

InLegalLLaMA: Indian Legal Knowledge Enhanced Large Language Model

Sudipto Ghosh^{1,*}, Devanshu Verma¹, Balaji Ganesan², Purnima Bindal¹, Vikas Kumar¹ and Vasudha Bhatnagar¹

¹Department of Computer Science, University of Delhi, India

²IBM Research India

Abstract

Large Language Models (LLM) are being increasingly used in many domains including legal and justice. General purpose models trained on web data are not performant enough on legal text analytics (LTA) tasks while fine tuning task specific models is expensive because of the annotation and compute costs. Pre-training domain or application specific models is increasingly popular. However pre-training LLMs in small domain corpora like Indian legal documents and judgements is challenging. We introduce our InLegalLLaMA model, along with the related training corpus, adapted for the Indian legal domain, that shows promise of improved performance on LTA tasks. We also propose a RAG-based framework for petition drafting that benefits from the legal language generation and reasoning abilities of large language models.

Keywords

Large Language Models, Knowledge Enhanced Models, Legal Text Analytics

1. Introduction

Like many other fields, Large Language Models (LLMs) are increasingly being used in the domain of Law and Justice. There have been works like Chalkidis et al. [1], Paul et al. [2] that have incorporated LLM embeddings for legal tasks. While general purpose LLMs are being tried in domain and country specific applications, such web scale models where the provenance of the training data cannot be easily established, may not be best suited for the purpose and face regulatory challenges. LLMs trained specifically for the domain corpus could be both compute efficient, trustworthy, and also perform well. Joshi et al. [3] introduced a benchmark for several tasks in the Indian legal domain.

Our motivation for training a India specific legal LLM is that tasks like petition drafting, case similarity prediction, judgement summarisation require knowledge infusion at different stages. As shown in Figure 1, these tasks are currently specialized tasks performed by legal professionals and researchers, and each of them includes several sub-tasks like legal NER, question answering, text-to-SQL. Infusing knowledge into LLMs has shown promise in general purpose tasks and could be useful for legal text analytics (LTA) as well. In particular, Agarwal et al. [4], Moiseev et al. [5], Agarwal et al. [6] have shown the effectiveness of additionally pre-training LLMs with external knowledge sources like Knowledge Graphs.

In this paper, we introduce InLegalLLaMA, a large language model enhanced with knowledge of the Indian legal domain and specifically designed for Indian legal text analytics tasks. The construction and use of a legal knowledge graph is central to the effectiveness of legal LLMs. This knowledge graph should include entities and relationships extracted from legal documents like judgments and legislation. They can be represented in a structured format of triples (subject, object, predicate) or other appropriate formats. The integration of this structured knowledge into LLMs will enable them to perform more accurate and context-aware analysis of legal text.

For creating an Indian legal knowledge graph, we build upon the work of Dhani et al. [7] who construct a knowledge graph on Indian court judgements and legislation. By integrating domain-specific

The First international OpenKG Workshop: Large Knowledge-Enhanced Models, August 03, 2024, Jeju Island, South Korea

*Corresponding author.

✉ sudipto.ghosh@scai.iitd.ac.in (S. Ghosh); dverma@cs.du.ac.in (D. Verma); bganesa1@in.ibm.com (B. Ganesan); pbindal@cs.du.ac.in (P. Bindal); vikas@cs.du.ac.in (V. Kumar); vbhatnagar@cs.du.ac.in (V. Bhatnagar)



© 2024 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

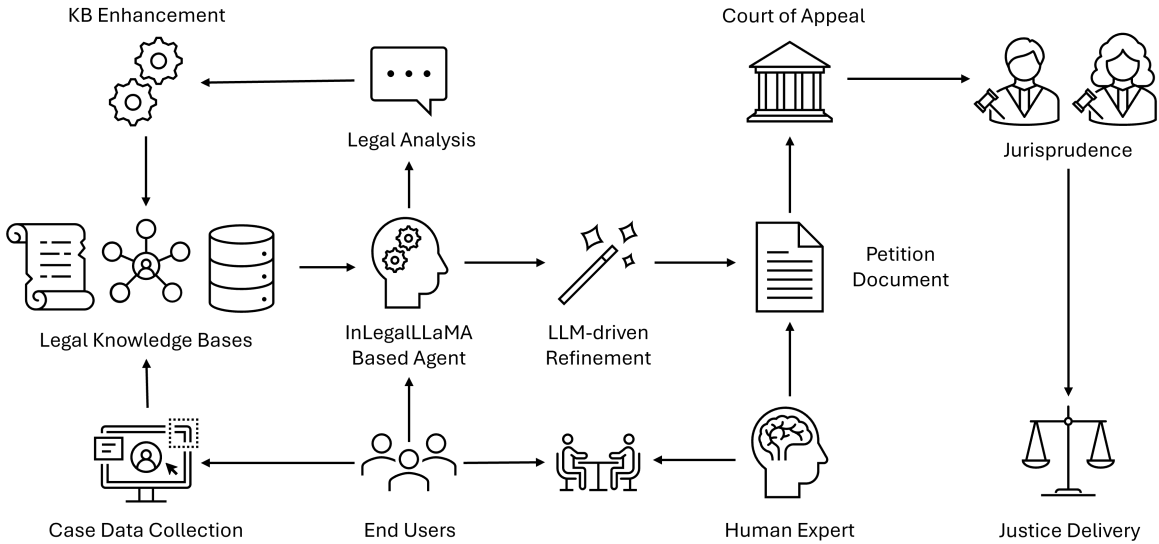


Figure 1: End-users interact with the LLM agent directly and through specialized data collection and interaction screens to exploit legal language generation and reasoning capabilities.

knowledge comprising Indian law and case documents, we expect InLegalLLaMA to be able to address several challenges inherent in legal documents, such as complex language, lengthy texts, prevalence of non-English terms, and unstructured information.

In addition to supporting legal professionals, InLegalLLaMA can also benefit law students, researchers, legislators, and citizens. Students and researchers can use the model to understand legal terms and concepts better. Legislators can use them for their discussions and improving laws. Finally even citizens not familiar with legal processes can use applications built using this model, in tasks such as drafting petitions and understanding legal notices. InLegalLLaMA aims to bridge the gap between advanced NLP techniques and the practical needs of the legal domain. By leveraging a robust legal knowledge graph, the model enhances the efficiency and accessibility of legal text analytics, contributing to more effective and timely delivery of justice.

This paper is organized as follows. In Section 2 we discuss related work, in Section 3 we introduce the InLegalLLaMA model, and in Section 4 we propose a RAG-based framework for petition drafting using InLegalLLaMA.

2. Related Work

There has been significant amount of work done in different methods to train LLMs with domain specific data. There are also few works on models trained on legal data.

Legal Knowledge Graph and datasets

Automatic Knowledge Graph Construction (AKBC) has been popular since the Knowledge Base Population (KBP) track Ji et al. [8] organized by the Text Analytics Conference (TAC) in 2010. Domain specific knowledge graphs Abu-Salih [9] still remain an ongoing research area. Dhani et al. [7], Jain et al. [10] discuss creating legal knowledge graphs using judgements and related documents from Indian courts. The role of human annotations in knowledge graph construction is also a well researched area. Chiticariu et al. [11] proposed a system to extract domain specific entities and relationships from documents. Vannur et al. [12] discussed fairness in personal knowledge base construction. We can characterize such methods as rule-based or rule-assisted knowledge base construction.

Guha et al. [13] introduced LegalBench, a benchmark for measuring legal reasoning in large language models. The Indian Legal Document Corpus published by Malik et al. [14] contains 35,000 Indian court

judgments and gold standard explanations for the Court Judgment Prediction and Explanation task. In this work, we introduce a new dataset comprising 10,000 Indian court judgments and legal statutes.

Knowledge Infusion

Chalkidis et al. [1] introduced the LegalBERT model that continues to be used for tasks on the legal data including our experiments in this work. Paul et al. [2] introduced InLegalBERT which is trained on Indian legal documents. Infusing knowledge into large language models has been discussed in several works. Two survey papers Wei et al. [15], Yang et al. [16] present different methods to infuse knowledge into large language models. Islam et al. [17] consume a knowledge graph for the entity generation task.

Agarwal et al. [4] created a method to translate knowledge graph triples into sentences for enhancing LLM pre-training. Moiseev et al. [5] and Agarwal et al. [6] then directly integrated these triples into T5 models, showing two effective paths for knowledge integration—via natural language or directly from triples. Vasisht et al. [18] took a different approach by using contextual text for embedding knowledge into models. dos Santos et al. [19] developed *Knowledge Prompts* for frequent Wikidata entities, refined to aid in triple prediction. Diao et al. [20] use adapters for efficient knowledge infusion into LLMs.

LegalBERT Chalkidis et al. [1], CaseLawBERT Zheng et al. [21] and JuriBERT Douka et al. [22] show the sustained interest of researchers in using language models for downstream legal tasks. However, these models are typically trained on European legal documents and do not perform well in the Indian context directly where there is more variability in the document structures and multi-linguality in legal documents. Under these circumstances, existing models do not work well out of the box and need additional training on local corpora.

Gururangan et al. [23] proposed extending the training phase of the language models with domain-specific datasets to realize domain adaption. Ibrahim et al. [24] empirically observe that it is possible to match performance of language models trained from scratch on a mix of original training corpora and incoming domain corpora, with a combination of novel learning rate scheduling strategies and replay of some portion of original corpora along with the training text from the target domain. Using a similar strategy, Yang et al. [25] continually pretrain a LLaMA-2 model with 10% replay of RedPajama, instruction tune it on a subset of LIMA as well as customized instructions and evaluate the model on plant science quizzes.

Legal Large Language Models

Works such as InLegalBERT and InCaseLawBERT Paul et al. [26] involve training the base models on Indian legal documents and achieve reasonable performance on certain tasks like legal statute identification, semantic segmentation and court judgment prediction. Much work still needs to be done to make LLMs useful in LTA tasks that need human expertise. We extend the pretraining phase of the LLaMA-2 foundation model Touvron et al. [27] on small-scale Indian legal domain corpora and instruction-tune it for a selected set of tasks in the Indian legal domain. We aim to use parameter-efficient fine-tuning methods for pretraining and instruction fine-tuning, and compare its performance on multiple datasets in tandem with other state-of-the-art models, with and without fine-tuning on domain corpora.

Joshi et al. [3] introduced the IL-TUR benchmark for Indian Legal Text Understanding and Reasoning. IL-TUR introduces monolingual (English, Hindi) and multilingual (9 Indian languages) legal domain-specific tasks from the point of view of understanding and reasoning over Indian legal documents. They present baseline models for each task and propose a community leader board. Joshi et al. [28] introduce the Prior Case Retrieval task to cite prior cases and proposes a solution using events extraction. Bhattacharya et al. [29], Belfathi et al. [30] explore Rhetorical Role Prediction in legal documents using transformer-based architectures.

3. InLegalLLaMA

In this section, we describe all the stages of training InLegalLLama model, and the experiments we conducted to evaluate its performance.

Knowledge Infusion

A legal knowledge graph can help students familiarize themselves with the legal terms and concepts. Such knowledge graphs can also be used to infuse knowledge into or fine tune large language models (LLMs) to fill gaps in such models where they may not have sufficient domain specific knowledge. We build on Dhani et al. [7] and Jain et al. [10] to create a legal knowledge graph by scraping the web for court cases, judgements, laws and other cases cited from the judgements etc. In particular, they use court repositories and other public sources in the Indian court system, who's provenance can be easily established. And any document that be removed from the training corpus upon request.

They retrieve legal documents from Indian court systems and use citations and similarity from Indi-anKanoon Sinha [31] and Casemine Yadav [32]. Next, they process the original documents using Stanza (Qi et al. [33]), extract entities and relations using SystemT (Chiticariu et al. [11]). They further annotate these documents using manually curated dictionaries as described in Vannur et al. [12].

Based on the above prior work, we too represent our Indian legal knowledge graph in triples format comprising of subject, object, and predicate. Representing knowledge graph triples and infusing them as a triple prediction task is quite well known. Agarwal et al. [4] generated natural language sentences from triples, and additionally pre-trained large language models with the generated sentences. Moiseev et al. [5] showed that we could directly infuse triples into large language models without having to generate sentences from the triples. Agarwal et al. [6] tried infusing triples from domain specific knowledge graphs into flan-t5 models.

Vasisht et al. [18] fine-tune LLMs on triple prediction and design prompts to probe the extent of knowledge infusion in LLMs. One of the limitations of knowledge infusion using triples, as described in Moiseev et al. [5] is the inability of the models to capture graph structure. Vasisht et al. [18] try to solve this by relying on the contextual information to help the model recollect other information associated with the triples. We mask one of the elements of the triples and pose cloze questions, while providing triples as context, to our model during the instruction-tuning phase to allow it to answer questions posed in a similar fashion.

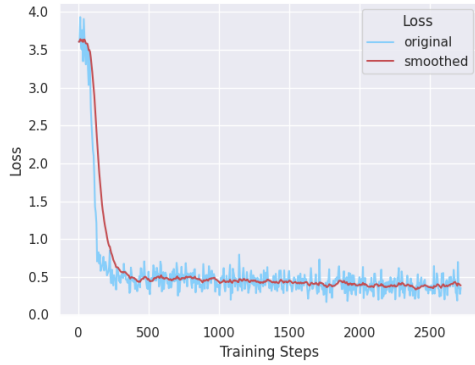
Following the idea that domain-adapted models tend to perform better on domain tasks than general language models which have never seen domain corpora Gururangan et al. [23], Ibrahim et al. [24], we extend the training phase of the LLaMA-2 foundation model on Indian legal text and instruction tune it for a couple of domain tasks. We use the training scripts from the LLaMA-Factory project Zheng et al. [34]. The base and the instruction-tuned versions of the model are publicly available on HuggingFace.^{1 2}

Dataset

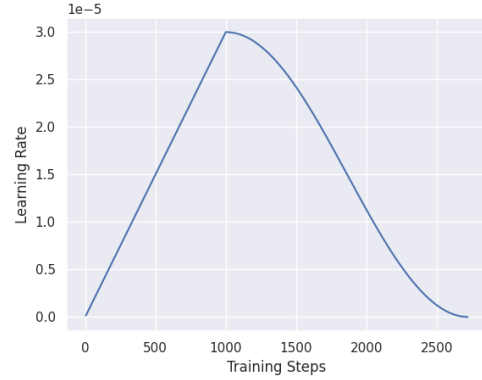
We make use of 10,000 legal documents from the Indian common law system, comprising of an equal number of (i) reportable court judgments (which are important and become binding on lower courts) and non-reportable court judgments (which are limited in their application to the specific case at hand) published by the Supreme Court of India, and (ii) legal statutes published in the Gazette of India by the Indian parliament and various state legislative institutions. We use this corpus to continue the pretraining phase of LLaMA-2. We preprocess the text using an in-house package to remove non-printable characters, stray sequences and most of the noise. For the instruction-tuning of the foundation model, we use the datasets from Vasisht et al. [18], Bhattacharya et al. [29] and Zhou et al. [36].

¹<https://huggingface.co/sudipto-ducs/InLegalLLaMA>

²<https://huggingface.co/sudipto-ducs/InLegalLLaMA-Instruct>

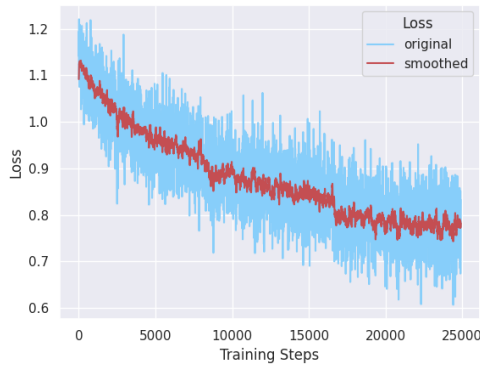


(a) Raw and exponentially smoothed training loss over steps during instruction tuning phase of InLegalLLaMA

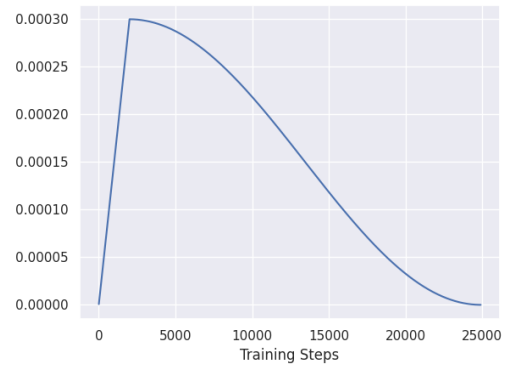


(b) Cosine LR Schedule Hoffmann et al. [35] with 1000-step linear rewarmup as used for instruction tuning

Figure 2: Training loss and learning rate schedule during instruction tuning phase of InLegalLLaMA



(a) Raw and exponentially smoothed training loss over steps during continual pretraining



(b) Cosine LR Schedule Hoffmann et al. [35] with 2000-step linear rewarmup as used for continual pretraining

Figure 3: Training loss and learning rate schedule during continual pretraining phase of InLegalLLaMA

Continual Pretraining

The foundation model is adapted to the Indian legal domain in hopes of realising benefits that domain adaptation brings. However, naively continuing to train on the new dataset from that training environment can result in poor adaptation on the incoming data and catastrophic forgetting of the capabilities and knowledge that the model holds. Instead of randomly initializing weights and training from scratch, model training is continued on the new pretraining dataset using the learning rate re-warmup and re-decay approach as suggested in the same work.

We continue training LLaMA-2 from the published model weights on the auto-regressive pretraining task with a new dataset of documents with 88,768,648 unique n -grams and 5% replay data from RedPajama TogetherAI [37] to avoid catastrophic forgetting, for around 24n900 steps. In order to do this efficiently due to resource constraints, we set a chunk size of 2,048 tokens and use LoRA ($r = 16$, $\alpha = 32$) for weight updates instead of full pretraining. We also make use of a learning rate schedule to speed up training with an initial linear re-warmup of 2,000 steps up to an η_{peak} of 3.0×10^{-4} followed by a cosine re-decay phase as recommended by Hoffmann et al. [35]. It took approximately 301 GPU hours for running the training process with 3 epochs on a single NVIDIA A6000. The training loss and learning rate over training steps are plotted in Figure 3.

Instruction Tuning

In order to enable the model to respond to specific instructions and queries, we use examples for in-context masked triple prediction (Vasishth et al. [18]) and legal sentence rhetorical role classification (Bhattacharya et al. [29]) tasks posed as instructions to the model. We also use LIMA instructions Zhou et al. [36] to prevent catastrophic forgetting. We perform supervised LoRA instruction-tuning with default LLaMA-2 prompt template, with η_{peak} of 3.0×10^{-5} and a 1000-step warmup. Instruction tuning took about six hours on the same node. The instruction tuning loss and LR schedule are depicted in Figure 2.

Evaluation

We evaluate the LLaMA-2 foundation models and InLegalLLaMA on in-context masked triple prediction Vasishth et al. [18] and legal sentence rhetorical role classification Bhattacharya et al. [29] tasks. We take held-out sets of task instances and report task metrics in Tables 1 and 2. We set out to observe whether we are able to match the performance of the off-the-shelf LLaMA-2 model on these Indian legal domain tasks, and do not seek to establish state-of-the-art in these tasks. We note that InLegalLLaMA performs better than LLaMA-2 in the domain tasks we consider. The promising results show that InLegalLLaMA may also be suited to other tasks in the Indian legal context and perform at par with the baselines.

Model	Hits@1	BLEU	ROUGE
Flan-T5	0.914	-	-
LLaMA-2-7B	0.925	94.951	94.927
InLegalLLaMA	0.984	98.224	99.191

Table 1

Comparative performance of InLegalLLaMA for in-context triple prediction task in terms of Hits@1, BLEU, ROUGE-L metrics

Model	P	R	F1
Hier-BiLSTM-CRF	0.652	0.552	0.578
BERT-BiLSTM-CRF	0.688	0.615	0.635
LLaMA-2-7B	0.620	0.553	0.571
InLegalLLaMA	0.669	0.573	0.585

Table 2

Comparative performance of InLegalLLaMA for legal sentence rhetorical role prediction task in terms of Precision (P), Recall (R) and F1-Score (F1) metrics

In Table 1, we can see the performance of our InLegalLLaMA model compares well against the baseline LLaMA-2-7B model as well as the performance of Flan-T5 model. However, the high performance of all the models indicates, the triple prediction task on this dataset may not be that hard. We plan to increase the size of the knowledge graph from which these triples are drawn.

On the other hand, rhetorical role prediction is a harder problem which has important applications in legal text analytics. Here our evaluation of the InLegalLLaMA model shows more promising results as shown in Table 2. Here our model is performing as well as supervised models for this task.

To extend this model to the more complex legal text analytics tasks described in the next section, we believe more extensive instruction tuning is required. We leave this as future work. Further, a code fine-tuned version of InLegalLLaMA will likely be needed for tasks like Text-to-SQL.

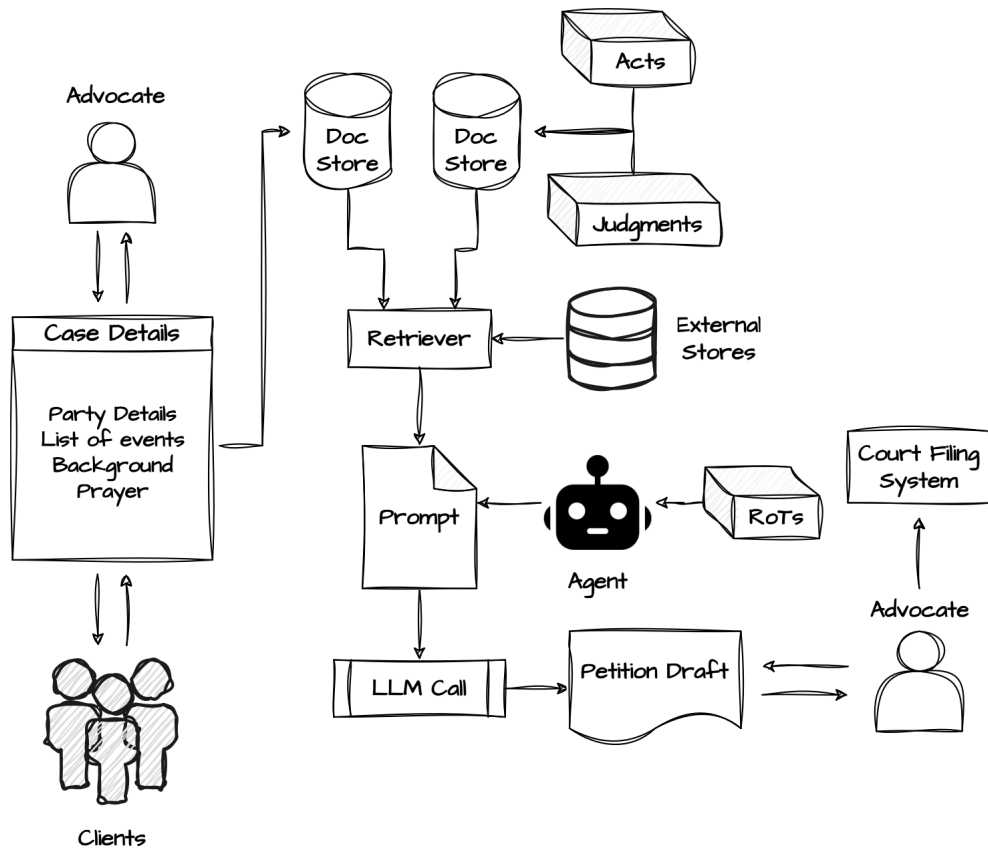


Figure 4: Retrieval Augmented Generation-based Framework for Petition Drafting

4. Petition Drafting

Court petitions are formal written requests submitted to a court, seeking a specific judicial action or ruling. Ghosh et al. [38] includes samples of petitions filed in Indian courts. They serve as a primary method for individuals or entities to initiate various legal proceedings, seek specific court orders, or appeal against decisions of lower courts or authorities. While individuals can file petitions themselves, it is common to engage lawyers due to the complexity of legal procedures.

Petition drafting is a task, that is inherently human centered, especially in the context of Indian court system. Indian courts have the concept of Public Interest Litigations (PILs), using which, any citizen can approach a court of law to seek relief on issues concerning the people. There are, of course, much larger number of people approaching the courts seeking redressal of their grievances.

Enabling people or their lawyers to write well-written petitions can go a long way in getting them access to justice. Given the backlog and the volume of petitions disposed by courts in India and in many other countries, poorly written petitions can add significant cost to both individuals and the society as a whole. Among other things, poorly written petitions could be those that leave out important pieces of information, addressed to the wrong courts or authorities, risk being dismissed as frivolous when in fact they are not.

We propose using LLMs to identify missing information in a petition. This is a qualitatively much harder task than writing a document which focuses on the writing style and presentation. Our task involves making LLMs to identify missing information that should typically be present in the petition. This can be designed as a conversational question answering task. This is closely related to the factuality related work in LLMs, since we do not want the model to ask trivial questions. The model needs to be able to identify salient information in a petition, and prompt the user to furnish any missing information.

For example, in a petition about a missing person which is a very sensitive but important judicial function, the petition is expected to provide the time when the person was last seen by a member of the public or a CCTV camera. While we expect this to be a multi-turn conversation, we currently focus on creating a question answering dataset and evaluating our LLaMA-2 model on the question answering task.

A typical Indian court petition includes (i) petitioner’s and respondent’s names and addresses, (ii) a detailed statement of facts and events leading to the petition, (iii) legal grounds and relevant laws supporting the petition, and (iv) specific relief or action sought from the court.

We propose a RAG solution for the task in Figure 4 which relies on the inputs of a trained advocate for generating the draft and employs a human-in-the-loop approach. It consists of four key stages:

1. **Template Selection:** Based on the case details and the court to which the appeal is to be made, a set of candidate templates is retrieved from a template store. It is essential to file petitions in the appropriate court with jurisdiction over the matter. The advocate selects the most appropriate template from the recommendations. This template acts as a structured outline for the petition and is used in the next stages.
2. **Content Generation:** The content generation phase is modelled as a parallel multi-hop question answering task Mavi et al. [39] over case details. Each section of the petition has to be written from a certain perspective. An LLM agent can use RAG, external tools and human input to acquire the required context for each section and generate the content by following certain rules of thumb (RoTs). The role of these RoTs are to ensure that the model only uses the relevant details to generate the content, and to steer the tone and depth of detail of the generations.
3. **Refinement and Integration:** Each section is to be refined to enhance readability, eliminate redundancies, ensure coherence and proper narrative. A human expert intervenes by accepting or rejecting these refinements. These sections are then merged into a cohesive document that follows the outline.
4. **Draft Evaluation:** The final phase involves a evaluation and iteration process where the petition draft is assessed through an elaborate evaluation strategy. If required, sections can be regenerated on demand. LLMs can be used to judge the quality in tandem with human expert(s) and the feedback can be used as align model generations using reinforcement learning Zheng et al. [40].

From template selection to an exhaustive evaluation, the framework deals with the intricacies of the petition drafting task. Human experts must monitor such a system to ensure that the generated petition draft is admissible in the court, complete, legally sound and does justice to the case at hand.

Conclusion

Our observations of the existing works and applications in Indian legal text analytics make it amply clear that we need to develop country and domain specific Large Language Models (LLMs) enhanced with knowledge graphs. In this work, we have introduced InLegalLLaMA, a set of LLMs enhanced with knowledge from an Indian legal knowledge graph. We show the performance of our model on tasks like tail prediction and rhetorical role prediction. We then discuss how our model is useful in more complex legal text analytics tasks like petition drafting, case similarity, judgement summarisation and legal question answering. We plan to work on code variants of InLegalLLaMA in the future that will help in Retrieval Augmented Generation (RAG) applications.

References

- [1] I. Chalkidis, M. Fergadiotis, P. Malakasiotis, N. Aletras, I. Androutsopoulos, LEGAL-BERT: The muppets straight out of law school, in: Findings of the Association for Computational Linguistics: EMNLP 2020, Association for Computational Linguistics, Online, 2020, pp. 2898–2904.
- [2] S. Paul, A. Mandal, P. Goyal, S. Ghosh, Pre-training transformers on indian legal text, 2022. doi:10.48550/ARXIV.2209.06049.
- [3] A. Joshi, S. Paul, A. Sharma, P. Goyal, S. Ghosh, A. Modi, Il-tur: Benchmark for indian legal text understanding and reasoning, in: Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2024, pp. 11460–11499.
- [4] O. Agarwal, H. Ge, S. Shakeri, R. Al-Rfou, Knowledge graph based synthetic corpus generation for knowledge-enhanced language model pre-training, in: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, Online, 2021, pp. 3554–3565.
- [5] F. Moiseev, Z. Dong, E. Alfonseca, M. Jaggi, SKILL: Structured knowledge infusion for large language models, in: Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, 2022, pp. 1581–1588.
- [6] A. Agarwal, S. Gawade, S. Channabasavarajendra, P. Bhattacharya, There is no big brother or small brother: knowledge infusion in language models for link prediction and question answering, in: Proceedings of the 19th International Conference on Natural Language Processing (ICON), Association for Computational Linguistics, New Delhi, India, 2022, pp. 204–211.
- [7] J. S. Dhani, R. Bhatt, B. Ganesan, P. Sirohi, V. Bhatnagar, Similar cases recommendation using legal knowledge graphs, 2024. 3rd Symposium on Artificial Intelligence and Law (SAIL).
- [8] H. Ji, R. Grishman, H. T. Dang, K. Griffitt, J. Ellis, Overview of the tac 2010 knowledge base population track, in: Third text analysis conference (TAC 2010), volume 3, 2010, pp. 3–3.
- [9] B. Abu-Salih, Domain-specific knowledge graphs: A survey, *Journal of Network and Computer Applications* 185 (2021) 103076.
- [10] S. Jain, P. Harde, N. Mihindukulasooriya, S. Gosh, A. Bisht, A. Dubey, Constructing a knowledge graph from indian legal domain corpus., in: TEXT2KG/MK@ ESWC, 2022, pp. 80–93.
- [11] L. Chiticariu, R. Krishnamurthy, Y. Li, S. Raghavan, F. Reiss, S. Vaithyanathan, SystemT: An algebraic approach to declarative information extraction, in: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, 2010, pp. 128–137.
- [12] L. S. Vannur, B. Ganesan, L. Nagalapatti, H. Patel, M. Tippeswamy, Data augmentation for fairness in personal knowledge base population, in: Trends and Applications in Knowledge Discovery and Data Mining: PAKDD 2021 Workshops, 2021 Proceedings 25, 2021, pp. 143–152.
- [13] N. Guha, J. Nyarko, D. Ho, C. Ré, A. Chilton, A. Chohlas-Wood, A. Peters, B. Waldon, D. Rockmore, D. Zambrano, et al., Legalbench: A collaboratively built benchmark for measuring legal reasoning in large language models, *Advances in Neural Information Processing Systems* 36 (2024).
- [14] V. Malik, R. Sanjay, S. K. Nigam, K. Ghosh, S. K. Guha, A. Bhattacharya, A. Modi, Ildc for cjpe: Indian legal documents corpus for court judgment prediction and explanation, in: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), 2021, pp. 4046–4062.
- [15] X. Wei, S. Wang, D. Zhang, P. Bhatia, A. Arnold, Knowledge enhanced pretrained language models: A comprehensive survey, *arXiv preprint arXiv:2110.08455* (2021).
- [16] J. Yang, G. Xiao, Y. Shen, W. Jiang, X. Hu, Y. Zhang, J. Peng, A survey of knowledge enhanced pre-trained models, *arXiv preprint arXiv:2110.00269* (2021).
- [17] S. M. Islam, A. Nagpal, B. Ganesan, P. K. Lohia, Fair data generation using language models with hard constraints, *CtrlGen Workshop* (2021).
- [18] K. Vasisht, B. Ganesan, V. Kumar, V. Bhatnagar, Infusing knowledge into large language models with contextual prompts, 2023. 20th International Conference on Natural Language Processing.
- [19] C. N. dos Santos, Z. Dong, D. Cer, J. Nham, S. Shakeri, J. Ni, Y. hsuan Sung, Knowledge prompts: In-

- jecting world knowledge into language models through soft prompts, 2022. [arXiv:2210.04726](https://arxiv.org/abs/2210.04726).
- [20] S. Diao, T. Xu, R. Xu, J. Wang, T. Zhang, Mixture-of-domain-adapters: Decoupling and injecting domain knowledge to pre-trained language models memories, [arXiv preprint arXiv:2306.05406](https://arxiv.org/abs/2306.05406) (2023).
- [21] L. Zheng, N. Guha, B. R. Anderson, P. Henderson, D. E. Ho, When does pretraining help? assessing self-supervised learning for law and the casehold dataset of 53,000+ legal holdings, in: Proceedings of the eighteenth international conference on artificial intelligence and law, 2021, pp. 159–168.
- [22] S. Douka, H. Abdine, M. Vazirgiannis, R. El Hamdani, D. Restrepo Amariles, JuriBERT: A masked-language model adaptation for French legal text, in: Proceedings of the Natural Language Processing Workshop 2021, 2021, pp. 95–101.
- [23] S. Gururangan, A. Marasović, S. Swayamdipta, K. Lo, I. Beltagy, D. Downey, N. A. Smith, Don't stop pretraining: Adapt language models to domains and tasks, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, 2020, pp. 8342–8360.
- [24] A. Ibrahim, B. Thérien, K. Gupta, M. L. Richter, Q. Anthony, T. Lesort, E. Belilovsky, I. Rish, Simple and scalable strategies to continually pre-train large language models, [arXiv preprint arXiv:2403.08763](https://arxiv.org/abs/2403.08763) (2024).
- [25] X. Yang, J. Gao, W. Xue, E. Alexandersson, Pllama: An open-source large language model for plant science, [arXiv preprint arXiv:2401.01600](https://arxiv.org/abs/2401.01600) (2024).
- [26] S. Paul, A. Mandal, P. Goyal, S. Ghosh, Pre-trained language models for the legal domain: A case study on indian law, in: Proceedings of the Nineteenth International Conference on Artificial Intelligence and Law, ICAIL '23, Association for Computing Machinery, 2023, p. 187–196.
- [27] H. Touvron, L. Martin, K. R. Stone, P. Albert, A. Almahairi, et al., Llama 2: Open Foundation and Fine-Tuned Chat Models, [arXiv preprint arXiv:2307.09288](https://arxiv.org/abs/2307.09288) (2023).
- [28] A. Joshi, A. Sharma, S. K. Tanikella, A. Modi, U-creat: Unsupervised case retrieval using events extraction, in: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2023, pp. 13899–13915.
- [29] P. Bhattacharya, S. Paul, K. Ghosh, S. Ghosh, A. Wyner, DeepRhole: Deep Learning for Rhetorical Role Labeling of Sentences in Legal Case Documents, *Artificial Intelligence and Law* (2023).
- [30] A. Belfathi, N. Hernandez, L. Monceaux, Harnessing gpt-3.5-turbo for rhetorical role prediction in legal cases, in: Legal Knowledge and Information Systems - JURIX 2023, volume 379 of *Frontiers in Artificial Intelligence and Applications*, IOS Press, 2023, pp. 187–196.
- [31] S. Sinha, IndianKanoon: Search Engine for Indian Law, <https://indiankanoon.org/>, 2008.
- [32] A. Yadav, Casemine: A granular mapping of indian case law, <https://www.casemine.com/>, 2013.
- [33] P. Qi, Y. Zhang, Y. Zhang, J. Bolton, C. D. Manning, Stanza: A python natural language processing toolkit for many human languages., In Association for Computational Linguistics (ACL) System Demonstrations. 2020., 2020.
- [34] Y. Zheng, R. Zhang, J. Zhang, Y. Ye, Z. Luo, Y. Ma, LlamaFactory: Unified Efficient Fine-Tuning of 100+ Language Models, [arXiv preprint arXiv:2403.13372](https://arxiv.org/abs/2403.13372) (2024).
- [35] J. Hoffmann, S. Borgeaud, A. Mensch, E. Buchatskaya, et al., Training Compute-Optimal Large Language Models, in: Advances in Neural Information Processing Systems, volume 35, Curran Associates, Inc., 2022, pp. 30016–30030.
- [36] C. Zhou, P. Liu, P. Xu, S. Iyer, J. Sun, Y. Mao, X. Ma, et al., LIMA: Less Is More for Alignment, [arXiv preprint arXiv:2305.11206](https://arxiv.org/abs/2305.11206) (2023).
- [37] TogetherAI, RedPajama: an Open Dataset for Training Large Language Models, 2023.
- [38] S. Ghosh, D. Verma, B. Ganesan, P. Bindal, V. Kumar, V. Bhatnagar, Human centered ai for indian legal text analytics, 2024. [arXiv:2403.10944](https://arxiv.org/abs/2403.10944).
- [39] V. Mavi, A. Jangra, A. Jatowt, Multi-hop Question Answering, [arXiv preprint arXiv:2204.09140](https://arxiv.org/abs/2204.09140) (2024).
- [40] L. Zheng, W.-L. Chiang, Y. Sheng, S. Zhuang, Z. Wu, Y. Zhuang, Z. Lin, Z. Li, D. Li, E. P. Xing, et al., Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena, [arXiv preprint arXiv:2306.05685](https://arxiv.org/abs/2306.05685) (2023).