Current Science
CS 4 (6), 927– 937 (2024)
Received 14 Oct 2024 | **Revised** 20 **Nov.** 2024 | **Accepted** 05 **Dec**. 2024 | **Online Available** 25 **Dec** 2024
https://doi.org/10.52845/CS/2024-4-6-16

Current Science

Current Science

OPEN ACCESS

ISSN (O) 2795-8639

ORIGINAL ARTICLE

A Continuous Quality Improvement Evaluation Approach for Extracting Legal Knowledge

Huangtao Zhao

Nanjing Audit University, Nanjing, Jiangsu, P.R. China

***Corresponding Author: Huangtao Zhao**

Abstract

Legal texts, particularly case precedents, are a rich source of legal knowledge that can be harnessed for intelligent legal work. The extraction of legal knowledge in the format of <head entity, relation, tail entity> is becoming an increasingly critical task within the legal domain. While deep learning-based named entity recognition and other approaches are available for extracting legal knowledge triplets from texts, effectively evaluating the quality of these extracted triplets remains a significant challenge. To tackle this challenge, this paper introduces a continuous quality improvement evaluation approach for the extraction of legal knowledge. This method involves segmenting the initial legal texts into several parts, employing natural language processing techniques to extract knowledge triplets from one segment at a time, and then manually evaluating these triplets. Correctly identified triplets are utilized to aid in the extraction process for subsequent segments. Through iterative application, this approach allows for the efficient and effective extraction of knowledge triplets from legal texts. An experimental study, utilizing 382 cases from the Caselaw Access Project, was undertaken to generate and evaluate legal knowledge triplets. The findings underscore the efficiency and effectiveness of the proposed approach in extracting legal knowledge triplets and significantly enhancing their accuracy. The extracted triplets could lay a foundational groundwork for constructing a legal knowledge graph.

Keywords: knowledge extraction; quality evaluation; legal knowledge; text analysis

Introduction

Legal texts encapsulate a wealth of intricate and comprehensive judicial experiences [1]. Systematically organizing these documents into a legal knowledge graph and dissecting their inferential mechanisms could markedly bolster the development of intelligent judicial systems. The initial step in constructing a legal knowledge graph entails the extraction of knowledge, which involves converting unstructured text into structured knowledge representations [2]. Among such representations, triplets serve as a prevalent format, typically embodying a subject, predicate, and object, denoted as (S, P, O) [3]. Nonetheless, given the highly specialized and complex nature of the legal field, extracting knowledge triplets from legal texts presents notable challenges. A persistent challenge lies in ensuring the completeness and freshness of the extracted triplets. To tackle these challenges, numerous researchers are delving into aspects such as contextual semantic dependencies, affective considerations, and temporal dimensions. Despite the advancements in knowledge extraction techniques, including deep learning-based named entity recognition, the quality of the extracted triplets may not consistently meet high standards. It is the high-quality triplets that endow a knowledge graph with practical applicative value. Consequently, devising an effective evaluation method for these extracted triplets represents another critical step in the construction of a legal knowledge graph. Manual evaluation stands as a potential strategy to ensure the accuracy of the evaluation process. However, the extensive volume of triplets that form the basis of a legal knowledge graph makes manual evaluation an impractical venture. This method is notably resource-intensive, laborious, and time-consuming, posing significant challenges to its feasibility on a large scale.

To mitigate this challenge, this paper introduces a continuous quality improvement evaluation methodology.

Initially, it partitions the original legal texts into distinct segments. Subsequently, tools such as LexNLP and spaCy are utilized to extract knowledge triplets from these segments. Following this extraction, the triplets undergo a manual evaluation phase. Those triplets deemed accurate are then leveraged to aid in the extraction process for subsequent segments. Through iterative cycles of this process, it becomes possible to extract high-quality knowledge triplets from legal texts both efficiently and effectively.

The structure of this paper is methodically arranged as follows: Section 2 delves into the existing literature and related works within the domain. Section 3 elaborates on the continuous quality improvement evaluation approach, including a detailed discussion of the basic triplet extraction algorithm. Section 4 details the experimental setup and outcomes derived from analyzing legal case texts sourced from the Caselaw Access Project. Section 5 offers an in-depth discussion of the experimental results, interpreting

their implications within the broader context of legal knowledge extraction. Finally, Section 6 concludes the paper by summarizing the key findings and contributions of this work.

2 Related Work

Presently, research on legal knowledge extraction remains relatively limited [4]. Nevertheless, the foundational structure of legal knowledge triplets does not markedly differ from general triplets. The primary distinction lies in the specificity of the content within entities and relations, tailored to the intricacies of the legal domain. The ongoing research endeavors outlined in this paper are dedicated to enhancing and assessing knowledge extraction, with a primary emphasis on two key facets: knowledge triplet extraction and the evaluation of triplet quality.

2.1 Research on Knowledge Triplet Extraction

Knowledge triplet extraction is the process of deriving structured knowledge in the form of triplets from unstructured plain text. In this paper, knowledge triplets are denoted as (h, r, t), where 'h' represents the head entity, 't' represents the tail entity, and 'r' represents the relationship between 'h' and 't' [5]. The extraction of knowledge triplets involves deducing (h, r, t) based on the information present in the sentence text. Early methods for knowledge triplet extraction relied on rule-based and statistical approaches, employing techniques such as Hidden Markov Models (HMM) [6] and Conditional Random Fields (CRF)

[7] within the realm of machine learning. These methods typically performed named entity recognition (NER) [8] and relation extraction (RE) [9] sequentially, ultimately leading to the extraction of triplets. Nevertheless, these early methods were encumbered by various limitations, including error propagation and redundancy in entity pairs. With substantial advancements in deep learning technologies, such as Convolutional Neural Networks (CNNs) [10], Recurrent Neural Networks (RNNs) [11], attention mechanisms [12], and the advent of pre-trained language models like BERT [13] and GPT [14], the landscape of entity and relation extraction research has undergone a significant transformation towards the adoption of joint models. Recent studies, exemplified by researchers like Zhang et al. [15], have harnessed pre-trained language models like BERT-WWM to acquire contextual and candidate entity-related embedding information for classification tasks, thereby achieving cross-document subject-predicate-object (SPO) extraction tasks. Incorporating convolutional neural networks, Cao et al. [16], for instance, introduced a pre-trained convolutional neural network model (R-BERT-CNN) tailored for entity-relation extraction tasks. This model integrates entity-level information into pre-trained models and utilizes CNNs to extract sentence-level information. Additionally, certain research models have incorporated transfer learning methods for fine-tuning, as observed in models like ULM-Fit [17], PharmKE [18], and ATLOP [19], aiming to enhance overall model convergence speed and accuracy. Significantly, sentiment analysis has emerged as a crucial focus, leading research efforts towards extracting sentiment triplets from text. This includes approaches such as Text Sentiment Triplet Extraction (TSTE) [20] and Aspect Sentiment Triplet Extraction (ASTE) [21].

2.2 Research on Triplet Quality Evaluation

Triplet quality evaluation constitutes a vital component of knowledge refinement [22]. As implied by its name, this process involves the assessment of the quality of triplets extracted from text, which typically manifest as structured representations of knowledge in the form of subject-predicate-object structures. Several well-established frameworks exist for evaluating data and information quality, including those proposed by Wang et al. [23], Stvila et al. [24], and Zaveri et al. [25] In assessing triplet quality, the evaluation process can be conducted across six primary dimensions: accuracy, consistency, completeness [26], freshness [27], trustworthiness [28], and availability [29]. Additionally, there are secondary evaluation dimensions [25] that may also influence the overall quality. Hence, it becomes imperative to engage in construction maintenance [30], synchronize updates [31], and rectify errors [32] to ensure the continual enhancement of triplet quality.

Generally, accuracy stands out as the most intuitively reflective measure of quality. Naumann et al. [33] equates accuracy to information quality (IQ). Wang et al. [29] propose that knowledge graph accuracy should be defined as the degree to which knowledge is correct, reliable, and proven to be error-free.

Various methods exist for evaluating knowledge graph accuracy. For example, Fõrber et al. [34] assess knowledge graph accuracy by examining the functional dependency between the attributes of entities, while Lei et al. [35] base their evaluation on the ratios of inaccurate labels, annotations, and classifications of entities. Considering the negative factors that can impact knowledge graph quality, such as errors in relationships, entities, and attributes, potentially diminishing knowledge graph accuracy, current research primarily focuses on identifying these negative factors through error detection. Approaches like the path ranking algorithm proposed by Lao et al. [36] and the method combining type features and path features into a relationship classifier suggested by Melo et al. [32] have proven effective in detecting errors in relationships. Additionally, the approach introduced by Paulheim et al. [37], based on the statistical distribution of entity attributes and types, aims to identify errors in entity type assertions. For errors in attributes, Golab et al. [38] and Koudas et al. [39] utilize multi-attribute consistency checks to identify errors in the numerical attributes of entities.

Furthermore, various other dimensions can significantly influence the accuracy aspect. Consistency-based error detection [40] and consistency checks [41] achieved through knowledge inference, for example, can contribute to

enhancing the accuracy of knowledge graphs. When contrasted with DBpedia's global update approach [42], employing local updates through statistical methods and deep learning techniques can further enhance the accuracy of predictions for knowledge updates. Noteworthy examples encompass Bayesian inference [43], stochastic modeling [44], and two time-aware knowledge update prediction models proposed by Jiang et al. [45].

Manual evaluation represents the most straightforward method of evaluation. If executed entirely by human effort, the implications in terms of time efficiency and labor costs are significant. To address this challenge, Gao et al. [46] introduced an efficient sampling method designed to offer robust statistical assurances for the accuracy of quality evaluations while simultaneously reducing the burden of manual assessment. Additionally, the correctness of triplets can be verified using data intrinsic to the knowledge graph itself. Li et al. [47] developed an innovative AttTucker model, which leverages the Transformer architecture to incorporate path-level information between entities in the knowledge graph for enhancing the evaluation of its quality. Their findings indicated notable improvements in F1 score and overall accuracy for tasks related to knowledge graph quality assessment. Nevertheless, the pursuit of high-quality triplets presents its own set of challenges, as errors within the knowledge base could compromise the efficacy of triplet evaluation. To counteract this, Ban et al. [48] integrated external consistency (EC) and internal consistency (IC) assessments to pinpoint potentially erroneous triplets. This methodology effectively diminishes the impact of incorrect knowledge, thereby significantly improving the reliability of triplet quality evaluations.

2.3 Summary

In conclusion, triplet quality evaluation plays a pivotal role in knowledge extraction tasks, significantly contributing to the credibility and utility of the extracted knowledge. The evolution of knowledge triplet extraction methods has witnessed a shift from early rule-based and statistical approaches to the adoption of sophisticated deep learning techniques, including BERT, CNN, RNN, and more. While methods for triplet extraction continue to evolve, there is a growing recognition of the importance of evaluating the quality of the extracted triplets. The dimensions of quality evaluation are an ongoing focus, with continuous exploration aimed at discovering more effective methods to enhance the overall quality. Methods based on sampling design or utilizing internal knowledge graph data have proven effective, yet may not consistently achieve the desired level of precision. On the other hand, manual evaluation offers higher precision but is less efficient. In response to these considerations, this paper advocates for a balanced approach that combines automated extraction of legal triplets with semi-automated quality evaluation. This approach seeks to achieve a judicious compromise between efficiency and precision in the extraction and assessment of legal knowledge triplets.

3 The Continuous Quality Improvement Evaluation Approach

3.1 The basic triplet extraction algorithm

In the architecture of the proposed evaluation methodology, an iterative cycle of triplet extraction and assessment is fundamental. Consequently, elucidating the basic triplet extraction algorithm is imperative, albeit the primary emphasis of this document is on the evaluation process rather than the extraction mechanism per se. Thus, the extraction algorithm leverages basic linguistic analyses—specifically, part-

of-speech tagging and syntactic order examination—to extract triplets. The principal operations of this algorithm are diagrammatically illustrated in Figure 1.

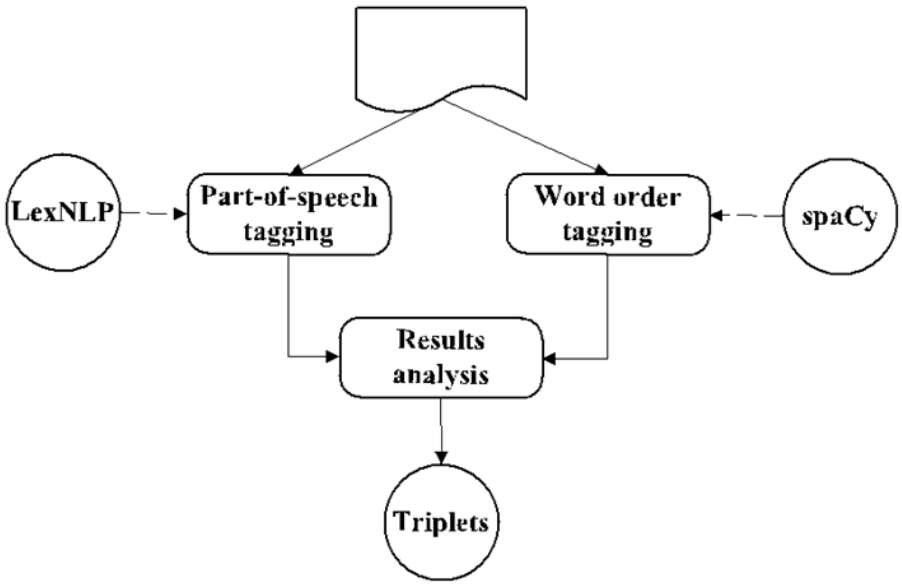

Figure 1 The main step of the basic triplet extraction algorithm

As depicted in Figure 1, the triplet extraction process relies on the utilization of LexNLP [49] for part-of-speech analysis and spaCy [50] for word order analysis within a given sentence. LexNLP serves as a specialized tool tailored for the nuanced characteristics of unstructured legal texts, proficiently discerning the part-of-speech for each word in a sentence. On the other hand, spaCy is a high-performance natural language processing tool adept at identifying the subject, predicate, and object in a sentence through meticulous word order analysis. The extraction of triplets is accomplished by scrutinizing the word orders of nouns and verbs within a sentence. To illustrate, consider the sentence "Appellant expected to prove by each of said jurors that he had formed an opinion that the defendant was guilty." Figure 2 provides a visual depiction of the triplet extraction process applied to this specific sentence, elucidating the sequential steps involved in identifying and constructing the subject, predicate, and object elements.

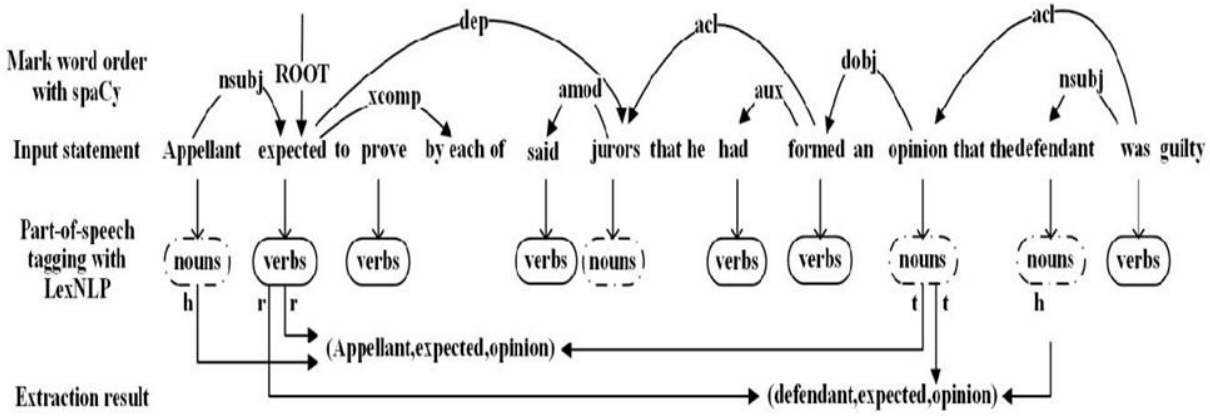

Figure 2 The triplet extraction procedure

In the outlined procedure, the annotations provided by spaCy play a crucial role in deciphering the syntactic structure of a sentence, thereby facilitating the automatic extraction of triplets. The annotations such as "ROOT", "nsubj" and "dobj" are instrumental in identifying the core components of the sentence structure—predicate, nominal subject, and direct object, respectively. This analytical framework leverages the syntactic roles determined by spaCy to classify words into their respective roles within a triplet. When a word that fulfills both the criteria of being a noun and having the "nsubj" (nominal subject) annotation is identified as the head entity of the triplet. Similarly, when a word that is identified as a noun and has the "dobj" (direct object) annotation is classified as the tail entity. Finally, a word that is recognized as a verb with the "ROOT" annotation, which signifies its role as the main verb or predicate in the sentence, is earmarked as the relationship component of the triplet. Applying these principles to the example sentence, two triplets: (Appellant, expected, opinion) and (defendant, expected, opinion) are extracted automatically. This method allows for the systematic extraction of structured knowledge in the form of triplets from unstructured text, streamlining the process of converting vast volumes of legal text into a more organized and accessible format for further analysis and application.

3.2 The Proposed Evaluation Approach

Quality evaluation stands as a pivotal step in knowledge extraction [46], constituting a central focus of this paper. Its objective is to ensure the accuracy and dependability of the extracted triplets. Manual triplet evaluation represents an alternative approach, characterized by high precision but inefficiency. To address this challenge, a continuous quality improvement evaluation method is introduced. In this methodology, triplet extraction and manual evaluation are conducted alternately. Triplets that fulfill specific weight screening conditions are systematically utilized to progressively refine the quality assessment of the triplets. The core procedure of this approach is visually represented in Figure 3.

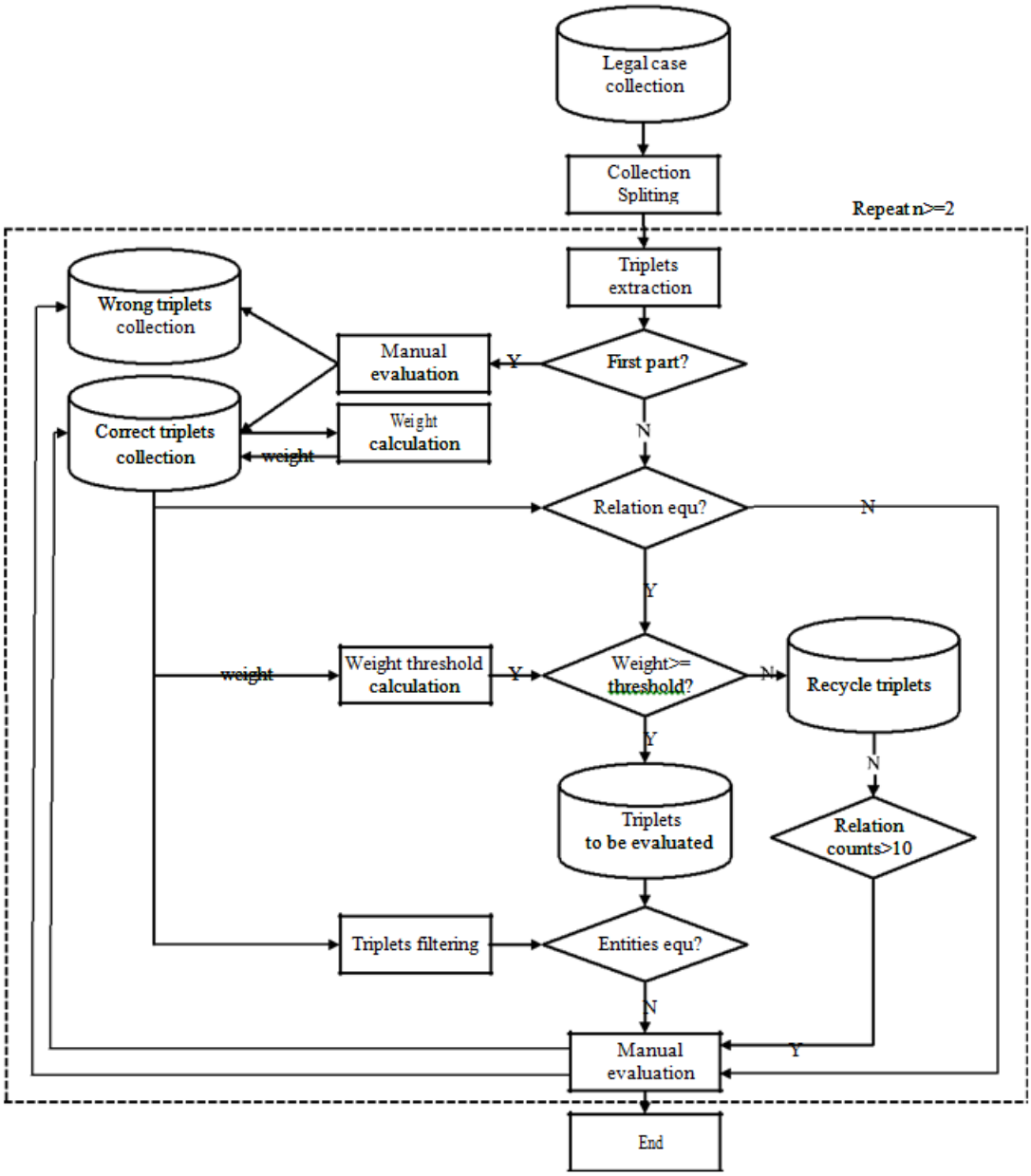

Figure 3 The main procedure of the quality evaluation of triplets

Figure 3 illustrates the process where the initial corpus of legal judgments is segmented into n equal parts (where n is greater than or equal to 2). A segment is selected at random to serve as the initial point of analysis, and the designated triplet extraction method is employed to extract triplets from this segment. Subsequent to extraction, these triplets undergo a rigorous manual review process. Within this process,

triplets deemed accurate are allocated to the "Correct Triplet Collection" whereas those identified as erroneous are relegated to the "Incorrect Triplet Collection". The evaluation of each triplet's validity hinges on its semantic integrity and its pertinence to the legal domain. For triplets classified under the "Correct Triplet Collection", the significance of each relation verb encapsulated within the triplets is quantitatively assessed employing the subsequent formula:

quantitatively assessed employing the subsequent formula:

(1)

Applying Formula (1), each relation verb in the correct triplet collection is assigned a corresponding weight value. Subsequently, the collection is updated to incorporate these weight values. Following this, triplets are extracted from any of the remaining parts, designated as the second part. The "Correct Triplet Collection" is employed for a comparative analysis with the newly extracted triplets. If the new triplets exhibit different relations compared to those in the "Correct Triplet Collection", they undergo manual reevaluation. Correct triplets are then integrated into the "Correct Triplet Collection", while incorrect ones are segregated into the "Wrong Triplet Collection". For triplets sharing the same relations as those in the "Correct Triplet Collection", weight thresholds are computed. These thresholds serve as filters to identify triplets requiring manual evaluation.

Assuming there are five relation words {n1, n2, n3, n4, n5}

with corresponding weights {0.2, 0.3, 0.5, 0.7, 0.8} in the "Correct Triplet Collection", and the same five relation words appear in the newly extracted triplets. If the weight threshold is set to 0.2, all triplets will be considered candidates for evaluation, leading to a relatively high ratio of candidate triplets in the newly extracted set. Conversely, with a threshold of 0.8, only triplets containing the word n5 will be considered candidates, resulting in a low ratio of candidate triplets. The ratio of candidate triplets is inversely proportional to weight, as higher-weighted relation words are more likely to be correct candidates in the "Correct Triplet Collection". Formula (1) establishes that weight represents the ratio of correct triplets, creating a proportional relationship. In cases where two inverse relations share the same weight, they intersect at a specific weight threshold. This threshold signifies the point where the probability of correct candidate triplets balances with the number of potential candidates.

Figure 4 illustrates the relationship between weight (w) and two key metrics: the ratio of candidate triplets (s_t), which decreases inversely with weight as represented by function f1(w), and the ratio of correct triplets (r_t), which increases in direct proportion to weight as depicted by function f2(w). In this coordinate system, an increase in weight leads to a reduction in the pool of candidate triplets for evaluation, as indicated by the declining trend of f1(w). This trend suggests that with higher weights, fewer triplets are flagged for review. Conversely, the ratio of correct triplets, as shown by the ascending trend of f2(w), suggests that triplets with higher weights have a greater likelihood of accuracy and inclusion in the "Correct triplet collection".

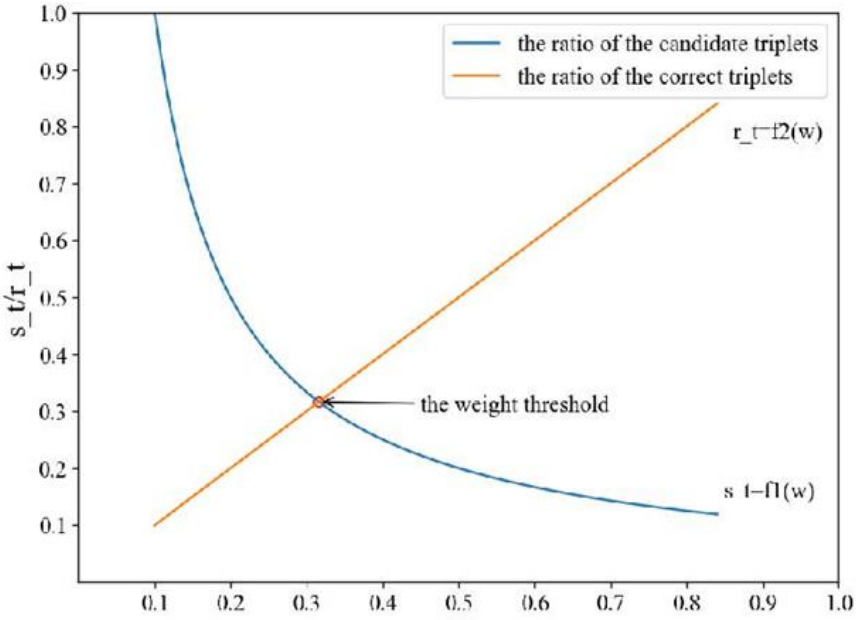

Figure 4 The method of finding the weight threshold

The intersection point depicted in Figure 4, where the inverse function of candidate triplet ratio and the direct function of correct triplet ratio converge, signifies the optimal weight threshold. This critical threshold ensures a balance between identifying potential candidates for evaluation and maintaining a high likelihood of correctness among these candidates. By applying this threshold, the selection process for evaluating triplets becomes more efficient: only those triplets whose relation weights are at or above the threshold, and whose entities differ from those already verified in the "Correct triplet collection," are subjected to further manual scrutiny. This ensures a focused review on potentially novel or incorrect entries. Conversely, triplets falling below this threshold are relegated to the "Recycle triplet collection" for additional consideration or refinement, acknowledging their current lower probability of correctness but not dismissing their potential value after reevaluation or adjustment in future knowledge extraction cycles. By iterating this process, both the "Correct triplet collection" and "Recycle triplet collection" steadily increase. This iterative mechanism enhances the efficiency and precision of the evaluation. As the "Correct triplet collection" enlarges, it facilitates a more automated and precise evaluation process. The larger collection provides a more comprehensive and diverse set of correct triplets, allowing the proposed approach to learn and identify correct triplets more effectively. To address the potential oversight of correct triplets in the "Recycle triplet collection", manual re-evaluation is initiated once the count of any same triplet in the collection reaches a predefined size, such as ten triplets. This ensures that no correct triplet is mistakenly excluded, allowing for their accurate inclusion into the "Correct triplet collection". Through this repetitive approach of extraction and evaluation, a high-quality set of knowledge triplets is progressively refined, bolstering the knowledge base's comprehensiveness and accuracy.

4 Experiment

4.1 Data sources

4.2 Experiment procedure and Experiment result

A total of 8,460 sentences were derived from the 382 legal cases, and these sentences were tokenized into a sum of 134,561 words. Subsequently, LexNLP and spaCy were employed for part-of-speech and word order tagging,

respectively. Based on the annotation results, a total of 2,855 entities and relationships were extracted. This automated extraction process resulted in the generation of 15,228 triplets, representing the relationships between entities in the legal text.

Initially, the 382 legal cases were divided into six segments, each comprising approximately 1,500 sentences for the purpose of conducting experiments. The first segment was selected for the automatic extraction of knowledge triplets, yielding a total of 1,999 triplets. Manual annotation of these triplets led to the identification of 508 correct triplets, and initial weight values for each relational verb were calculated to establish an initial weight table. Utilizing the continuous quality improvement evaluation method previously described, the triplets extracted from the first data segment were refined using the initial weight table. This refinement process resulted in 1,602 triplets that satisfied the weight filtering criteria, of which 421 were deemed correct.

Subsequently, the same procedure was applied to the second dataset segment, which led to the automatic generation of 1,787 triplets. It was noted that 1,574 of these triplets passed the weight filtering criteria, with 288 identified as correct. Table 1 showcases the weight values assigned to certain relational verbs, derived from the analysis of both the first and second data segments.

Table 1: Relational verb weight values obtained from the data.

The first part of data		The second part of data	
Relational verbs	Weight values	Relational verbs	Weight values
shown	0.48855	shown	0.509677
is	0.112933	is	0.154976
given	0.363636	given	0.438596
answered	0.634146	answered	0.622642
think	0.3	think	0.369048
held	0.329787	held	0.449612
be	0.104651	be	0.12
convicted	0.258065	convicted	0.404762
asked	0.458015	asked	0.539394
had	0.415385	had	0.425532
said	0.352941	said	0.373494

Table 1 highlights discrepancies in the weight values of each relational verb between the two experiments. This variation is attributed to the gradual increase in the total number of triplets formed by each relational verb, as each dataset comprises approximately over 2000 triplets. The evolving count of correct triplets further contributes to these changes, ultimately resulting in variations in the weight values until a weight threshold is reached. Following these observations, experiments were conducted on the remaining four data segments, with specific results depicted in Figure 5.

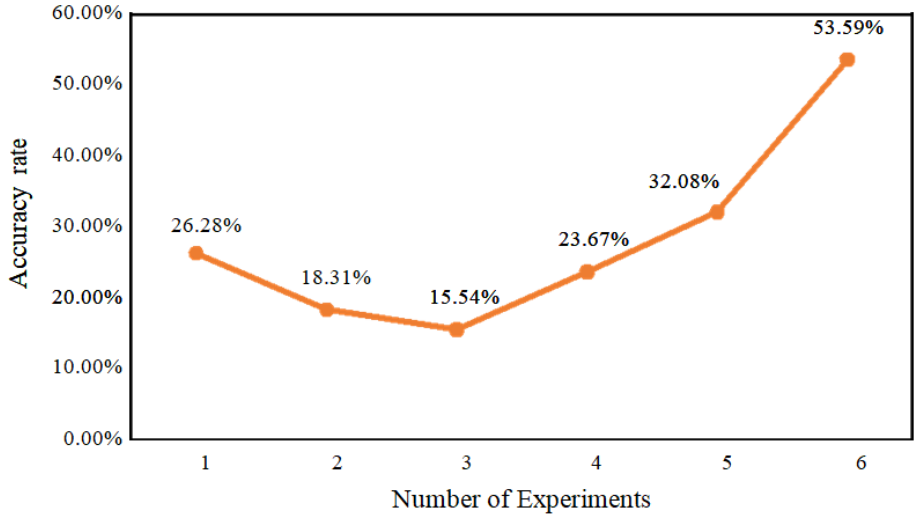

Figure 5 Experimental results using the continuous improvement assessment model

Observing from Figure 5, discernible improvements in accuracy are evident, particularly in the last three experiments, where the increase is notably pronounced. The results from the model evaluation in these latter experiments surpass those of the preceding rounds, underscoring the effectiveness of the proposed progressive quality evaluation method in enhancing the accuracy of triplets.

5 Discussion

The outcomes largely aligned with anticipations, yet a notable decrease in accuracy was observed in the initial three experiments. This decline could stem from a low weight threshold assigned to certain relational verbs in the second or third data sets. The rationale behind these low thresholds is that, during the first or second experiments, these verbs appeared infrequently or were predominantly associated with incorrect triplets, adversely impacting the accuracy of the data in the second or third segment.

Subsequently, each of the six segments underwent manual annotation. The process began with automated extraction of knowledge triplets, followed by employing a multi-person evaluation method for manual annotation, enhancing the data's reliability and consistency. A triplet was deemed correct only if the majority of evaluators agreed on its accuracy. Sets of incorrect triplets were excluded. Figure 6 displays the outcomes from the human evaluation experiment conducted across the six segments.

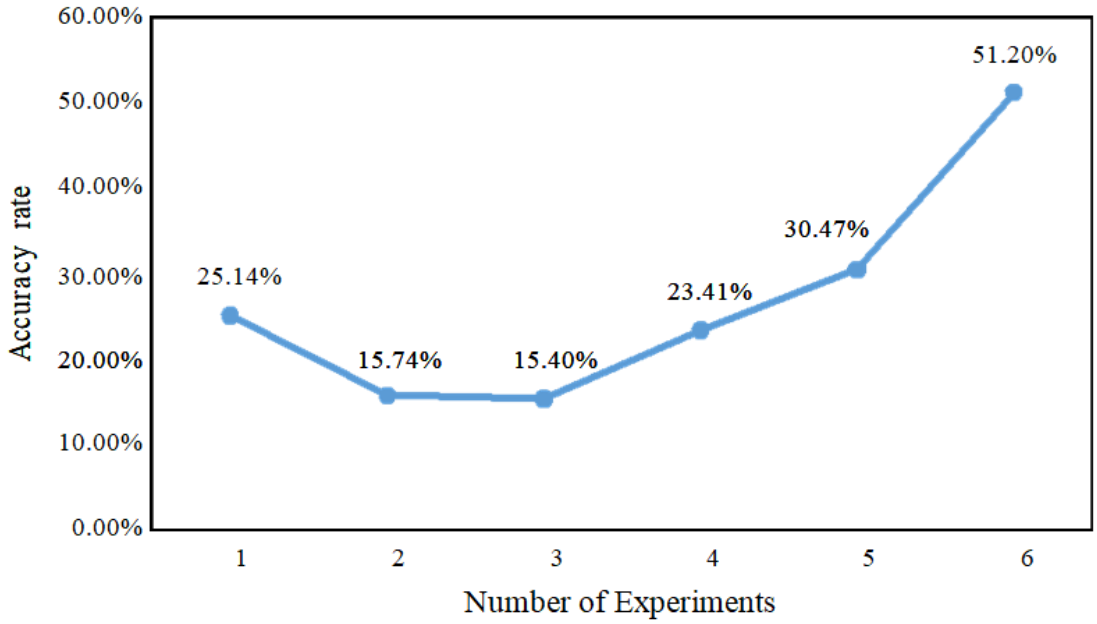

Figure 6 Experimental results using manual evaluation

Based on the outcomes depicted in Figure 6, there's a noticeable decline in accuracy during the first three segments, suggesting an initial scarcity of correct triplets within these portions. This pattern hints at a level of randomness inherent in the data selection process. Notably, the manual evaluation results for the third segment stood at only 15.40%, marking the lowest performance across all six experiments. Consequently, a significant number of triplets from this segment were relegated to the recycle collection post-model annotation, underscoring the efficacy of the model's rescue mechanism. According to this protocol, any relational verb recurring over 10 times within the recycle collection is subject to human re-evaluation. This strategy is instrumental in salvaging a considerable number of correct triplets, thereby bolstering the model's robustness.

Additionally, it is worth noting that in each experiment, there is no assurance that a relational verb will precisely appear in the recycle triplet collection more than 10 times. The occurrences might be, for instance, only 3 or 6 times. Consequently, any triplets in the recycle triplet collection that haven't undergone reevaluation before each experiment are retained. This mechanism of cumulative evaluation enables the approach more reliable, ultimately enhancing the completeness of triplets. Table 2 presents the experimental results for selected he relational verbs in the initial two experiments and the third experiment.

Table 2: Results 0f the first three experiments with some relational verbs

The results of the first two experiments				The results of the third experiment	
Relational verbs	Sum	Number of correct	Weight threshold	The number of newly generated triplets	New correct number
called	64	2	0.03125	18	3
been	25	1	0.04	6	2
was	302	31	0.102649	54	8
presented	31	2	0.064516	13	9
alleged	56	4	0.071429	36	5
decided	46	4	0.086957	8	1
find	46	4	0.086957	21	8
render	22	2	0.090909	0	0

authorize	22	2	0.090909	4	0
occurred	22	1	0.045455	0	0

Table 2 presents various relational verbs with lower and less frequent weight thresholds, along with the count of newly generated triplets in the corresponding column. However, it's noteworthy that the number of triplets identified as correct is minimal, or even nonexistent. This phenomenon is attributed to the impact of the weight threshold or the presence of inaccurate data itself, where a lower weight threshold results in the filtration of more triplets. For example, considering the current weight threshold for the verb "been" is 0.04, if hypothetically there are 100 relevant triplets newly generated, only 4 would surpass the threshold after filtering, leaving the remaining 96 in the recycle triplet collection. It is evident that the weight values of many triplets formed by certain relational verbs are listed as 0, although not explicitly mentioned. Nevertheless, these triplets, consisting of relational verbs with zero-weight values, should not be dismissed. As the experiments unfold, the weight values of these relational verbs will gradually reach the threshold. However, it's crucial to acknowledge the subjectivity and randomness inherent in manual evaluation, and the potential impact of data randomness on the experiment results should not be overlooked.

As the evaluation progresses, a consistent increase in accuracy is observed across subsequent experiments. This improvement is directly linked to the foundation established by correctly identified triplet sets in the initial experiments. The corresponding weight values for relational verbs gradually increase, leading to a reduction in the number of filtered triplets. Relational verbs identified in the initial experiments approach a threshold in subsequent iterations, resulting in an increased number of correct triplets. For newly emerging relational verbs, their weight values rise progressively, expanding the size of the triplet set. It is foreseeable that, over time, the accuracy achieved by the proposed model will surpass that of manual evaluation. Through multiple rounds of evaluation iterations, the progressive quality evaluation method consistently identifies and incorporates new correct triplets, achieving high accuracy. The experimental results robustly demonstrate the effectiveness and potential of the proposed method in generating high-quality triplets for legal research.

6 Conclusion

This paper introduces a continuous improvement evaluation method for extracting legal knowledge, specifically focusing on assessing the quality of extracted triplets. This marks an initial exploration into evaluating the quality of triplets extracted in the context of legal knowledge. The experiment, based on 382 cases from the Caselaw Access Project, aims to generate and evaluate legal knowledge triplets. The results showcase the efficiency and accuracy of the proposed approach in generating high-quality legal knowledge triplets. The analysis of experimental outcomes indicates that this approach effectively enhances the triplets' accuracy while minimizing human and resource input. By employing this method, a set of high-quality triplets can be efficiently extracted from legal texts, providing foundational groundwork for constructing a high-quality legal knowledge graph. Future research will focus on optimizing knowledge extraction and evaluation methods by exploring the integration of deep learning or large-scale models such as BERT and ChatGPT to further enhance the construction of high-quality legal knowledge graphs.

7 Acknowledgement

This work is funded by the Planning Fund Project of Humanities and Social Sciences Research of Ministry of Education (No:23YJA870009), the Significant Project of Jiangsu College Philosophy and Social Sciences Research (No: 2021SJZDA153), and the Qing Lan Project of Jiangsu College.

References

1. Kingston, S., AV, S., MS, B., & Rajagopal, M. K. Comparative study on Judgment Text Classification for Transformer Based Models. arXiv 2023 arXiv:2306.01739. https://doi.org/10.48550/arXiv.2306.01739.
2. Huang, P., Zhao, X., Fang, Y., Zhu, H., & Xiao, W. End-to-end Knowledge Triplet Extraction Combined with Adversarial Training. Journal of Computer Research and Development 2019, 56(12): 2536-2548. https://doi.org/10.7544/issn1000-1239.2019.20190640.
3. Liu, S., d'Aquin, M., & Motta, E. Measuring Accuracy of Triples in Knowledge Graphs. In: Gracia,J., Bond, F., McCrae, J., Buitelaar, P., Chiarcos, C., Hellmann, S. (eds) Language, Data, and Knowledge. LDK 2017. Lecture Notes in Computer Science(), vol 10318. Springer, Cham. https://doi.org/10.1007/978-3-319-59888-8_29.
4. Zheng, Y., Zhu, D., Wu, H., & Peng, X. Overview on Knowledge Graph Question Answering. Computer Systems & Applications 2022,31(04):1-13. https://doi.org/10.15888/j.cnki.csa.008418.
5. Liu, Z., Sun, M., Lin, Y., & Xie, R. Knowledge Representation Learning: A Review. Journal of Computer Research and Development 2016, 53(2): 247-261. https:// doi.org/10.7544/issn1000-1239.2016.2016 0020.
6. Mor, B., Garhwal, S., & Kumar, A. A Systematic Review of Hidden Markov Models and Their Applications. Arch Computat Methods Eng 2021, 28, 1429–1448. https://doi.org/10.1007/s11831-020-0 9422-4.
7. Sutton, C., & McCallum, A. 2012. An introduction to conditional random fields. Foundations and Trends® in Machine Learning 2012,4(4):267-373. https://doi.org/10.1561/2200000013.
8. Mohit, B. 2014. Named entity recognition. In Natural language processing of semitic languages. Berlin, Heidelberg: Springer Berlin Heidelberg. 2014, pp. 221-245. https ://doi.org/10.1007/978-3-642-45358-8_7.
9. Pawar, S., Palshikar, G. K., & Bhattacharyya, P. Relation extraction: A survey. arXiv 2017

arXiv:1712.05191. https ://doi.org/10.48550/arXiv. 17 12.05191.
10. Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., ... & Chen, T. Recent advances in convolutional neural networks. Pattern recognition 2018, 77: 354-377. https://doi.org/10.1016/j.patcog. 20 17.10.013.
11. Salehinejad, H., Sankar, S., Barfett, J., Colak, E., & Valaee, S. Recent advances in recurrent neural networks. arXiv 2017 arXiv:1801.01078. https://doi. org/10.4855 0/arXiv.1801.01078.
12. Niu, Z., Zhong, G., & Yu, H. A review on the attention mechanism of deep learning. Neurocomputing 2021, 452: 48-62. https://doi.org/10.1016/j.neucom. 2021. 03.091.
13. Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. North American Chapter of the Association for Computational Linguistics. arXiv 2018, ar Xiv1810. 04805. https://doi.org/10.18653/v1/N19-1423.
14. Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. Improving language understanding by generative pre-training 20 18.
15. Zhang, Z. Cross Document SPO Extraction with BERT-WWM Pre-training. Computer Applications and Software 2023, 06: 181-186+215. https:// doi. org/10.3969/j.issn.100 0-386x.2023.06.028.
16. Cao, W., & Xu, X. Entity Relationship Extraction Based on R-BERT-CNN. Computer Applications and Software 2023, 04: 222-229. https://doi.org/10.39 69/j.issn. 1000-386x.2023.04.036.
& Howard, J., & Ruder, S. Universal language model fine-tuning for text classification. In I. Gurevych & Y. Miyao (Eds.), Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics 2018; (Volume 1: Long Papers) (pp. 328-339). Melbourne, Australia: Association for Computational Linguistics. https://doi.org/10.18653/v1/P1 8-1031.
17. Jofche, N., Mishev, K., Stojanov, R., Jovanovik, M., Zdravevski, E., & Trajanov, D. PharmKE: Knowledge Extraction Platform for Pharmaceutical Texts Using Transfer Learning. Computers 2023, 12(1): 17. https://doi.org/10.3390/computers1201 0017.
18. Zhou, W., Huang, K., Ma, T., & Huang, J. Document-level relation extraction with adaptive thresholding and localized context pooling. In Proceedings of the AAAI conference on artificial intelligence, 2021; 35(16), 146 12-14620. https://doi.org/10.16 09/aaai. v35i16.17717.
19. Chen, Z. Research on Text Sentiment Triplet Extraction Algorithm. Huazhong University of Science &Technology, 2023. https://doi.org/10.27157/d.cnki. ghzku. 2021 .005374.
20. Li, Y., Wang, F., & Zhong, S. H. A More Fine-Grained Aspect–Sentiment–Opinion Triplet Extraction Task. Mathematics 2023, 11(14):3165. https://doi.org/ 10.3390/math 11 143165.
21. Paulheim, H. Knowledge graph refinement: A survey of approaches and evaluation methods. Semantic web 2017, 8(3): 489-508. https:// doi.org/10. 3233/SW-160218.
22. Wang, X., Chen, L., Ban, T., Usman, M., Guan, Y., Liu, S., ... & Chen, H. Knowledge graph quality control: A survey. Fundamental Research 2021, 1(5): 607-626.
24. Stvilia, B., Gasser, L., Twidale, M. B., & Smith, L. C. A framework for information quality assessment. Journal of the American society for information science and technology 2007, 58(12): 1720-1733. https: //doi.org/10.1002/asi.20652.
25. Zaveri, A., Rula, A., Maurino, A., Pietrobon, R., Lehmann, J., & Auer, S. Quality assessment for linked data: A survey. Semantic Web 2016, 7(1): 63-93. https://doi.org/10.3233/SW-150175.
26. Issa, S., Adekunle, O., Hamdi, F., Cherfi, S. S. S., Dumontier, M., & Zaveri, A. Knowledge Graph Completeness: A Systematic Literature Review. IEEE Access 2021, 9: 31322-31339. https://doi.org/10. 1109/ACCESS.2021.3056622.
27. Färber, M., Bartscherer, F., Menne, C., & Rettinger, A. Linked data quality of dbpedia, freebase, opencyc, wikidata, and yago. Semantic Web 2018, 9(1): 77-129. https://doi.org/10.3233/SW-170275.
28. Jayawardene, V., Sadiq, S., & Indulska, M. An analysis of data quality dimensions. ITEE Technical Report 2013-01 and 2015-02. School of Information Technology and Electrical Engineering, The University of Queensland. Australia, 2015.
29. Wang, R. Y., & Strong, D. M. Beyond accuracy: What data quality means to data consumers. Journal of management information systems 1996, 12(4): 5-33. https://doi.org/10.1080/07421222.1996. 115 18099.
30. Lin, Y., Liu, Z., Sun, M., Liu, Y., & Zhu, X. Learning entity and relation embeddings for knowledge graph completion. In Proceedings of the AAAI conference on artificial intelligence, February 2015; 29(1). https://doi.org/10.1609/aaai.v29i1.9491.
31. Liang, J., Zhang, S., & Xiao, Y. How to keep a knowledge base synchronized with its encyclopedia source. In Proceedings of the 26th International Joint Conference on Artificial Intelligence, 2017; 3749–3755. AAAI Press. https://doi.org/10.24963/ijcai. 20 17/524.
32. Melo, A., & Paulheim, H. Detection of relation assertion errors in knowledge graphs. In Proceedings of the Knowledge Capture Conference, December 2017; 22, 1-8. https://doi.org/10.1145/3148011.3148 033.
33. Naumann, F. (Ed.). Quality-driven query answering for integrated information systems. Lecture Notes in Computer Science 2002, volume 2261. https://doi.org /10.1007/3-540-45921-9.
34. Fürber, C., & Hepp, M. Swiqa–a semantic web information quality assessment framework. In V. K. Tuunainen, M. Rossi & J. Nandhakumar (eds.), ECIS. 2011, 76. https://aisel.aisnet.org/ecis2011/76.
35. Lei, Y., Uren, V., & Motta, E. A framework for evaluating semantic metadata. In Proceedings of the 4th international conference on Knowledge capture, October 2007; 135-142. https://doi.org/10.1145/129 8406.1298431.
36. Lao, N., Mitchell, T., & Cohen, W. Random walk inference and learning in a large scale knowledge base.

In Proceedings of the 2011 conference on empirical methods in natural language processing, July 2011; 529-539. https://aclanthology. org/D11-1049.
37. Paulheim, H., & Bizer, C. Improving the quality of linked data using statistical distributions. International Journal on Semantic Web and Information Systems (IJSWIS) 2014, 10(2): 63-86. https://doi. org/10.40 18 /ijswis.2014040104.
38. Golab, L., Karloff, H., Korn, F., Saha, A., & Srivastava, D. Sequential dependencies. Proceedings of the VLDB Endowment 2009, 2(1): 574-585. https:// doi.org/10.147 78/1687627.1687693.
39. Koudas, N., Saha, A., Srivastava, D., & Venkatasu bramanian, S. Metric functional dependencies. In 2009 IEEE 25th International Conference on Data Engineering. IEEE, 2009; 1275-12 78. https://doi.org /10.1109/ICDE.2009.219.
40. Li, H., Li, Y., Xu, F., & Zhong, X. Probabilistic error detecting in numerical linked data. In: Chen, Q., Hameurlain, A., Toumani, F., Wagner, R., Decker, H. (eds) Database and Expert Systems Applications. Globe DEXA 2015 2015. Lecture Notes in Computer Science(), 2015, 9261. Springer, Cham. https://doi. org/10.1007/978-3-319-22 849-5_5.
41. Paulheim, H. Knowledge graph refinement: A survey of approaches and evaluation methods. Semantic web 2017, 8(3): 489-508. https:// doi.org/10. 3233/SW-160218.
42. Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., & Ives, Z. Dbpedia: A nucleus for a web of open data. In: Aberer, K., et al. The Semantic Web. ISWC ASWC 2007 2007. Lecture Notes in Computer Science, November 2007; 4825. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-76298-0_52.
43. Barnett, V. Review of An Introduction to Bayesian Inference and Decision. by R. L. Winkler. Journal of the Royal Statistical Society. Series A (General) 1973, 136(3): 452–454. https://doi.org/10.2307/2345002.
44. Pinsky, M., & Karlin, S. An introduction to stochastic modeling. Academic press. 2010. https://doi.org/ 10. 1016/C2009-1-61171-0.
45. Jiang, T., Liu, T., Ge, T., Sha, L., Chang, B., Li, S., & Sui, Z. Towards time-aware knowledge graph completion. In Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, December 2016.1715-1724. https://aclanthology.org /C16-1161.
46. Gao, J., Li, X., Xu, Y. E., Sisman, B., Dong, X. L., & Yang, J. Efficient knowledge graph accuracy evaluation. Proc. VLDB Endow 2019,12(11): 16 79–1691. https://doi.org/10.14778/3 342263 .3342642.
47. Li, H., Shi, Z., Pan, C., Zhao, D., & Sun, N. Cybersecurity knowledge graphs construction and quality assessment. Complex & Intelligent Systems 2023, 10: 1201–1217. https://doi.org/10.1007/s40747-023-01205-1.
48. Ban, T., Wang, X., Chen, L., Wu, X., Chen, Q., & Chen, H. Quality evaluation of triples in knowledge graph by incorporating internal with external consistency. IEEE transactions on neural networks and learning systems 2024, 35(2): 1980–1992. https://doi.org/10.1109/TNNLS.2022.3186033.
49. Bommarito, Michael James and Katz, Daniel Martin and Detterman, Eric, LexNLP: Natural Language Processing and Information Extraction For Legal and Regulatory Texts June 6 2018. Available at SSRN: https://ssrn.com/abstract=3192101 or http://dx.doi. org/ 10.2139/ssrn.3192101.
50. Vasiliev, Y. Natural language processing with Python and spaCy: A practical introduction. No Starch Press. 2020. https://books.google.com.tw/books?id=btQgq AAACAAJ.