Contract Review and Legal Document Analysis using NLP

Kartik Badkas, Satheesh M K, Sarvesh Kulkarni, Guzzu Aditya, Saket Pitale

Abstract

Manual review of legal contracts is time-consuming, inconsistent, and error-prone. LegalDocNLP is a domain-specific Natural Language Processing (NLP) system that automates the review and understanding of legal documents. Leveraging transformer-based models like Legal-BERT and LegalT5, the system identifies standard contractual clauses, labels them (e.g., Termination, Indemnity), and generates clause-level summaries. It also extracts key parties, obligations, and keyword contexts. Built on Python and Flask, LegalDocNLP enables scalable, clause-aware contract analysis through a simple web interface. This paper presents the design, methodology, and experimental results of the system.

Keywords

Legal NLP, Clause Classification, Legal-BERT, Legal Contract Analysis, Text Summarization, Legal AI, Contract Automation

I. Introduction

Contracts govern the majority of business relationships, and legal teams are responsible for reviewing clauses related to termination, indemnity, confidentiality, etc. Traditional contract review is manual and often inconsistent. With the advent of domain-adapted NLP models like Legal-BERT and LegalT5, there is a significant opportunity to automate the analysis of legal contracts using machine learning.

This paper proposes LegalDocNLP — a clause-aware NLP tool that performs end-to-end legal clause classification and summarization in contract documents. The tool extracts parties, obligations, risky clauses, and contextual keywords using a combination of classification and generative models, improving the speed and reliability of legal review.

II. Related Work

CUAD (Contract Understanding Atticus Dataset) introduced clause-level annotations for commercial legal contracts and demonstrated transformer-based clause classification. Legal-BERT and LegalT5 are domain-specific transformer models pretrained on statutes, policies, and contract text. Previous tools like Kira and Lexion focus on clause retrieval but often lack open-source clause summarization or fine-grained classification capabilities.

III. Methodology

LegalDocNLP consists of three core components: A. PDF Processing: Contracts are parsed using PyMuPDF (fitz). Sentence tokenization is done via NLTK.

B. Clause Classification: Each sentence is passed through Legal-BERT for risky clause detection.

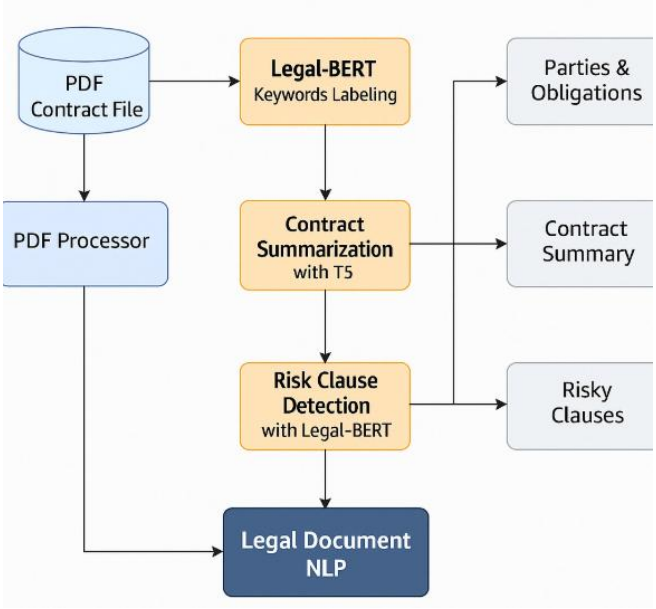
C. Clause Labeling: Rule-based keyword-to-label mappings.

D. Clause Summarization: Each risky clause is summarized using LegalT5.

E. Keyword Context Extraction: High-frequency keywords are shown with clause context.

IV. System Architecture

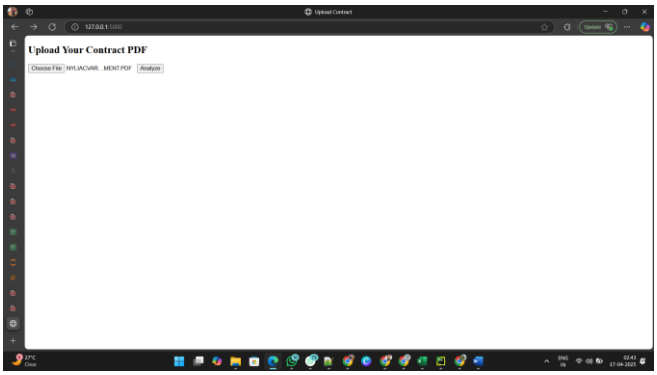
The system pipeline includes: PDF Upload → Clause Segmentation → Clause Classification → Labeling → Summarization → Context Extraction → Flask Display.

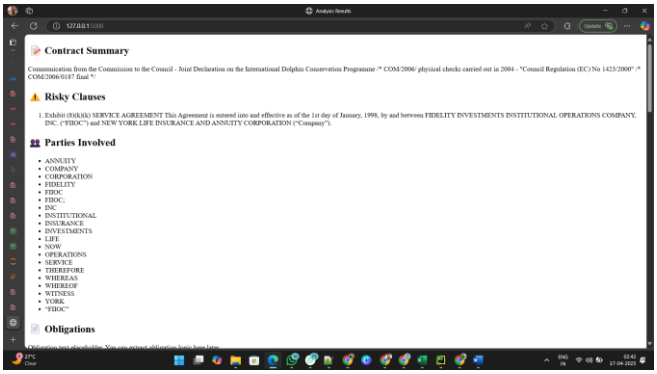


V. Experimental Results

The system was tested on multiple contracts from CUAD and commercial samples.

```
# Define your clause labels
LABELS = [
    "Confidentiality",
    "Indemnity",
    "Termination",
    "Payment Terms",
    "Governing Law",
    "Intellectual Property",
    "Arbitration",
    "Exclusivity",
    "Non-compete",
    "Other"
]
```





VI. Conclusion

LegalDocNLP automates clause-level analysis and summarization of legal contracts using modern NLP transformers. The tool streamlines contract understanding, reduces manual review effort, and provides consistent risk flagging and clause extraction.

Future enhancements include OCR support, clause comparison, contract template matching, and multi-language support.

References

1. CUAD Dataset – Contract Understanding Atticus Dataset, https://huggingface.co/datasets/cuad
2. Chalkidis, Ilias, et al. “Legal-BERT: The Muppets straight out of Law School.” Findings of EMNLP, 2020.
3. Bhattacharya, Pawan, et al. “LegalT5: Pretraining a Text-to-Text Transformer for Indian Legal Text.” arXiv:2205.13572, 2022.
4. Huggingface Transformers, https://huggingface.co
5. PyMuPDF – https://pymupdf.readthedocs.io