Data Augmented Pipeline for Legal Information Extraction and Reasoning

Nguyen Minh Phuong Nguyen Ha Thanh May Myo Zin Ken Satoh

ROIS-DS Center for Juris-Informatics, Research Organization of Information and Systems, Tokyo, Japan

1. Overall Deep PROLEG System

The PROLEG knowledge representation language, as outlined by (Satoh 2023; Satoh 2023; Nguyen et al. 2022), is designed to enable lawyers to utilize the legal reasoning system by offering a **minimal legal language that is sufficient for reasoning**. The Deep PROLEG system comprises three major modules:

- (1) **Natural Language Perceiver**: A neural semantic parser is designed and trained to receive a natural *legal case* and parses it to the *facts* in legal knowledge representation language.
- (2) **PROLEG Reasoner**: The *logical rules* of all legal contracts or agreements are installed in a *Symbolic Reasoner using the PROLEG language*. The facts that were obtained from the legal cases outputted in the previous step are transferred to this symbolic reasoner to *verify the truth value of the goal expression*.
- (3) **Inference Explainer**: This module tracks the logical inference in the symbolic reasoner and *visualizes the inferencing flow*, which supports easily inspecting the reasoning process.

Finally, the Deep PROLEG system assists lawyers or courts in evaluating legal cases alongside contract documents to swiftly determine **the entailment of these cases with the established contracts**. Additionally, the system outputs detailed results of the inference flow and corresponding truth values for each reasoning step.

Data Example

Table 1. Examples of Augmented Templates and Slot Holders

Slot holders 1	{"Object": "the house", "Accessory": "garage A", "OriginalOwner": "sarah", "Creditor": "john", "Obligator": "alex" }
Slot holders 2	{"Object": "the apartment", "Accessory": "balcony C", ... }
Template 1	After {OriginalOwner} inherited {Object} from {Creditor}, {Creditor} came across {Obligator} at {Object}, who had erected {Accessory}. {Creditor} requested {Obligator} to leave {Object} and dismantle {Accessory}. In response, {Obligator} asserted that they rented {Object} from {OriginalOwner}, thus claiming rights over {Accessory}. Will {Creditor} be able to reclaim {Object}?
Template 2	During a visit to {Object}, {Creditor} discovered {Obligator} residing there and having constructed {Accessory}, which was inherited ...

PROLEG facts	`original_ownership({OriginalOwner},{Object}).`
	`transfer({OriginalOwner},{Creditor},{Object}).`
	`occupancy({Obligator},{Object}).`
	`existence_of_accessory({Accessory},{Object}).` ...

5. Results and Performance Evaluation

Our proposed method (`http://136.187.109.3:5003/`) significantly reduces the human effort required to implement new legal domains from scratch. To this end, the Deep PROLEG system is working effectively with the performance of more than **95% accuracy** over the augmented dataset containing **5000 legal cases** within **four kinds of contracts and 20 different legal slot holders** (distribution is depicted in Figure 1).

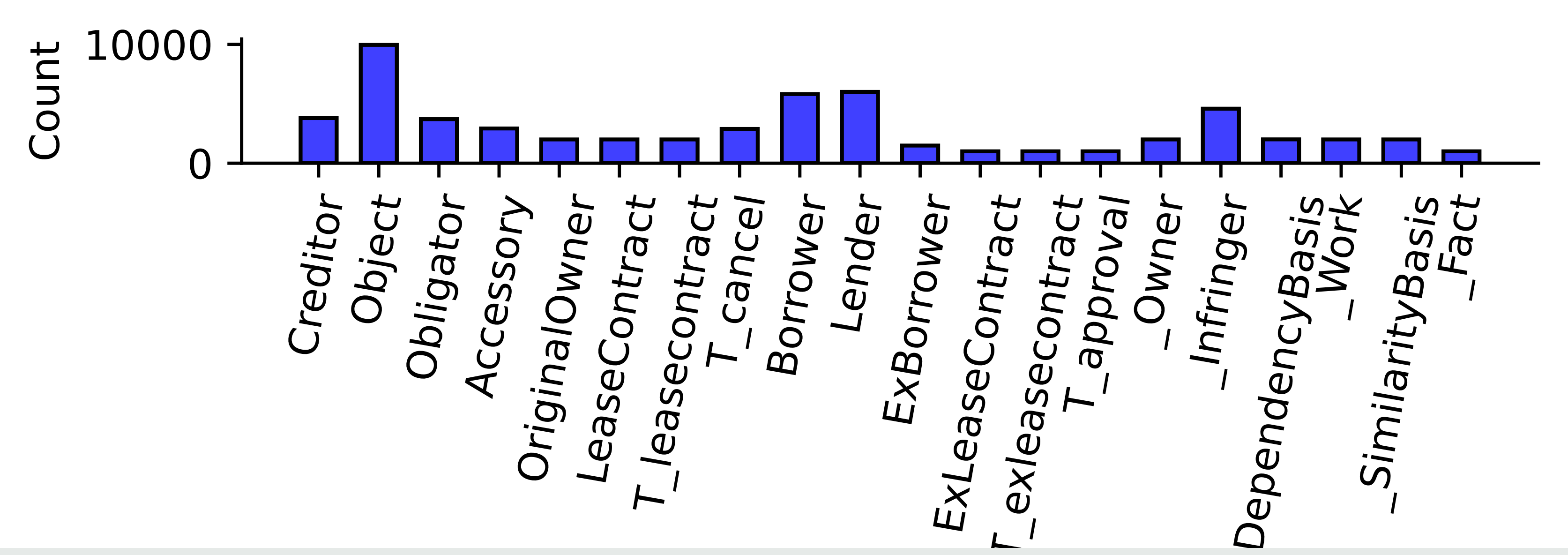


Figure 1. The distribution of legal slot holders.

2. Challenges of Scaling Up

The process of adding a new contract to the Deep PROLEG system involves two main steps (Figure 2):

- (1) manually **installing the set of PROLEG clauses** (grounding) that are implied in the contract and
- (2) **retraining a neural semantic parser model** to convert a legal case query into a set of facts that support logical inference.

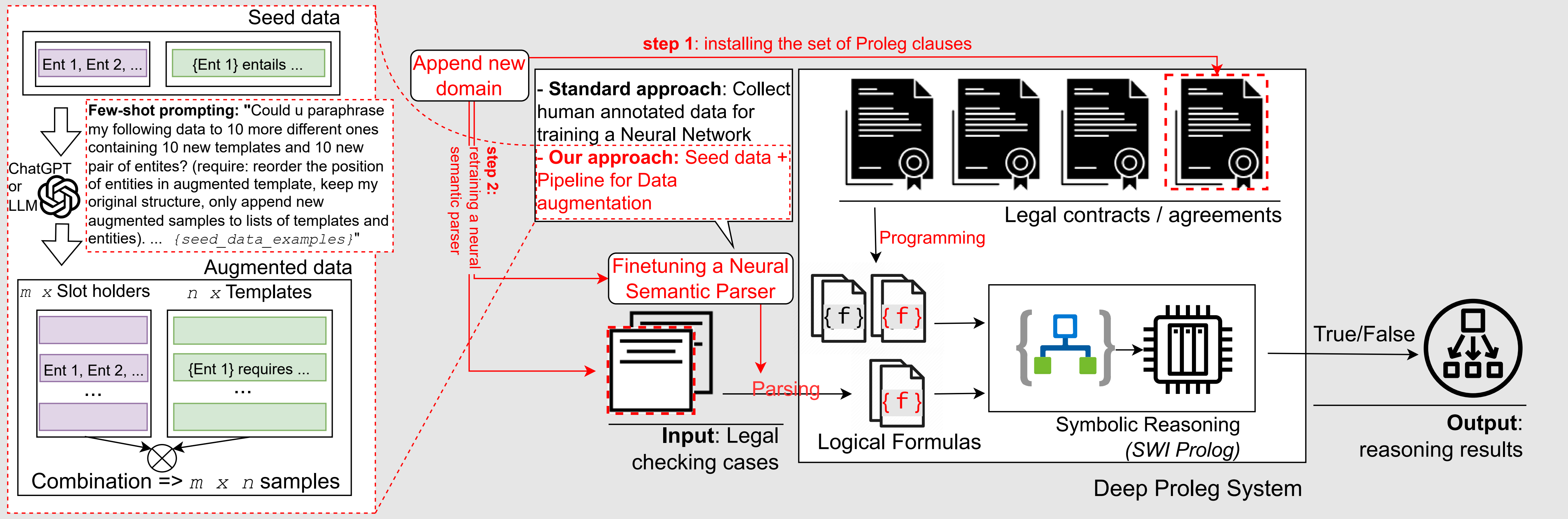
The second step demands significant effort from experts and is more time-consuming. Previous approaches using heuristic rules (Jia and Liang 2016; Wang et al. 2015) for data augmentation lack the generalization to adapt to complicated legal cases.

3. Motivation

This work presents a **pipeline for data augmentation** leveraging **few-shot prompting technique** (Brown et al. 2020) and Large Language Models (LLMs). The proposed method is both **simple** and **effective**, **significantly reducing the manual effort required for data annotation** while enhancing the robustness of Information Extraction systems. Furthermore, the method is **generalizable**, making it applicable to various NLP tasks beyond the legal domain.

4. Deep PROLEG Framework

We posit that legal cases can be deconstructed into two types of information: *templates* and *slot holders*. In essence, *each combination of a set of slot holders and a template produces a single legal case sample.* A slot holder may represent an entity name or a text span with specific significance (Table 1). We provide one or two templates and entities as seed data, which then guide LLMs to generate additional templates and sets of entities, as depicted in Figure 2.


Seed data
Ent 1, Ent 2, ...
{Ent 1} entails ...
ChatGPT or LLM
Augmented data
m x Slot holders
n x Templates
Ent 1, Ent 2, ...
...
{Ent 1} requires ...
...
Combination => m x n samples
Append new domain
step 2: retraining a neural semantic parser
step 1: installing the set of Proleg clauses
Finetuning a Neural Semantic Parser
Input: Legal checking cases
Parsing
Programming
Legal contracts / agreements
{ f }
{ f }
{ f }
Logical Formulas
Symbolic Reasoning (SWI Prolog)
Deep Proleg System
True/False
Output: reasoning results

Few-shot prompting: "Could u paraphrase my following data to 10 more different ones containing 10 new templates and 10 new pair of entites? (require: reorder the position of entities in augmented template, keep my original structure, only append new augmented samples to lists of templates and entities). ... `{seed_data_examples}`"

- **Standard approach**: Collect human annotated data for training a Neural Network

- **Our approach:** Seed data + Pipeline for Data augmentation

Figure 2. The architecture of the Deep PROLEG system. The components related to the process of adding a new domain are highlighted in red.

References

Satoh, K. (2023). "PROLEG: Practical Legal Reasoning System". In: *Prolog: The Next 50 Years*. Ed. by D. S. Warren, V. Dahl, T. Eiter, M. V. Hermenegildo, R. Kowalski, and F. Rossi. Cham: Springer Nature Switzerland, pp. 277–283. ISBN: 978-3-031-35254-6. DOI: 10.1007/978-3-031-35254-6_23. URL: https://doi.org/10.1007/978-3-031-35254-6_23.

Nguyen, H.-T., F. Nishino, M. Fujita, and K. Satoh (Dec. 2022). "An Interactive Natural Language Interface for PROLEG". In: ISBN: 9781643683645. DOI: 10.3233/FAIA220484.

Brown, T., B. Mann, N. Ryder, and et al. (2020). "Language Models are Few-Shot Learners". In: *Advances in Neural Information Processing Systems*. Vol. 33. Curran Associates, Inc., pp. 1877–1901.

Jia, R. and P. Liang (Aug. 2016). "Data Recombination for Neural Semantic Parsing". In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Berlin, Germany: ACL, pp. 12–22.

Wang, Y., J. Berant, and P. Liang (July 2015). "Building a Semantic Parser Overnight". In: *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th IJCNLP (Volume 1: Long Papers)*. Beijing, China: ACL, pp. 1332–1342. DOI: 10.3115/v1/P15-1129.