# A Knowledge Graph Construction Method for Legal Documents

Yan Li[1,2,3], Huanpu Yin[1,2,3], Ruotong Li[1,2,3], and Haisheng Li[1,2,3(✉)]

[1] School of Computer and Artificial Intelligence, Beijing Technology and Business University, Beijing 100048, China
[2] Beijing Key Laboratory of Big Data Technology for Food Safety, Beijing 100048, China
[3] National Engineering Laboratory For Agri-product Quality Traceability, Beijing 100048, China
lihsh@th.btbu.edu.cn

**Abstract.** "China Judgments Online" is a platform that collects and stores court judgments from various levels of courts nationwide, providing a vast amount of legal cases and rulings. Leveraging these information to promote legal transparency, fairness, and efficiency holds significant research value. And knowledge graph is a tool for organizing and representing knowledge graphically. By constructing relevant legal knowledge graphs, the entities, relationships, and attributes in legal data are structured to help legal practitioners better understand and analyze the relationships and characteristics between court judgments. Therefore, we propose a bottom-up approach to construct a legal knowledge graph. Firstly, we crawl relevant legal data from a wide variety of websites. Then, we employ different methods to extract entities and their relationships from both structured and unstructured data. Finally, we integrate these two sets of data through entity matching and attribute matching, and visualize the constructed knowledge graph using Neo4j.

**Keywords:** Legal knowledge graph · Bottom-up approach · Neo4j

## 1 Introduction

The platform "China Judgments Online"[1] serves as a platform for publishing effective judgment documents from people's courts at all levels in China. It has already exceeded a quantity of 100 million documents. With the continuous growth of this number, the content of these cases has also experienced an explosive growth trend. Faced with such a massive volume of judgment documents, effectively extracting and managing the legal knowledge contained within them has become a crucial problem in urgent need of resolution [1].

[1] https://wenshu.court.gov.cn/.


Y. Jia et al. (Eds.): CISC 2024, LNEE 1283, pp. 62–70, 2024.
https://doi.org/10.1007/978-981-97-8650-3_7

The knowledge graph [2–4] is a graphical model used to organize and represent structured knowledge. It establishes a network structure by using entities as nodes and relationships between entities as edges, enabling the visualization of the associations between pieces of knowledge. Therefore, knowledge graphs serve as effective tools for managing information within legal judgments. Depending on their application domain, knowledge graphs can be categorized into general knowledge graphs [5] and domain-specific knowledge graphs [6–8]. Since Google proposed the Knowledge Graph in 2012 [9], several general-purpose knowledge graphs have emerged. Microsoft Research Asia introduced a knowledge base called "Probase" [10], which automatically extracts over 2.7 million concepts from a corpus of 1.68 billion web pages. Leipzig University and Mannheim University jointly built a multilingual knowledge base named "DBpedia" [11], which encompasses 30 languages and over 2.6 million entities. Researchers at the Max Planck Institute for Informatics constructed "YAGO" [12], which comprises approximately 10 million entities and 120 million facts. While general-purpose knowledge graphs have found extensive applications in academia and industry, covering a vast amount of knowledge, they face difficulties in providing domain-specific services due to the lack of domain-specific knowledge. Therefore, for specific domains, it is necessary to construct knowledge graphs with domain-specific expertise to offer relevant services. Yu et al. [13] built a knowledge graph in the food domain. Wolfram Research developed a mathematics domain knowledge graph called "WolframAlpha" [14]. These knowledge graphs contain a wealth of data from various domains. However, in the field of legal documents, knowledge graphs are still deficient in managing and characterizing, like, the knowledge structure of legal cases and the use of specialized terminology [15].

Integrating legal case knowledge into knowledge graphs can reveal the connections and interactions among different cases. This aids in understanding the commonalities, differences, and trends among cases, providing a more comprehensive reference for legal research and decision-making. It also empowers intelligent legal services with enhanced capabilities. Lawyers, judges, and the public can receive faster and more accurate legal consultations, legal analysis, and case judgment predictions through structured legal case data.

During the process of knowledge graph construction, there are two common approaches: top-down and bottom-up [16]. The top-down approach [17] starts from the highest-level concepts and gradually refines the branches to ensure the integrity of the hierarchical structure. In this approach, the overall framework and concepts are initially built, and then the collected entities are added to their respective concepts one by one. On the other hand, the bottom-up approach [18,19] begins with low-level entities and gradually abstracts them to form higher-level concepts. In this approach, higher-level concepts and relationships are formed incrementally by discovering and extracting the relationships and interactions among the low-level entities.The bottom-up approach can start from the collected legal documents and adjust the structure of the knowledge graph based on the trends and evolution observed in the continuously expanding legal dataset. This ensures the knowledge graph is firmly grounded in the actual legal data. So, in this paper, we adopt the bottom-up approach to construct a knowledge graph in the legal domain.

Unlike general types of named entities (e.g., persons, institutions and places), entities in the legal domain involve many proper names (e.g., case types and trial procedure). The design of subject types and relationships is crucial in the legal domain. Therefore, to address these issues, we propose a framework for the construction of a legal knowledge graph, and our contribution points are as follows:

- We crawled the relevant data from the "China Judicial Documents" Network and constructed the corresponding corpus.
- We design a resource description framework to extract the corresponding triples from structured text.
- We fuse knowledge from different information sources to refine the constructed knowledge graph, which is stored and visually displayed via Neo4j.

## 2 Method

The key technologies involved in constructing a legal knowledge graph include data acquisition, knowledge extraction, and knowledge fusion. Finally, we demonstrates the constructed knowledge graph through Neo4j, which is continuously improved through incremental iteration to build a better legal knowledge graph. The specific process we follow for constructing the knowledge graph is illustrated in Fig. 1.

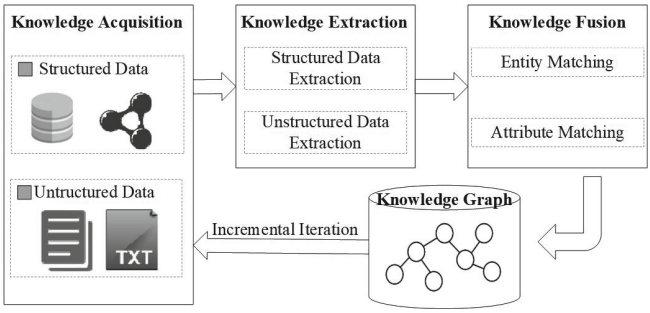


**Fig. 1.** The overall process of knowledge graph construction.

### 2.1 Data Acquisition

Data acquisition [20] is a crucial step in building a knowledge graph, conducting data analysis, and enabling intelligent decision-making. During the data acquisition process, it is necessary to gather information and entities relevant to the knowledge graph and organize and combine them into a suitable representation for the knowledge graph.

The website "China Judgments Online" is a platform established by the Supreme People's Court for publishing legal judgments on the internet, providing access to effective judgments from courts at all levels across the country.

Legal cases typically include information such as case name, cause of action, case type, legal basis, and so on. We used Python web crawling techniques to crawl relevant legal information from the "China Judgments Online" website, which mainly includes structured information and unstructured information. This helped us build a legal information corpus. For the extracted structured information, we performed data cleaning to remove irrelevant or inconsistent data elements to ensure data quality. We also standardized the representation of the data to maintain consistency across the entire dataset. For the extracted unstructured information, we segmented the text and added appropriate markup to identify the basic semantic units such as paragraphs, sentences, and terms. Then, we resolved the semantic ambiguities in the text, such as synonyms and relations of reference. We used the processed corpus as the dataset for constructing the knowledge graph. We stored all the processed data in the form of TXT or CSV files. The data acquisition process is illustrated in Fig. 2.

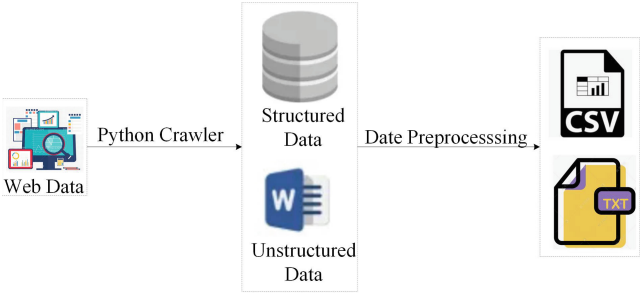


**Fig. 2.** The overall process of data acquisition.

### 2.2 Knowledge Extraction

One of the key steps in constructing a legal knowledge graph is knowledge extraction [21], which involves collecting data from multiple sources and adding it to the knowledge graph based on rules. In the process of knowledge extraction, we need to define the types of relations. The data we crawled mainly includes case number, case name, case type, and cause of action as entities. For case number and case name, one case number corresponds to a unique case name, thus we will define the relationship between case number and case name as ASSOCIATION. In addition, we have defined a total of 9 types of relations, and each relationship also has corresponding attribute values. The specific information of the defined relationship attributes is shown in Table. 1, where "domain" represents the class represented by the entity on the left side of the relationship, and "range" represents the class represented by the right side of the relationship.

**Table 1.** Relational properties of the legal knowledge graph.

| Relation | Domain | Range | Description |
|---|---|---|---|
| ASSOCIATION | Case number | Case name | Each case number corresponds to a specific case name |
| INVOLVES | Case number | Party | Representing the parties involved in a case |
| HELD_BY | Case number | Trial court | Representing the court where the case is held, with each case being tried in a specific court |
| BASED_ON | Case number | Legal basis | Representing the legal basis for the implementation of the case |
| HEARD_BY | Case number | Judge | Representing the judge who handles the case |
| HAS_REASON | Case number | Cause of action | Representing the cause of the case, where each case has a specific cause |
| HAS_TYPE | Case number | Case type | Representing the type of the case |
| HAS_PROCEDURE | Case number | Trial procedure | Representing the procedural steps of the case, where each case has its corresponding procedural steps |
| RELEASED_ON | Case number | Release date | Each case has a specific date of release |

*Structured Data Extraction.* Structured data refers to data organized in a specific format and follows defined rules, comprising clearly defined fields and values. Resource Description Framework (RDF) provides a flexible way to represent and link information from different data sources, serving as a foundation for data integration and inference. To extract the proper nouns and relationship entities in the legal domain, we designed a resource description framework triple graph model to better describe the associations and semantic meanings between legal data. The entity labels are divided into 10 types, including case number, trial procedure, case type, legal basis, and so on. Based on the crawled data, we further categorized the trial procedure into criminal first-instance trial and formal second-instance trial. We also divided the case type into criminal case, civil case, and administrative case. Each entity has its own set of attribute values. The designed resource description framework triple graph model is illustrated in Fig. 3. In the relationships, each relationship also has corresponding attribute values.

*Unstructured Data Extraction.* Unstructured data refers to data that does not conform to the traditional relational database structure. It does not follow a fixed data schema or tabular structure and lacks explicit predefined patterns or architectures. [22] Extracting knowledge from unstructured data into specific corpora involves the use of different techniques. Firstly, there is entity extraction, also known as named entity recognition, which aims to identify named entities with specific meanings in the text, including general entities and proprietary entities in the legal domain. Secondly, there is relation extraction, which primarily involves identifying entity pairs in the text and determining the rela-

tionship types and attributes between them using methods such as rules, machine learning, or deep learning. Lastly, there is attribute extraction, which involves identifying and extracting attribute values with specific semantic meanings from the text.

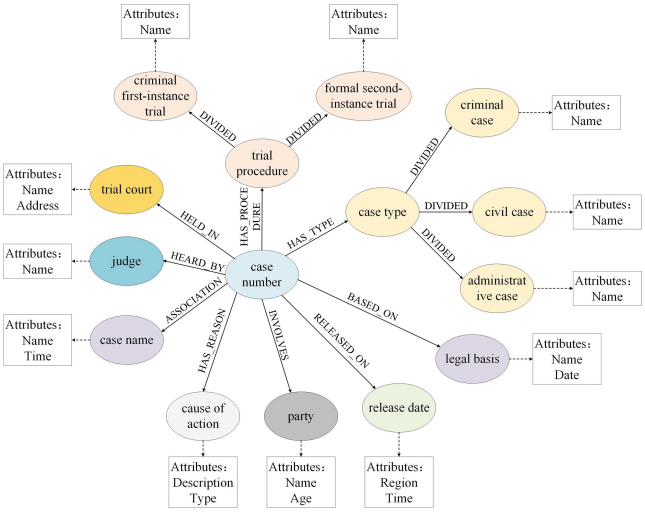


**Fig. 3.** Resource description ternary diagram overall framework diagram. Where the colors indicate different entity types and the white boxes indicate the attribute values that the entities have.

### 2.3 Knowledge Fusion

Knowledge fusion ensures the quality of the knowledge graph by eliminating redundancy, misconceptions, and conceptual ambiguity through entity matching and attribute matching [23]. Entity matching identifies and aligns entities with similar or identical semantics across different knowledge sources, avoiding the duplication of representing the same entity. This helps eliminate redundant information. Attribute matching resolves naming differences and inconsistent representations of attributes in different knowledge sources. By mapping attributes with the same semantics to a unified representation, it eliminates conceptual ambiguity and misconceptions. In this paper, we utilized the Word2Vec model, which considers contextual information, as the entity's vector representation. The chord distance $D_s(A, B)$ is used to determine the distance between two words represented by non-zero vectors $A$ and $B$ in an n-dimensional vector space. Assuming that the angle between $A$ and $B$ is defined as $\theta$, the cosine similarity measure $D_c(A, B)$ is calculated as follows:

$$D_c(A, B) = 1 - \cos(\theta) = 1 - D_s(A, B) \tag{1}$$

$$D_s\left(A,B\right)=\frac{\sum_{i=1}^{n}A_i\times B_i}{\sqrt{\sum_{i=1}^{n}A_i^2}\times\sqrt{\sum_{i=1}^{n}B_i^2}} \tag{2}$$

## 3 Storage and Visualization

Neo4j is a popular graph database management system widely used for constructing and managing knowledge graphs. In a knowledge graph, Neo4j can store and represent relationships between entities, forming a graph structure. This graph structure effectively captures complex associations and dependencies among entities. Therefore, we have chosen Neo4j to visualize and store the legal knowledge graph we have constructed. Figure 4 illustrates a portion of our constructed legal knowledge graph.

**Fig. 4.** A partial segment of the legal knowledge graph is depicted, where different colors represent different entity types. For example, blue represents parties involved, yellow represents publication dates, and so on.

## 4 Conclusion

This paper constructs a dataset in the legal field based on the data of "China Judgment and Decision Network", and chose the bottom-up knowledge graph construction mode with the help of the related technology of knowledge graph construction, corresponding relationship types are designed for entities. The implementation leverages the Neo4j graph database for visualizing the legal knowledge graph. This initiative lays the foundation for a legal intelligent question-answering system.

**Acknowledgments.** The study was supported by the Scientific Research Program of Beijing Municipal Education Commission (KZ202110011017), the Beijing Natural Science Foundation (No.4244078), and the National Natural Science Foundation of China (No. 62277001).

## References

1. Filtz, E.: Building and processing a knowledge-graph for legal data. In: The Semantic Web: 14th International Conference, ESWC 2017, Portorož, Slovenia, May 28–June 1, 2017, Proceedings, Part II 14, pp. 184–194. Springer (2017). https://doi.org/10.1007/978.3.319.58451.5.13
2. Liu, M., Zhao, C., Peng, X., Siming, Yu., Wang, H., Sha, C.: Task-oriented ML/DL library recommendation based on a knowledge graph. IEEE Trans. Software Eng. (2023). https://doi.org/10.1109/2023.3285280
3. Qian, Y., Pan, L.: Variety-aware GAN and online learning augmented self-training model for knowledge graph entity alignment. Inform. Process. Management **60**(5), 103472 (2023). https://doi.org/10.1016/2023.103472
4. Sui, Y., Feng, S., Zhang, H., Cao, J., Liang, H., Zhu, N.: Causality-aware enhanced model for multi-hop question answering over knowledge graphs. Knowl. Based Syst. **250**, 108943 (2022). https://doi.org/10.1016/2022.108943
5. Bollacker, K., Cook, R., Tufts, P.: Freebase: a shared database of structured general human knowledge. AAAI **7**, 1962–1963 (2007). https://doi.org/10.5555/1619797.1619981
6. Wang, J.: Knowledge graph analysis of internal control field in colleges. Tehnički vjesnik **27**(1), 67–72 (2020). https://doi.org/10.17559/20191004092659
7. Sarika, J., Pooja, H., Nandana, M., Sudipto, G., Abhinav, D., Ankush, B.: Constructing a knowledge graph from Indian legal domain corpus. In: Text2KG,: International Workshop on Knowledge Graph Generation from Text. Co-located with the ESWC (2022). https://doi.org/10.17559/251724819
8. Wang, D., Li, H., Wang, W., Qiao, L.: Cross-modal knowledge graph construction for multiple food additives. In: Chinese Intelligent Systems Conference, pp. 839–847. Springer (2022). https://doi.org/10.1007/978.981.19.6226.4.80
9. Singhal, A., et al.: Introducing the knowledge graph: things, not strings. Official Google Blog. **5**(16) (2012). https://www.scribd.com/document/481868810/Introducing-the-Knowledge-Graph-things-not-strings. https://doi.org/10.17559/481868810
10. Wang, Z., Huang, J., Li, H., Liu, B., Shao, B., Wang, H., Wang, J., Wang, Y., Wentao, W., Xiao, J., et al.: Probase: a universal knowledge base for semantic search. Microsoft Research Asia (2010). https://www.cs.sjtu.edu.cn/kzhu/papers/probase-demo.pdf
11. Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., Hellmann, S.: Dbpedia-a crystallization point for the web of data. J. Web Semant. **7**(3), 154–165 (2009). https://doi.org/10.1016/2009.07.002
12. Amarilli, A., Galárraga, L., Preda, N., Suchanek, F.M.: Recent topics of research around the YAGO knowledge base. In: Web Technologies and Applications: 16th Asia-Pacific Web Conference, APWeb 2014, Changsha, China, September 5–7, Proceedings 16, pp. 1–12. Springer (2014). https://doi.org/10.1007/978.3.319.11116.2.1

13. Haoze, Yu., Li, H., Mao, D., Cai, Q.: A relationship extraction method for domain knowledge graph construction. World Wide Web **23**(2), 735–753 (2020). https://doi.org/10.1007/s11280.019.00765.y
14. Necesal, P., Pospıšil, J.: Experience with teaching mathematics for engineers with the aid of wolfram alpha. Proceed. World Cong. Eng. Comp. Sci. **1**, 271–274 (2012). https://doi.org/10.5555/27740912
15. Haoze, Yu., Li, H., et al.: A knowledge graph construction approach for legal domain. Tehnički vjesnik **28**(2), 357–362 (2021). https://doi.org/10.17559/20201119084338
16. Zhao, Y., Zhang, B., Gao, D.: Construction of petrochemical knowledge graph based on deep learning. J. Loss Prev. Process Ind. **76**, 104736 (2022). https://doi.org/10.1016/2022.104736
17. Liang, H., Peng, X., Zhao, N., Duan, L., Yu, P., Fu, D.: An approach of top-down electric generation knowledge graph construction. In: IOP Conference Series: Earth and Environmental Science, vol. 661, p. 012021. IOP Publishing (2021). https://doi.org/10.1088/1755.1315/661/1/012021
18. Yang, W., Yang, S., Wang, G., Liu, Y., Jing, L., Yuan, W.: Knowledge graph construction and representation method for potato diseases and pests. Agronomy **14**(1), 90 (2023). https://doi.org/10.3390/14010090
19. Wang, Y., Cheng, Y., Qi, Q., Tao, F.: IDS-KG: an industrial dataspace-based knowledge graph construction approach for smart maintenance. J. Ind. Inf. Integr. **38**, 100566 (2024). https://doi.org/10.1016/2024.100566
20. Meng, X., Jing, B., Wang, S., Pan, J., Huang, Y., Jiao, X.: Fault knowledge graph construction and platform development for aircraft PHM. Sensors **24**(1), 231 (2023). https://doi.org/10.3390/24010231
21. Papaluca, A., Krefl, D., Suominen, H., Lenskiy, A.: Pretrained knowledge base embeddings for improved sentential relation extraction. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop, pp. 373–382 (2022). https://doi.org/10.18653/v1/2022.acl-srw.29
22. Wang, Yu., Tong, H., Zhu, Z., Li, Y.: Nested named entity recognition: a survey. ACM Trans. Knowl. Discov. Data (TKDD) **16**(6), 1–29 (2022). https://doi.org/10.1145/3522593
23. Mapetu, J.P.B., Kong, L., Chen, Z.: A dynamic VM consolidation approach based on load balancing using Pearson correlation in cloud computing. J. Supercomput. **77**(6), 5840–5881 (2021). https://doi.org/10.1007/s11227.020.03494.6