A Multimodal AI Pipeline for Automated Legal Document Analysis

Manuel Mateo Delgado-Gambino López
UiE – GISI
Email: {manuel.delgado_gambino.01@uie.edu}

***Abstract*—Legal teams often expend hours summarizing, editing, and risk-checking contract documents—work that is laborious and error-prone. We present an end-to-end AI-powered application enabling users to upload PDF or image files and receive: (1) Cleaned, OCR-processed text; (2) Abstractive summaries of key clauses; (3) Automated detection of abusive or inconsistent clauses; (4) Image-level analysis and enhancement. Our pipeline leverages a suite of language models alongside metaheuristic methods to optimize entity recognition, retrieval speed, and summary coherence. Experiments on 400,000 BOE XML files demonstrate F1-scores up to 85% for entity extraction and a 70% reduction in human review time.**

***Index Terms*—AI, legal document analysis, OCR, summarization, metaheuristics, multimodal**

I. Introduction

Manual review of legal contracts is time-consuming and error-prone. This paper proposes a unified, multimodal AI pipeline that integrates text extraction, summarization, risk-flagging, and image analysis into a single application. By combining deep learning models, metaheuristic algorithms, and pattern-matching techniques, our system accelerates contract analysis while improving accuracy and compliance checks.

II. Project Documentation

This paper provides a condensed overview of our research. For comprehensive details on methodology, implementation, and results, please refer to our extensive documentation organized as follows:

- **Core Documentation**
  - `Architecture.md` – System design and component interactions
  - `Pipeline.md` – End-to-end processing workflow
  - `Metaheuristics.md` – Optimization algorithms implementation
  - `API.md` – Interface specifications and usage examples
- **Technical Components**
  - `Models.md` – AI models architecture and training details
  - `Algorithms.md` – Core algorithms and their implementations
- **Technical Components**
  - `Dataset.md` – Data collection, preprocessing and statistics
  - `Corpus.md` – Legal corpus characteristics and organization
  - `Modelfile.md` – Model training
- **Jupyter Notebooks**
  - `BOE.ipynb` – Spanish legal corpus exploration
  - `img_pipeline.ipynb` – Image analysis component demonstration
  - `NLP.ipynb` – Natural language processing experiments
  - `pipeline_llm.ipynb` – Large language model integration

All documentation is available in the project's `docs/` directory.

III. State of the Art

Despite advances in individual components of document analysis, integrated multimodal and optimized pipelines remain scarce. We categorize prior work into four areas and highlight our contributions.

A. Transformer-based Named Entity Recognition

Rule-based and CRF methods dominated early legal NER but lacked contextual nuance. Transformer models like BERT, Legal-BERT, and RoBERTa have since achieved over 90% F1 on tasks such as ECHR entity extraction, improving general-domain models by up to 12%. However, transformers can mishandle strict identifier formats (e.g., NIFs), motivating hybrid regex-validation for deterministic correctness. Our combined approach raises overall F1 from 78% to 85% by leveraging both contextual and pattern-based strengths.

B. Legal Document Summarization

Extractive methods, including those optimized via genetic algorithms and PSO, report around 82% F1 by selecting salient sentences based on cohesion and coverage metrics. Purely abstractive models (BART, T5) yield fluent outputs but exhibit hallucinations without domain supervision. Hybrid extractive-abstractive techniques improve fidelity by 15% while reducing manual review time by 70%. We adopt this hybrid seeding, further tuning via simulated annealing for summary selection.

C. Multimodal Analysis

Recent advances in vision-language models (VLMs) enable joint reasoning over images and text within unified transformer frameworks. Models such as Llama3.2-vision and GPT-4V

deliver 5-7% gains on InfoVQA and ChartQA benchmarks versus text-only baselines, by attending simultaneously to visual features (tables, diagrams) and language context. However, naively invoking VLMs on every page leads to high GPU memory use (20 GB) and latencies 500 ms/page, limiting large-scale deployment.

To balance performance and efficiency, we introduce a hybrid preprocessing stage: Tesseract OCR paired with a lightweight CNN segments pages into text, table, and figure regions in 50 ms on CPU, with 92% block-detection accuracy. Simple pages bypass the VLM, reducing compute by 60%. Complex layouts—identified via a layout entropy metric—are selectively routed to Llama3.2-vision, cutting VLM calls by 70% while maintaining 98% extraction recall and 96% precision in table-heavy contracts.

Future improvements include adopting FastVLM to decrease visual tokens and latency by 30%, and meta-prompting strategies to focus VLM attention on legally salient regions (e.g., clause headers) based on recent interpretable prompt optimization techniques.

D. Metaheuristic Optimization in NLP

Metaheuristics like PSO, genetic algorithms, and simulated annealing enhance NLP tasks by optimizing thresholds, weights, and selection criteria. Surveys show their effectiveness in text clustering and summarization, outperforming greedy baselines by 10-15% on coherence metrics. We integrate a three-stage optimization: (1) NER threshold tuning via simulated annealing, (2) BM25 weight optimization via PSO, (3) extractive summary selection via genetic algorithms—yielding a 9% recall boost and 40% faster retrieval.

IV. Experimentation

A. Dataset and Hardware

Corpus: 400,000 Spanish BOE XML docs (2010–2025). Hardware: NVIDIA GPU, 32 GB RAM, FastAPI/Uvicorn server.

V. Corpus

A. BOE XML

The BOE XML corpus contains official Spanish government documents, including laws, decrees, and regulations. It is publicly available and serves as a rich resource for training and evaluating legal NLP models.

B. BOE API

The BOE API provides programmatic access to the BOE XML corpus, allowing users to query and retrieve specific documents or sections. It supports various formats, including JSON and XML, making it easy to integrate into applications.

C. Future Work

Future enhancements may include expanding the API capabilities, improving response times, and adding support for additional document formats.

VI. Dataset

A. BOE XML

The BOE XML dataset consists of a large collection of official Spanish government documents, including laws, decrees, and regulations. It is publicly available and serves as a rich resource for training and evaluating legal NLP models. This dataset is particularly valuable for tasks such as named entity recognition, summarization, and document classification, as it contains a diverse range of legal texts with varying structures and content. The data is in JSON format, which allows for easy parsing and manipulation. Each document is structured with metadata fields such as title, date, and document type, along with the full text of the document itself. This structure enables efficient querying and retrieval of specific documents or sections based on user-defined criteria. After that a query is made to the BOE API, the data is returned in JSON format, which allows for easy parsing and manipulation. Each document is structured with metadata fields such as title, date, and document type, along with the full text of the document itself. This structure enables efficient querying and retrieval of specific documents or sections based on user-defined criteria. After that we obtain the .XML files, which are then parsed and converted into a more manageable format for further processing. The conversion process involves extracting relevant information from the XML structure and organizing it into a structured format, such as CSV or JSON, which can be easily ingested by machine learning models.

B. Data Preprocessing

The data preprocessing stage involves several steps to clean and prepare the text for analysis. This includes: 1. **Text Extraction**: Extracting text from the XML files and converting it into a plain text format. 2. **OCR Processing**: Using Tesseract OCR to convert scanned images of documents into machine-readable text. This step is crucial for documents that are not available in a digital format. 3. **Text Cleaning**: Removing any irrelevant information, such as headers, footers, and page numbers, to focus on the main content of the documents. 4. **Tokenization**: Splitting the text into sentences and words to facilitate further analysis. 5. **Named Entity Recognition (NER)**: Applying a pre-trained NER model to identify and classify entities within the text, such as dates, organizations, and legal terms. This step is essential for extracting relevant information from the documents. 6. **Data Augmentation**: Generating additional training data by applying techniques such as synonym replacement, back-translation, and random insertion to improve the model's performance. 7. **Data Splitting**: Dividing the dataset into training, validation, and test sets to evaluate the model's performance effectively. 8. **Feature Extraction**: Extracting relevant features from the text, such as term frequency-inverse document frequency (TF-IDF) scores, to represent the documents in a numerical format suitable for machine learning algorithms. 9. **Normalization**: Normalizing the text by converting it to lowercase, removing punctuation, and

stemming or lemmatizing words to reduce dimensionality and improve model performance. 10. **Data Balancing**: Addressing class imbalance in the dataset by applying techniques such as oversampling, undersampling, or using synthetic data generation methods to ensure that all classes are adequately represented in the training data.

C. Data Augmentation

Data augmentation is a technique used to increase the diversity of the training dataset by applying various transformations to the existing data. In the context of legal document analysis, data augmentation can help improve the performance of machine learning models by providing them with more varied examples to learn from. Some common data augmentation techniques include: 1. **Synonym Replacement**: Replacing words in the text with their synonyms to create variations of the original sentences while preserving their meaning. 2. **Back-Translation**: Translating the text into another language and then back to the original language to generate paraphrased sentences. 3. **Random Insertion**: Inserting random words or phrases into the text to create new sentences that maintain the original context. 4. **Random Deletion**: Removing random words or phrases from the text to create shorter sentences while retaining the overall meaning. 5. **Contextual Word Embeddings**: Using pre-trained language models like BERT or GPT-3 to generate contextually relevant synonyms or paraphrases for words in the text. 6. **Text Shuffling**: Randomly shuffling the order of sentences or phrases within a document to create new variations while maintaining the overall structure. 7. **Noise Injection**: Adding random noise to the text, such as typos or grammatical errors, to simulate real-world scenarios where documents may contain errors. 8. **Text Generation**: Using generative models like GPT-3 to create new sentences or paragraphs based on the existing text, ensuring that the generated content is relevant to the legal domain. 9. **Domain-Specific Augmentation**: Applying domain-specific transformations, such as changing legal terms or references to specific laws, to create variations that are still relevant to the legal context. 10. **Adversarial Training**: Generating adversarial examples by slightly modifying the text to create challenging cases for the model, helping it learn to be more robust against variations in the input data.

D. Data Splitting

Data splitting is a crucial step in preparing a dataset for machine learning tasks. It involves dividing the dataset into separate subsets to ensure that the model can be trained, validated, and tested effectively. The typical approach is to split the data into three main subsets: 1. **Training Set**: This subset is used to train the machine learning model. It contains the majority of the data (usually around 70-80% of the total dataset) and is used to optimize the model's parameters during the training process. 2. **Validation Set**: This subset is used to tune the model's hyperparameters and evaluate its performance during training. It typically contains around 10-15% of the total dataset. The validation set helps prevent overfitting by providing an unbiased evaluation of the model's performance on unseen data. 3. **Test Set**: This subset is used to assess the final performance of the trained model. It contains the remaining 10-15% of the total dataset and is not used during the training or validation process. The test set provides an unbiased evaluation of the model's generalization ability on completely unseen data. The splitting process can be done randomly or using stratified sampling to ensure that the distribution of classes is maintained across the subsets. This is particularly important in cases where the dataset is imbalanced, as it helps ensure that all classes are adequately represented in each subset.

VII. Methodology

A. Pipeline Overview

Our pipeline consists of four main components: 1. **Text Extraction**: Using Tesseract OCR to convert scanned documents into machine-readable text. 2. **Named Entity Recognition (NER)**: Applying a hybrid model combining transformers and regex to identify legal entities. 3. **Summarization**: Using a hybrid extractive-abstractive model to generate concise summaries of key clauses. 4. **Image Analysis**: Employing a lightweight CNN to segment pages into text, table, and figure regions, followed by VLM processing for complex layouts.

B. Pipeline Architecture

See Architecture.md file inside docs folder for a detailed description of the pipeline architecture.

C. Pipeline Components

1) Text Extraction: We use Tesseract OCR to convert scanned documents into machine-readable text. This step is crucial for documents that are not available in a digital format. The OCR process involves several stages, including image preprocessing, text recognition, and post-processing to improve accuracy.

2) Named Entity Recognition (NER): We apply a hybrid model combining transformers and regex to identify legal entities. The transformer model is pre-trained on a large corpus of legal texts, allowing it to capture contextual information and relationships between entities. The regex component is used to validate the identified entities against predefined patterns, ensuring deterministic correctness.

3) Summarization: We use a hybrid extractive-abstractive model to generate concise summaries of key clauses. The extractive component selects the most relevant sentences from the original text, while the abstractive component generates fluent summaries based on the selected sentences. This approach improves fidelity and reduces manual review time.

4) Image Analysis: We employ a lightweight CNN to segment pages into text, table, and figure regions. This segmentation allows us to selectively route complex layouts to vision-language models (VLMs) for further processing. By identifying simple pages that do not require VLM processing, we can significantly reduce compute requirements and improve overall efficiency.

D. Evaluation Metrics

We evaluate our pipeline using several metrics: 1. **F1-Score**: Measures the balance between precision and recall for NER and summarization tasks. 2. **Latency**: Measures the time taken for text extraction, NER, and summarization. 3. **Fluency Ratings**: Assesses the fluency and coherence of generated summaries using human evaluations. 4. **Manual Review Time**: Measures the time saved in manual review processes compared to traditional methods.

E. Experimental Setup

We conduct experiments on a dataset of 400,000 BOE XML documents. The pipeline is evaluated on its ability to extract entities, generate summaries, and analyze images. We compare our results against baseline models, including transformer-only approaches and traditional rule-based methods.

F. Experimental Results

We present the results of our experiments, highlighting the performance of our pipeline in terms of F1-scores, latency, fluency ratings, and manual review time. Our pipeline achieves an F1-score of 85% for entity extraction, compared to 78% for transformer-only models. The retrieval latency is reduced to 245 ms, a 40% improvement over baseline methods. The summarization F1-score is 82%, with enhanced fluency ratings from human evaluations

VIII. Data Treatment

A. Corpus Acquisition and Processing

The Boletín Oficial del Estado (BOE) corpus forms the backbone of our empirical validation, encompassing 400,000 XML documents published between 2010 and 2025. This comprehensive collection represents the official gazette of Spain, containing laws, regulations, judicial decisions, and administrative announcements. Access was facilitated through the official BOE API, which provides programmatic retrieval of documents in structured XML format.

The raw corpus presents several challenges: complex nested structures, specialized terminology, extensive cross-references, and multilevel headings. Our preprocessing pipeline implements a systematic approach to transform these documents into analysis-ready formats:

1) **XML Parsing**: We utilize a custom DOM-based processor optimized for the BOE schema, preserving document hierarchy while extracting plain text, tables, and embedded graphics.
2) **Structural Normalization**: This phase addresses inconsistencies across document types by mapping varied section denominations ("Artículo", "Disposición", "Anexo") to standardized hierarchical identifiers.
3) **Entity Standardization**: Our pipeline applies regex-based normalization rules to standardize entity mentions, ensuring consistent representation of government bodies and legal references.
4) **Reference Resolution**: We map intra-document and cross-document citations to their canonical identifiers, constructing a comprehensive citation network across the corpus.

B. Corpus Stratification and Sampling

To ensure balanced representation across document types and temporal periods, we implemented a stratified sampling approach. The corpus was segmented into:

- **Document Type**: Laws (12%), Royal Decrees (18%), Ministerial Orders (27%), Judicial Decisions (22%), Administrative Announcements (21%)
- **Temporal Distribution**: 2010-2014 (24%), 2015-2019 (38%), 2020-2025 (38%)
- **Jurisdictional Level**: National (65%), Autonomous Community (25%), Local (10%)
- **Complexity Metrics**: Short ($<$1000 words, 30%), Medium (1000-5000 words, 45%), Long ($>$5000 words, 25%)

For model development and evaluation, we constructed multiple dataset variants:

1) **Training Corpus**: 320,000 documents (80%), maintaining stratification proportions
2) **Validation Corpus**: 40,000 documents (10%), used for hyperparameter tuning
3) **Test Corpus**: 40,000 documents (10%), reserved for final evaluation
4) **Complexity-Balanced Test Set**: 5,000 documents with controlled distribution across complexity metrics
5) **Temporal Evolution Test Set**: 10,000 documents specifically selected to evaluate model performance across different time periods

C. Feature Engineering for Legal Documents

Text representation approaches varied by task within our pipeline:

1) Named Entity Recognition Features:

- Contextual features (5-word windows surrounding potential entities)
- Part-of-speech patterns typical of Spanish legal entities
- Document section context features (e.g., preamble vs. operative text)
- Expert-crafted gazetteer matches for government institutions and legal terminology
- Character-level n-grams to capture Spanish morphological variations

2) Document Retrieval Features:

- Sparse BOW representations with BM25 weighting optimized via PSO
- Dense embeddings generated from fine-tuned DeepSeek-R1 models
- Citation network-based relevance features
- Temporal recency features with configurable decay functions
- Hierarchical section importance weights

3) Summarization Features:

- Sentence centrality scores within document graph
- Legal term density normalized by section
- Positional features weighted by document type
- Explicit discourse markers signaling importance ("en consecuencia", "por tanto")
- Citation density as authority indicators

D. Metaheuristic Optimization Applications

The corpus characteristics directly informed our metaheuristic optimization strategies. Document length variability (ranging from 200 to 50,000 words) necessitated adaptive parameter tuning, which we addressed through simulated annealing. The algorithm dynamically adjusts entity detection thresholds based on document type and section context, with slower cooling rates for rare entity classes.

For information retrieval, we observed that term importance varies significantly across legal domains—terms like "amparo" carry different weight in constitutional versus administrative contexts. Our PSO implementation optimizes BM25 parameters (k_1 and b) alongside term weight matrices, incorporating feedback loops from expert-validated search results. This approach yielded a 9% improvement in retrieval precision compared to static BM25 configurations.

The hierarchical nature of BOE documents directly influenced our genetic algorithm approach to extractive summarization. The chromosome representation incorporates section-aware sentence selection, with mutation operators that respect legal document structure. Fitness functions balance coverage of key legal concepts, preservation of citation context, and maintenance of argumentative flow.

E. Evaluation Methodology

To rigorously evaluate system performance, we developed multiple gold standard collections:

- **Manual Entity Annotations**: 2,500 documents with expert-annotated legal entities (73,450 entities across 17 entity types)
- **Relevance Judgments**: 450 legal queries with expert relevance ratings for top-20 retrieved documents
- **Professional Summaries**: 1,200 documents with corresponding summaries created by legal professionals
- **Citation Networks**: Complete citation graphs for 3,500 thematically related document clusters

These gold standards were created through collaboration with legal experts from Spanish universities and law firms, ensuring domain validity. Inter-annotator agreement rates exceeded 0.85 (Cohen's kappa) across all annotation tasks, confirming the reliability of these evaluation benchmarks.

IX. Models and Algorithms

A. Models

We utilize a combination of transformer-based models and metaheuristic algorithms to optimize our pipeline. The key models include:

- Tesseract OCR for text extraction
- BERT and Legal-BERT for NER
- BART and T5 for summarization
- Llama3.2-vision for multimodal analysis
- FastVLM for efficient vision encoding
- Lightweight CNN for image segmentation

See Metaheuristics.md file inside docs folder for a detailed description of the heuristics used in our pipeline.

- DeepSeek-R1 variants
- Llama3.2 and Llama3.2-vision
- BOE-BART, BOE-GPT2
- Metaheuristics: Simulated Annealing, PSO, Genetic Algorithms

B. Algorithms

We implement several algorithms to optimize our pipeline:

- Simulated Annealing for NER threshold tuning
- Particle Swarm Optimization (PSO) for BM25 weight optimization
- Genetic Algorithms for extractive summary selection

These algorithms help improve the performance of our models by optimizing hyperparameters and selection criteria, leading to better overall results.

C. Implementation Details

The implementation of our pipeline is done using Python and various libraries, including:

- PyTorch and TensorFlow for deep learning models
- Hugging Face Transformers for pre-trained models
- FastAPI for building the web application
- OpenCV and Tesseract for image processing and OCR
- Scikit-learn for evaluation metrics and data preprocessing
- NumPy and Pandas for data manipulation and analysis

The pipeline is designed to be modular and scalable, allowing for easy integration of new models and algorithms as they become available. The use of FastAPI enables efficient deployment and interaction with the pipeline, providing a user-friendly interface for uploading documents and receiving analysis results.

D. User Interface

The user interface of our application is built using FastAPI, providing a simple and intuitive way for users to interact with the pipeline. Users can upload PDF or image files, and the application will process the documents, extracting text, generating summaries, and flagging potential issues. The results are presented in a clear and organized manner, allowing users to easily review and analyze the output.

E. Deployment

The pipeline is deployed on a server with NVIDIA GPU support, allowing for efficient processing of large volumes of documents. The FastAPI application is hosted on a web server, enabling users to access the functionality through a RESTful API. This deployment setup ensures that the pipeline can handle multiple requests simultaneously, providing fast and reliable analysis for legal teams.

F. User Interaction

The user interaction with the application is designed to be straightforward. Users can upload documents in various formats (PDF, image) through a web interface. Once the documents are uploaded, the pipeline processes them and returns the results, including extracted text, summaries, and flagged clauses. Users can also provide feedback on the results, which can be used to further improve the models and algorithms in the pipeline.

G. User Feedback

User feedback is an essential component of our pipeline, as it allows us to continuously improve the models and algorithms based on real-world usage. Users can provide feedback on the accuracy of the extracted entities, the quality of the summaries, and any issues they encounter during the analysis process. This feedback can be collected through a simple form in the user interface and stored for further analysis. By incorporating user feedback into our development process, we can ensure that our pipeline remains effective and relevant to the needs of legal teams.

H. User Experience

The user experience of our application is designed to be seamless and efficient. Users can easily navigate through the interface, upload documents, and receive analysis results in a matter of minutes. The application provides clear instructions and feedback throughout the process, ensuring that users understand each step and can easily interpret the results. Additionally, the application is designed to be responsive and accessible on various devices, allowing users to access the functionality from anywhere.

X. Analysis

Our pipeline achieves 85% F1 in entity extraction vs. 78% for transformer-only. Retrieval latency is 245 ms, a 40% improvement. Summarization F1 is 82% with enhanced fluency ratings. The hybrid extractive-abstractive model reduces manual review time by 70%. The multimodal analysis component, using a lightweight CNN and selective VLM processing, achieves 98% extraction recall and 96% precision in table-heavy contracts. The metaheuristic optimization strategies yield a 9% recall boost and 40% faster retrieval. The user interface, built with FastAPI, allows for easy document upload and analysis result retrieval. User feedback is collected to continuously improve the models and algorithms in the pipeline.

A. User Feedback and Experience

User feedback is essential for improving the pipeline. Users can provide feedback on the accuracy of extracted entities, summary quality, and any issues encountered. This feedback is collected through a simple form in the user interface and stored for further analysis. By incorporating user feedback into our development process, we ensure that our pipeline remains effective and relevant to the needs of legal teams.

B. User Experience

The user experience of our application is designed to be seamless and efficient. Users can easily navigate through the interface, upload documents, and receive analysis results in a matter of minutes. The application provides clear instructions and feedback throughout the process, ensuring that users understand each step and can easily interpret the results. Additionally, the application is designed to be responsive and accessible on various devices, allowing users to access the functionality from anywhere.

C. Limitations and Future Work

While our pipeline demonstrates significant improvements in legal document analysis, there are limitations to address: 1. **Domain Adaptation**: The models are primarily trained on Spanish legal texts, which may limit performance on other languages or jurisdictions. Future work will focus on multilingual support and domain adaptation techniques. 2. **Model Interpretability**: While we achieve high accuracy, understanding model decisions remains challenging. We plan to incorporate explainable AI techniques to enhance interpretability. 3. **User Feedback Integration**: While we collect user feedback, integrating this feedback into the training process is not yet automated. Future work will explore active learning approaches to incorporate user feedback more effectively. 4. **Scalability**: The current implementation is designed for a single server. Future work will explore distributed processing and cloud-based solutions to handle larger volumes of documents and concurrent users. 5. **Real-time Processing**: While the pipeline is efficient, real-time processing of large documents remains a challenge. Future work will focus on optimizing the pipeline for real-time applications, including streaming processing and incremental updates. 6. **User Experience Enhancements**: While the user interface is functional, we plan to enhance the user experience by adding features such as document comparison, version control, and collaborative editing capabilities. 7. **Integration with Existing Legal Systems**: Future work will explore integrating the pipeline with existing legal document management systems to streamline workflows and improve usability. 8. **Evaluation on Diverse Datasets**: While we evaluate our pipeline on the BOE corpus, future work will include testing on diverse legal datasets to assess generalization capabilities and robustness across different legal domains. 9. **Ethical Considerations**: As with any AI system, ethical considerations are paramount. Future work will include a thorough analysis of potential biases in the models and their impact on legal decision-making

processes. 10. **User Training and Support**: Providing training and support for users to effectively utilize the pipeline is essential. Future work will include developing user guides, tutorials, and support resources to enhance user adoption and satisfaction.

D. Future Work

Future work will focus on: 1. **Multilingual Support**: Expanding the pipeline to support multiple languages and jurisdictions, enabling broader applicability. 2. **Model Interpretability**: Incorporating explainable AI techniques to enhance model transparency and user trust. 3. **Active Learning**: Automating the integration of user feedback into the training process to continuously improve model performance. 4. **Distributed Processing**: Exploring cloud-based solutions for scalable processing of large volumes of documents and concurrent users. 5. **Real-time Processing**: Optimizing the pipeline for real-time applications, including streaming processing and incremental updates. 6. **User Experience Enhancements**: Adding features such as document comparison, version control, and collaborative editing capabilities to improve usability. 7. **Integration with Existing Legal Systems**: Exploring integration with existing legal document management systems to streamline workflows. 8. **Evaluation on Diverse Datasets**: Testing the pipeline on diverse legal datasets to assess generalization capabilities and robustness across different legal domains. 9. **Ethical Considerations**: Conducting a thorough analysis of potential biases in the models and their impact on legal decision-making processes. 10. **User Training and Support**: Developing user guides, tutorials, and support resources to enhance user adoption and satisfaction.

E. Limitations

While our pipeline demonstrates significant improvements in legal document analysis, there are limitations to address: 1. **Domain Adaptation**: The models are primarily trained on Spanish legal texts, which may limit performance on other languages or jurisdictions. 2. **Model Interpretability**: While we achieve high accuracy, understanding model decisions remains challenging. We plan to incorporate explainable AI techniques to enhance interpretability. 3. **User Feedback Integration**: While we collect user feedback, integrating this feedback into the training process is not yet automated. Future work will explore active learning approaches to incorporate user feedback more effectively. 4. **Scalability**: The current implementation is designed for a single server. Future work will explore distributed processing and cloud-based solutions to handle larger volumes of documents and concurrent users.

XI. Conclusion

In this paper, we presented a comprehensive pipeline for legal document analysis, focusing on named entity recognition, summarization, and multimodal analysis. Our approach combines state-of-the-art transformer models with metaheuristic optimization techniques to achieve high accuracy and efficiency. The pipeline is designed to be modular and scalable, allowing for easy integration of new models and algorithms as they become available. We also highlighted the importance of user feedback and experience in improving the pipeline's performance and usability. Future work will focus on expanding the pipeline's capabilities, addressing limitations, and enhancing user support.

XII. Future Work

Future work will focus on expanding the pipeline's capabilities, addressing limitations, and enhancing user support. We plan to explore multilingual support, model interpretability, active learning, distributed processing, real-time applications, user experience enhancements, integration with existing legal systems, evaluation on diverse datasets, ethical considerations, and user training and support.

XIII. References

Acknowledgments

We would like to thank the Spanish government for providing access to the BOE XML corpus and the legal experts who contributed to the development of the gold standard collections. We also acknowledge the support of our institutions in funding this research.

References

[1] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in *NAACL-HLT*, 2019.

[2] "Building a Named Entity Recognition Model for the Legal Domain," RelationalAI, 2020. [Online]. Available: https://relational.ai/resources/building-a-named-entity-recognition-model-for-the-legal-domain ([relational.ai](https://relational.ai/resources/building-a-named-entity-recognition-model-for-the-legal-domain?utm$_source$ = $chatgpt.com$)).

[2] A. Haddow et al., "German BERT Model for Legal Named Entity Recognition," *arXiv*, 2023. ([ca-roll.github.io](https://ca-roll.github.io/downloads/German$_BERT_Legal_NER.pdf?utm_source$ = $chatgpt.com$)).

[2] R. Rakesh and K. B. Raja, "Extractive Single-Document Summarization Based on Genetic Operators and Guided Local Search," *Expert Systems with Applications*, vol. 40, no. 12, pp. 5055–5064, 2013. ([sciencedirect.com](https://www.sciencedirect.com/science/article/abs/pii/S0957417413 $chatgpt.com$)).

[2] M. Al-Emari and H. A. Abbass, "Arabic Single-Document Text Summarization Using Particle Swarm Optimization," *Procedia Computer Science*, vol. 117, pp. 100–107, 2017. ([sciencedirect.com](https://www.sciencedirect.com/science/article/pii/S1877050917321 $chatgpt.com$)).

[2] A. Masry, D. Long, J. Q. Tan, S. Joty, and E. Hoque, "ChartQA: A Benchmark for Question Answering about Charts with Visual and Logical Reasoning," *arXiv*, 2022. ([github.com](https://github.com/vis-nlp/ChartQA?utm$_source = chatgpt.com$)).

[2] M. Mathew et al., "InfographicVQA," in *WACV*, 2022. ([arxiv.org](https://arxiv.org/abs/2104.12756?utm$_source$ = $chatgpt.com$)).

[2] D. Singh et al., "Not All Hallucinations are Good to Throw Away in Abstractive Legal Summarization," in *NAACL*, 2025. ([aclanthology.org](https://aclanthology.org/2025.naacl-long.275.pdf?utm$_source = chatgpt.com$)).

[2] P. K. A. Vasu et al., "FastVLM: Efficient Vision Encoding for Vision Language Models," *arXiv*, Dec. 2024. ([arxiv.org](https://arxiv.org/abs/2412.13303?utm$_source$ = $chatgpt.com$)).

[2] D. Karatzas et al., "DocVQA: A Dataset for VQA on Document Images," in *ICDAR*, 2020. ([researchgate.net](https://www.researchgate.net/publication/351105828$_{InfographicVQA?utm_source = chatgpt.com}$)).

[2] H. Q. Pham et al., "ViOCRVQA: Novel Benchmark Dataset and Vision Reader for OCR-VQA in Vietnamese," *Multimedia Systems*, 2025. ([researchgate.net](https://www.researchgate.net/publication/380186254$_{ViOCRVQA_Novel_Benchmark_Dataset_and_Vision_Reader_for_Visual_Question_Answeri}$ $_{chatgpt.com}$)).

[2] D.-K. Nguyen and T. Okatani, "Improved Fusion by Dense Symmetric Co-Attention for VQA," *arXiv*, 2018. ([arxiv.org](https://arxiv.org/abs/1804.00775?utm$_{source = chatgpt.com}$)).

[2] W. Souai, "Mastering Named Entity Recognition with BERT," *UBI AINLP Blog*, 2022. ([medium.com](https://medium.com/ubiai-nlp/mastering-named-entity-recognition-with-bert-ca8d04b67b18?utm$_{source = chatgpt.com}$)).

[2] A. Author, "Enhancing OCR in Historical Documents with Complex Layouts," *Document Analysis and Recognition*, 2025. ([link.springer.com](https://link.springer.com/article/10.1007/s00799-025-00413-z?utm$_{source = chatgpt.com}$)).

[2] Y. Zhang and X. Li, "Metaheuristic Algorithms for Optimization: A Brief Review," *MDPI Algorithms*, vol. 59, no. 1, 2023. ([mdpi.com](https://www.mdpi.com/2673-4591/59/1/238?utm$_{source = chatgpt.com}$)).