Enhancing Natural Language Understanding in Large Language Models by Symbolic Representation

Bingqian Li[1,*], Baiyang Song[2,*] and Yi Zhou[3,†]

[1]ShanghaiTech University, Shanghai, China
[2]University of Science and Technology of China, Hefei, China
[3]University of Science and Technology of China, Hefei, China

Abstract

This paper presents the Symbolically Enhanced Neural Inference Framework (SENIF), which enhances the natural language understanding (NLU) capabilities of large language models (LLMs) such as GPT-4 by combining large language models with symbolic representations. The proposed method aims to improve the performance of LLMs by enabling them to infer based on formalized statements. The framework employs Assertional Logic (AL) as its foundational representation. Initially, the framework translates natural language utterances into logical expressions after developing a Concept-Operator diagram (CO) within the domain. We propose a zero-shot parser that enables smaller language models to yield high-quality parsing results for a given Concept-Operator Diagram. We then design a Chain-of-Thought (CoT) prompt that utilizes both the original text and the parsing results from the preceding step as inputs. Experimental results show that LLMs, like GPT-4, can greatly benefit from these high-quality parsing results. Our framework exhibits substantial improvement in GPT-4's performance, elevating the most challenging measure, C@90, by 46.67% (40% $\rightarrow$ 86.67%). Meanwhile, we have also verified its feasibility in modeling in different fields and medium language models. This research provides a promising direction for enhancing the inference capabilities of large language models.

Keywords

Domain Knowledge, Semantic Parsing, Symbolic Representation,

1. Introduction

Natural Language Understanding (NLU) is a challenging task, even for the most advanced and powerful language models. This task entails a comprehensive understanding, often requiring not only the syntactic structure of the language but also semantic meanings, contextual cues, and pragmatic factors. This intricate nature of language comprehension presents a formidable challenge even for large models such as ChatGPT or GPT-4.

Human comprehension of the world is a synthesis of perception and cognition, indicating that our understanding is not purely based on data-driven processes [1]. Rather, it involves a combination of learned knowledge, experiences, and symbolic reasoning. Therefore, it stands to reason that mixing symbolic representations into large language models may enhance the language understanding capabilities of large models [2, 3]. By integrating symbolic representations, models may be able to better encode and utilize abstract, high-level concepts and relationships inherent in language.

Both formal reasoning and language models exhibit imperfections in language understanding. Formal reasoning, despite its proficiency in concept comprehension and inference, is often hindered by generalization issues, impeding its practical application. In contrast, large language models, despite their expansive coverage, often fail to accurately capture complex reasoning processes, limiting their reliability. We could even say that the accuracy of language models in machine reading comprehension tasks relies more on suitable QA pairs, rather than a genuine understanding of the question. This point is emphasized and robustly tested by the ZEST benchmark, which is why we have chosen to focus our efforts on this dataset [4].

In light of these challenges, we first use the CO Diagram based on assertion logic to achieve the symbolic representation of domain prior knowledge, then we use a CoT prompt-based approach to incorporate it into the neural network, this method can integrate the generalization and fuzzy matching capabilities of language models with the precision of formal representations. This innovative strategy significantly improves model performance on tasks related to language understanding.

Moreover, to efficiently obtain formal representations in an open domain, we present a semantic parser for assertional logic [5]. This algorithm confers several advantages, including swift cross-domain migration, ease of improvement, and independence from annotated data. Addressing these core challenges in the field of semantic parsing is of utmost importance.

To validate our claims, we apply our proposed methodology to approximately 200 examples extracted from the ZEST benchmark. We further annotate about 400 assertions in assertional logic to evaluate the performance of our zero-shot parser. Meanwhile, we used a subset of ZEST for automatic and hasty modeling, and fine-tuned llama3 based on the data parsed from this CO Diagram. Our experiments show two key insights: 1) formal reasoning is an essential complement to neural inference ($40.00\% \rightarrow 73.33\%$), 2) high-quality parsing results are key to benefitting the language model ($40.00\% \rightarrow 86.67\%$). Our approach is effective for quick and dirty domain modeling and also for fine-tuning on moderate models. However, if the parsing and reasoning processes are suboptimal, they may potentially decrease the performance in Machine Reading Comprehension (MRC) significantly ($30.00\% \rightarrow 6.67\%$ for turbo).

In conclusion, our contributions are as follows:

1. We introduce the Symbolically Enhanced Neural Inference Framework (SENIF), which mimics the way humans process semantics and cleverly combines the powerful capabilities of language models with symbolic representations. This innovative blend leverages the strengths of the former's a generalization and fuzzy matching capabilities, along with the precision

KiL'24: Workshop on Knowledge-infused Learning co-located with 30th ACM KDD Conference, August 26, 2024, Barcelona, Spain
*Corresponding author.
† Both authors contributed equally to this research.
libq2022@shanghaitech.edu.cn (B. Li); baibaibai@mail.ustc.edu.cn (B. Song); yi_zhou@ustc.edu.cn (Y. Zhou)

of the latter, to markedly improve model performance on NLU tasks.

2. A semantic parser for assertional logic is proposed to facilitate the efficient translation of natural language into formal representations in an open domain. It achieves state-of-the-art performance on a semantic parsing dataset annotated with assertional logic.

2. Background

2.1. Concept-Operator Diagram

The Concept-Operator Diagram (CO diagram) is a graphical representation of a knowledge representation model that is based on assertional logic. In this logic, knowledge is represented in the form of "$a = b$", where a and b are either atomic individuals or compound individuals. There are three components of its syntax: individual, concept, and operator. Concepts are represented as rectangles in the diagram, while operators are represented as diamonds. Since individuals only represent specific instances of concepts, they are not typically included in a CO diagram.

Figure 1 is an illustration of the CO diagram. The concept is represented by a rectangle and the operator by a diamond, and we capitalize concept names for the sake of distinguishing between concepts and operators, especially when written as logical expressions. In this figure, 'NUMBER' refers to the set of numbers in mathematics, such as $1, 5.201, \frac{1}{3}$, and so on. While the 'addition' represents a logical operation or a logical relation or a map from LHS to RHS. The logical expression corresponding to Figure 1 is *addition* (NUMBER, NUMBER) = NUMBER. The semantics is that the sum of two numbers equals another number. An example of this operator is $2 + 3 = 5$.

Concepts and operators can be nested and considered as individuals as well. Additionally, CO Diagram serves for assertional logic, which possesses higher-order logic expressiveness at least. This allows for representing complex relationships and rules like the Pythagorean theorem, which is challenging for tuple-based KBs.

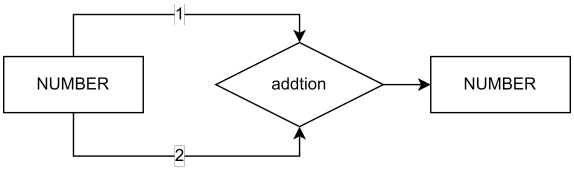


Figure 1: A simple example for CO diagram.

Compared to traditional entity-relationship (E-R) models, the CO model has several advantages. The E-R model can only describe existing data, while the CO model is capable of expressing logical relationships between concepts, such as that classic example $2 + 3 = 5$. This logical expression is difficult to represent in an E-R diagram but can be easily represented in a CO diagram. The numbers on the arrows in the CO diagram indicate the order of the concepts in the operator, with "2" being the first input and "3" being the second input in the example of $2 + 3 = 5$.

The CO model is an expressive model that enhances traditional data models by enabling reasoning and inference capabilities. Moreover, It overcomes the limitation of the traditional model, which is unable to perform inference. This enables the CO model to be used for modeling various types of concepts and their relationships to describe wide knowledge.

The CO diagram is a powerful tool for representing knowledge in a way that is both intuitive and expressive. It allows for logical relationships to be expressed clearly and concisely.

3. Methodology

3.1. Pipeline

The whole steps of the proposed SENIF are shown in Figure 2. To enhance the performance of the traditional method that leverages language models for NLU tasks, our research introduces symbolic representations and simple reasoning into the existing framework. The central hypothesis is that by infusing these two elements, the model can handle higher-level, abstract thoughts that often elude pre-trained language models, therefore improving overall performance.

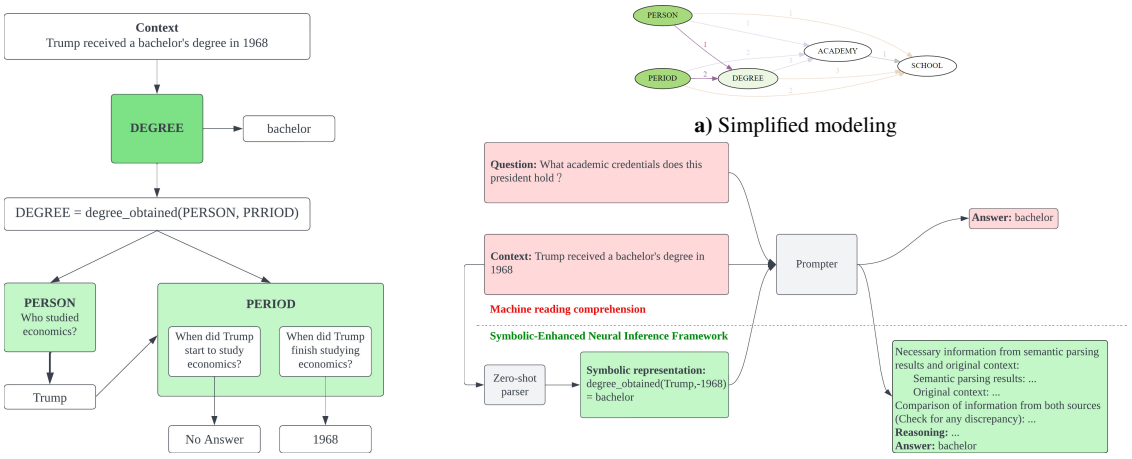
- **Domain-specific CO diagram** We construct a domain-specific CO diagram based on the collected domain information text, which contains the necessary meta-knowledge in a domain.
- **Parsing based on CO diagram** Our parsing procedure is conducted based on a predefined domain-specific CO diagram, as shown in Figure 2a and Figure 3.
  Allow for generalization, we have designed a zero-shot parser to handle it (Figure 2b). We treat the parsing task as a combination of Named Entity Recognition (NER) and MRC tasks.
- **Integrating symbolic representation and reasoning** Therefore, we incorporate an additional semantic parsing dimension to the existing inputs of question and context. Moreover, we designed a chain-of-thought prompt that effectively integrates these three inputs (question, context, and semantic parsing results) for further analysis, as illustrated in Figure 2c.

3.2. Domain-specific CO Diagram

To begin with, we need to build a corpus from https://www.whitehouse.gov/about-the-white-house/presidents/ to model the presidential domain of the ZEST benchmark, which contains concise but essential information about the presidents. The information gathered from the website can be used to abstract the core concepts and extract the relationship called operator. And this information is in natural language format and does not require any annotation or processing.The operator helps algorithms understand how the different concepts are related to each other, and they help algorithms integrate domain-specific knowledge.

Based on this corpus, we use both manual processing and large language model automatic processing to abstract concepts and operators from natural language, and expand outward with different conceptual relationships, ultimately establishing a model that covers this field and meets modeling quality standards.

The criteria for modeling quality include less semantic information loss, simplicity, etc. We will now explore some of these criteria in detail to help understand how they can be achieved in modeling the education experience of presidents.



a) Simplified modeling

b) Zero-shot parser

c) Adding the parsing and reasoning process

Figure 2: The pipeline of Symbolic-Enhanced Neural Inference Framework (SENIF).

The first example is for less semantic information loss. Compared the "*resident_place* (PERSON) = PLACE, *resident_period* (PERSON) = PERIOD" and correct one "*resident_info* (PERSON, PERIOD) = PLACE" for contexts like "The family lived in Lamar until Harry was ten months old". The first one will lose the dependencies between a certain place and a certain period. In other words, the inference system will be confused if there're multiple places and periods of residence.

For simplicity, too many variables would make the model difficult to extract and infer. For instance, "*school_of* (PERSON) = SCHOOL and *belong_to* (CLASS) = SCHOOL" are better than "*class_of* (PERSON, SCHOOL) = CLASS" because the information of the latter can be derived from the easier former. Another example is "*birth_date* (PERSON) = DATE and *birth_place* (PERSON) = PLACE" versus "*birth_info* (PERSON, DATE) = PLACE". We prefer the first one because they have the same semantics as long as life only has once.

Achieving all quality criteria simultaneously at the same time is near impossible. We need to balance them well to achieve the best model. This balance is different in different fields and it requires experimentation in the modeling field.

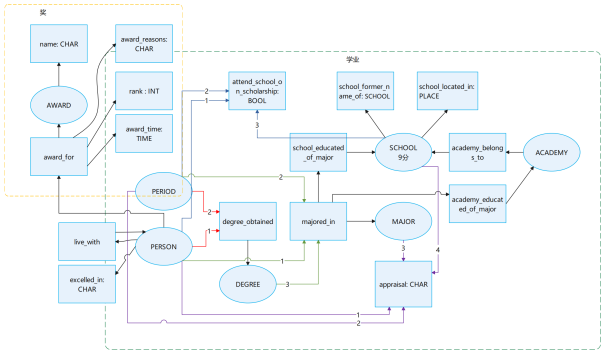


Figure 3: The part of our CO diagram.

3.3. Zero-shot Semantic Parser

Most existing semantic parsing datasets are limited to parsing short sentences and single facts [6, 7]. Although MIVS [8] has introduced a semantic parsing dataset for multiple facts, it is essentially a compilation of single-fact datasets, making it relatively mechanical and challenging to apply to real-world scenarios. So we developed a simple zero-shot semantic parsing.

3.3.1. Two-stage algorithm

This paper presents a semantic parsing process that is controlled by a given CO diagram and designed for an open-domain task. This parsing process is difficult to accomplish using traditional algorithms or even advanced language models such as ChatGPT or Davinci without finetune.

We use a two-stage algorithm. In the first stage, we utilize an open-domain named entity recognition (hereafter referred to as OpenNER) model to recognize individuals with certain concepts, while in the second stage, a MRC system is applied to fill variables for certain operators that are related to concepts identified in stage one. The MRC process is based on question templates generated automatically. This two-stage approach allows us to capture the relationships between individuals and individuals more accurately and efficiently. In this paper, we use UIE [9]) and DeBERTa-v3-large-squad2 as the base model.

3.3.2. Templates for MRC step

The MRC system starts with a pre-compiled set of templates, where each template corresponds to a specific operator. The MRC system can answer questions like "Who is six feet tall?" by using the template "Who is [HEIGHT] tall?", the template corresponds to questions asking for the person with a particular height. Therefore, it's necessary to construct templates automatically in a zero-shot scenario.

Capitalizing on the advancements in in-context learning, it has become feasible to generate question-answer templates for each operator. Thus completing the final step towards constructing a parser for a given CO diagram, with almost complete automation and without annotations.

The generation process is prompted by the combination of instruction, chain-of-thought, and standard prompting, which we have found to achieve an appropriate balance between quality and variety. We present a brief overview of this schema in Table 1. We found that this combination is better

Table 1
question templates generation for operators.

Whole prompt	
An instruction and a CoT	Your aim is given the question templates for every function and its variables. For example, the input '*age_of*': ['PERSON0', 'AGE1'] indicates... The semantics is...Only after an question template is given, we can suppose that value can be obtained and use it in next template... For instance, you can only use AGE1 in the first step ...
some standard prompts	Q: write question template for formula: '*age_of*': ['PERSON0', 'AGE1'] A: ['Whose age was {age1}?'] ... Q: write question template for formula: "{**Operator**}"

than only using instruction or the chain-of-thought prompt with more examples.

In fact, the number of incorrect templates during the generation process is higher than that of correct ones. But fortunately, some hard constraints can be employed to detect all faults when using the prompt shown in Table 1:

- The number of question templates for each operator should be equal to the number of concepts that need to be filled.
- Every question template is only permitted to use concepts with known values because they are queried one by one.

The complete generation process involves the following steps:

1. Set the temperature to 0.0 and maximal tries to 20.
2. Alternate between using the text-davinci-003 and gpt-3.5-turbo models to generate the templates.
3. Verify the results using the aforementioned hard constraints. If the templates do not pass the test, the temperature is increased by 0.1 and the process is repeated.
4. Repeat steps 2–3 until the correct question templates are generated or the maximal number of tries is reached.

As a result of this schema, correct templates can always be generated if they pass the constraints, with only two operators failing. The absence of templates for a few operators is insignificant in practice.

Moreover, the davinci model is more reliable than the turbo model in precise scenarios, which are consistent with observations when they are used as baselines for zero-shot semantic parsing.

3.4. Case study for Symbolic-Enhanced Neural Inference Framework

Finally, a case study will be applied to introduce the whole steps of SENIF (Figure 2). Consider the question "What academic credentials does this president hold?" and the context "Trump received a bachelor's degree in 1968.".

Suppose that we've construct a CO diagram (Figure 2a), and then zero-shot parser will extract the structural information by a two-stage algorithm (Figure 2b):

1. Identify the degree concept and its individual 'bachelor', and turn to fill the "degree_obtained (PERSON, PERIOD) = DEGREE".
2. Query MRC models by automatically generated templates and get the **symbolic representation**": *degree_obtained* (Trump,-1968) = bachelor".

Next, the generative models will receive the question, context, and symbolic representations as inputs (Figure 2c). The inference process is then completed in five steps: identifying the primary information, selecting the relevant knowledge, synthesizing the original context with the parsing results, performing reasoning, and finally, providing the answer.

4. Experimental Setup

4.1. Datasets and Metrics

Datasets In order to demonstrate the practical significance of our framework and preliminarily explore the potential of integrating symbolic logic reasoning with large language model, we selected a subset of approximately 200 question-answer pairs from the ZEST dataset to test within the specific domain that we manually modeled. This test comprises approximately 200 question-answer problems. With its innovative scoring mechanism (C@K) and challenging problem design, ZEST effectively measures the performance of models in truly understanding the questions, rather than merely obtaining correct answers by chance due to input pairs that happen to fit modelnetwork well. Meanwhile, because our methodology is related to the parsing quality, we need a dataset for the analysis of the parsing quality.

Due to the lack of a publicly available benchmark to assess the performance of semantic parsing for assertional logic, our study has undertaken the annotation of a dataset of 400 assertions to serve as the test dataset. Notably, our approach to semantic parsing does not require the use of training datasets. To improve the reliability of the evaluation, it has some differences in detail, see the appendix B.1.

Furthermore, to quickly verify the effectiveness of our method in other fields, we selected all questions matching the prompt words from the training set of the ZEST benchmark, and used a large model for zero-shot modeling (different from the previous manual plus automatic modeling), including questions in various fields such as the president, national parks, and dog breeds. We tested about 800 question-answer pairs in the modeling of this field to verify the versatility of our method.

Metrics for the NLU task In line with the metrics employed in the foundational study by [4], we utilize Mean F1, C@75, and C@90 for assessment. In this benchmark, each question is associated with around 20 ⟨context, answer⟩ pairs.

Table 2
Comparison on ZEST samples.

Models		Performance		
		Mean	C@75	C@90
Finetuned models	BART-large	51	30	20
	T5-3B	70	60	50
	T5-11B	73	70	60
Davinci	+ few-shot prompt	64	40	10
	+ parsing prompt	54	30	6.67
	+ SENIF (ours)	64.33	36.67	10
Turbo	+ few-shot prompt	73	50	30
	+ parsing prompt	67.67	30	6.67
	+ SENIF (ours)	84	76.67	33.33
GPT-4	+ few-shot prompt	88.67	90	40
	+ parsing prompt	93.66	**100**	73.33
	+ SENIF (ours)	**97**	**100**	**86.67**

The Mean F1 denotes the average F1 score, while C@A represents a specialized evaluation metric where an algorithm only receives 1 score if the average F1 score across approximately 20 ⟨question, context⟩ pairs surpasses the A%.

Metrics for parsing task We present our findings by comparing the precision and recall measures, using the exact match condition, as employed in the SQuAD 2.0 [10] benchmark. Specifically, we perform a variable-wise matching of all assertions, assigning a score of 1 when they're the same and 0 otherwise. The maximal score across all the gold assertions is then determined as the final score. It should be noted that a score of 0 is assigned in instances where the operators do not match, as this implies a lack of consistency in the underlying semantics.

Due to the limitation of zero resources, we have employed NER and QA models to extract facts that align with the semantics of the original context. We do not refine these facts by considering whether they correspond to the original sentences or merely possess similar semantics. For instance, given the context "Alice is the mother of Bob." the facts "mother_of (Bob) = Alice" and "child_of (Alice) = Bob" are both correct, although the latter is not an original sentence. However, this inherent deficiency does not have any practical implications and can even be regarded as advantageous, as it alleviates the difficulties associated with reasoning.

In order to incorporate these accurate facts into the computation of precision and recall metrics, an inference system has been developed to augment the given parsing outcomes. A notable observation is that more extensive language models yield a greater quantity of supplementary facts. This can be ascribed to the superior inference capabilities of larger models, which possess the ability to generate novel facts when processing contexts.

The details of the inference system and ablation experiments are shown in section B.2.

4.2. Baselines

In this study, we evaluate our proposed algorithm by comparing it with the state-of-the-art baselines of ZEST (BART and T5) and the most powerful generative models: Text-Davinci-003, GPT-Turbo-3.5, and GPT-4, all renowned for their few-shot and zero-shot learning capabilities. To ensure a fair comparison and reproducibility, we maintain similar parameters and prompts across different models GPT-family, including temperature (0.0), max_tokens (2048), and a '\n' stop marker. The complete prompts used can be found in Appendix C.2. The training details of BART and T5 can be found in Appendix B.3.

Due to the non-determinacy of generative models, we repeated each experiment three times, then report the mean value.

5. Results and Analysis

5.1. NLU task performance

To demonstrate the superiority of our proposed SENIF in enhancing the language understanding capabilities of large models, we conducted a comparison with the advanced generative models in the NLU task. In the experiments, we employed three types of prompts:

- Using a few-shot prompt, requiring the model to directly respond to the question;
- Utilizing a CoT prompt, which necessitates that the model first parse input through formal expressions, followed by inference and response. We anticipate that this methodology will enhance both the reliability and interpretability of reading comprehension tasks.
- Using the almost same prompt, but replace the parsing results by our zero-shot parser (SENIF).

As evidenced in Table 2, our scheme outperforms the baseline method considerably in the test examples. It is important to note that our proposed approach not only focuses on reading comprehension tasks but just views it as merely one means for validating its effectiveness. The success reveals the feasibility of integrating symbolic logic with neural network-based inference.

Second, it can be observed that the prompt requiring the model to first parse input before answering the question yields weaker results compared to the simple prompt for davinci and turbo. We believe this can be attributed to two main factors:

- The second type of prompt does not provide sample data for the model to learn from the context;
- Insufficiently skilled and reliable parsing results may interfere with the model's output.

However, it is worth noting that by replacing the parsing step with our algorithm's parsing results, a significant improvement can be achieved. We believe this demonstrates the

potential for incorporating symbolic reasoning to enhance inference reliability by language model (The ZEST dataset assessing whether the model genuinely comprehends the questions), but this improvement is reliant on high parsing accuracy – an observation that shares a similar conclusion with CoT's success, which is dependent on the model's accuracy in terms of consistency and fact-based output.

5.2. Evaluation of Semantic Parsing

Table 3
Comparison between models of GPT family and ours on semantic parsing task.

Models	precision	recall	F1
gpt-turbo-3.5	25.63	32.04	28.47
text-davinci-003	38.98	26.30	31.41
GPT-4	56.59	38.38	45.73
Ours	**66.03**	**41.53**	**50.99**

To verify the relationship between our method and parsing quality, we tested the parsing quality of the GPT family and our method.Table 3 presents an overview of the performance of semantic parsing by GPT models and ours.

In our experiments, the proposed model with only about 700M parameters demonstrates a significant performance improvement, achieving approximately a 40.40% increase in precision compared to turbo while surpassing the recall performance of davinci by a 15.23% increase. Notably, Turbo and Davinci models struggle to achieve high precision and recall scores simultaneously, whereas our model attains state-of-the-art results in both aspects.

We attribute this enhancement primarily to the alignment between the assertional logic and our structure. More importantly, these results suggest the potential for driving existing knowledge representation towards greater complexity and controllability (stemming from the construction of the modeling process), ultimately aiding in constructing a more sophisticated knowledge base. This approach holds promise to address challenges faced in knowledge computation that arise from inconsistencies between knowledge representation and knowledge bases, as well as reducing high resource demands for semantic parsing associated with specific or complex languages.

To show the relationship between NLU and parsing performance, we plot the performance difference on the ZEST dataset before and after incorporating the parsing step, with respect to the performance of baseline models on parsing data. From Figure 4, a positive correlation could be observed: parsing results with high precision is a key element for the validity of extra formal steps, and precision is more important than the recall score by comparing Figure 4a and Figure 4b. This finding provides further evidence supporting the claim that our framework relies on the precision of symbolic representation, in conjunction with the fuzzy matching capabilities of large language model, to enable broader reasoning. This observation is in line with our initial hypothesis.

5.3. Generalization Experiment

To quickly and comprehensively validate the generalizability of our approach, we automatically model additional domains from the ZEST benchmark in zero-shot scenarios. Our approach to achieving rapid domain-specific modeling involves the following steps:

Table 4
Performance of SENIF in other fileds.

Models		Performance		
		Mean	C@75	C@90
Turbo	+ few-shot prompt	40	12	0
	+ parsing prompt	37	11	0
	+ SENIF (ours)	**42**	**16**	0

- **Entity Extraction:** Identification of all entities within the text for subsequent concept formation.
- **Entity to Concept:** Abstraction of entities into specific real-world concepts. For example, the entity "red" is abstracted into the concept "COLOR".
- **Relation Extraction:** Identification and extraction of relevant relationships between the extracted entities and their corresponding concepts.

To enhance the quality of modeling, we applied filter conditions to the final results using the prompts detailed in Appendix C.3. We counted the frequency of all concepts, removing concepts and corresponding operators that appeared too infrequently. Additionally, we filtered out operators with identical meanings based on semantic similarity.

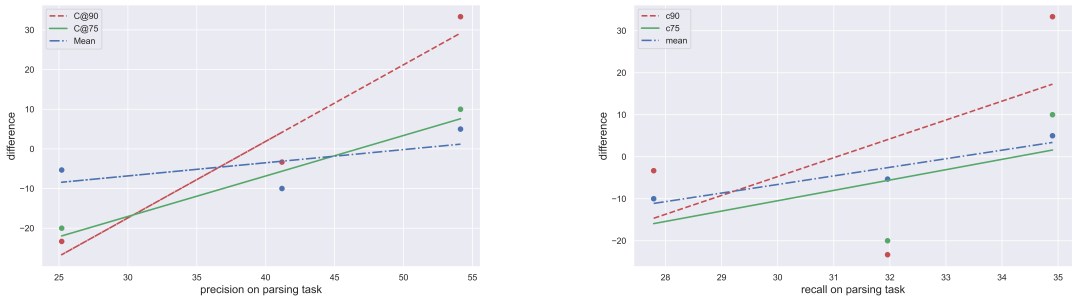
As shown in Table 4, our method consistently achieves optimal results even with rough modeling. This not only verifies the superior generalization capability of our approach but also highlights the potential of combining symbolic language with large language model.

At the same time, we analyzed the reasons for the decline in performance when it was extended to other fields: compared with the manually constructed precise domain CO graph, the quality of zero-shot modeling is significantly worse than that of the manually constructed domain CO graph, and it has obvious problems such as semantic loss and high complexity. For example, for the sentence "Malamutes were thought to be bred by the Malemiut Inupiaq people of Alaska's Norton Sound region.", automatic modeling tends to focus more on the main part of the sentence, that is, modeling "(ANIMAL) be_bred_by(PERSON)" from the sentence, but there is another important semantics in this sentence: (PERSON) live_in(PLACE). These situations lead to a drop in performance in other areas, which also verifies the importance of high-quality domain knowledge in model reasoning.

Furthermore, in order to prove that other models can also combine symbols to improve their language understanding ability, we fine-tune LLaMA3 on the lora framework and use a zero-shot parser to parse data built from automatically generated CO-Diagrams. We use the zero-shot parser to process a subset of the training set in the ZEST benchmark, a total of 700 question-answer pairs, and use this as the fine-tuning dataset. We fine-tune llama3 in two forms: question-answer pairs (Q/A) and question-answer pairs plus our parsing results (Q/A/R). In the Table 5 we can see that our method continued to achieve superior performance in the fine-tuned LLaMA3, this suggests that models can benefit from domain knowledge or structured knowledge.

6. Related work

NLU in symbolic AI AI system based on logic are skilled in reasoning and have a deep understanding of concepts. Past researchers try to construct elaborate representation



(a) (b)

Figure 4: The relationship between performance on parsing task and NLU task

Table 5
The performance of SENIF on other models

Models	Performance		
	Mean	C@75	C@90
llama3-8b-instruct(Q/A)	40	9	0
llama3-8b-instruct(Q/A/R)	**46**	**20**	0

frameworks such as knowledge base [11, 12], axiom system for highly specialized domains like pouring water [13, 14] and so on. However, these systems struggle with the issue of over-generalization and are difficult to acquire.

NLU in LLMs On the other hand, language models have powerful universal capabilities for many downstream tasks [15, 16], but they lack a true understanding of the world and are weak in reasoning [17, 18], [19, 20]. LLMs might only use patterns [19], the suitable input pair [4], or take shortcuts [21] to infer, without truly understanding the background context.

Symbolic-enhanced systems Therefore, researchers have made numerous efforts to combine traditional AI with language models. Approaches include neuralizing rule-based system [22, 23], neural module network [24, 25], soft or hard symbolic constraints [26, 3], formal reasoning-based system [27] and so on. Despite these attempts, these methods have yet to successfully combine the advantages of symbolism and connectionism, often relying too heavily on the capabilities of one over the other. We believe that the most beneficial elements of these two technology pathways are the fuzzy matching capability of large language model and the high precision of symbolic systems. Our work focuses on merging these elements within advanced generative models. We use symbolic representation to provide precise knowledge and language models to enable universal inference.

7. Conclusion

We have explored an innovative approach (SENIF) for augmenting the comprehension capabilities of large language models. Our findings suggest that integrating symbolic representation into LLMs significantly improves the NLU ability, offering promising directions for future advancements in the field.

Further, the introduction of a zero-shot parser designed for the CO diagram is another significant contribution of our work. The parser's capacity for quick cross-domain migration, ease of enhancement, and independence from annotated data make it a potent tool for translating natural language into formal representations, a critical step in improving NLU tasks.

We conduct empirical validation on the NLU examples and our own annotated semantic parsing dataset. The results offer strong evidence of our approach's efficacy, while our findings also underscore its potential for cross-domain applicability.

8. Limitations

Our approach works well in zero-shot scenarios and naturally benefits from the enhancement of NER and MRC models without additional effort. However, in the process of using information extraction for approximate semantic parsing, it will also be troubled by reasoning efficiency, redundancy of extraction, and the congenital gap between them, which will affect the further expansion of scale and accuracy. Meanwhile, our zero-shot parsing algorithm will be affected by scale. When facing Large-scale domain knowledge CO Diagrams, its complexity will affect the reasoning speed.

Furthermore, the challenge of multi-step reasoning tasks remains unresolved for large language model. Therefore, it is imperative to pursue further investigations based on the proposed framework in order to integrate the capabilities of large language model more deeply into the reasoning process.

References

[1] K. Mahowald, A. A. Ivanova, I. A. Blank, N. Kanwisher, J. B. Tenenbaum, E. Fedorenko, Dissociating language and thought in large language models: a cognitive perspective, arXiv preprint arXiv:2301.06627 (2023).

[2] H. Zhang, Z. Liu, C. Xiong, Z. Liu, Grounded conversation generation as guided traverses in commonsense knowledge graphs, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online, 2020, pp. 2031–2043. URL: https://aclanthology.org/2020.acl-main.184. doi:10.18653/v1/2020.acl-main.184.

[3] K. Zhou, K. Zheng, C. Pryor, Y. Shen, H. Jin, L. Getoor, X. E. Wang, Esc: Exploration with soft commonsense constraints for zero-shot object navigation, arXiv preprint arXiv:2301.13166 (2023).

[4] O. Weller, N. Lourie, M. Gardner, M. E. Peters, Learning from task descriptions, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, Online, 2020, pp. 1361–1375. URL: https://aclanthology.org/2020.emnlp-main.105. doi:10.18653/v1/2020.emnlp-main.105.

[5] Y. Zhou, From first-order logic to assertional logic, in: T. Everitt, B. Goertzel, A. Potapov (Eds.), Artificial General Intelligence, Springer International Publishing, Cham, 2017, pp. 87–97.

[6] M. Moradshahi, V. Tsai, G. Campagna, M. S. Lam, Contextual semantic parsing for multilingual task-oriented dialogues, arXiv preprint arXiv:2111.02574 (2021).

[7] S. Gupta, R. Shah, M. Mohit, A. Kumar, M. Lewis, Semantic parsing for task oriented dialog using hierarchical representations, arXiv preprint arXiv:1810.07942 (2018).

[8] H. Xu, R. Cao, S. Zhu, S. Jiang, H. Zhang, L. Chen, K. Yu, A birgat model for multi-intent spoken language understanding with hierarchical semantic frames, arXiv preprint arXiv:2402.18258 (2024).

[9] Y. Lu, Q. Liu, D. Dai, X. Xiao, H. Lin, X. Han, L. Sun, H. Wu, Unified structure generation for universal information extraction, in: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Dublin, Ireland, 2022, pp. 5755–5772. URL: https://aclanthology.org/2022.acl-long.395. doi:10.18653/v1/2022.acl-long.395.

[10] P. Rajpurkar, R. Jia, P. Liang, Know what you don't know: Unanswerable questions for squad, arXiv preprint arXiv:1806.03822 (2018).

[11] D. B. Lenat, M. Prakash, M. Shepherd, Cyc: Using common sense knowledge to overcome brittleness and knowledge acquisition bottlenecks, AI magazine 6 (1985) 65–65.

[12] R. Speer, J. Chin, C. Havasi, Conceptnet 5.5: An open multilingual graph of general knowledge, in: Proceedings of the AAAI conference on artificial intelligence, volume 31, 2017.

[13] E. Davis, Pouring liquids: A study in commonsense physical reasoning, Artificial Intelligence 172 (2008) 1540–1578.

[14] E. Davis, Logical formalizations of commonsense reasoning: a survey, Journal of Artificial Intelligence Research 59 (2017) 651–723.

[15] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al., Language models are few-shot learners, Advances in neural information processing systems 33 (2020) 1877–1901.

[16] S. Bubeck, V. Chandrasekaran, R. Eldan, J. Gehrke, E. Horvitz, E. Kamar, P. Lee, Y. T. Lee, Y. Li, S. Lundberg, H. Nori, H. Palangi, M. Tulio Ribeiro, Y. Zhang, Sparks of Artificial General Intelligence: Early experiments with GPT-4, arXiv e-prints (2023) arXiv:2303.12712. doi:10.48550/arXiv.2303.12712. arXiv:2303.12712.

[17] Z. Yuan, H. Yuan, C. Tan, W. Wang, S. Huang, How well do Large Language Models perform in Arithmetic tasks?, arXiv e-prints (2023) arXiv:2304.02015. doi:10.48550/arXiv.2304.02015. arXiv:2304.02015.

[18] B. Zhou, D. Khashabi, Q. Ning, D. Roth, "going on a vacation" takes longer than "going for a walk": A study of temporal commonsense understanding, 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, Hong Kong, China, 2019, pp. 3363–3369. URL: https://aclanthology.org/D19-1332. doi:10.18653/v1/D19-1332.

[19] C. Durt, T. Froese, T. Fuchs, Against ai understanding and sentience: Large language models, meaning, and the patterns of human language use, 2023. URL: http://philsci-archive.pitt.edu/21983/.

[20] A. Lenci, Understanding natural language understanding systems. a critical analysis, arXiv preprint arXiv:2303.04229 (2023).

[21] Y. Lai, C. Zhang, Y. Feng, Q. Huang, D. Zhao, Why machine reading comprehension models learn shortcuts?, in: Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, Association for Computational Linguistics, Online, 2021, pp. 989–1002. URL: https://aclanthology.org/2021.findings-acl.85. doi:10.18653/v1/2021.findings-acl.85.

[22] S. Li, H. Xu, Z. Lu, Generalize symbolic knowledge with neural rule engine, ArXiv abs/1808.10326 (2018).

[23] C. Jiang, Y. Zhao, S. Chu, L. Shen, K. Tu, Cold-start and interpretability: Turning regular expressions into trainable recurrent neural networks, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2020, pp. 3193–3207.

[24] J. Andreas, M. Rohrbach, T. Darrell, D. Klein, Neural module networks, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 39–48.

[25] K. Yi, J. Wu, C. Gan, A. Torralba, P. Kohli, J. Tenenbaum, Neural-symbolic vqa: Disentangling reasoning from vision and language understanding, Advances in neural information processing systems 31 (2018).

[26] N. Tandon, B. Dalvi, J. Grus, W.-t. Yih, A. Bosselut, P. Clark, Reasoning about actions and state changes by injecting commonsense knowledge, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Brussels, Belgium, 2018, pp. 57–66. URL: https://aclanthology.org/D18-1006. doi:10.18653/v1/D18-1006.

[27] A. Rajasekharan, Y. Zeng, P. Padalkar, G. Gupta, Reliable natural language understanding with large language models and answer set programming, ArXiv abs/2302.03780 (2023).

A. Modeling results for CO diagram

A.1. Concepts and Operators

Concepts are shown in table 6 while operators are shown in table 7.

Concepts	Explanation
ACADEMY	A section of an university or college
AGE	The length of time a person has lived, typically measured in years
AWARD	A prize or recognition is given for achievement or merit
BOOL	A data type that can hold one of two values, typically "true" or "false"
DATE	A specific day
DEGREE	An academic title awarded for completion of a program of study
DESCENT	Refers to a person's ancestry or ethnic background
GENDER	The state of being male or female (or non-binary)
HEIGHT	The vertical measurement of a person
ILLNESS	A medical condition or disease
INT	A data type that can hold integer values
MAJOR	The main subject of study in a college or university program
NATIONALITY	The status of belonging to a particular country or nation
PARTY	A group or organization with shared beliefs or goals, often in a political context
PERIOD	A specific time period
PERSON	A human being
PLACE	A location or area
PRESIDENT	The head of a country or organization, often in a political context
PROFESSION	A person in a certain period of responsibility, generally refers to the profession, work, occupation, job, career, position, etc
RACE	A Human populations
RANK	A position in a hierarchy or order of importance
SCHOOL	An institution for education, often refer to university and college

Table 6: Concepts

Operator	Explanation	LHS	RHS
degree_obtained	Indicates the degree obtained by a person during a period	PERSON, PERIOD_S, PERIOD_T	DEGREE
majored_in	Indicates the major subject studied by a person during a period and while obtaining a certain degree	PERSON, PERIOD_S, PERIOD_T, DEGREE	MAJOR
school_educated_of	Indicates the school where a person obtained a certain degree during a period	*school_educated_of* (PERSON, PERIOD_S, PERIOD_T, DEGREE)	SCHOOL
academic_educated_of	Indicates the academic institution where a person obtained a certain degree during a period	*academy_educated_of* (PERSON, PERIOD_S, PERIOD_T, DEGREE)	ACADEMIC
academy_belongs_to	Indicates the school where an academy or department belongs to	ACADEMY	SCHOOL
school_located_in	Indicates the location of a school	SCHOOL	PLACE
school_former_name_of	Indicates the former name of a school	PERSON	PERSON
death_date	Indicates the date of death of a person	PERSON	DATE
birth_date	Indicates the date of birth of a person	PERSON	DATE
birth_place	Indicates the place of birth of a person	PERSON	PLACE
GetHeight	Indicates the height of a person	PERSON	HEIGHT
resident_in	Indicates the place of residence of a person during a period	PERSON, PERIOD_S, PERIOD_T	PLACE
died_in	Indicates the place where a person died	PERSON	PLACE
father_of	Indicates the father of a person	PERSON	PERSON
mother_of	Indicates the mother of a person	PERSON	PERSON
spouse_of	Indicates the spouse of a person	PERSON	PERSON
son_of	Indicates the son of a person	PERSON	PERSON
daughter_of	Indicates the daughter of a person	PERSON	PERSON
sibling_of	Indicates the sibling of a person	PERSON	PERSON
gradeparent_of	Indicates the grandparent of a person	PERSON	PERSON
grandchild_of profession_of	Indicates the grandchild of a person Profession of a person during a period	PERSON PERSON, PERIOD_S, PERIOD_T	PERSON PROFESSION
which_president_rank_of	Rank of a president	PRESIDENT	RANK
race_of	Race of a person	PERSON	RACE
gender_of	Gender of a person	PERSON	GENDER
nationality_of	Nationality of a person	PERSON	NATIONALITY
descent_of	Descent of a person	PERSON	DESCENT
which_children_rank_of	Rank of a child in a family	PERSON	RANK
party_affiliation_of	Political party affiliation of a person	PERSON	PARTY
alias_of	Alias or nickname of a person	PERSON	NAME
age_of	Age of a person	PERSON	AGE
illness_of	Illness of a person during a period	PERSON, PERIOD_S, PERIOD_T	ILLNESS
studied_subject_of	Major studied by a person during a period	PERSON, PERIOD_S, PERIOD_T	MAJOR
someone_nominate_someone_for_profession	Nominate someone for a profession during a period	PERSON, PERIOD_S, PERIOD_T, PERSON	PROFESSION
number_of_children	Number of children of a person	PERSON	INT
number_of_grandchildren	Number of grandchildren of a person	PERSON	INT
is_married	Whether a person is currently married	PERSON	BOOL
is_divorced	Whether a person is currently divorced	PERSON	BOOL
start_time_of_period	Start time of a time period	PERIOD_S, PERIOD_T	DATE
terminal_time_of_period	Terminal time of a time period	PERIOD_S, PERIOD_T	DATE
year_of_date	Year of a date	DATE	INT
month_of_date	Month of a date	DATE	INT
day_of_date	Day of a date	DATE	INT
award_for	Award received by a person	PERSON	AWARD
award_time	Date when a person received an award	*award* (For the convenience of annotating, we set LHS as "PERSON, AWARD" this time)	DATE

Continued on next page

Table continued from previous page

Operator	Explanation	LHS	RHS
live_with	Person who lived with another person during a period	PERSON, PERIOD_S, PERIOD_T	PERSON
great_grandson_of	Represents the relationship of being a great grandson of someone.	PERSON	PERSON
child_of	Represents the relationship of being a child of someone.	PERSON	PERSON
nickname	Represents the nickname given to a person.	PERSON	NAME
brother_of	Represents the relationship of being a brother of someone.	PERSON	PERSON
sister_of	Represents the relationship of being a sister of someone.	PERSON	PERSON
uncle_of	Represents the relationship of being an uncle of someone.	PERSON	PERSON
aunt_of	Represents the relationship of being an aunt of someone.	PERSON	PERSON
succeeded_by	Represents the succession of a person by another person in a certain profession at a specific date.	PERSON	PROFESSION, DATE, PERSON
married_with	Represents the marriage between two persons that occurred on a specific date.	PERSON	DATE, PERSON
son-in-law_of	Represents the relationship of being the son-in-law of someone.	PERSON	PERSON

Table 7: Operators

B. Details of evaluation

B.1. Restriction for operators

Certain operators may possess ambiguities that are not aligned with the annotation standard. For instance, the *alias_of* operator is designed to capture distinct names used by an individual in varying periods or circumstances, such as nicknames, former names, pseudonyms, etc. However, we notice that the full name and its abbreviation may also be regarded as the alias of a person, as exemplified by Barack Hussein Obama II, Barack Hussein Obama, Barack Obama, and Obama. Recording such information may be meaningless and challenging to label without omissions. Consequently, these operators are omitted when calculating the precision and recall score.

Meanwhile, two operators encountered failure during the template generation step: "succeeded_by" and "someone_nominate_someone_for_profession". To make a fair comparison without manual intervention, we refrained from creating the corresponding question templates. As a result, these two operators were excluded from the evaluation.

B.2. Inference system

We utilize 29 rules about family relationships and personal information to generate complete semantics, please see Table 8. Table 9 indicates the relevant ablation experiments.

B.3. Training settings for BART and T5

For BART-large, we use the same setup as in the [4]. However, for T5-3B and 11B models, as we did not have access to TPUs, we replicate the experiments using 4x3090 24G GPUs and 2xA800 80G GPUs. It was observed that when running under these resource constraints, the setup described in the paper employing 16x8 TPUs yielded poor results (even worse than BART-large). Therefore, we opted for an alternative configuration that produced the best performance for these two baselines. Specifically, an initial learning rate of 5e-5 was employed for 3 epochs during the training process (in fact, the best performance for T5-11B is the one after two epoch training). Moreover, we also set batchsize as 32 but achieve it by batchsize=1 and gradient_accumulation_steps=32. This is because we find that any optimization may result in T5 not converging, so it is significantly limited by memory.

C. Complete prompts

C.1. Templates generation prompt

We generate MRC templates with the prompt provided in Table 10:

sibling_of (A) = B → *sibling_of* (B) = A
child_of (A)=B ∧ *child_of* (A)=C → *sibling_of* (B)=C
¬ *resident_in* (A, any, any) = A ∧ *birth_place* (A) = B → *resident_in* (A, '', '') = B
resident_in (A, '', '') = B ∧ *birth_date* (A) = B ∧ *birth_place* (A) = C → *resident_in* (A, B, '') = C
father_of (A) = B → *child_of* (B) = A
mother_of (A) = B → *child_of* (B) = A
spouse_of (A) = B → *spouse_of* (B) = A
sister_of (A) = B → *sibling_of* (A) = B
brother_of (A) = B → *sibling_of* (A) = B
mother_of (A) = B ∧ *father_of* (A) = C → *spouse_of* (B) = C
mother_of (A) = C ∧ *mother_of* (B) = C → *sibling_of* (A) = B
father_of (A) = C ∧ *father_of* (B) = C → *sibling_of* (A) = B
son_of (A) = C ∧ *spouse_of* (A) = B → *son_of* (B) = C
daughter_of (A) = C ∧ *spouse_of* (A) = B → *daughter_of* (B) = C
son_of (A) = B → *child_of* (A) = B
daughter_of (A) = B → *child_of* (A) = B
child_of (A) = B ∧ *spouse_of* (A) = C → *child_of* (C) = B
profession_of (A, any, any) = president → *gender_of* (A) = man
gender_of (A) = man ∧ *spouse_of* (A) = B → *gender_of* (B) = woman
son_of (A) = B → *gender_of* (B) = man
daughter_of (A) = B → *gender_of* (B) = woman
father_of (A) = B → *gender_of* (B) = man
mother_of (A) = B → *gender_of* (B) = woman
brother_of (A) = B → *gender_of* (B) = man
sister_of (A) = B → *gender_of* (B) = woman
child_of (A) = B ∧ *gender_of* (B) = man → *son_of* (A) = B
child_of (A) = B ∧ *gender_of* (B) = woman → *daughter_of* (A) = B
child_of (A) = B ∧ *gender_of* (A) = man → *father_of* (B) = A
child_of (A) = B ∧ *gender_of* (A) = woman → *mother_of* (B) = A

Table 8
Inference rules for metric function

models	precision	recall	F1
gpt-turbo-3.5	24.83 (0.83)	32.04 (0.96)	27.97 (0.82)
text-davinci-003	38.60 (0.31)	26.19 (0.10)	31.21 (0.17)
GPT-4	56.18 (1.17)	**38.33** (1.80)	45.56 (1.63)
Ours	**60.14**	37.89	**46.48**

Table 9
Ablation experiments for inference system (without inference)

C.2. Parsing baselines

The prompt used for semantic parsing task is given in Table 11.

C.3. Auto modeling

The templates we use for automatic modeling are provided in 12

C.4. Downstream task baselines

In this section, we present the prompts utilized for the baseline and our semantic parsing results, with the differences between these two prompts highlighted in red for easy identification (Table 13 and Table 14). Our objective was to facilitate fair comparisons; thus, we intentionally introduced only subtle discrepancies in the first set of prompts. These modifications were primarily focused on incorporating our parsing results into the first prompt.

Moreover, due to the inclusion of a lengthy text (i.e., parsing results) at the end of a prompt may potentially confuse the language model and cause it to lose track of its tasks, we incorporated reminders ("Follow above ... the question") to maintain consistency and ensure that all steps are successfully executed.

For the few-shot prompt, please see the table 15 below.

Your aim is given the question templates for every function and its variables. For example, the input 'age_of': ['PERSON0', 'AGE1'] indicate the formula age_of (PERSON0) = AGE1. The semantics of this formula is The age of PERSON0 is AGE1. The value ['PERSON0', 'AGE1'] are the elements related to this function. Concretely, the last one ('AGE1' in this case) is always defined as the output while others (['PERSON0'] in this case) are input (s) for the function. You can always suppose that we've gotten the value of the output ('AGE1' in this case), so just write question templates for the previous elements (['PERSON0'] in this case) one by one and from first to the last but one. Only after an question template is given, we can suppose that value can be obtained and use it in next template. Otherwise, you only can use the output variable and input variables with given templates. To avoid the confusion about two same variable names, an unique id from 0 to n-1 is added to the end of variable names. The question order is 0, 1,... n-2 when we have n elements.
For instance, you can only use AGE1 in the first step to construct template for PERSON0 (no 0-th variable in your first template). After that, you are allowed to use 0-th variable and output variable to design the next template (no 1-th variable in your second template) if there are any variables left over, and so on.
Q: write question template for formula: 'age_of': ['PERSON0', 'AGE1']
A: ['Whose age is age1?']
Q: write question template for formula: 'birth_date': ['PERSON0', 'DATE1']
A: ['Who was born in date1?']
Q: write question template for formula: 'death_date': ['PERSON0', 'DATE1']
A: ['Who died in date1?']
Q: write question template for formula: 'GetHeight': ['PERSON0', 'HEIGHT1']
A: ['Whose height is height1 tall?']
Q: write question template for formula: 'degree_obtained': ['PERSON0', 'PERIOD_S1', 'PERIOD_T2', 'DEGREE3']
A: ['Who has recieved degree3?', 'When did person0 start degree3?', 'When did person0 recieved degree3?']
Q: write question template for formula: 'majored_in': ['PERSON0', 'PERIOD_S1', 'PERIOD_T2', 'DEGREE3', 'MAJOR4']
A: ['Who majored in major4?', 'When did person0 start to study major4?', 'When did person0 graduate in major4?', 'What degree was person0 study for major4 in period_s1-period_t2?']
Q: write question template for formula: 'school_located_in': ['SCHOOL0', 'PLACE1']
A: ['Which school is located in place1?']
Q: write question template for formula: 'borned_in': ['PERSON0', 'PLACE1']
A: ['Who borned in place1?']
Q: write question template for formula: 'father_of': ['PERSON0', 'PERSON1']
A: ['Whose father is person1?']
Q: write question template for formula: 'mother_of': ['PERSON0', 'PERSON1']

Table 10
Prompt for question templates generation

Please parsing the given context into structured data by preset templates. The information that cannot be covered by templates should be ignored. I'll give you the templates and some examples. Then you should parsing the next context.
Note that you are only allowed to use the words or phrases in the context.
The templates are following:
{'degree_obtained': ['PERSON0', 'PERIOD_S', 'PERIOD_T', 'DEGREE']}
...
Context: Nixon's visit to China in 1972 eventually led to diplomatic relations between the two nations.
Answer: ['operator': 'visited_place', 'PERSON0': 'Nixon', 'PERIOD_S1': '1972', 'PERIOD_T2': '', 'PLACE3': 'China']
Context: The black bear is a common inhabitant of Olympic National Park, and North America, in general.
Answer: ['operator': 'inhabitant_animal_of', 'PLACE0': 'Olympic National Park', 'ANIMAL1': 'black bear','operator': 'inhabitant_animal_of', 'PLACE0': 'North America', 'ANIMAL1': 'black bear']
Context: Dachshunds have a wide variety of colors and patterns, the most common one being red.
Answer: ['operator': 'common_color_of', 'ANIMAL0': 'Dachshunds', 'COLOR1': 'red']
Context: Six Trump campaign advisers and staff were indicted and five pled guilty to criminal charges. Answer: []

Table 11
Prompt for semantic parsing (baselines)

Please imitate the following example to extract operators from the given text, and only the answers are output, taking into account all meanings of the coverage statement.
The process of operator extraction is as follows: 1. First identify the named entities in the sentence. For example, for the sentence 'Dachshunds have a wide variety of colors and patterns, the most common one being red', identify Dachshunds, red. 2. To extend entities into categories, try to think of extensions to larger and actually existing categories in nature, such as Dalmatians not extending to dogs but to animals, and Lincoln Park not extending to parks but to places.Don't use non-existent concepts like "danger" and "DESCRIPTION". 3. Extract the operator $\text{common}_{color_of from it and get common\_color\_of(ANIMAL)} = COLORExample$:
Context: Dwight David 'Ike' Eisenhower (EYE-zn-how-r; October 14, 1890 – March 28, 1969), GCB, OM was an American army general and statesman who served as the 34th president of the United States from 1953 to 1961.
Answer: ['birth_date(PERSON) = DATE', 'death_date(PERSON) = DATE', 'profession_of(PERSON,PERIOD_START, PERIOD_TERMINAL) = PROFESSION', 'which_president_rank_of(PRESIDENT) = RANK']
Context: The most common animals observed around Rim Drive are golden-mantled ground squirrels, Canada jays and an assortment of butterflies and bees. Black bear sightings are more common in autumn and late spring.
Answer: ['common_observed_in(PERIOD,PLACE) = ANIMAL']
Context: The English White Terrier is the failed show ring name of a pricked-ear version of the white fox-working terriers that have existed in Great Britain since the late 18th century.
Answer: ['existed_in(PERIOD,ANIMAL) = PLACE', 'ring name(ANIMAL) = ANIMAL']
Context: Black bear \u2013 Ursus americanus. The black bear is a common inhabitant of Olympic National Park, and North America, in general. They are smaller and darker than the grizzly bear and the brown bear. Females typically weigh between 100 and 400 lbs, while males weigh between 250 and 600 lbs.
Answer:

Table 12
prompt for auto modeling

Given the question '{question}' and the original context '{context}', please:
1. Identify the main concepts and relationships involved in the question. Provide the semantic parsing results of the context based on this ontologies, in first-order logic form.
2. Select necessary information from both the semantic parsing results and the original context.
3. Compare the information from these two sources. If there is a discrepancy, resolve it by deciding which source is likely to be more accurate.
4. Combine the verified pieces of information and present your line of formal reasoning in first order logic.
5. Output the answer without any extra details by "Answer:{answer}" format. The answer should be yes, no, n/a or a brief phrase from the input words based on the question and context. n/a means no answer."'

Table 13
Prompt for semantic parsing (baselines)

For a given question '{question}', the original context '{context}', and corresponding semantic parsing results (at the end), please:
1. Identify the main concepts and relationships involved in the question.
2. Select necessary information from both the semantic parsing results and the original context.
3. Compare the information from these two sources. If there is a discrepancy, resolve it by deciding which source is likely to be more accurate.
4. Combine the verified pieces of information and present your line of formal reasoning in logic.
5. Output the answer without any extra details by "Answer:{answer}" format. The answer should be yes, no, n/a or a brief phrase from the input words based on the question and context. n/a means no answer.
Semantic parsing results:{parsing_results}
Follow above five steps exactly to complete the question

Table 14
Prompt for adding our semantic parsing

Give an answer from 'yes, no, n/a', or a brief phrase from the input words based on question and context, n/a means no answer.

Question: After leaving office, where did this president go to retire?
Context: Dwight David 'Ike' Eisenhower (EYE-z\u0259n-how-\u0259r; October 14, 1890 \u2013 March 28, 1969), GCB, OM was an American army general and statesman who served as the 34th president of the United States from 1953 to 1961. Following the war, he served under various generals and was promoted to the rank of brigadier general in 1941. After the United States entered World War II, Eisenhower oversaw the invasions of North Africa and Sicily before supervising the invasions of France and Germany. After the war, he served as Army Chief of Staff (1945\u20131948), as president of Columbia University (1948\u20131953) and as the first Supreme Commander of NATO (1951\u20131952). While Eisenhower was stationed in Texas, he met Mamie Doud of Boone, Iowa. Eisenhower was mostly reluctant to discuss his death. Their second son, John Eisenhower (1922\u20132013), was born in Denver, Colorado. John served in the United States Army, retired as a brigadier general, became an author and served as U.S.
Answer: n/a

Question: Are bear sightings common at this national park?
Context: Black bear \u2013 Ursus americanus. The black bear is a common inhabitant of Olympic National Park, and North America, in general. They are smaller and darker than the grizzly bear and the brown bear. Females typically weigh between 100 and 400 lbs, while males weigh between 250 and 600 lbs.
Answer: yes

Question: Are bear sightings common at this national park?
Context: This area is thickly forested. Moose and, less commonly, bears can be seen if they are near the road; otherwise, wildlife sightings are fairly rare. The road rises up to Mile 9, eventually breaking out of spruce forest and into a low alpine zone of tall bushes and sporadic trees. Moose frequent this stretch during the autumn (mid-August to mid-September). Caribou and bears can occasionally be seen, especially toward Savage River (Mile 15). Mountain-dwelling critters, like marmots, pika and Dall sheep are sometimes seen on Healy Ridge and Mount Margaret.
Answer: no

Question: What public offices did this president run for and win?
Context: Johnson won election to the United States Senate from Texas in 1948 after winning the Democratic Party's nomination by an extremely narrow margin with fraudulent votes that were manufactured by friendly political machines. He was appointed to the position of Senate Majority Whip in 1951. He became the Senate Minority Leader in 1953 and the Senate Majority Leader in 1955. At the same time as his vice presidential run, Johnson also sought a third term in the U.S. Senate. According to Robert Caro, 'On November 8, 1960, Lyndon Johnson won election for both the vice presidency of the United States, on the Kennedy\u2013Johnson ticket, and for a third term as senator (he had Texas law changed to allow him to run for both offices).
Answer: United States Senate|vice presidency

Question: ...
Context: ...
Answer:

Table 15
Few-shot prompt