NitiGraph: Open Thai Legal Knowledge Graph

Naphaphol Sawatvong[1], Patipon Wiangnak[2], Thiti Phutaarmart[2] and Natthawut Kertkeidkachorn[2]

[1]Chiang Mai University, Thailand
[2]Japan Advanced Institute of Science and Technology, Japan

Abstract

Access to reliable legal information underpins access to justice, compliance, and the systems built on both. Thai law resists it. Legislation is published as consolidated documents whose amendment history sits in footnotes rather than in queryable form, and nothing in a court opinion states reliably which provision the judgment turned on. The substantial Thai legal knowledge graphs are production platforms that do not release their graphs. We present NitiGraph, a knowledge graph of the four codes at the centre of Thai legal practice, 3,138 provisions in 4,853 dated versions together with the 44,217 Supreme Court decisions that cite them and the 115,240 citation edges between them. Its ontology ties each amendment to an event naming the instrument that authorised it rather than to a difference between two texts, and every version traces back to the published document behind it. The two layers are kept disjoint and meet only through a shared, reified citation, so the decisions that apply a provision are one hop away in either direction. Construction is deterministic, converting the amendment record the sources already state, and every assertion carries the source file, byte range and checksum behind it. A retrieval check on 150 bar-examination sub-questions finds that expanding the retrieved provisions through the graph raises recall at 20 from 0.371 to 0.474 over text retrieval alone.

Keywords

Thai law, Legal knowledge graph, Court decisions, Ontology, Legal information retrieval

1. Introduction

Answering a question of Thai law requires both the text of a provision as it stood when the facts arose and the decisions that have applied it. Neither is straightforward to obtain. Thai codes are amended by separate instruments over decades, so a provision is not one text but a chain of dated texts, and the consolidated editions published by the Office of the Council of State (OCS) record that history in footnotes rather than in a queryable form. Decisions are published as free text, and nothing in an opinion states reliably which provision the judgment turns on, since a number in the reasoning may be one the court applied, one it rejected, or one quoted from a decision under discussion.

Thai legal knowledge graphs already exist, but the substantial ones are production systems whose graph and code are not released [1, 2], so the whole graph cannot be inspected and its construction cannot be reproduced. A second gap is in the model itself. Neither published account describes the amending instrument or the gazette record behind a change as entities in the graph, so a version cannot be traced back to the act that produced it.

This paper offers a resource with formal semantics that names the amending instrument behind every version and carries per-assertion provenance. A short retrieval check reports what the graph adds over a text corpus, and the full evaluation is reported separately.

Our contributions are summarised as follows:

1. An ontology for Thai law that separates a provision from its dated versions, ties each version to the instrument that introduced it and each amendment to the event that authorised it, materialises the sub-provision units Thai judgments pinpoint, and joins legislation to case law through a single shared, reified citation.

IJCKG 2026 Posters, Demos and Workshops, November 19–21, 2026, Bangkok, Thailand
n.sawatvong@gmail.com (N. Sawatvong); w.patipon@jaist.ac.jp (P. Wiangnak); p.thiti@jaist.ac.jp (T. Phutaarmart); natt@jaist.ac.jp (N. Kertkeidkachorn)
© 2026 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073
published 2026-11-06

CEUR-WS.org/Vol-4316/pd-short-10.pdf

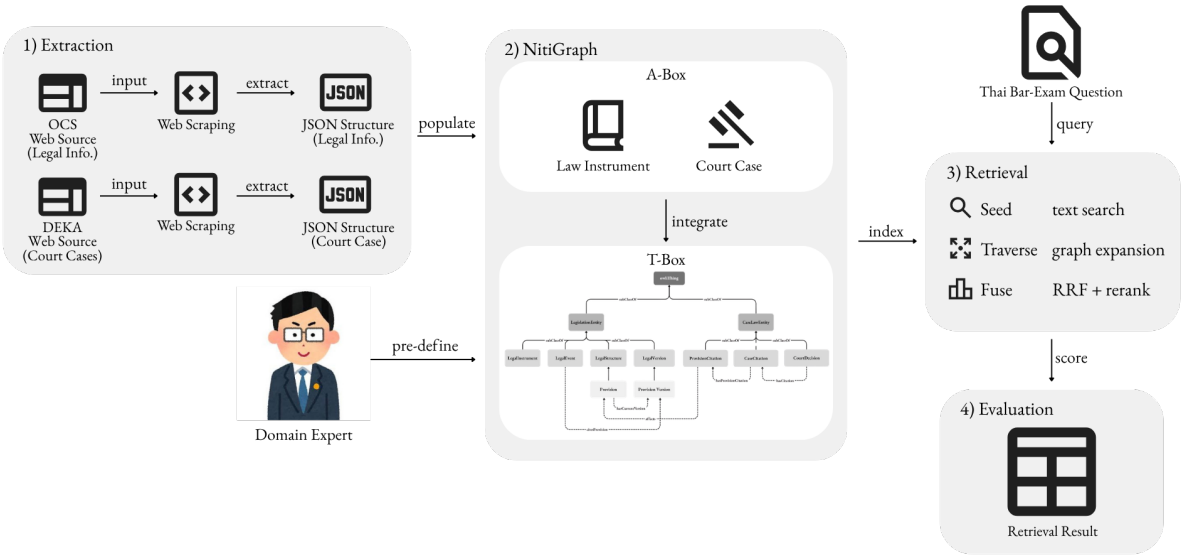


Figure 1: An overview of construction and retrieval pipeline.

2. A populated graph covering the four core Thai codes with their amendment history, the 44,217 Supreme Court decisions that cite them and the 115,240 citation edges that join the two, built by a deterministic pipeline rather than inferred from text.

2. Related Work

Legal knowledge graphs and provision-level versioning. Legal knowledge graphs publish legislation, decisions and the relations between them as linked, queryable data. The pattern is established outside Thailand. The Austrian legal knowledge graph links legislation to case law through the ELI and ECLI identifier standards [3], and Semantic Finlex publishes Finnish legislation and case law as linked open data [4], extended in later work to carry statute history [5]. Shared vocabularies such as LKIF-Core supply reusable classes for common legal concepts [6], and Akoma Ntoso's change-management model represents an amendment as a typed event acting on a target rather than as a difference observed between two texts [7]. NitiGraph claims none of this as new, since it follows the Akoma Ntoso design and specialises it into ten typed subproperties of `affects`. Vuong et al. [8] build a heterogeneous graph connecting Vietnamese legal cases to the laws they involve, without the dated versions and amendment provenance NitiGraph carries.

Thai legal knowledge graphs. For Thai law, NitiBench [9] supplies the reference retrieval benchmark, whose CCL test set holds 3,729 single-label queries over 21 pieces of legislation and whose tax set holds 50 rulings with a mean of 2.62 gold provisions each. The closest resource to ours is FourCorners, described in two papers by the same group, one presenting the knowledge graph [1] and the other the platform deployed on it [2]. It covers more laws and more decisions than NitiGraph, 4,118 laws in 6,561 tracked versions with 87,394 Supreme Court decisions [1], and its authors claim the first unified temporal knowledge graph for Thai legal data, which we do not contest. Two differences remain, and they are what this paper addresses. First, that account describes temporal versioning and citations targeting timeless nodes [1], but not the amending instrument or the gazette record as entities, and the platform resolves the version behind a court citation from the decision's publication year [2]. NitiGraph instead

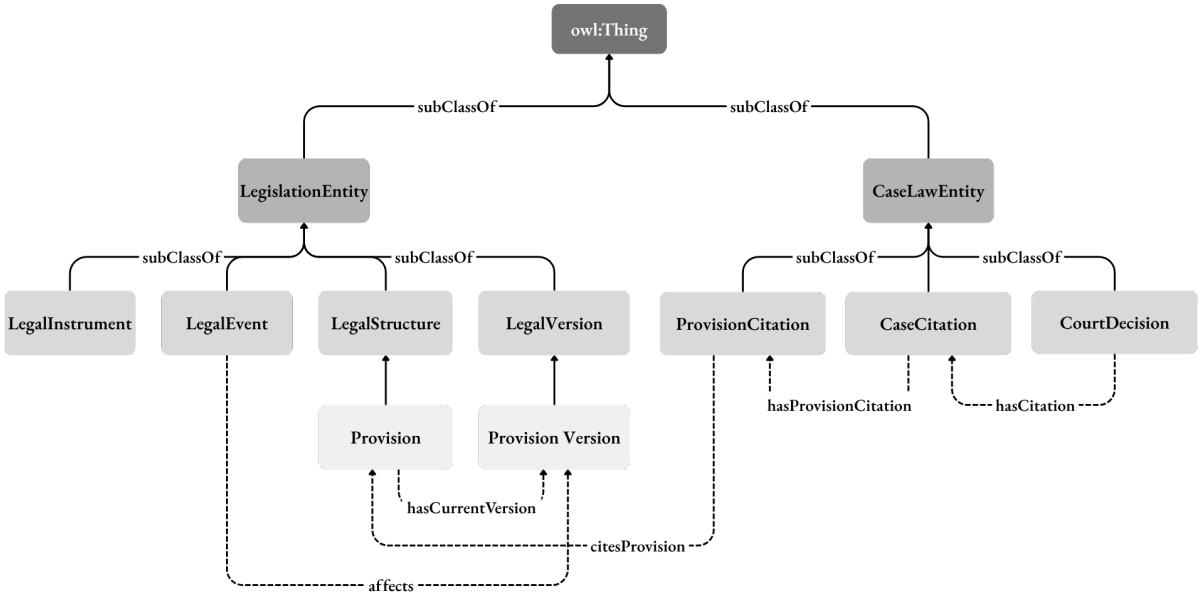


Figure 2: An overview of the main concepts of the NitiGraph ontology, an excerpt in which intermediate classes are elided. The two layers are disjoint and meet only through the citation classes.

asserts each version from an event naming the instrument that authorised it. Second, the knowledge graph and the application code are not open-sourced [2], which limits independent inspection and reproduction. Both differences complement that work rather than compete with it.

3. Open Thai Legal Knowledge Graph

NitiGraph couples legislation, modelled with its amendment history, to the judgments that apply it. It covers the four codes at the centre of Thai legal practice, namely the Civil and Commercial Code (CCC), the Civil Procedure Code, the Criminal Procedure Code and the Penal Code, as 3,138 provisions in 4,853 dated versions together with the 44,217 Supreme Court decisions that cite them. These four are the codes whose amendment history the Office of the Council of State publishes in a form a pipeline can convert. Extending the same model to the rest of Thai legislation is under way and is reported separately.

3.1. Ontology Design

The ontology has 132 classes, 96 object and 104 datatype properties, four disjointness blocks, two property chains, two transitive and 28 functional properties, every term labelled in Thai and English.[1] Its top level is two disjoint buckets, `LegislationEntity` and `CaseLawEntity`, with only `LegalAuthority`, `LegalDomain` and `LegalSource` outside both.

Temporal model. A provision is separated from its time-bounded `ProvisionVersion` instances. The functional `hasCurrentVersion` identifies the one in force and `nextVersion` chains the rest, so a point-in-time query resolves to exactly one text per provision per date. The paragraph and subsection nodes that judgments pinpoint hang off the version whose text they segment. Amendment is asserted rather than inferred. A `LegalEvent` links by `affects` and ten specialised subproperties to everything it acts on, and by `causedByInstrument` to the instrument that authorises it. A new version therefore

[1]The Thai section, *matra*, is the class `Article` $\sqsubseteq$ `Provision`, labelled "Section" in English; we write *provision* throughout.

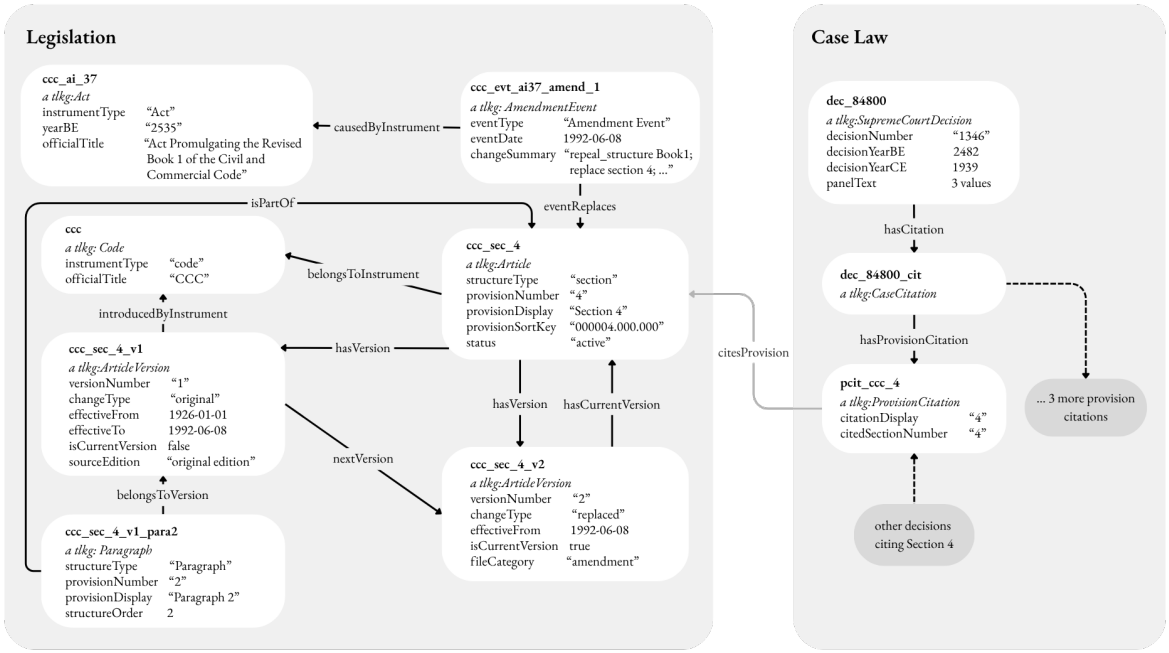


Figure 3: An example of legislation and case law individuals.

requires an authoritative legal change with a citable instrument behind it, never a textual difference observed between two consolidated documents.

The citation hinge. The layers meet in two hops, from `CourtDecision` to `CaseCitation` to `ProvisionCitation` to `Provision`, composed into a shortcut by two property chains. A `ProvisionCitation` is one node per cited provision and pinpoint and is `shared`, so every decision applying CCC section 420 points at the same node and the inverse question is one hop rather than a scan. Citations resolve to the provision, never to a version, because the corpus does not record which text was in force and the decision date does not recover it, since Thai reporters cite the provision as it stood when the cause of action arose, which may precede the judgment by years. No citation is therefore pinned to a version by inference.

3.2. Graph Construction

For each code the base edition and every amendment edition published by the Office of the Council of State[2] were extracted into a per-instrument JSON export carrying the structural tree, every provision with its ordered version chain, the amendment events with their targets, and the Royal Gazette footnotes. RDF conversion is deterministic, with no text-level guessing, because those chains, dates and event links are explicit in the source. Every version retains its source edition, gazette citation and an `evidenceId` giving source file, byte range and checksum, so any assertion traces back to the document behind it.

The case layer takes the 44,217 decisions, from a compilation of 72,988, that cite at least one of the four codes. Their curated metadata names the provisions applied, which is what the citation edges materialise, and 4,800 of the 4,936 citation nodes, a share of 0.972, resolve to a provision in the graph. Table 1 summarises the result.

3.3. Application

Retrieval runs in three stages. 1) Seed. BM25 over PyThaiNLP-segmented provision text and BGE-M3 dense retrieval [10], fused by reciprocal rank fusion [11]. This first stage is deliberately the one a

[2] https://searchlaw.ocs.go.th, retrieved 5 August 2026.

Table 1: Size and coverage of the knowledge graph.

Legislation		*Case law*	
Instruments	150	Supreme Court decisions	44,217
Provisions	3,138	Judicial proceedings	12,941
Provision versions	4,853	Litigant participations	82,888
with > 1 version	1,146	Shared provision citations	4,936
Sub-provision units	11,591	resolving to a provision	4,800
Structural nodes	408	Citation edges	115,240
Legal events	496	distinct provisions reached	2,440
Gazette references	3,118	Mean citations per decision	2.61

Table 2: Comparison of provision retrieval results.

Condition	R@20	Hit@20	MRR@20
Closed book, qwen3:14b	0.083	—	—
Text retrieval	0.371	0.673	**0.352**
Text retrieval with graph expansion	**0.474**	**0.760**	0.334

text-only baseline runs, so that anything gained later comes from the graph. 2) Traverse. Three channels expand the seeds, to the decisions citing them, to the other provisions those decisions cite, and to the seeds' structural siblings. 3) Fuse. The channels are combined by rank rather than by scores that share no scale, and decisions, but not provisions, are reranked by a cross-encoder.

4. Experiment

A legal knowledge graph is worth building only if it retrieves law that a text corpus does not. We check that on retrieval alone, with no generator in the loop, and report the full evaluation separately.

The benchmark is Thai bar-examination and judicial-entrance papers with official model answers, normalised into 1,209 scoreable sub-questions, of which a stratified random sample of 150 is scored. Those items carry 420 gold provision citations, a median of two per item.

Two indexes hold the same law. The text index contains 3,093 in-force provisions and 40,904 decisions with BM25 over PyThaiNLP-segmented text[3]. The graph index projects the graph into 3,073 provisions with text, 44,217 decisions and 115,321 citation edges. Both reach 419 of the 420 gold provisions, so a difference in recall is a difference in structure and not in inventory. All intervals are cluster bootstrap over items with 20,000 resamples, never over citations, which are nested within items.

Table 2 reports recall, hit rate and mean reciprocal rank at 20, with recall pooled over all 420 gold citations rather than averaged per question. As a floor, a closed-book model asked to name the provisions it would apply, with no documents, reaches 0.083, and because it commits to a set rather than to a ranking, no rank-based measure applies to it. Text retrieval reaches 0.371. Expanding the retrieved provisions through the graph, to the decisions that cite them, to the other provisions those decisions cite, and to their structural siblings, reaches 0.474, an improvement of 0.102 with a cluster bootstrap interval from 0.052 to 0.152, and it raises hit rate from 0.673 to 0.760. Mean reciprocal rank falls slightly, from 0.352 to 0.334, so the traversal adds provisions the text search missed rather than reordering the ones it already found.

5. Conclusion and Future Work

NitiGraph holds the four core Thai codes with their amendment history and the 44,217 Supreme Court decisions that apply them, joined by a shared, reified citation, under an ontology that asserts each

[3] https://pythainlp.org

version from the source's own record of the change rather than inferring it from text. In a first retrieval check, expanding the retrieved provisions through the graph raises recall by 0.102 over text retrieval alone.

Three limits belong to the sources. The decision corpus effectively ends at B.E. 2557 (2014), so recent case law is absent, and it records no decision-to-decision citations, so `citesDecision` is defined but empty. The consolidated editions the OCS publishes for most instruments carry only the current snapshot, so the amendment history in the graph is the four codes' own. Questions under other statutes or on post-2014 case law therefore fall outside the resource. One limit belongs to the work, namely that retrieval was measured without a generator in the loop.

Three of these are what we intend to address next. The amendment history of the remaining instruments can be recovered from the footnote strings the editions do carry, which name the amending act in prose rather than as a link, and turned into the same typed events. Recent decisions can be added by the same converter without changing the model. The third needs a model behind the retrieval, graded against a per-item checklist derived from the official model answers, so that the question becomes whether the right provisions entered the reasoning. The ontology, the graph and the construction pipeline will be made publicly available.

Declaration on Generative AI

During the preparation of this work, the authors used the AI assistant in OpenAI Prism and Claude (Anthropic) for assistance with drafting, rewording and grammar correction. The authors reviewed and revised the resulting content and take full responsibility for the final content of the paper.

References

[1] P. Akarajaradwong, S. Nutanong, C. Chaksangchaichot, FourCorners: A production knowledge graph unifying Thailand's legal system, in: Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Industry Track), 2026, pp. 1808–1818.

[2] P. Akarajaradwong, T. Nopparatbundit, T. Saeteng, K. Phoncharoen, S. Nutanong, C. Chaksangchaichot, FourCorners: Grounded Thai legal research over a temporal knowledge graph, in: Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (System Demonstrations), 2026, pp. 463–472. URL: https://aclanthology.org/2026.acl-demo.46/.

[3] E. Filtz, S. Kirrane, A. Polleres, The linked legal data landscape: Linking legal data across different countries, Artificial Intelligence and Law 29 (2021) 485–539. doi:10.1007/s10506-021-09282-8.

[4] A. Oksanen, M. Tamper, J. Tuominen, E. Mäkelä, A. Hietanen, E. Hyvönen, Semantic Finlex: Transforming, publishing, and using Finnish legislation and case law as linked open data on the web, in: Knowledge of the Law in the Big Data Age, volume 317 of *Frontiers in Artificial Intelligence and Applications*, IOS Press, 2019, pp. 212–228. doi:10.3233/FAIA190023.

[5] A. Valleala, E. Hyvönen, H. Rantala, J. Tuominen, Using legislative change history of statutes based on a linked open data knowledge graph, in: Knowledge Graphs and Semantic Web (KGSWC 2025), volume 16373 of *Lecture Notes in Computer Science*, Springer, Cham, 2026, pp. 256–264. doi:10.1007/978-3-032-13109-6_18.

[6] R. Hoekstra, J. Breuker, M. Di Bello, A. Boer, The LKIF core ontology of basic legal concepts, in: Proceedings of the Workshop on Legal Ontologies and Artificial Intelligence Techniques, volume 321 of *CEUR Workshop Proceedings*, 2007, pp. 43–63.

[7] M. Palmirani, Legislative change management with Akoma-Ntoso, in: Legislative XML for the Semantic Web, Springer, 2011, pp. 101–130. doi:10.1007/978-94-007-1887-6_7.

[8] T.-H.-Y. Vuong, M.-Q. Hoang, T.-M. Nguyen, H.-T. Nguyen, H.-T. Nguyen, Constructing a knowledge graph for Vietnamese legal cases with heterogeneous graphs, in: 2023 15th

International Conference on Knowledge and Systems Engineering (KSE), IEEE, 2023, pp. 1–6. doi:10.1109/KSE59128.2023.10299526. arXiv:2309.09069.

[9] P. Akarajaradwong, P. Pothavorn, C. Chaksangchaichot, P. Tasawong, T. Nopparatbundit, K. Pratai, S. Nutanong, NitiBench: Benchmarking LLM frameworks on Thai legal question answering capabilities, in: Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, 2025, pp. 34304–34327. URL: https://aclanthology.org/2025.emnlp-main.1739/.

[10] J. Chen, S. Xiao, P. Zhang, K. Luo, D. Lian, Z. Liu, M3-Embedding: Multi-linguality, multi-functionality, multi-granularity text embeddings through self-knowledge distillation, in: Findings of the Association for Computational Linguistics: ACL 2024, 2024, pp. 2318–2335.

[11] G. V. Cormack, C. L. A. Clarke, S. Buettcher, Reciprocal rank fusion outperforms Condorcet and individual rank learning methods, in: Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval, 2009, pp. 758–759.