Identifying Logical Patterns in Text for Reasoning

Pauline Armary, Cheikh-Brahim El-Vaigh, Antoine Spicher, Ouassila Labbani Narsis, Christophe Nicolle

Identifying Logical Patterns in Text for Reasoning

Pauline Armary
CIAD UR 7533
Université de Bourgogne, UB, Anabasis Assets
F-21000 Dijon, France
pauline.armary@u-bourgogne.fr

Cheikh-Brahim El-Vaigh
CIAD UR 7533
Université de Bourgogne, UB
F-21000 Dijon, France
Cheikh-Brahim.El-Vaigh@u-bourgogne.fr

Antoine Spicher
R&D Department
Anabasis Assets
Saint-Denis, France
spicher@anabasis-assets.com

Ouassila Labbani Narsis
CIAD UR 7533
Université de Bourgogne, UB
F-21000 Dijon, France
ouassila.narsis@u-bourgogne.fr

Christophe Nicolle
CIAD UR 7533
Université de Bourgogne, UB
F-21000 Dijon, France
cnicolle@u-bourgogne.fr

***Abstract*—Translating unstructured text into logical format is a key challenge for building ontologies automatically and addressing deductive inference. Most of the approaches have tackled the identification of concepts and relations in text, but few of them have addressed the most complex axioms like class expression subsumption. This work proposes DeLIR, a neuro-symbolic approach to identify complex logical patterns in text by combining a grammatical translation of dependency parsing trees and a fine-tuned Large language Model (LLM). DeLIR combines the strength of the parsing accuracy provided by a grammatical approach and pattern flexibility provided by a fine-tuned LLM. We evaluated our approach on FOLIO dataset for both translation capacity and inference capability. Our grammatical approach has a perfect parsing accuracy and combining the grammatical approach with LLMs improves the LLMS translation capacity: tinyLlama, T5-small-text2logic, Llama-7B and Mistral-7B. We also evaluate the inference capacity of the different LLMs. Mistral-7B, while being smaller than the state-of-the-art approach using GPT-4, presents similar results to predict the correct inference labels.**

***Index Terms*—Ontology Learning, Translation to Logic, Natural Language Inference**

I. Introduction

Ontology captures a sound semantic machine-understandable description of digital content or expert knowledge to link information together and allows automatic deductive reasoning. Classically, a domain ontology is constructed manually by a dedicated work of ontologists and knowledge engineers through a careful understanding of the domain. These human experts also embed constraints of the domain in the ontology, in the form of axioms and rules, which are required to perform deductive reasoning and draw inferences based on first-order logic. Nevertheless, building an ontology manually can be tedious and time-consuming. Therefore, the task of *Ontology Learning* (OL) seeks to overcome the knowledge acquisition bottleneck in ontology construction by automating the extraction of ontological elements from data. The primary motivation is to minimize the manual effort and cost associated with building ontologies and keep pace with rapidly evolving knowledge, e.g. [1]–[7]. If the identification of classes and relations within data is a well-studied topic, integrating axioms and rules within the ontology remains an open challenge identified in the literature [1], [5].

Overall, the literature on OL identifies two distinct tasks for learning axioms: 1)The inductive task [8], [9] consists of inferring causal relationships based on the statistical observation of numerous correlations. 2) The translation task [10] involves transforming information from textual to first-order logic. The main complexity consists in resolving language ambiguities when converting unstructured text to an ontology. Our work seeks a translation-based approach for learning general axioms from textual documents.

Several approaches have been proposed in the literature to address the task of OL from text [2], including early ontology learning systems that used natural language processing, and machine learning techniques, to extract ontological elements from text. Moreover, with the explosion of the large language models (LLM), several approaches have been proposed using LLM to learn ontologies from text [11], reaching the non-taxonomic relations extraction level. Meanwhile, first-order logic axioms are learned in [12], [13] from the text and translated into first-order logic for making inferences. These two approaches seek to learn general axioms from text using neuro-symbolic approaches that combine a fine-tuned LLM with dedicated prompts and a first-order logic prover to translate text to a first-order logic for deductive reasoning purposes.

Our work proposes DeLIR, a neuro-symbolic approach that uses grammatical parsing of sentences based on universal dependencies [14]. The idea is first to perform a grammatical translation of the input text to respect the syntax and semantics. The output of the grammatical translation is used as expert rules to train an LLM allowing to control the learning process

We thank the French National Research and Technology Association (ANRT) for their financial support provided through the CIFRE fellowship [Grant number 2023/0276]. This work was performed using HPC resources from GENCI–IDRIS [Grant 2023-AD011013058R1]

of the LLM to avoid hallucination. The proposed grammatical translation comprises three main tasks: identifying named entities in a sentence, associating the pronouns and their co-references to the appropriate variable or individual, and transforming grammatical patterns into first-order logic motifs using tree traversals algorithms on Dependency Parsing Tree (DPT).

In order to validate our approach, we compare it to different well-known LLMs such as T5, LLama, or Mistral [15]–[17], and show that combining these LLMs with our grammatical approach leads to the best accuracy score when translating text documents to first-order logic formula. Moreover, we tested our approach on the inference task, which consists of evaluating a logical conclusion from a set of premises. Our approach improves the inference accuracy for small LLMs such as tinyLlama-1B and reaches similar accuracy to the state-of-the-art models using smaller LLMs (Mistral-7B). The experiments are conducted on the public dataset FOLIO [18].

The main contributions of the paper can be summarized as follows:

1) A novel state-of-the-art neuro-symbolic grammar-based approach that allows translating text to first-order logic formula with a syntactic accuracy of 97.9%;
2) A hybrid translation approach combining the neuro-symbolic approach with foundational LLM models, which we show to be more robust and accurate for translating text to first-order logic using both BLEU and ROUGE metrics;
3) A detailed experiment showing the accuracy of our approach on the public translation and reasoning tasks proposed in FOLIO.

The paper is organized as follows: sec. II introduces some key concepts for our work, sec. IV presents the description of our proposed solution, sec. V details the various experiments we have conducted, and sec. VI proposes a discussion of the results and their limits.

II. Background

Ontology learning can be seen as learning specific first-order logic axioms that are translated into an ontology model. This section first explains how first-order logic axioms can be translated into ontology data models, such as the Web Ontology Language (OWL). Moreover, we also explain hereunder the tasks that we tackle in this work, namely the translation of the text into first-order logic axioms and natural language reasoning, as well as the used evaluation metrics.

A. Learning first-order logic axioms

World Wide Web Consortium (W3C) defines two standards for expressing ontologies: the Web Ontology Language (OWL) [1] and RDF/S, OWL being the most expressive and the mostly used one to integrate axioms within an ontology. OWL is grounded on Description Logics (DLs), a family of formal languages, which is a subset of first-order logic defined through different profiles to reach balances between expressiveness and reasoning complexity. Table I presents the common axioms in Description Logic with their equivalence in first-order logic. From this table, it is possible to translate first-order logic axioms to Description Logic.

[1]https://www.w3.org/TR/owl2-syntax/

TABLE I: Equivalence between Description Logics Axioms and First-Order Logic Formulas.

DL component	**Axiom Classification**	**First-Order Logic**
Concept Assertion	Instances	$C(x)$
Role Assertion	Instances	$r(x, y)$
Negation	Class Expression Sub.	$\neg C(x)$
Conjunction	Class Expression Sub.	$C(x) \wedge D(x)$
Disjunction	Class Expression Sub.	$C(x) \vee D(x)$
Existential Restriction	Class Expression Sub.	$\exists y[r(x, y) \wedge C(y)]$
Universal Restriction	Class Expression Sub.	$\forall y[r(x, y) \rightarrow C(y)]$
Concept Inclusion (GCI)	Class Axiom	$\forall x[C(x) \rightarrow D(x)]$
Concept Equivalence	Class Axiom	$\forall x[C(x) \leftrightarrow D(x)]$
Disjoint Concept	Class Axiom	$\forall x[C(x) \rightarrow \neg D(x)]$
Transitivity	Relation Axiom	$\forall x \forall y \forall z[(r(x, y) \wedge r(y, z) \rightarrow r(x, z)]$
Reflexivity	Relation Axiom	$\forall x[r(x, x)]$
Symmetry	Relation Axiom	$\forall x \forall y[r(x, y) \rightarrow r(y, x)]$
Role Inclusion	Relation Axiom	$\forall x \forall y[r(x, y) \rightarrow s(x, y)]$
Inverse role	Relation Axiom	$r(y, x)$

The different axioms may be classified into three categories based on the W3C classification: Class Axiom, Object Property Axiom, and Class Expression Subsumption. Class axioms correspond to class restrictions (inclusion, equivalence, disjoint concepts). Properties axioms are restrictions over relations (transitivity, reflexivity, symmetry, role inclusion, inverse role). Class expression subsumption combines several class expression axioms (negation, conjunction, disjunction, existential, and universal restriction) with general concept inclusion (GCI) to create complex formulas. Finally, instances are identified with the attribution of an individual to a concept or a role (concept assertion and role assertion).

Identifying axioms within texts consists of translating the text into its equivalent first-order logic formula and capturing within those formulas the logical patterns that correspond to their Description Logic equivalent. Doing so may control the level of expressiveness and decidability to target for a given data model.

B. Translation Task

The translation task from Natural Language text to logic axioms can be formulated as follows: Let P be a set of sentences $\{p_1, p_2, p_3, ...p_n\}$ expressed in natural language. The translation task consists in finding the appropriate first-order logic set of formulas $\phi = \{\phi_1, \phi_2, \phi_3, ...\phi_n\}$ so that for a given translation score f, $f(P, \phi)$ is maximized.

This task is evaluated based on two aspects: the quality of the translation and the accuracy of the parsing for the generated logic formula.

The translation quality is assessed using the well-known machine-translation metrics: sacreBLEU and ROUGE. SacreBLEU [19] is a shareable variation of the BLEU [20] score, which assesses the precision of the proposed translation by comparing the shared number of bi-grams and n-grams over the whole translation, normalizing the distribution to provide a score between 0 and 100. ROUGE [21] is a metric defined in text summarization where identifying all the elements is more important than presenting them in the same order. It computes the number of n-grams in the reference text, which also occurs in the generated text. BLEU is mainly oriented on the precision while ROUGE is primarly oriented on the recall but can be adapted to compute the precision also and the F1-score as well. We use both the BLEU and F1-score version of ROUGE in our evaluation.

The parsing accuracy is assessed using a logic parser provided by the nltk library [2], which can be integrated afterward with theorem provers like Prover9 [3]. We assessed the achievement of the parsing for each sentence with a binary value (1 if parsed, 0 if not) and computed the percentage of parsed sentences over the dataset (score between 0 and 100).

C. Reasoning Task

The reasoning task consists of evaluating a set of conclusions sentences C based on a set of premises. The evaluation results can be either True, False or Undefined. Therefore, the reasoning task can be defined as follows: given a set of premises P (resp. it first-order logic formulas ϕ) and a conclusion C (resp. it first-order logic formulas ξ), is it possible to deduce C from P (is there a model that satisfies $\phi \vdash \xi$). In practice, this task is implemented as a classification task that takes P (resp. ϕ) and C (resp. ξ) and outputs the labels True, False, or Undefined. The evaluation is performed using the accuracy metric which determines the number of correct answer over all the predicted labels.

III. Related Work

This section introduces an overview of the prominent methods from the literature. We first discuss the task of ontology learning from text and then the task of natural language inference.

A. Ontology Learning from Text

Most of the current approach to address the challenge of identifying axioms and rules within text relies on lexico-syntactic approaches [22]–[26], using the grammatical information and key lexical elements to identify the hypotheses. Meanwhile, [10] paved the way towards Neural Networks and Deep Learning, proposing an approach for neural machine translation. Most of the studies focus on the identification of class axioms and class expression (negation, union, intersection, general class inclusion (GCI), equivalence, cardinality, value restriction, universal and existential quantification) [10], [23]–[26]. Nevertheless, [22], [27] address Rules beyond the expression of the OWL axioms, mostly relying on the Semantic Web Rule Language (SWRL), a combination of OWL and the Datalog RuleML sublanguage based on Horn-rules.

Moreover, the approach in [24] presents a strategy based on the Grammatical Dependency Parsing Tree combined with the identification of relevant words expressing specific logic patterns ('only', 'before', 'between'), which are mapped to their corresponding logic structure (expressed in Description Logic), or to identified predicate (between is given a specific predicate). On the other hand, [10] tackles the task as a neural network translation and uses a Recurrent Neural Network (RNN) trained on labeled data to learn Description Logic axioms, mainly class expressions from definition sentences. However, those attempts do not cover a large variety of axioms and rules at once but cover different subsets of axioms.

B. Natural Language Inference

Several works have proposed a system to address the first-order logic inference task based on FOLIO dataset [18], either as a full neural system to perform the reasoning with LLM [28] using Chain-of-Thought or as a neuro-symbolic system combining both LLM and symbolic reasoners [12], [13]. The LINC system presented in [12] proposes an LLM for translation, combined with Prover9 as their first-order logic prover. LogicLM combines different symbolic solvers and approaches (Logic Programming, First-Order Logic Prover, Constraint Optimization and SMT Solver), with the translation provided by LLM. Both approaches compare their results to zero-shot inference from the LLM GPT-4 and Chain-of-Thought techniques. SymCoT [28] takes a full neural approach and improves the Chain-of-Thought techniques by splitting it into four dedicated modules: translating into first-order-logic, planning the inference step by step, solving the inference task and verifying the accuracy of the inference before giving the final answer.

However, most of those works focus on the inference task, only addressing the translation task as a mean for inference, in the case of neuro-symbolic systems. Yet, the complexity of the translation task was identified as a key challenge by [12], [13], [18], especially when sentences are very close to natural language, as in FOLIO. One of the crucial aspects to address is the variation of translation proposed by neural systems like LLM, which does not keep consistency in translating the set of sentences into the same logical equivalent and often generates translations that have errors in the logical syntax, detrimental for a symbolic parser. The work in [12] tried to tackle the problem by asking an LLM to generate the translation several times and taking the mean evaluation provided by the symbolic reasoner at the end of the inference process. This solution, while being efficient to mitigate the syntactic errors generated by LLM, may not be robust enough when addressing a large corpus of natural language sentences, which increases the time complexity and the probability of syntactic errors.

[2]https://www.nltk.org/howto/inference.html

[3]http://www.cs.unm.edu/mccune/prover9/

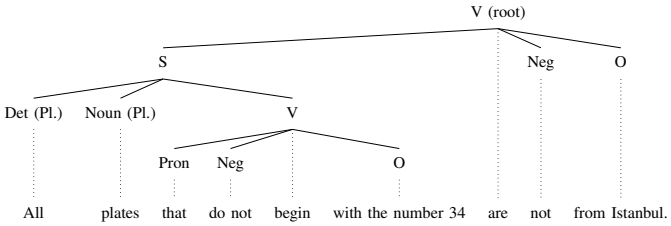

Fig. 1: Parsing of the sentence's grammatical dependencies.

IV. Methodology

A. Grammatical Dependency

Our algorithm is grounded on grammatical parsing of sentences based on universal dependencies [14]. Universal Dependencies (UD) is a framework that defines grammatical annotations across different human languages. It defines the hierarchical structure of clauses through an asymmetric binary relation from the head to the dependant. Through all the dependencies, the whole sentence is parsed as a tree from the root corresponding to the central proposition's verb. The UD framework provides a standard for defining large language treebank corpora across 150 languages. Those treebanks provided an important dataset that ground the current improvement of grammatical parsing. Large Language Models reached the current state-of-the-art specifically trained on those corpora, especially transformers like BERT [29].

Our algorithm defines a recursive path over the tree of dependency relations, as defined in the universal dependency standard [14], following the definitions hereunder.

Definition 1. A *dependence parsing tree* (DPT) $\langle w, p, C \rangle$ is:

- a *token* w, which is a simple sequence of characters,
- a *part of speech* p among UD tags,
- a collection C of children DPT, each with a *dependence* among UD dep.

In more precise steps, our algorithm travels bottom-up across the whole tree of dependencies by recursively identifying the syntactic pattern within the graph. Fig. 1 presents how our algorithm identifies patterns within the grammatical dependency tree, like Subject Verb Object (SVO) clauses or relative clauses attached to a noun. When all the syntactic patterns have been identified, the first-order logic formula is created bottom-up from the lower leaves to the root of the tree, using the compositional quality of the first-order logic formula.

A key challenge to transform a sentence into logic is the identification and appropriate association of individuals and variables, to address them with the appropriate logical form. We address it with the construction of co-reference cluster, via a state-of-the-art coreference resolution algorithm [30], which provides reference cluster following the definition below:

Definition 2. A *coreference cluster* $\langle i, T \rangle$ is:

- a *identifier* i, which is a simple sequence of numbers and characters which uniquely identify the cluster,

TABLE II: Grammatical Patterns and First-Order Logic translation

Grammatical Patterns	**FOL translation**	**Axioms**
S V O	$S(x) \wedge O(y) \wedge v(x,y)$	Conjunction
S NEG V O	$S(x) \wedge O(y) \wedge \neg v(x,y)$	Negation
S_1 [and] S_2 V O	$[S_1(x) \wedge O(y_1) \wedge v(x,y_1)] \wedge [S_2(z) \wedge O(y_2) \wedge v(z,y_2)]$	Conjunction
S_1 [or] S_2 V O	$[S_1(x) \wedge O(y_1) \wedge v(x,y_1)] \vee [S_2(z) \wedge O(y_2) \wedge v(z,y_2)]$	Disjunction
Noun [which/that] V O	$\exists y[N(x) \wedge O(y) \wedge v(x,y)]$	Class expression with Ex. restriction
S V Attr	$S(x) \rightarrow A(x)$	Concept Inclusion
IF S V_1 O_1, S V_2 O_2	$[S(x) \wedge O_1(y) \wedge v_1(x,y)] \rightarrow [O_2(z) \wedge v_2(x,z)]$	General Concept Inclusion
S[Plural] V O	$\forall x[S(x) \rightarrow (O(y) \wedge v(x,y))]$	Universal restriction
S[NE] V O[NE]	$v(\alpha, \beta)$	Relation assertion
S[NE] V Attr	$A(\alpha)$	Concept Assertion

- a collection T of disjoint DPT which refers to the same individual or variable.

The co-reference algorithm, therefore, associates all DPT, with a noun or personal pronoun as its root, to a unique identifier. The co-reference cluster is essential to keep the coherence of the individual and variable across the text. In the context of translating rules, quantification requires bounding all co-referent variables to the same quantifier, especially for long and complex rules. Our algorithm associates this identifier of the co-reference cluster to the appropriate unique variable or individual when entities are identified. To identify entities in the text, we rely on current NER algorithm [31] to identify the different kinds of entities. Most of the state-of-the-art algorithms associate different tags to the entities recognized: person, location, organization, work-of-art, money, numerical values, languages, etc. Depending on the desired model for the ontology, the different kinds of entities may be addressed as instances or as other types of logical forms, integrating a larger set of entities (integrating money or languages as instances may depend on the modeling choices).

Bounding of the variable to the appropriate quantifier (universal or existential) happens during the parsing, with two strategies: universal quantifiers were identified within the sentence through grammatical patterns, while existential quantifiers were assigned to all unbound variables within the parsing. The first strategy was conducted by identifying patterns in the subject noun phrase to capture generic plurals, as presented in table II, with the corresponding DL axioms. On the other hand, all the variables that were not bounded by a universal quantifier or were not identified as instances were bound to an existential quantifier to meet the requirements of first-order logic.

The other logical operators and patterns (negation $\neg$, conjunction $\wedge$, disjunction $\vee$, concept inclusion $\rightarrow$, etc) are identified within the Dependency Parsing Tree (DPT) through the grammatical patterns presented in table II.

The main grammatical pattern Subject Verb Object (S V O) is translated into a conjunction of classes and relations.

Objects of the verb are identified either as direct objects or indirect objects through a preposition. In this latest case, the preposition is considered part of the verb and integrated within the constructed relation. Grammatical conjunction (and, or) are translated into conjunction and disjunction of conjunction of predicates. Negation identified in the sentence (as negation before the verb or a negative subject such as 'nobody') is translated as negation of the relation. Noun phrases with relative clauses are treated like propositions to construct class expressions, the variable being bound within the whole expression by the type of determinant before the noun (as universal or existential quantifiers).

Concept Inclusion is identified through verbs or verbal expressions that expect an attribute (be, seem, look like, etc.), as the verb doesn't express a specific relation apart from the inclusion of the subject in the attribute scope. General Concept Inclusion is captured through a conditional proposition SVO pattern, with a specific marker (if, when, since, etc.) associated with the main proposition. The coreference clusters are mainly useful in those cases to associate the pronouns between the two propositions. Finally, named entities in the sentences are treated as relation assertion and class assertion.

Therefore, our current algorithms covers all Class Expression Subsumption axioms (negation, conjunction, disjunction, universal and existential quantification, general class inclusion). Concept equivalence may be addressed if two sentences express the inclusion of two concepts within one another ('All A is a B. All B is an A.'). However, this is quite rare in textual documents. Overall, our algorithms covers the profile of the Description Logic family $\mathcal{ALC}$.

B. Large Language Models Fine-tuning

Large Language Models (LLMs) with the Transformer architecture have defined a new state-of-the-art for natural language processing, particularly in information extraction and translation tasks. The Transformer architecture relies on transfer learning techniques for addressing specific tasks: the model is trained using vast corpora on a generic task with a self-supervised approach (predicting the next word).

LLMs can be categorized into three main types: encoders, decoders, and encoder-decoder models. Each type has distinct architectures and applications.

Encoder models that are designed to generate a representation of the input text. These models are beneficial for tasks that require understanding and encoding the input into a fixed-size vector representation, such as the model BERT [29] and its different variations. This model provides the DPT parsing required for the grammatical approach presented above.

Decoder models generate text based on an input prompt. These models are commonly used for tasks that require text generation, such as machine translation and text completion. A well-known example of a decoder is GPT (Generative Pre-trained Transformer) and its different versions, such as GPT-2, GPT-3, and GPT-4. In this work, we rely on Mistral [16] and Llama [17] as decoder models for the reasoning task. Mistral and Llama are both open-source models with much smaller sizes by opposition to GPT-3 and GPT-4.

Finally, encoder-decoder models are designed to transform one sequence into another. These models are beneficial for tasks that involve sequence-to-sequence transformation, such as translation and summarization. In this work, we use the well-known T5 (Text-To-Text Transfer Transformer) [15] for the translation.

These different LLMs are foundational models that can be fine-tuned afterward on a more specific task with a smaller corpora, by updating their weights. This fine-tuning approach reduces the need for vast corpora and resources to train the LLM for each specific task, allowing the models to be more accurate while remaining trainable for particular tasks.

Meanwhile, even if these models (T5, Llama, and Mistral) are not as big as GPT-4 (1.7 trillion parameters), they can not be fully fine-tuned since doing so may completely change all the parameters of the models (and their accuracy), and take a huge computational time. In order to solve these issues, we rely on the new architecture LoRA (Low-Rank Adaptation) [32] for fine-tuning Large Language Models such as BERT, GPT or T5. This architecture freezes the pre-trained model weights before the fine-tuning and creates a rank-decomposition matrices. These matrices are updated during the fine-tuning before being added to the model weights matrix. This approach reduces the number of trainable parameters by 10,000 times and greatly improves the accuracy.

V. Experiments

We conducted our experiments on FOLIO dataset to evaluate both the translation and reasoning tasks.

FOLIO was introduced in 2022 by [18] as a dataset to benchmark the capacity of natural language systems (and Large Language Models in particular) to perform deductive reasoning over a set of natural language sentences. This dataset comprises, for each testing case, a set of premises, a conclusion, and a label asserting if the conclusion follows from the premises (True), if it doesn't (False), or if there is not enough information to conclude (Uncertain). Each premise and conclusion are presented in natural language (English sentences) and first-order logic. Table III presents an example of the dataset. It was constructed, partly manually by logic students and partly automatically following deductive templates, to test the capacity of understanding complex natural language sentences, with many topics and syntactic formulation, and the capacity to perform complex deductive reasoning, with several inference steps required to address the conclusion. FOLIO was constructed to assess two different tasks: an inference task and a translation task. The inference task evaluates the capacity to predict the appropriate label (True, False, Uncertain) given the set of natural language sentences in the premises and conclusion. The translation task evaluates the capacity to formalize the meaning of a natural language sentence in a first-order logic form to perform deductive reasoning.

In the example provided in Table III, the conclusion 'Tom's license plate is from Istanbul' is labeled as False, since Tom's

TABLE III: Example of Premises and Conclusion in FOLIO

Premises	1. All vehicle registration plates in Istanbul begin with 34. 2. Plates that do not begin with the number 34 are not from Istanbul. 3. Joe's vehicle registration plate is from Istanbul. 4. Tom's license plate begins with the number 35. 5. If a license plate begins with the number 35, it does not begin with the number 34.
Premises FOL	$\forall x(VehicleRegistrationPlateIn(x, istanbul) \rightarrow BeginWith(x, num34))$ $\forall x(\neg BeginWith(x, num34) \rightarrow \neg FromIstanbul(x))$ $\exists x(Owns(joe, x) \wedge VehicleRegistrationPlateIn(x, istanbul))$ $\exists x(Owns(tom, x) \wedge BeginWith(x, num35))$ $\forall x(BeginWith(x, num35) \rightarrow \neg BeginWith(x, num34))$
Conclusion	Tom's license plate is from Istanbul.
Conclusion FOL	$\exists x(Owns(tom, x) \wedge VehicleRegistrationPlateIn(x, istanbul))$
Label	False

license plate begins with the number 35 (P4), and doesn't begin with number 34 (P5) and plates which don't begin with number 34 are not from Istanbul (P2). Therefore, Tom's licence plate is not from Istanbul, and the conclusive assertion is False based on this fact.

A. Translation Task

We have conducted four experiments (presented in Fig. 2 and reported in Tab. IV) to assess different systems' capacity to translate a natural language sentence to a first-order logic formula.

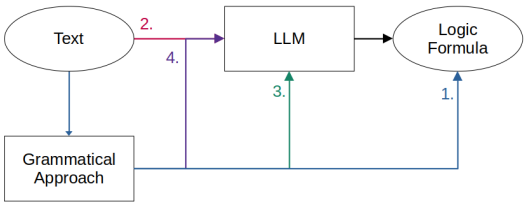

Fig. 2: Experiments set-up combining the Grammatical approach and LLM fine-tuning. Each path is a different set-up to translate text to FOL formula. The different experiments are detailed hereunder.

Experiment 1. (exp_1 in blue) is executed using the grammatical approach presented in sec. IV identifying logical patterns within DPT. We used the grammatical parsing module provided by spacy [4], based on the transformer model for English (en-core-web-trf) which is trained with RoBERTa [33]. This module provided the POS tagging, dependency parsing, and Named-Entity Recognition for our experiment. We complete our pipeline with a final module for co-reference resolution [30]. This pipeline transforms each sentence into a Dependency Parsing Tree (with NER tag and co-reference cluster), which is parsed by our algorithm to generate the first-order logic formulas.

Experiment 2. (exp_2 in red) consists in fine-tuning LLM models (t5-base, t5-small-text2log, mistral-7B, TinyLlama-1B, and Llama-7B) to translate text into first-order logic formula. We trained the models while providing the text of each sentences of the premises as an input and the FOL formula supplied by the FOLIO dataset as the target.

Experiment 3. (exp_3 in green) is conducted in the same way, but as an input, we provided the translated FOL formulas provided by the first experiment's result with the grammatical approach, keeping the FOL formulas provided by FOLIO as the target.

Finally, experiment 4 (exp_4 in purple) consists of fine-tuning the LLM models while providing both the natural language text and the FOL formulas generated by the first experiment with the grammatical approach, it combines exp_2 and exp_3.

B. Reasoning Task

We also fine-tuned several open-source LLMs (TinyLlama-1B, Mistral-7B, Llama-7B, Mixstral-8x7B, and Llama-3-70B) for the reasoning task as classification. The input is the premises and the conclusion, and the output is the labels True, False, or Uncertain.

Two evaluations have been conducted and reported in Table V to assess the quality of our models. First, we evaluate the models in zero-shot learning by inferring the labels on FOLIO validation data without any training. Then, except for the model Llama-3-70b, the models are fine-tuned using LoRA architecture on the FOLIO train set and are evaluated on the validation set. The evaluation of models is based on the accuracy of the reasoning (accuracy of the predicted labels).

All the fine-tuning experiments were conducted using Python's Transformers and Pytorch libraries. The results of the different models for these two tasks are discussed in the next section.

VI. Results & Discussion

A. Translation task

Table IV presents the results of our conducted experiments for the translation task.

TABLE IV: Results of the experiment on translation to logic

Models	BLEU	ROUGE	Parsing
Grammatical approach	36.64	56.79	**97.88**
Text2logic Text	31.98	51.70	14.94
Text2logic Grammar	32.39	50.72	15.83
Text2logic Text + Grammar	31.56	51.33	19.36
T5-base Text	44.16	63.96	31.57
T5-base Grammar	41.63	58.42	25.11
T5-base Text + Grammar	**44.45**	**64.11**	32.18
TinyLlama Text	31.40	58.88	4.24
TinyLlama Grammar	30.84	60.01	3.97
TinyLlama Text + Grammar	36.17	62.97	1.079
Llama7b Text	33.24	61.52	2.91
Llama7b Grammar	38.31	62.79	2.65
Llama7b Text + Grammar	38.09	63.86	2.12
Mistral Text	30.84	60.01	3.98
Mistral Grammar	34.48	62.24	1.59
Mistral Text + Grammar	32.57	62.98	2.92

[4]https://spacy.io/models/en

Among the different methods, the grammatical approach achieved the best Parsing score (97.88%), showing its robustness and accuracy for generating first-order logic formulas. This approach should in theory reach 100% and the delta in performance is due to some errors in the generation of the DPT by the Transformer model below (RoBERTa) which are not captured during the parsing.

The grammatical approach alone shows a BLEU score of 36.64% and a ROUGE score of 56.79%, whereas the T5-base model with fine-tuning over the text achieved the third best BLEU score of 44.16% and the second best ROUGE score(63.96%). Fine-tuning the T5 model using the grammatical approach output gives a lower BLEU and ROUGE scores than the T5-base alone but better than the grammatical approach. However, combining T5 fine-tuning with both the Text and the grammatical translation approach output gave the highest results overall with a 44.45% BLEU score and 64.11% ROUGE score. While the improvement is small since the T5-base is very good for the translation task, this result shows the grammar complements this T5 encoder-decoder model.

We observed an improvement, when integrating the grammatical approach output for the LLMs (exp_4), in the scores of all the decoder LLM (TinyLlama, Llama-7B, and Mistral-7B) by up to +5% on the BLEU and +4% on the ROUGE. This improvement suggests that the grammatical approach is more effective for generative models (decoders) as it allows for the control of the generated tokens. However this approach doesn't reach a score of parsing equivalent to the grammatical approach.

The lower translation score of the grammatical approach may be explained by the narrow coverage of syntactic and logical patterns. In particular, some cases around cardinality restriction (e.g. 'the only animals are rabbits and squirrels') have not been integrated within our handled grammatical rules. There is inherently a lack of flexibility in the grammatical approach to account for grammatical structure, which has not been defined within the patterns. However, this approach ensures that the expected logic formula syntax will be respected and that the logical pattern will be translated in the expected form. There is a much higher level of control over the translation output by maintaining the statistical prediction of the LLM to the lowest level of the pipeline (tagging, ner and co-reference), in area where the state-of-the-art is already well established. On the other hand, fine-tuning a T5 model builds on the flexible capacity of LLMs to process structures that have not been defined and seen previously, despite its lack of precision on the syntactic parsing.

B. Reasoning task

The results for the reasoning task are presented in Table V. The zero-shot learning evaluation shows an increase of the accuracy as function of the number of parameters, large models have great ability for label inference. Mistral (7B) which is the smallest in Table V has an accuracy of 37.% while GPT-4 (1.76T) which is the largest gave an accuracy of 61.3%. We noticed also that LLama-3 (70B) while having less number of parameters than GPT-3 (175B) achieved a higher accuracy (+4%) than the later. Finally the Mixtral (8x7B) which is a mixture of experts version of Mistral is better than Mistral.

TABLE V: The evaluation of the inference task on FOLIO using 0-shot learning and fine-tuning. The column size gives the number of parameters, which can be in millions (M), billions (B), or Trillions (T).

Model	**Size**	**Accuracy (%)**
zero-shot learning		
Mistral	7B	37.4
Mixstral	8x7B	51.2
Llama-3	70B	57.6
GPT-3.5-Turbo [18]	175B	53.1
GPT-4 [18]	1.76T	61.3
Fully supervised fine-tune		
BERT-large [18]	340M	59.0
RoBERTa-large [18]	340M	62.1
Flan-T5-Large [18]	783M	65.9
TinyLlama	1B	60.6
Mistral	7B	77.3
Mixstral	8x7B	75.9
Logic-LM [13]	-	78.1
LINC [12]	-	73.1
Fully supervised translation + inference		
TinyLlama + exp_2	1B	61.1
TinyLlama + exp_4	1B	63.1

We also report the results of the fine-tuned models. We notice that the decoder models (Mistral, Llama) are better than the encoder models (BERT, RoBERTa) and the encoder-decoder T5, this is due to the large size of the decoder models. The two best models are Logic-LM [13], which is based on GPT-4 (1.7T parameters) and Mistral (7B parameters). The fine-tuned Mistral(7B) that we propose here for the inference task is relatively small compared to the GPT-4, it achieved similar results.

Finally, we used the fine-tuned model on the translation task (exp_2 to exp_4) and fine-tuned them on the inference task. We did not notice an improvement for 7B parameters models like Mistral or LLama. Nevertheless, the TinyLlama (exp_4) achieved an improvement of 3% (63.1%) compared to the TinyLlama baseline model (60.6%) that was fine-tuned for the inference task only. This final fine-tuned TinyLlama with only 1B parameters is better than the zero-shot learning of the GPT-4 model with 1.76T parameters (+2%).

VII. Conclusion

This paper presents a neuro-symbolic approach to address the translation of a natural language text into logical formulas for integrating axioms in ontology and making inferences. Our approach combines a grammatical identification of logical patterns in the Dependencies Parsing trees (DPT), and a neural approach by fine-tuning Large Language Models on the FOLIO dataset.

The grammatical approach provides a translation with an accurate syntactic parsing, however it doesn't capture logical patterns which have not been specified beforehand. On the other hand, the fine-tuned T5 model performs better on the translation task by identifying more logical patterns, however it

doesn't reach a satisfying level for logical parsing. The combination of both approaches, while improving the accuracy of the translation and parsing on small (Text2Logic, TinyLlama) and large (T5, Mistral) LLMs, doesn't reach the level of syntactic accuracy of the grammatical approach alone.

Moreover, we fine-tuned different LLM on a natural language inference task, with a zero-shot learning approach and a fully supervised fine-tuning. The fine-tuned Mistral 7B achieved results close to the Logic-LM approach based on GPT-4, with 1000× less number of parameters.

As future work, we will consider designing a join loss function that allows to perform at once text to FOL translation and FOL reasoning. We are also considering the exploration of other logical pattern within the grammatical dependency parsing tree, to identify cardinality restriction, for instance.

REFERENCES

[1] W. Wong, W. Liu, and M. Bennamoun, "Ontology learning from text: A look back and into the future," *ACM Computing Surveys*, vol. 44, pp. 1–36, Aug. 2012.

[2] M. Somodevilla García, D. Vilariño Ayala, and I. Pineda, "An overview of ontology learning tasks," *Computación y Sistemas*, vol. 22, no. 1, pp. 137–146, 2018.

[3] M. N. Asim, M. Wasim, M. U. G. Khan, W. Mahmood, and H. M. Abbasi, "A survey of ontology learning techniques and applications," *Database*, vol. 2018, p. bay101, Jan. 2018.

[4] R. Lourdusamy and S. Abraham, "A Survey on Methods of Ontology Learning from Text," vol. 9, (Cham), pp. 113–123, Springer International Publishing, 2020. Book Title: Intelligent Computing Paradigm and Cutting-edge Technologies Series Title: Learning and Analytics in Intelligent Systems.

[5] A. C. Khadir, H. Aliane, and A. Guessoum, "Ontology learning: Grand tour and challenges," *Computer Science Review*, vol. 39, p. 100339, Feb. 2021.

[6] F. N. Al-Aswadi, H. Y. Chan, and K. H. Gan, "Automatic ontology construction from text: a review from shallow to deep learning trend," *Artificial Intelligence Review*, vol. 53, pp. 3901–3928, Aug. 2020.

[7] S. Yuan, J. He, M. Wang, H. Zhou, and Y. Ren, "A review for ontology construction from unstructured texts by using deep learning," in *International Conference on Internet of Things and Machine Learning (IoTML 2021)*, vol. 12174, pp. 315–322, SPIE, Apr. 2022.

[8] D. Fleischhacker and J. Völker, "Inductive learning of disjointness axioms," in *OTM Conferences*, 2011.

[9] T. H. Nguyen and A. G. B. Tettamanzi, "Learning Class Disjointness Axioms Using Grammatical Evolution," in *Genetic Programming* (L. Sekanina, T. Hu, N. Lourenço, H. Richter, and P. García-Sánchez, eds.), vol. 11451, pp. 278–294, Cham: Springer International Publishing, 2019. Series Title: Lecture Notes in Computer Science.

[10] G. Petrucci, M. Rospocher, and C. Ghidini, "Expressive ontology learning as neural machine translation," *Journal of Web Semantics*, vol. 52-53, pp. 66–82, Oct. 2018.

[11] H. Babaei Giglou, J. D'Souza, and S. Auer, "Llms4ol: Large language models for ontology learning," in *International Semantic Web Conference*, pp. 408–427, 2023.

[12] T. X. Olausson, A. Gu, B. Lipkin, C. E. Zhang, A. Solar-Lezama, J. B. Tenenbaum, and R. P. Levy, "Linc: A neurosymbolic approach for logical reasoning by combining language models with first-order logic provers," in *The 2023 Conference on Empirical Methods in Natural Language Processing*.

[13] L. Pan, A. Albalak, X. Wang, and W. Y. Wang, "Logic-lm: Empowering large language models with symbolic solvers for faithful logical reasoning," in *The 2023 Conference on Empirical Methods in Natural Language Processing*.

[14] M.-C. de Marneffe, C. D. Manning, J. Nivre, and D. Zeman, "Universal Dependencies," *Computational Linguistics*, vol. 47, pp. 255–308, 07 2021.

[15] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *Journal of Machine Learning Research*, vol. 21, no. 140, pp. 1–67, 2020.

[16] A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. d. l. Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, *et al.*, "Mistral 7b," *arXiv preprint arXiv:2310.06825*, 2023.

[17] M. GenAI, "Llama 2: Open foundation and fine-tuned chat models," *arXiv preprint arXiv:2307.09288*, 2023.

[18] S. Han, H. Schoelkopf, Y. Zhao, Z. Qi, M. Riddell, L. Benson, L. Sun, E. Zubova, Y. Qiao, M. Burtell, D. Peng, J. Fan, Y. Liu, B. Wong, M. Sailor, A. Ni, L. Nan, J. Kasai, T. Yu, R. Zhang, S. Joty, A. R. Fabbri, W. Kryscinski, X. V. Lin, C. Xiong, and D. Radev, "FOLIO: Natural Language Reasoning with First-Order Logic," Sept. 2022. arXiv:2209.00840 [cs].

[19] M. Post, "A call for clarity in reporting BLEU scores," in *Proceedings of the Third Conference on Machine Translation: Research Papers*, (Belgium, Brussels), pp. 186–191, Association for Computational Linguistics, Oct. 2018.

[20] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pp. 311–318, 2002.

[21] C.-Y. Lin, "ROUGE: A package for automatic evaluation of summaries," in *Text Summarization Branches Out*, (Barcelona, Spain), pp. 74–81, Association for Computational Linguistics, July 2004.

[22] W. Zaouga and L. B. A. Rabai, "A Decision Support System Based on Ontology Learning from PMI' Project Risk Management," in *Intelligent Systems Design and Applications* (A. Abraham, V. Piuri, N. Gandhi, P. Siarry, A. Kaklauskas, and A. Madureira, eds.), Advances in Intelligent Systems and Computing, (Cham), pp. 732–742, Springer International Publishing, 2021.

[23] C. Kacfah Emani, C. Ferreira Da Silva, B. Fiès, and P. Ghodous, "NALDO: From natural language definitions to OWL expressions," *Data & Knowledge Engineering*, vol. 122, pp. 130–141, July 2019.

[24] B. Gyawali, A. Shimorina, C. Gardent, S. Cruz-Lara, and M. Mahfoudh, "Mapping Natural Language to Description Logic," in *The Semantic Web* (E. Blomqvist, D. Maynard, A. Gangemi, R. Hoekstra, P. Hitzler, and O. Hartig, eds.), vol. 10249, pp. 273–288, Cham: Springer International Publishing, 2017. Series Title: Lecture Notes in Computer Science.

[25] K. A. Mathews and P. S. Kumar, "Extracting Ontological Knowledge from Textual Descriptions through Grammar-based Transformation," in *Proceedings of the Knowledge Capture Conference*, (Austin TX USA), pp. 1–4, ACM, Dec. 2017.

[26] R. R. D. Azevedo, F. Freitas, R. Rocha, J. A. A. D. Menezes, C. M. D. O. Rodrigues, and M. C. Gomes, "Representing Knowledge in DL ALC from Text," *Procedia Computer Science*, vol. 35, pp. 176–185, 2014.

[27] K. Yordanova, "From Textual Instructions to Sensor-based Recognition of User Behaviour," in *Companion Publication of the 21st International Conference on Intelligent User Interfaces*, (Sonoma California USA), pp. 67–73, ACM, Mar. 2016.

[28] J. Xu, H. Fei, L. Pan, Q. Liu, M.-L. Lee, and W. Hsu, "Faithful Logical Reasoning via Symbolic Chain-of-Thought," June 2024. arXiv:2405.18357 [cs].

[29] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186, 2019.

[30] S. Otmazgin, A. Cattan, and Y. Goldberg, "F-coref: Fast, accurate and easy to use coreference resolution," in *AACL*, 2022.

[31] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, and C. Dyer, "Neural architectures for named entity recognition," in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (K. Knight, A. Nenkova, and O. Rambow, eds.), (San Diego, California), pp. 260–270, Association for Computational Linguistics, June 2016.

[32] J. E. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, and W. Chen, "Lora: Low-rank adaptation of large language models," *ArXiv*, vol. abs/2106.09685, 2021.

[33] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.