complex texts. Increasing complexity of the sentences in the texts challenges the extraction capabilities of the methods even further. Increasing size of the documents also entails further scalability issues. Running these methods on document level becomes less feasible with large document sizes, so instead one could opt for separation of the text into chunks.

The quantitative results indicated that the sentence based approach is better suitable for the task of ontology extraction than the other two approaches. This is because the amount of extracted classes and individuals matched the ground truth the best, the overall F1 score was the highest, as well as the average degree score matched the ground truth the closest. However, the qualitative evaluation showed that some mistakes are not always reflected in quantitative results. Incorrect triples can contain correct classes, individuals, or properties. Hallucinations can also occur and are difficult to distinguish from the knowledge present in the use case. For larger use cases, qualitative evaluation is not always feasible, which increases the need for more extensive quantitative evaluation.

Finally, the general inconsistency of the LLM outputs are still a limitation of this method. While this inconsistency is most notable in the direct approach, all approaches yielded varying extracted ontologies. The variety in output impacts reliability of the results. This makes the use of GPT-4o less suitable for applications in general pipelines, especially in domains or use cases where consistency is critical.

6. Conclusion and Future Work

Natural Language Processing techniques, such as Large Language Models, can aid the automatic creation of ontologies, a process known as ontology learning. In this paper, we explore three different prompting approaches to extract ontologies from a news message using GPT-4o, including a direct approach, a sequential approach and a sentence based approach. The results show that, while classes and individuals can be extracted with some success, the generated ontologies do not yet match the quality of a manually created one. This can partly be attributed to development choices and differences in prompting strategies. Sentence based prompting yields results closest to the manually developed ontology, indicating its effectiveness for high-detail extraction. Including more details in the prompt, such as types, can also improve the results, but requires modelling expertise from the user. Despite the generated ontologies not fully matching manually created ones, they are comprehensive models that include classes, individuals, and properties. This shows that a Large Language Model such as GPT-4o can

provide a new perspective on ontology learning.

For future work, a valuable addition to this work would be to research the potential of LLMs for extending an existing ontology. Additionally, a comparison between different types of LLMs for this task would be insightful. Furthermore, as GPT-4o shows promising results on the extraction of classes, individuals and their properties, the next step is the extraction of rules and axioms. By researching the extent of LLM capabilities to tackle the extraction of these top layers of ontologies, LLMs could further advance the field of ontology learning.

Acknowledgments

We extend our gratitude to the NATO Science & Technology Organization Centre for Maritime Research & Experimentation (CMRE) for providing us with a practical use case, and we extend special thanks to Dr. Pawel Kowalski for his guidance in the maritime domain and previous work. Additionally, we would like to thank the *TrustLLM* and *RVO Causal Relations in Behavioural Modelling* projects for their support.

References

[1] P. Hitzler, K. Janowicz, F. Lecue, On the role of knowledge graphs in explainable AI, Semantic Web 11 (2020) 41–51. doi:10.3233/SW-190374.

[2] R. Confalonieri, G. Guizzardi, On the Multiple Roles of Ontologies in Explainable AI, arXiv preprint arXiv:2311.04778 (2023).

[3] H. Babaei Giglou, J. D'Souza, S. Auer, LLMs4OL: Large language models for ontology learning, in: International Semantic Web Conference, Springer, 2023, pp. 408–427.

[4] M. De Boer, J. Verhoosel, Creating and evaluating data-driven ontologies, International Journal on Advances in Software 12 (2019).

[5] Y. Zhu, X. Wang, J. Chen, S. Qiao, Y. Ou, Y. Yao, S. Deng, H. Chen, N. Zhang, LLMs for knowledge graph construction and reasoning: Recent capabilities and future opportunities, arXiv preprint arXiv:2305.13168 (2023).

[6] R. M. Bakker, D. L. Di Scala, From Text to Knowledge Graph: Comparing Relation Extraction Methods in a Practical Context, in: First International Workshop on Generative Neuro-Symbolic AI, co-located with ESWC 2024, Hersonissos, Crete, Greece, 2024.

[7] P.-L. Huguet Cabot, R. Navigli, REBEL: Relation extraction by end-to-end language generation, in: Findings of the Association for Computational Linguistics: EMNLP 2021, Association for Computational Linguistics, 2021, pp. 2370–2381.

[8] S. Efeoglu, A. Paschke, Retrieval-Augmented Generation-based Relation Extraction, arXiv preprint arXiv:2404.13397 (2024).

[9] R. Studer, V. R. Benjamins, D. Fensel, Knowledge engineering: Principles and methods, Data & knowledge engineering 25 (1998) 161–197.

[10] A. Rector, N. Drummond, M. Horridge, J. Rogers, H. Knublauch, R. Stevens, H. Wang, C. Wroe, OWL Pizzas: Common errors & common patterns from practical experience of teaching OWL-DL, in: Proceedings of the European Knowledge Acquisition Workshop (EKAW-2004), Springer Verlag, Northampton, England, 2004.

[11] J. Graça, M. Mourao, O. Anunciação, P. Monteiro, H. S. Pinto, V. Loureiro, Ontology building process: the wine domain, in: Proc. of the 5th Conf. of EFITA, 2005.
[12] P. Buitelaar, P. Cimiano, B. Magnini, Ontology learning from text: methods, evaluation and applications, volume 123 of Frontiers in Artificial Intelligence and Applications, IOS press, 2005.
[13] L. Zhou, Ontology learning: state of the art and open issues, Information Technology and Management 8 (2007) 241–252.
[14] W. Wong, W. Liu, M. Bennamoun, Ontology learning from text: A look back and into the future, ACM computing surveys (CSUR) 44 (2012) 1–36.
[15] M. N. Asim, M. Wasim, M. U. G. Khan, W. Mahmood, H. M. Abbasi, A survey of ontology learning techniques and applications, Database, The Journal of Biological Databases and Curation 2018 (2018) 1–24. doi:10.1093/database/bay101.
[16] A. C. Khadir, H. Aliane, A. Guessoum, Ontology learning: Grand tour and challenges, Computer Science Review 39 (2021) 100339.
[17] R. Du, H. An, K. Wang, W. Liu, A short review for ontology learning from text: Stride from shallow learning, deep learning to large language models trend, arXiv preprint arXiv:2404.14991 (2024).
[18] P. Gamallo, M. Gonzalez, A. Agustini, G. Lopes, V. S. De Lima, Mapping syntactic dependencies onto semantic relations, in: Proceedings of the ECAI workshop on machine learning and natural language processing for ontology engineering, 2002, pp. 15–22.
[19] F. Giunchiglia, I. Zaihrayeu, Lightweight ontologies, Technical Report DIT-07-071, University of Trento Department of Information and Communication Technology, 2007.
[20] F. Fürst, F. Trichet, Heavyweight ontology engineering, in: On the Move to Meaningful Internet Systems 2006: OTM 2006 Workshops: OTM Confederated International Workshops and Posters, AWeSOMe, CAMS, COMINF, IS, KSinBIT, MIOS-CIAO, MONET, OnToContent, ORM, PerSys, OTM Academy Doctoral Consortium, RDDS, SWWS, and SeBGIS 2006, Montpellier, France, October 29-November 3, 2006. Proceedings, Part I, Springer, 2006, pp. 38–39.
[21] A. Hari, P. Kumar, WSD based ontology learning from unstructured text using transformer, Procedia Computer Science 218 (2023) 367–374.
[22] F. N. AL-Aswadi, H. Y. Chan, K. H. Gan, From ontology to knowledge graph trend: Ontology as foundation layer for knowledge graph, in: Iberoamerican Knowledge Graphs and Semantic Web Conference, Springer, 2022, pp. 330–340.
[23] Y. Tian, G. Chen, Y. Song, X. Wan, Dependency-driven relation extraction with attentive graph convolutional networks, in: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), 2021, pp. 4458–4471.
[24] M. Y. Jaradeh, K. Singh, M. Stocker, A. Both, S. Auer, Information extraction pipelines for knowledge graphs, Knowledge and Information Systems 65 (2023) 1989–2016.
[25] T. 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. Ziegler, J. Wu, C. Winter, D. Amodei, Language models are few-shot learners, Advances in neural information processing systems 33 (2020) 1877–1901.
[26] S. Hao, B. Tan, K. Tang, H. Zhang, E. P. Xing, Z. Hu, Bertnet: Harvesting knowledge

graphs from pretrained language models, arXiv preprint arXiv:2206.14268 (2022).
[27] R. S. I. Wilson, J. S. Goonetillake, A. Ginige, W. A. Indika, Ontology quality evaluation methodology, in: International Conference on Computational Science and Its Applications, Springer, 2022, pp. 509–528.
[28] M. McDaniel, V. C. Storey, Evaluating domain ontologies: clarification, classification, and challenges, ACM Computing Surveys (CSUR) 52 (2019) 1–44.
[29] R. M. Bakker, M. H. T. de Boer, Dynamic Knowledge Graph Evaluation, TechRxiv preprint (2024). Under review.
[30] C. Niklaus, M. Cetto, A. Freitas, S. Handschuh, A survey on open information extraction, arXiv preprint arXiv:1806.05599 (2018).
[31] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, Advances in neural information processing systems 26 (2013).
[32] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, arXiv preprint arXiv:1810.04805 (2018).
[33] S. Min, X. Lyu, A. Holtzman, M. Artetxe, M. Lewis, H. Hajishirzi, L. Zettlemoyer, Rethinking the role of demonstrations: What makes in-context learning work?, in: EMNLP, 2022.
[34] H. Khorashadizadeh, N. Mihindukulasooriya, S. Tiwari, J. Groppe, S. Groppe, Exploring in-context learning capabilities of foundation models for generating knowledge graphs from text, arXiv preprint arXiv:2305.08804 (2023).
[35] D. Alivanistos, S. B. Santamaría, M. Cochez, J.-C. Kalo, E. van Krieken, T. Thanapalasingam, Prompting as probing: Using language models for knowledge base construction, 2023. arXiv:2208.11057.
[36] J. Z. Pan, S. Razniewski, J.-C. Kalo, S. Singhania, J. Chen, S. Dietze, H. Jabeen, J. Omeliyanenko, W. Zhang, M. Lissandrini, R. Biswas, G. de Melo, A. Bonifati, E. Vakaj, M. Dragoni, D. Graux, Large language models and knowledge graphs: Opportunities and challenges, 2023. arXiv:2308.06374.
[37] R. M. Bakker, G. J. Kalkman, I. Tolios, D. Blok, G. A. Veldhuis, S. Raaijmakers, M. H. de Boer, Exploring knowledge extraction techniques for system dynamics modelling: Comparative analysis and considerations, in: Proceedings of the Benelux Conference on Artificial Intelligence (BNAIC), 2023.
[38] J. H. Caufield, H. Hegde, V. Emonet, N. L. Harris, M. P. Joachimiak, N. Matentzoglu, H. Kim, S. Moxon, J. T. Reese, M. A. Haendel, et al., Structured prompt interrogation and recursive extraction of semantics (spires): A method for populating knowledge bases using zero-shot learning, Bioinformatics 40 (2024).
[39] G. Rossiello, M. F. M. Chowdhury, N. Mihindukulasooriya, O. Cornec, A. M. Gliozzo, Knowgl: Knowledge generation and linking from text, in: Proceedings of the AAAI Conference on Artificial Intelligence, volume 37, 2023, pp. 16476–16478.
[40] N. Heist, S. Hertling, H. Paulheim, KGrEaT: a framework to evaluate knowledge graphs via downstream tasks, in: Proceedings of the 32nd ACM International Conference on Information and Knowledge Management, 2023, pp. 3938–3942.
[41] H. Chen, G. Cao, J. Chen, J. Ding, A practical framework for evaluating the quality of knowledge graph, in: Knowledge Graph and Semantic Computing: Knowledge Computing and Language Understanding: 4th China Conference, CCKS 2019, Hangzhou, China,

August 24–27, 2019, Revised Selected Papers 4, Springer, 2019, pp. 111–122.
[42] N. Mihindukulasooriya, S. Tiwari, C. F. Enguix, K. Lata, Text2kgbench: A benchmark for ontology-driven knowledge graph generation from text, in: International Semantic Web Conference, Springer, 2023, pp. 247–265.
[43] C. Guéret, P. Groth, C. Stadler, J. Lehmann, Assessing linked data mappings using network measures, in: The Semantic Web: Research and Applications: 9th Extended Semantic Web Conference, ESWC 2012, Heraklion, Crete, Greece, May 27-31, 2012. Proceedings 9, Springer, 2012, pp. 87–102.