LELDR: Learning Encompassed Strategic Framework for Legal Document Recommendation Incorporating Semantics

Gerard Deepak[1,†], Navya Joshy[2,*,†] and Samiksha Shukla[3,†]

[1]BMS Institute of Technology and Management,Bengaluru,India
[2]Manipal Institute of Technology Bengaluru, Manipal Academy of Higher Education, Bengaluru, India
[3]Symbiosis University of Applied Sciences Indore, Bengaluru, India

Abstract

In the era of the Web 3.0, there is a growing need for a document recommendation system that caters to highly specialized domains, such as legal studies. This paper proposes a legal document recommendation framework that generates legal ontologies and integrates them with existing upper domain ontologies. It also strategically builds knowledge graphs and incorporates information from static repositories that draw from several heterogeneous sources of domain knowledge. This helps boost the concentration of auxiliary knowledge within the model. It also incorporates Deep Belief Networks and Logistic Regression as classification infrastructures to classify the static knowledge repository and the dataset, respectively. Explicit Semantic Analysis, K-L divergence, Shannon-Wiener Index computation with differential threshold and step deviance measures are employed in the proposed framework along with Red Deer Optimization for metaheuristics-driven optimization, making it the best-in-class approach with an overall precision of 95.07%, an F-measure of 96.41%, and an FDR of 0.05.

Keywords

Document Recommendation System, Legal Document, Ontology, Deep Belief Networks, Red Deer Optimization,

1. Introduction

The rise of Web 3.0 signifies a transformative shift in online engagement, addressing the limitations of the current web and aiming to deliver a more efficient, secure, and user-centric experience. Web 3.0 prioritizes seamless data integration across platforms to enable smarter, context-aware applications. While Web 3.0 and Semantic Web are sometimes used interchangeably, the former spans broader advancements, whereas the latter refers specifically to technologies and standards aimed at helping machines understand and interpret the meaning of information on the internet. With the increasing complexity of online data, Semantic Web-compatible technologies have become crucial for efficient data retrieval as traditional

MSW-2024: 4th International Workshop on Multilingual Semantic Web, December 10, 2024, Paris, France

[*]Corresponding author.

[†]These authors contributed equally.

gerard.deepak@bmsit.in (G. Deepak); navya.joshy02@gmail.com (N. Joshy); samiksha.shukla@gmail.com (S. Shukla)

0000-0003-0466-2143 (G. Deepak); 0009-0007-1793-1951 (N. Joshy); 0000-0003-2690-8175 (S. Shukla)

methods have become inefficient. They link and structure data in a machine-understandable way, contributing to accurate and relevant information retrieval.

A tangible application of the Semantic Web is seen in document recommendation systems, which use semantic technologies to comprehend relationships and context within documents, enabling them to recommend relevant content to users. The documents in the legal field are complicated, extensive, and demand specialized knowledge; hence, a legal document recommendation system aids legal professionals and researchers in analysing semantics and identifying relevant case laws, statutes, and precedents, thereby improving the efficiency and comprehensiveness of legal research. Thus, the integration of Semantic Web technologies and document recommendation systems in the context of legal documents enhances the accessibility, interpretation, and utilization of the vast legal information available on the Web for informed decision-making.

1.1. Motivation

The primary motivation is the need for a document recommendation framework for highly specialized domains like legal sciences. There is also a dire need for a strategic framework for legal document recommendations that integrates auxiliary knowledge sources and aligns with Web 3.0 standards, as most existing frameworks fail to meet these standards. To address this, this paper increases the strength of classification by using the hybridization of a machine learning and a deep learning classifier at different stages in the architectural pipeline. In addition to this, there are very few frameworks that are operable in an environment that is highly cohesive and dense with knowledge. Owing to this, the proposed legal document recommendation framework focuses on integrating ontologies and knowledge graphs while incorporating auxiliary knowledge from static, standardized knowledge repositories, which is the central focus of the paper.

1.2. Contributions

The framework integrates existing upper-domain legal ontologies with newly generated ontologies based on dataset entities to build a strong foundation of knowledge. Additionally, further generation of knowledge graphs using Google's Knowledge Graph (KG) API makes it a very strong knowledge encompassment framework, which is quite novel. Apart from this, a static domain knowledge repository comprising of legal e-books, glossary-indexed metadata, case reports, judgement reports, and legal expert reports also adds to the density of auxiliary knowledge, this too is novel. Explicit Semantic Analysis (ESA), Shannon-Wiener Index, and K-L Divergence with differential threshold and step deviance measures are used to compute semantic similarity for quantitative semantic reasoning, thereby making it quite innovative. Red Deer Optimization serves as a nature-inspired, metaheuristic driven best-in-class optimization strategy. The Logistic Regression (LR) classifier and hybridized Deep Belief Network (DBN) classifier are used at two different stages in the proposed architectural pipeline in order to provide a very strong learning infrastructure, which also makes it quite novel.

1.3. Organization

Section 1 depicts the Introduction, Section 2 addresses the Related Works, Section 3 presents the Proposed System Architecture, Section 4 delves into the Implementation, Section 5 presents the Performance Evaluation and Results, Section 6 presents a Discussion followed by Section 7 which is the Conclusion, and Section 8 depicts References.

2. Related Works

Yang et al. [1] developed LegalGNN, a graph neural network framework addressing challenges in legal recommendation through unified content and structure integration, user query incorporation, and relational attention mechanisms, enhancing performance significantly. Dhanani et al. [2] have presented a framework for a Legal Document Recommendation System (LDRS) utilizing graph clustering and Doc2Vec for efficient identification of relevant legal judgments, addressing scalability concerns by limiting pairwise similarity computations. Sleimi et al. [3] proposed a novel model in the Requirements Engineering (RE) field, utilizing natural language processing (NLP) techniques to achieve automated template recommendations for legal requirements elicitation. Trivedi et al. [4] proposed a model for summarizing verbose and unstructured Indian legal case documents and retrieving similar cases, utilizing a support vector classifier trained on pre-1970s Indian Supreme Court data to enhance efficiency by focusing on crucial case paragraphs for retrieval.

Zheng et al. [5] introduce LawRec, a recommendation framework utilizing the models of BERT and Skip-RNN to incorporate legal provisions and case descriptions, effectively recommending laws and regulations for cases. Thomas et al. [6] suggest a framework called QuickCheck that facilitates efficient retrieval of relevant legal case opinions by searching through entire texts, analysing citation networks, and employing an advanced ranking model hierarchy supported by a comprehensive legal taxonomy and editorial case summaries. Liu et al. [7] introduce a method utilizing the TextRank algorithm to determine similarity among texts describing criminal facts and legal cases, effectively extracting key features for recommending comparable legal cases, demonstrating better performance than leading-edge experiments with 1,000 theft-related legal judgment documents. Gerard et al. [8] introduced a Bi-LDR, a semantically driven legal document recommendation system utilizing logistic regression and Long Short-Term Memory (LSTM) classifiers, enriched by semantic and entity similarity computations, achieving high accuracy and F-measure with minimal false discovery rate.

Dhani et al. [9] put forth a framework that outlines a solution for predicting similar nodes in a legal knowledge graph, addressing challenges related to node type selection and feature identification in downstream graph tasks. Roopak et al. [10] introduce Onto Judy which utilizes a static judicial domain ontology, structural topic modelling, and random forest classification to achieve highly accurate recommendations for judicial cases by integrating semantic similarity computation and user preferences from the CAIL2018 dataset. Shankhdhar et al. [11] proposed a Legal Semantic Web project that utilizes Resource Description Framework (RDF)-based court case repositories and web semantics to facilitate proactive legal decision-making, enabling lawyers to efficiently filter and extract relevant judgments for improved argumentation. Lu et al. [12] introduced a content recommendation system based on issues in the legal domain, leveraging metadata and user behaviour data to enable precise topic detection, cluster association, and labelling, producing high-quality recommendations comparable to those generated by human experts across diverse legal document types.

3. Proposed System Architecture

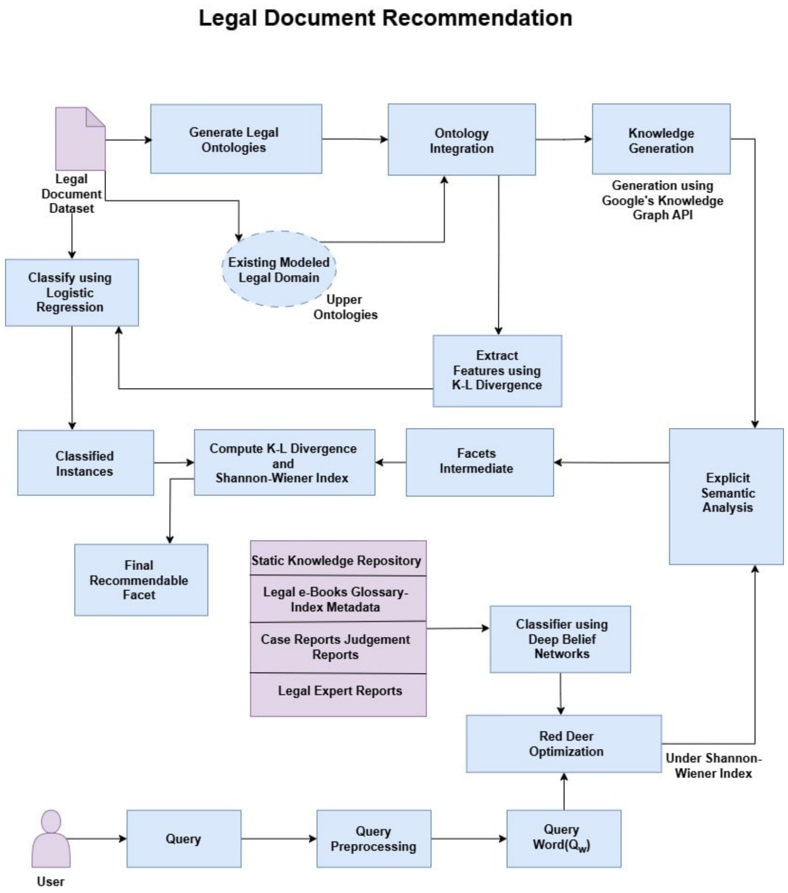
Legal Document Recommendation
Legal Document Dataset
Generate Legal Ontologies
Ontology Integration
Knowledge Generation
Generation using Google's Knowledge Graph API
Existing Modeled Legal Domain
Upper Ontologies
Classify using Logistic Regression
Extract Features using K-L Divergence
Classified Instances
Compute K-L Divergence and Shannon-Wiener Index
Facets Intermediate
Explicit Semantic Analysis
Final Recommendable Facet
Static Knowledge Repository
Legal e-Books Glossary-Index Metadata
Case Reports Judgement Reports
Legal Expert Reports
Classifier using Deep Belief Networks
Red Deer Optimization
Under Shannon-Wiener Index
User
Query
Query Preprocessing
Query Word(Q_w)

Figure 1: LELDR Architecture Model

Figure 1 portrays the architecture diagram for the proposed legal document recommendation framework, which encompasses strategic knowledge-driven learning and reasoning at its core. Since the framework revolves around the legal domain, it employs a dataset of legal documents. The dataset categories and their associated keywords are subjected to the generation of legal ontologies—structured representations of legal concepts—and an automatic ontology generator called OntoCollab is used as the tool of choice.

Also, existing legal domain ontologies spanning up to seven levels are developed through consultations with domain experts, legal experts and professors of law across several subdomains of law like the law of torts, criminal law, etc. Glossaries are derived from e-books and they serve as the foundation for these ontologies, which enables the creation of a detailed seven-level structure and the formulation of an upper ontology. Both the existing legal ontologies and the legal domain upper ontologies are integrated. Integration happens randomly and strategically by finding common concepts. If there are no common concepts, then Shannon's Entropy is found, and the nodes with a similar or almost equal Shannon's Entropy or information measure are aggregated and the ontology is integrated. For the integrated ontology, its entities are subjected to knowledge graph generation through the KG API, where knowledge graphs and subgraphs of the existing Google KG API are loaded. The integrated ontology is also subjected to feature extraction, for which K-L divergence is applied with a step deviance of 0.10. The step deviance is chosen as such to keep it more stringent, and the LR classifier utilizes these features to classify the actual legal document dataset. The instances classified by the LR classifier are then utilized in the model. Subsequently, a static knowledge repository is used as metadata for the legal domain. This comprises of crawled case reports and judgement reports from the World Wide Web, as well as data from several socio-legal experts, lawyers, law professors, legal expert reports, and e-book indices which are crawled or manually extracted. This static knowledge repository is extensively large owing to the large number of entities, and therefore it can't be handled as it is; rather, it is classified by implementing a strong deep learning classifier, namely the Deep Belief Networks. The DBN sorts through the static knowledge repository of the legal domain while the user query undergoes query preprocessing.

Queries are the primary input in the document recommendation framework. The query preprocessing involves tokenisation, lemmatisation, stop word removal, and Named Entity Recognition (NER). Once the user query is pre-processed, the individual query words (Qw) are obtained, and the query words are enriched by subjecting them to Red Deer Optimization, which is an optimization algorithm that utilizes metaheuristic principles. Owing to the stringency and scientific factors involved in the Red Deer Optimisation algorithm, it is used to enhance the query words under the Shannon-Wiener Index used as an objective function. The Shannon-Wiener Index is initially adjusted to a step deviance of 0.10 owing to the stringency of Red Deer Optimization as well as the inherent strength of the index as a result of it being supported by scientific principles. However, the step deviance is relaxed to 0.15 and is not made that stringent to allow for more flexibility.

Entities that come out of this pipeline are subjected to Explicit Semantic Analysis with the knowledge graphs generated through the KG API. ESA is done until every node in the generated knowledge graph and subgraph is covered at least once. ESA is set to a median threshold of 0.15, as relevance is already computed to generate intermediate facets. These intermediate facets are further subjected to the computation of the K-L Divergence of step deviance of 0.10 and the Shannon-Wiener Index of step deviance of 0.10. Both step deviances are made very stringent owing to the large number of entities in the intermediate facets. It is computed with the classified instances of the dataset that come out of the LR classifier, to yield the final recommendable facets, which are organised in the increasing order of the Shannon-Wiener Index. These recommendable facets are recommended as the user clicks on them; these facets are correlated with the categories and keywords in the document dataset, and the documents are yielded to the user by correlating. If satisfied, the search concludes; otherwise, the current user's facet selections are noted and utilized as inputs to continue this process until no additional user clicks are registered.

The formula for Shannon's entropy is expressed as:

$$H = \sum_{i=1}^{S} (-P_i \times \ln P_i) \tag{1}$$

From Eqn (1), it is inferred that H is the Shannon Entropy, Pi represents the fraction of the population consisting of a particular species i, ln is the natural log, S signifies the total count of species encountered, and Σ denotes the summation of species 1 to S.

The Kullback-Leibler (K-L) Divergence measures the difference between two probability distributions over the variable x itself.

$$D_{KL}(P||Q) = \sum_{x \in X} P(x) \log\left(\frac{P(x)}{Q(x)}\right) \tag{2}$$

From Eqn (2), it is inferred that P(x) and Q(x) represent the probability distributions of a discrete random variable x. P(x) represents the true or precisely calculated theoretical data distribution, while Q(x) stands for a theory or approximation of P(x).

Logistic Regression classifiers are a key machine learning algorithm that excels in binary classification tasks by determining the probability of an instance belonging to a specific class. Using input features known as independent variables, the algorithm assigns probability scores to potential classes and predicts the output class based on the highest probability. The sigmoid function transforms predicted values into probabilities between 0 and 1, adhering to probability theory constraints. The process of LR involves a linear combination of input features with weights and a bias term that generates raw scores that the sigmoid function converts into probabilities. A threshold of 0.5 is commonly set for class predictions, with probabilities exceeding this indicating positive class membership. Model training adjusts weights and biases iteratively through optimization methods to minimize disparities between predicted probabilities and actual class labels in the training dataset. LR offers user-friendly implementation, interpretability, and efficient training for quick classification of unfamiliar instances, particularly performing well with linearly separable datasets.

A Deep Belief Network is a unique machine learning algorithm that combines unsupervised learning principles with neural network architecture to create a distinctively deep structure. Unlike traditional neural networks, DBNs have multiple hidden layers that form a hierarchical structure by incorporating simple unsupervised networks such as Restricted Boltzmann Machines (RBMs) or autoencoders in a layered design of stacking sub-networks where one of the sub-network's hidden layers becomes the visible layer for the next, allowing the extraction of intricate patterns. DBNs employ layer-wise pre-training, typically through a Greedy learning algorithm, which involves learning generative weights layer by layer and establishing relationships between variables in one layer and those in the layer above. This process iteratively repeats for additional layers, and each RBM layer is separately trained using a Contrastive Divergence algorithm with positive and negative phases. The Greedy learning technique sequentially trains each RBM until the entire DBN is established, showcasing its efficiency in capturing complex patterns and features. DBNs offer a hierarchical structure with inherent advantages such as automatic feature extraction, reduced reliance on manual feature engineering, and proficiency in diverse tasks such as image recognition, video sequence analysis, etc.

The Red Deer Optimization Algorithm is a nature-inspired metaheuristic algorithm used for optimization tasks, integrating exploration and exploitation phases for effective solution space navigation. It begins with the random initialisation of potential solutions as Red Deer arrays feature layer-wise pre-training through a Greedy learning algorithm, mirroring natural male Red Deer behaviour for competitive edge enhancement through roaring. The subsequent classification into commanders and stags leads to a dynamic fighting phase, with commanders engaging in contests to form harems. The superior solution, determined by the objective function (OF), replaces the commander in this process. The harem establishment phase allocates hinds to commanders based on their OF values through proportional division. The mating process involves commanders selecting hinds as parents for the next generation, showcasing an innovative genetic evolution approach. Commanders may expand territory by attacking other harems, demonstrating adaptability to dynamic scenarios. The algorithm concludes with the strategic selection of the next generation through either retaining all-male Red Deer or selecting hinds and offspring through a fitness tournament or roulette wheel, allowing for flexibility based on iterations, the best solution's quality, or a specified time interval.

The Shannon-Wiener Index measures the diversity of species in a community and is calculated as:

$$H = -\sum_{i=1}^{R} p_i \ln p_i \tag{3}$$

In Eqn (3), H denotes the Shannon-Wiener Index, where pi is the proportion of the entire community made up of species i.

Explicit Semantic Analysis represents texts as vectors by utilizing a document corpus as a knowledge base.

$$S(x, y) = \frac{x \cdot y}{||x||||y||} = \frac{\sum_{i=1}^{N} x_i y_i}{\sqrt{\sum_{i=1}^{N} x_i^2}\sqrt{\sum_{i=1}^{N} y_i^2}} \tag{4}$$

In Eqn (4), N denotes the number of documents. Here, x and y are vectors used to compute the relatedness of two words.

4. Implementation

The proposed LELDR framework was carried out using Python3, with Google Colaboratory as the preferred IDE for the implementation. The legal ontologies were generated using OntoCollab and ontology integration was achieved utilizing agents designed by using AgentSpeak. The static knowledge repository was formalized by crawling case reports, judgements and legal expert reports from Web 3.0's customized crawlers. E-books were also downloaded from several repositories, and then they were pre-processed to yield only the indexes and glossaries. The Red Deer Optimization was implemented using a multi-agent setup, again developed using AgentSpeak. Semantic similarity computation and semantic similarity step deviance computation were done using K-L divergence and Shannon-Wiener Index with differential threshold and step deviance, this too was achieved using an agent designed using AgentSpeak. The implementation was done using a 6th generation i7 processor as part of the system requirements.

A single large dataset was created, comprising of multiple datasets such as the Dataset of Legal Documents[13] provided by the Deutsches Forschungszentrum für Künstliche Intelligenz (German Research Center for Artificial Intelligence), the Document Classification for Legal Firms dataset[14] and the Named Entity Recognition for Legal Documents dataset[15] given by Globose Technology Solutions (GTS), and OpenLegalData (2022 - Corpus)[16] provided by Jan Oliver Rüdiger. The implementation was done based on customized annotations created with a specialized annotator tool along with documents from the Web related to these annotations. The keywords of dataset entities were crawled and further annotated to formalize a seemingly large annotated dataset of the Web 3.0. Continuous experiments were then conducted.

Algorithm 1 Legal Document Recommendation Algorithm

```
Input: Legal Document Dataset D, Pre-existing Legal Ontologies O, User Query Q, and Static Knowledge Repository R
Output: Recommended Legal Documents
Begin
    Step 1: Pre-processing
        DCAT ⟵Extract_categories(D)
        DKET ⟵Extract_Keywords(D)
    Step 2: Ontology Generation
        Generated_Ontologies ⟵OntoCollab(DCAT, DKET)
    Step 3: Ontology Integration
        Integrated_Ontologies ⟵Integrate_Ontologies(O, Generated_Ontologies)
    Step 4: Node Aggregation
        Aggregated_Nodes ⟵Aggregate_Similar_Nodes(Integrated_Ontologies, Shannon_Entropy_H)
    Step 5: Knowledge Graph Generation
        Knowledge_Graphs ⟵Generate_Knowledge_Graphs(Aggregated_Nodes)
    Step 6: Divergence Computation
        for each graph in Knowledge_Graphs do
            KLD ⟵Compute_KLDivergence(graph, step_deviance=0.10)
        end for
    Step 7: Document Classification
        Classified_D ⟵Logistic_Regression_Classifier(D)
    Step 8: Repository Classification
        Classified_R ⟵Deep_Belief_Networks_Classifier(R)
    Step 9: Query Processing
        Processed_Q ⟵Tokenize_Lemmatize_RemoveStopWords_NER(Q)
    Step 10: Optimization
        Optimized_Q ⟵Red_Deer_Optimization(Processed_Q, Shannon_Wiener_Index_H, step_deviance=0.15)
    Step 11: Semantic Analysis
        ESA_Results ⟵Explicit_Semantic_Analysis(Optimized_Q, Knowledge_Graphs)
    Step 12: Facet Computation
        for each facet in ESA_Results do
            KLD ⟵Compute_KLDivergence(facet, step_deviance=0.10)
            Shannon_Index ⟵Compute_Shannon_Wiener_Index(facet)
        end for
    Step 13: Facet Organization
        Organized_Facets ⟵Organize_Facets_By_Shannon_Index(ESA_Results)
    Step 14: Recommendation Generation
        Recommended_Documents ⟵Correlate_Facets_With_Categories_Keywords(Organized_Facets, Classified_D)
        Yield: Recommended_Documents
End
```

Algorithm 1 enhances legal document retrieval by integrating and refining multiple data sources and processing steps. It begins by extracting categories and keywords from the dataset and generating legal ontologies using OntoCollab. These are then integrated with pre-existing ontologies through strategic matching and Shannon's Entropy-based aggregation. From the integrated ontology, the knowledge graphs are generated, and the features are extracted using K-L divergence. The document dataset is classified by a LR classifier, while a static knowledge repository is processed with DBNs for enhanced metadata classification. The user queries then undergo preprocessing and optimization via Red Deer Optimization, adjusted for the Shannon-Wiener Index. ESA is used to analyse the optimized queries against knowledge graphs, which then produce intermediate facets. These facets are refined through K-L Divergence and Shannon-Wiener Index calculations which are correlated with document categories and keywords to generate recommendations. The framework continues to refine recommendations based on user interactions until satisfactory results are achieved.

5. Performance Evaluation and Results

The proposed LELDR framework's performance, namely the Learning-Encompassed Strategic Framework for Legal Document Recommendation that incorporates semantics-oriented AI, is assessed considering precision, recall, accuracy, and F-measure percentages as potential metrics and the false discovery rate (FDR) as an auxiliary metric that quantifies the false positives. The reason for using these metrics as strategic primary metrics is because they quantify the relevance of results.

Table 1
Comparison of Performance of the proposed LELDR with other approaches

Model	Average Precision %	Average Recall %	Average Accuracy %	Average F-Measure %	FDR
LegalGNN [1]	90.22	92.45	91.34	91.32	0.10
LDRS [2]	90.78	98.89	94.84	94.66	0.10
ARTLR [3]	92.78	93.08	92.93	92.93	0.08
Proposed LELDR	**95.07**	**97.79**	**96.43**	**96.41**	**0.05**

In Table 1, it is indicated that the proposed LELDR has yielded the highest precision percentage of 95.07%, the highest average recall percentage of 97.79%, the highest accuracy percentage of 96.43%, the highest average F-measure percentage of 96.41%, and the lowest FDR of 0.05.

To assess the performance of the proposed LELDR framework, it is compared against baseline models, namely LegalGNN, LDRS, and ARTLR, respectively, which are also legal document or template recommendation frameworks. The LELDR outperforms all these models by a significant margin.

The LEDLR is a legal document framework which incrementally encompasses legal knowledge by synthesizing it or by using cognitive knowledge, which is available in the present structure of Web 3.0. It has generated legal ontologies as cognitive knowledge instances derived from datasets and integrates these with existing legal domain upper ontologies. These upper ontologies are widely accepted, community-contributed, and verified by legal domain expert. The integrated ontology is used to generate the knowledge graph using the Google KG API, which is crowdsourced, verified, and relies on knowledge contributions from the community. Subsequently, strategic models like LR as a classifier are used to classify the dataset, and K-L Divergence along with the Shannon-Wiener Index is used to compute the semantic similarity and relatedness This approach enables strategic quantitative semantic-oriented reasoning and learning through similarity measures. The ESA also adds to the semantic similarity computation strategy within the proposed framework. Subsequently, the presence of the DBNs to classify the legal e-books, case reports, judgement reports, legal expert reports, and the indexed metadata from the legal e-book glossaries constitutes the static knowledge repository, which not only adds to the amount and density of auxiliary knowledge but also makes it quite atomic. The presence of the DBN and the LR classifier provides a very strong infrastructure. Red Deer optimization with the Shannon-Wiener Index as the objective function helps in optimizing the framework. The meta-heuristic optimization strategy helps in yielding the most efficient solution from the intermediate solution, thereby yielding the best-in-class legal documents which are domain-specific.

The LegalGNN is a document recommendation framework which uses a very strong neural network, specifically the graph neural network, which has been enhanced in the proposed framework. This model uses unified legal content and structural representation to achieve feature fusion in a heterogeneous legal information network which connects the knowledge graph with contextual features. This results in a very strong enriched knowledge base supported by deep learning strategies. However, the unification of the enriched knowledge with the graph neural network doesn't happen because of the absence of a semantic bridge, semantic similarity computation, and an optimization strategy. Moreover, the auxiliary knowledge encompassing the model is limited as it comes from limited sources, and it also lacks a multi-source auxiliary knowledge encompassment strategy. These make the LegalGNN model less comprehensive as compared to the proposed framework.

The LDRS is a legal document recommendation system that uses cluster-based pairwise similarity computation methods, where a judgements addition network is built on clustering the judgements. The Doc2vec model helps in computing the semantic similarity. Both pairwise similarity and graph-based clustering are incorporated in the model. However, semantic similarity capabilities can be strengthened further. As the auxiliary knowledge integration is quite weak in the model, the learning mechanisms for them are also extensively absent. The learning infrastructure has to be strengthened, and hence, the LDRS model lacks when compared to the proposed framework.

The ARTLR automatically recommends templates for legal requirements. It uses NLP rules, where a list of requirements is a template for relevant legal recommendation. However, the model lacks a strong learning infrastructure and semantics-oriented reasoning is absent. An NLP rule-based infrastructure does not suffice well, and auxiliary knowledge encompassment is also quite sparse. Therefore, the performance of this framework may not fully meet expectations when compared to the proposed framework.

Compared to all the baseline models, the proposed framework has a very strong learning infrastructure in terms of LR and DBN classifier that classifies the dataset as well as the static knowledge repository. The repository respectively provides a very strong learning or classification infrastructure. The presence of ontology generation from the dataset combined with the integration of legal existing upper ontology to synthesize knowledge graphs and further application of ESA to DBN classified static knowledge store repositories helps in a very strong provision of auxiliary knowledge and also makes it more permeable into the model. The static knowledge repository includes legal e-books, glossary index metadata, case reports, judgement reports and legal expert reports, ensuring a very strong knowledge encompassment paradigm in the model, thereby narrowing down the semantic gap between the knowledge in the current web and the knowledge that permeates into the model. The DBN and the LR classifier put together give a strong classification infrastructure, and the Red Deer optimization with Shannon-Wiener Index helps in metaheuristic optimization. K-L divergence and Shannon-Wiener Index add to the improvement of quantitative semantics-oriented reasoning. Henceforth, the proposed model outdoes all the baseline models to serve as the best in-class approach.

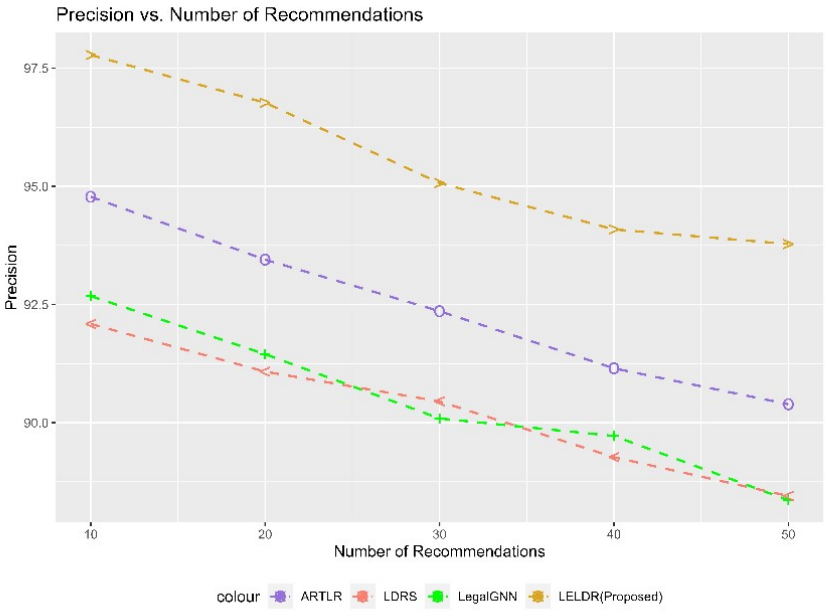

Figure 2: Line graph showing Precision Percentage Vs No. of Recommendations

Figure 2 illustrates the curve of precision percentage versus the distribution of the number of recommendations, which indicates that the LELDR occupies the top position in the graph, followed by ARTLR, while LegalGNN and LDRS occupy the lowest position in the hierarchy. The reason why the LELDR is the highest in the hierarchy is because the LELDR framework has a unique approach of incrementally synthesizing legal knowledge, integrating diverse ontologies, and utilizing strategic models like Logistic Regression and Deep Belief Networks. This comprehensive strategy, coupled with optimization using the Red Deer algorithm, results in a robust system tailored for legal document classification with a focus on semantic understanding. The LegalGNN doesn't perform as well because of its failure to effectively unify enriched legal knowledge with the graph neural network. The absence of a semantic bridge, semantic similarity computation, and a multi-source auxiliary knowledge strategy hinders its ability to connect diverse legal information networks, resulting in a limited and less comprehensive model compared to the proposed framework. The LDRS model underperforms due to its reliance on cluster-based pairwise similarity computation and graph-based clustering without adequate semantic strengthening. The model lacks a robust infrastructure due to weak auxiliary knowledge encompassment, limited incorporation of learning models, and an overall need for strengthening the semantic similarity computation. The ARTLR model's deficiency compared to the proposed framework arises from its reliance on NLP rules without a robust learning infrastructure and the absence of semantics-oriented reasoning. The sparse auxiliary knowledge it contains further limits its performance.

6. Discussion

The LELDR framework in its current form demonstrates significant performance while offering an efficient solution for legal document recommendation. However, it can be further enhanced by targeting several areas for future development. One such area for improvement is expanding the framework's generalizability across diverse legal jurisdictions and languages. While the current implementation performs well with specific datasets, the framework can be extended to handle legal documents from various legal systems, including civil and common law jurisdictions, and incorporate multilingual capabilities as well. This will enable the system to support a wider range of legal texts ensuring its effectiveness across different legal and linguistic contexts. Additionally, the potential biases in the auxiliary knowledge sources used by the framework need to be addressed. The system relies on legal case reports, expert annotations, and domain-specific glossaries, and hence there is a risk that these sources may unintentionally reflect certain legal biases or perspectives. Future work thus will have to include a comprehensive review and refinement of these knowledge sources to ensure that they represent diverse legal viewpoints. Furthermore, ways to mitigate the impact of errors in legal recommendations need to be explored, such as incorporating techniques like explainable AI (XAI) which provides transparency in the decision-making process and enable users to better understand the reason behind recommendations. This will help enhance the reliability and accountability of the system, particularly in high-stakes legal environments where the consequences of errors can be significant.

7. Conclusion

This paper puts forward a novel framework for legal document recommendation that integrates the legal upper-domain ontologies and synthesizes legal ontologies from the perspective of the dataset. The integration of ontologies and knowledge graph synthesis through the Google Knowledge Graph API and the encompassment of a standard static knowledge repository help meet the density of knowledge requirement for yielding the best-in-class recommendations through knowledge-centric reasoning. Classification of the static knowledge repositories is achieved using the DBNs and LR classifier that supports feature-controlled machine learning methods and classifies the legal document dataset. The Explicit Semantic Analysis and Shannon-Wiener Index encompassment in the proposed model, along with K-L divergence, helps in quantitative semantic-oriented reasoning through semantic relatedness computation. The metaheuristics are encompassed by the Red Deer Optimization algorithm, which serves as the best-in-class optimization strategy to yield the most optimal solution set. An overall recall of 97.79%, accuracy of 96.43%, and FDR of 0.05 is achieved by the proposed framework making it a leading-edge framework for legal document recommendation.

References

[1] J. Yang, W. Ma, M. Zhang, X. Zhou, Y. Liu, S. Ma, Legalgnn: Legal information enhanced graph neural network for recommendation, ACM Transactions on Information Systems 40 (2021) 1–29.

[2] J. Dhanani, R. Mehta, D. Rana, Legal document recommendation system: A cluster based pairwise similarity computation, Journal of Intelligent
Fuzzy Systems 41 (2021) 5497–5509.

[3] A. Sleimi, M. Ceci, M. Sabetzadeh, L. C. Briand, J. Dann, Automated recommendation of templates for legal requirements, in: IEEE International Requirements Engineering Conference, 2020, pp. 158–168. doi:10.1109/RE48521.2020.00027.

[4] A. Trivedi, A. Trivedi, S. Varshney, V. Joshipura, R. Mehta, J. Dhanani, Extracted summary based recommendation system for indian legal documents, in: International Conference on Computing, Communication and Networking Technologies, 2020, pp. 1–6.

[5] M. Zheng, B. Liu, L. Sun, Lawrec: Automatic recommendation of legal provisions based on legal text analysis, Computational Intelligence and Neuroscience (2022).

[6] M. Thomas, T. Vacek, X. Shuai, W. Liao, G. Sanchez, P. Sethia, T. Custis, Quick check: A legal research recommendation system, in: NLLP@ KDD, 2020, pp. 57–60.

[7] Y. Liu, X. Luo, X. Yang, Semantics and structure based recommendation of similar legal cases, in: IEEE International Conference on Intelligent Systems and Knowledge Engineering, 2019, pp. 388–395.

[8] D. Gerard, S. Vamsi, M. G. Siddharth, M. Ushasree, N. Ramanathan, P. Kesava, N. R. Santhanavijayan, Bi-ldr: A bi-classification model for legal document recommendation using knowledge synthesis approach, NeuroQuantology 20 (2022) 321.

[9] J. S. Dhani, R. Bhatt, B. Ganesan, P. Sirohi, V. Bhatnagar, Similar cases recommendation using legal knowledge graphs, arXiv preprint (2021). arXiv:2107.04771.

[10] N. Roopak, G. Deepak, Ontojudy: a ontology approach for content-based judicial recommendation using particle swarm optimisation and structural topic modelling, in: Data Science and Security: Proceedings of IDSCS 2021, Springer Singapore, 2021, pp. 203–213.

[11] G. K. Shankhdhar, V. K. Singh, M. Darbari, D. Yagyasen, P. Shukla, Legal semantic web-a recommendation system, International Journal of Applied Information Systems 7 (2014) 21–27.

[12] Q. Lu, J. G. Conrad, Bringing order to legal documents-an issue-based recommendation system via cluster association, in: International Conference on Knowledge Engineering and Ontology Development, volume 2, 2012, pp. 76–88.

[13] OpenDataLab, Dataset of legal documents, ???? URL: https://opendatalab.com/OpenDataLab/Dataset_of_Legal_Documents.

[14] GTS AI, Document classification for legal firms, ????. URL: https://gts.ai/case-study/document-classification-for-legal-firms/.

[15] GTS AI, Named entity recognition for legal documents, ????. URL: https://gts.ai/case-study/named-entity-recognition-for-legal-documents/.

[16] J. O. Rüdiger, Openlegaldata (2022 - corpus), Dataset (Text corpus), 2023. URL: https://live.european-language-grid.eu/catalogue/corpus/22980, version unspecified.