TheoremView: A Framework for Extracting Theorem-like Environments from Raw PDFs

Shrey Mishra[1 [0009−0004−2357−9593]], Neil Sharma[2 [0009−0004−2357−9593]], Antoine Gauquier[1 [0009−0005−9573−6364]], and Pierre Senellart[1,3 [0000−0002−7909−5369]]

[1] DI ENS, ENS, CNRS, PSL University, Inria, Paris, France
shrey.mishra@ens.psl.eu, antoine.gauquier@ens.psl.eu, pierre@senellart.com
[2] Malaviya National Institute of Technology
neil.sharma3000@gmail.com
[3] Institut Universitaire de France (IUF)

Abstract. This paper presents TheoremView, a novel framework for extracting proofs and theorems from raw PDF scientific papers without requiring LaTeX source files. Our approach combines three modalities (**font**, **text**, and **vision**) with sequential modeling to capture long-term dependencies and layout information. By eliminating OCR preprocessing, TheoremView reduces computational overhead for real-time applications while providing robust automated theorem extraction. Our framework is publicly available at https://gitlab.di.ens.fr/mishra/sys-demo, with a demonstration video at https://youtu.be/4IkDeN4f_w4.

Keywords: Theorem extraction · Multimodal learning · Document analysis · Machine learning · Natural language processing

1 Introduction

1.1 Motivation for Theorem Extraction

In contemporary scientific research, articles are primarily published as PDFs, and many search engines index entire papers instead of specific scientific results. This paper contributes to TheoremKB [3], a project focused on building a knowledge base of mathematical results across different fields of science. The objective is to improve the accessibility of relevant information for researchers, allowing for more effective retrieval and utilization of scientific knowledge. TheoremKB offers several key advantages:

- **Enhanced Accessibility:** Streamlines the retrieval of specific proofs and theorems, allowing quick access to targeted mathematical results compared to traditional full-text search engines
- **Facilitated Knowledge Discovery:** Helps researchers uncover connections between disparate mathematical results and their applications, such as exploring NP-hardness in relation to the vertex cover problem

- **Identification of Theorem Interdependencies:** Determines which theorems are used in the proofs of others, essential for assessing the impact of errors in foundational results
- **Support for Automated Reasoning:** Provides a foundation for developing AI systems

1.2 Prior work on Theorems and Proofs Extraction

Previous attempts to address this task include the work presented in [3], which focused on initial explorations of extraction from PDFs framed as object detection and text classification problems. This approach utilized font visuals and text modalities but operated only at the text-line level. Subsequent research, such as [1], refined the methodology by incorporating contextual information surrounding paragraphs and employing multimodal systems to unify the extraction model. The TheoremView framework offers a user interface to visualize the results extracted by various models in an end-to-end system that directly takes PDFs as input and displays the extracted results. It is designed modularly, allowing users to select which model to utilize for extraction, thereby leveraging different modalities that highlight the strengths and weaknesses of each approach. This flexibility enables users to run models on low-compute hardware, such as systems without GPU instances, for inference. The primary objective of this paper is to present an easy-to-use interface that facilitates preprocessing and inference in a modular manner.

2 Methodology

We propose a modular approach to extract raw information from PDFs. We utilize **Grobid** [7] and **pdfalto** [8] to convert the documents into valid XML formats. The XML generated by Grobid organizes the content into paragraphs, while the XML from pdfalto provides details segmented into text lines along with associated font information. We then employ a merging script to correlate the font information with each paragraph extracted by Grobid. This process yields a CSV file structured by paragraphs, where each row includes the spatial location of the paragraph on the page (indicating the page number as well as vertical and horizontal coordinates), the textual content extracted from Grobid, and the font information used within those paragraphs. For a schematic diagram on data pipeline refer to Figure 1.

Once the information is stored in CSV format, we process the font information using an **LSTM** [5] model, where each font is encoded as a unique token to train the network. Simultaneously, we utilize the bitmap image rendering of each paragraph to train an **EfficientNetV2** model [9]. Additionally, we employ a pretrained from scratch **RoBERTa** language model [2], on scientific corpus, to make predictions based on the tex modality. Subsequently, we integrate all three trained models into a unified multimodal architecture, freezing their weights of each modality backbone and adding additional layers to capture intermodality

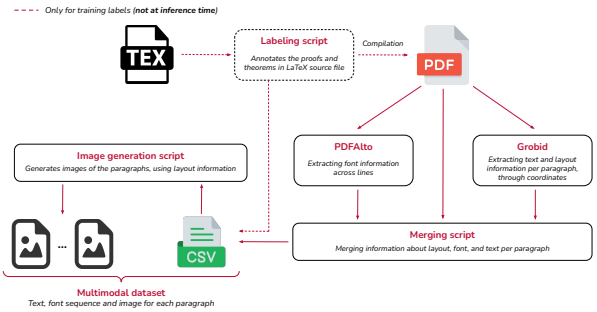


Fig. 1: Data pipeline for extracting and processing information from PDFs.

interactions through mechanisms like Gated Multimodal Units (GMU) [6] or cross-modality attention similar to ViLBERT [12] that capture intermodality dependencies. Refer to Figure 2 for feature extraction.

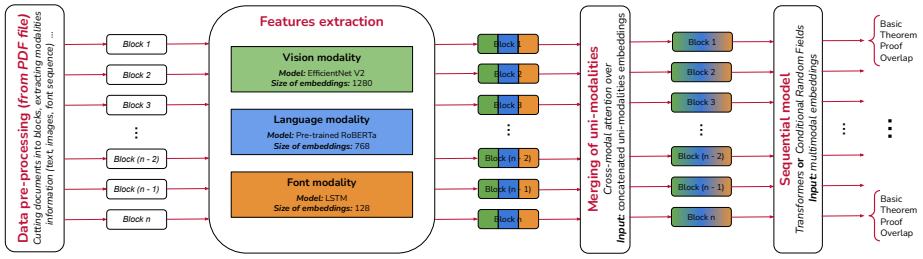


Fig. 2: Model inference pipeline (adding the sequential paragraph component)

With a set of base features extracted from either the unimodal or multimodal approaches, we generate features for all paragraphs within the PDF. This process incorporates normalized page information, normalized coordinate data for each paragraph, and paragraph embeddings derived from the raw features just before the softmax layer. To capture sequential information across multiple paragraphs, we train a Conditional Random Field (CRF) [10] or Transformer layer on top of the extracted features. The goal is to utilize relative information to contextualize each paragraph and accurately determine its label. Our model categorizes paragraphs into four major classes: (1) **Proof-like**, (2) **Theorem-like**, (3) **Basic** (neither proof nor theorem), and (4) an **Overlap** reject class that arises from preprocessing discrepancies.

3 Demonstration Scenario

The TheoremView demo interface, built using Streamlit, follows a modular architecture with distinct functional components. The frontend allows users to upload

PDFs or select from cached samples for metadata processing using Grobid and pdfalto tools, with results stored as CSV files. Users can run various ML models (unimodal or multimodal) through a pipeline that calculates processing time, generates bounding boxes, creates cropped images of theorems/proofs, and produces analytical graphs. The system implements efficient caching using pickle files for frequently accessed PDFs and ML model results.

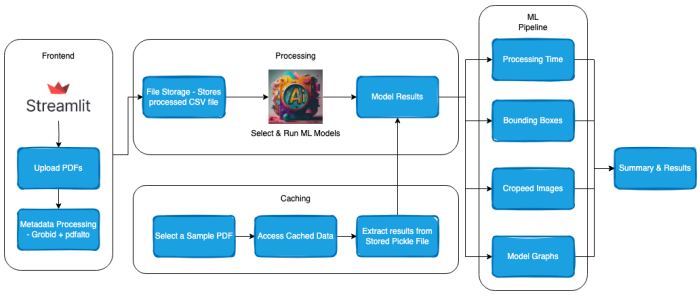
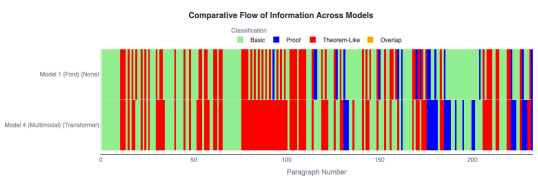


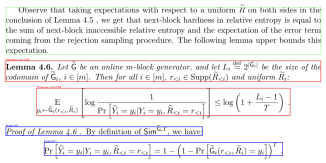
Fig. 3: Systems architecture of the various UI components

The UI of the demo is organized into several segments (for an overview see Figure 3), each serving a specific function:

1. **Upload and Process**: Users can upload PDFs or select from cached examples. The system processes PDFs using Grobid and pdfalto, converts pages to bitmap images, and merges XML outputs to generate a preprocessed `data.csv` file.
2. **Predict and Preview**: Users can select unimodal or multimodal base models, with optional sequential processing using CRF, Transformer, or none, offering 12 possible combinations. Results can be previewed or downloaded (see Figure 4).
3. **Summary and Statistics**: Provides breakdown of inference time for current and cached runs, enabling comparative analysis.



(a) Flow of information across paragraphs



(b) Visualization of predictions annotated on the PDF

Fig. 4: Visualizing model predictions

References

1. Mishra, S., Gauquier, A., Senellart, P.: Modular Multimodal Machine Learning for Extraction of Theorems and Proofs in Long Scientific Documents (Extended Version). arXiv preprint arXiv:2307.09047 (2024). https://arxiv.org/abs/2307.09047
2. Mishra, S.: Multimodal Extraction of Proofs and Theorems from the Scientific Literature. PhD thesis, Université Paris Sciences & Lettres (2024). https://hal.science/tel-04665528
3. Mishra, S., Pluvinage, L., Senellart, P.: Towards extraction of theorems and proofs in scholarly articles. In: Healy, P., Bilauca, M., Bonnici, A. (eds.) DocEng '21: ACM Symposium on Document Engineering 2021, Limerick, Ireland, August 24-27, 2021, pp. 25:1–25:4. Association for Computing Machinery, New York, NY, USA (2021). https://doi.org/10.1145/3469096.3475059
4. Mishra, S., Brihmouche, Y., Delemazure, T., Gauquier, A., Senellart, P.: First steps in building a knowledge base of mathematical results. In: Proceedings of the Fourth Workshop on Scholarly Document Processing (SDP 2024), pp. 165–174 (2024)
5. Hochreiter, S., Schmidhuber, J.: Long short-term memory. In: Neural Computation 9, 8 (1997)
6. Arevalo, J., Solorio, T., Montes-y Gómez, M., A González, F.: Gated multimodal networks. In: Neural Computing and Applications 32 (2020).
7. GROBID: https://github.com/kermitt2/grobid, GitHub (2008–2024). swh:1:dir:dab86b296e3c3216e2241968f0d63b68e8209d3c
8. pdfalto: https://github.com/kermitt2/pdfalto, GitHub (2017–2024). swh:1:dir:4b5e8b8c8e3c3216e2241968f0d63b68e8209d3c
9. Tan, M., Le, Q.V.: EfficientNetV2: Smaller Models and Faster Training. In: Proceedings of the 38th International Conference on Machine Learning, PMLR 139, pp. 10096–10106 (2021)
10. Lafferty, J., McCallum, A., Pereira, F.C.N.: Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In: Proceedings of the Eighteenth International Conference on Machine Learning, ICML '01, pp. 282–289. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (2001)
11. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is All you Need. In: Advances in Neural Information Processing Systems, pp. 5998–6008 (2017)
12. Lu, J., Batra, D., Parikh, D., Lee, S.: ViLBERT: Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language Tasks. In: Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, Vancouver, BC, Canada, pp. 13–23 (2019)