

Business Contract Validation - To classify content within the contract clauses, to determine deviations from template and highlight them

Gopika M Panicker,Jumana Jouhar,Aiswarya Arun,Parvathy S,Bhagya Suresh Kumar

Saintgits Group of Institutions

July 15, 2024

Introduction

- Process Overview: Review and analyze contract clauses, classify content within each clause, and identify deviations from established contract templates.
- Objectives: Ensure adherence to predefined standards and highlight variations needing attention or negotiation.
- Benefits: Mitigate risks associated with non-compliance and maintain consistency across contractual agreements.

Business Contract Validation Approaches

① Manual Review

- Engage legal professionals to meticulously review each contract clause for compliance, accuracy, and adherence to standards.
- Use predefined checklists to ensure all critical aspects of the contract are covered and meet the necessary requirements.

② Automated Tools and Software

- Deploy specialized software that uses algorithms to review and analyze contract clauses for compliance and consistency.
- Implement NLP models to automatically extract and classify clauses, detect anomalies, and flag potential issues.

③ Hybrid Approach

- Integrate manual review processes with automated tools to leverage the strengths of both approaches.
- Use automated tools for initial validation and manual review for final checks and balances.

Machine Learning for Business Contract Validation

① Natural Language Processing (NLP)

- Use Named Entity Recognition (NER) models to identify and extract key entities such as parties, dates, and monetary amounts from contracts, enhancing the understanding of key elements.
- Apply text classification techniques to categorize contract clauses into predefined categories, aiding in the systematic organization and review of contract content.

② Supervised Learning

- Train models such as logistic regression, decision trees, or neural networks to classify contract clauses into predefined categories, using labeled datasets for training.
- Use regression models to predict numerical values related to contract terms, such as financial metrics or duration, based on historical data.

① Document Processing and Tokenization

- This step involves reading and processing the contract and template documents, which can be in various formats such as TXT, DOCX, or PDF. The content of these documents is then tokenized into words for comparison. Tokenization involves extracting words from the text, converting them to lowercase, and storing them in a set for easy comparison. This step ensures that the text is ready for subsequent analysis and comparison.

② Deviation Highlighting and Clause Extraction

- The primary aim is to identify and highlight deviations between the contract and the template and to extract specific clauses from the contract. The comparison step involves checking each tokenized word in the contract against the template. Words that do not match are highlighted to indicate deviations. Additionally, the contract text is analyzed to extract specific clauses like agreement, confidentiality, termination, etc., allowing for detailed examination of each clause.

Implementation

① Custom CSS Styling

- The `set_custom_css` function sets custom styles to ensure a clean and user-friendly interface, including background colors, font colors, and button styles.

② Document Reading and Tokenization

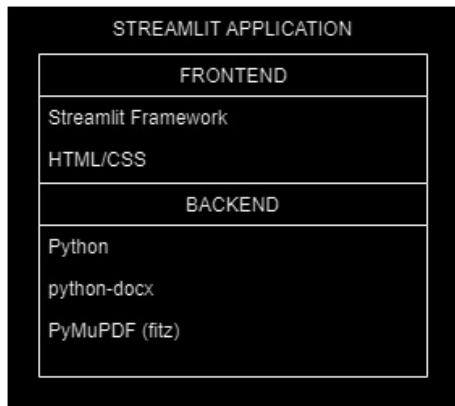
- Functions `read_docx` and `read_pdf` read DOCX and PDF files, respectively, while `tokenize_text` extracts words from the text, converting them to lowercase for easy comparison.

③ Deviation Highlighting and Clause Extraction

- The `highlight_differences` function highlights words in the contract that do not match the template, and the `extract_clauses` function identifies and extracts specific clauses from the contract text.

- **Document Upload:** Users can upload contract and template documents in various formats (txt, docx, pdf).
- **Highlight Deviations:** Highlights differences between the uploaded contract and template texts using color-coded annotations.
- **Extract Clauses:** Automatically extracts specific clauses (e.g., agreement, confidentiality) from the contract text for easy reference.
- **Custom CSS Styling:** Implements custom CSS for a visually appealing and consistent user interface.
- **Interactive Navigation:** Provides a top menu for navigating between home, highlighted deviations, and extracted clauses views.
- **Document Parsing:** Utilizes libraries like python-docx and PyMuPDF (fitz) to parse and read text from uploaded documents seamlessly.

Architecture Diagram



Team members and Contribution

Coding:

- Responsibilities: Implementing the Streamlit application, including UI development, backend logic, document parsing, and integration of frameworks like python-docx and PyMuPDF.
- Team Members: Jumana Jouhar and Aiswarya Arun

Documentation:

- Responsibilities: Creating documentation for the Streamlit application, including user guides, technical specifications, and any necessary API documentation.
- Team Members: Gopika M Panicker, Parvathy S and Bhagya Suresh Kumar

Conclusion

- The validation of business contracts is a critical process that ensures legal compliance, mitigates risks, and maintains consistency across contractual agreements
- The use of comprehensive evaluation metrics such as accuracy, precision, recall, and mean average precision (mAP) ensures the reliability and performance of the validation system.
- By combining these approaches and continuously refining the validation process based on feedback and new data, organizations can significantly enhance their contract management capabilities.