Modular Multimodal Machine Learning for Extraction of Theorems and Proofs in Long Scientific Documents

Shrey Mishra, Antoine Gauquier, Pierre Senellart

Modular Multimodal Machine Learning for Extraction of Theorems and Proofs in Long Scientific Documents

Shrey Mishra
DI ENS, ENS, CNRS, PSL University, Inria
Paris, France
shrey.mishra@ens.psl.eu

Antoine Gauquier
DI ENS, ENS, CNRS, PSL University, Inria
Paris, France
antoine.gauquier@ens.psl.eu

Pierre Senellart
DI ENS, ENS, CNRS, PSL University, Inria & IUF
Paris, France
pierre@senellart.com

Abstract

We address the extraction of mathematical statements and their proofs from scholarly PDF articles as a multimodal classification problem, utilizing text, font features, and bitmap image renderings of PDFs as distinct modalities. We propose a modular sequential multimodal machine learning approach specifically designed for extracting theorem-like environments and proofs. This is based on a cross-modal attention mechanism to generate multimodal paragraph embeddings, which are then fed into our novel multimodal sliding window transformer architecture to capture sequential information across paragraphs. Our approach demonstrates performance improvements obtained by transitioning from unimodality to multimodality, and finally by incorporating sequential modeling over paragraphs.

CCS Concepts

• **Information systems** → **Information extraction**.

Keywords

scholarly articles, information extraction, multimodal classifiers

1 Introduction

Scholarly articles in mathematical fields typically include theorems (and other theorem-like environments) along with their proofs. This paper builds upon our previous work [11], which aimed to transform scientific literature from a collection of PDF articles into an open knowledge base (KB) centered around theorems. In this paper, we concentrate primarily on the extraction aspect of the pipeline introduced in [11]. We conduct an in-depth exploration of diverse multimodal methodologies and assess the impact of modeling long-term paragraph sequences.

To clarify, in the whole of this paper we use *theorem* in the same sense as it is used in LaTeX (say, by the `\newtheorem` command): a theorem-like environment is a structured statement, possibly numbered, formatted in a specific way and used to represent a formal (usually mathematical) statement: it can be a theorem, a lemma, a proposition, etc., but also a definition, a formal remark or an example. By *theorem* we mean any statement of this kind. By *proof* we mean what would typically be rendered in LaTeX in a `proof` environment: a proof or proof sketch of a result.

We approach the theorem–proof identification problem by designing an approach based on multimodal machine learning that classifies each paragraph of an article into *basic*, *theorem*, and *proof* labels, based on the scientific language, on typographical information, and on visual rendering of PDF documents. Additionally, we take into account information about the *sequence* of paragraph blocks, normalised spatial coordinates and page numbers along with page breaks, to exploit the fact that the label of a paragraph heavily relies on that of the preceding (or following) ones.

We provide the following contributions in this paper, summarized in Figure 1: (i) Three unimodal (vision, text, font information) models for the theorem–proof identification problem relying on modern machine learning techniques (CNNs, transformers, LSTMs) with a focus on reasonably efficient models as opposed to very large ones; note that the text modality approach relies on pretraining a language model specific to our corpus, which may have applications beyond our task. (ii) A multimodal late fusion model that combines the features of all three modalities. (iii) A block sequential approach, based on a transformer model, that can be used to improve the performance of any unimodal and multimodal model by capturing dependencies between blocks. (iv) An experimental evaluation on a dataset of roughly 200k English-language papers from arXiv, with a separate validation dataset of 3.5k papers (amounting to 529k paragraph blocks).

We present in Section 2 the three unimodal models. We then discuss in Section 3 how to combine them into a multimodal model, and how to add support for information about block sequences. We further provide a description of our dataset in Section 4. Experimental results on all unimodal and multimodal models are presented in Section 5.

An extended version of this work is available [12] with discussion of the related work, details on different models, and experiments. We also refer to the PhD thesis of the first author [10] for additional details on our methodology and results. The code, data, and models supporting this paper are accessible at https://github.com/mv96/mm_extraction.

2 Unimodal Models

We now present the methodology of our three unimodal models: a pretrained transformer (RoBERTa-based) language model for text extracted for each paragraph of the PDF; an EfficientNetv2M [14] CNN for vision on the bitmap rendering of each PDF paragraph; and an LSTM model trained on font information sequences within each paragraph. For a technical reason explained in Section 4, the problem is formulated as a four-class classification: in addition to the three target *basic text*, *theorem*, *proof*, we employ a reject *overlap* class.

Text Modality. We pretrain a language model from scratch on a 50k vocabulary size (with byte-pair encoding), similar to the configuration of RoBERTa base (124M) [7]. While masking 15%

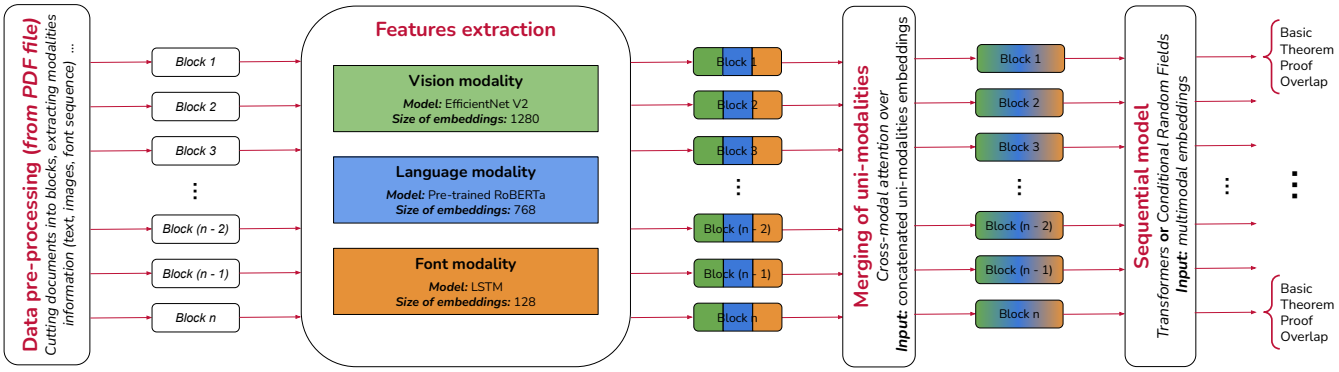

Figure 1: Overall model Inference pipeline

of tokens we kept the configuration similar to original RoBERTa ($L = 12, H = 768, A = 12$), but on a different vocabulary. The model used dynamic masking and was trained on masked language modeling loss. After pretraining, the model is fine-tuned for our classification task.

Vision Modality. CNNs, pivotal in image classification and as backbones in visual-language tasks, typically benchmark on ImageNet and CIFAR for top-1% accuracy. Our project, targeting the identification of mathematical symbols and the layout of paragraph blocks to discern proofs and theorems, necessitates model training from scratch. Distinct markers like the term "Proof" in unique fonts and the QED symbol, crucial yet overlooked by text modalities, guide our focus.

One specificity of vision approach for classification block is that images come in widely different **aspect ratios**. Traditional interpolation methods, though prevalent for adjusting natural images to a uniform resolution, unsuitably modify the geometry of text, symbols, and fonts in our context. Based on corpus analysis, we establish a fixed resolution of (400×1400) pixels. This size accommodates over 80% of our paragraphs, with larger images being cropped and smaller ones padded to maintain this standard without altering their intrinsic visual properties. This approach aligns with recommendations against scale variance [15] and parallels the preprocessing strategy used in the Nougat paper [1], which also maintains a constant aspect ratio to suit specific model inputs. Our method ensures the preservation of textual image integrity by avoiding the pitfalls of resizing, opting instead for cropping or padding to fit our predetermined resolution criteria.

To counteract the issue of **white backgrounds** in scientific texts, which can hinder CNN performance as noted by studies [5], we invert image colors to mimic the MNIST dataset's white-on-black text presentation. This approach prevents max-pooling operations in CNNs from mistakenly prioritizing the background, thereby maintaining focus on the textual content.

EfficientNet comes with several variants (B0–B7) where B7 has the largest receptive field due to compound scaling. We select in our experiments a base network (B0), a medium-sized network (B4) and the largest network (B7). EfficientNetV2 also comes with different sizes. We focused on the small (EfficientNetV2s) and medium-sized (EfficientNetV2m) models.

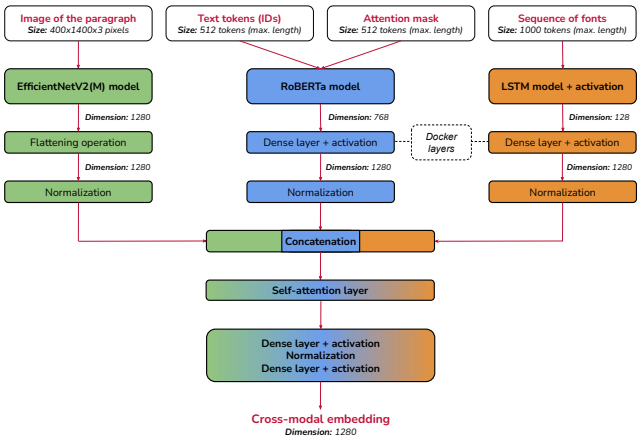

Figure 2: Cross-modal attention architecture

Font Modality. The last modality we consider is styling information present in the PDF in terms of the sequence of fonts (font family and font size) used in a specific paragraph. This information can be obtained using the pdfalto tool[1], which produces a list of fonts used in a given document, and associates each text token to a particular font. Fonts are usually standard LaTeX fonts, such as `cmr10` for Computer Modern Roman in 10 point.

From the training data, we build a font vocabulary of 4 031 unique fonts including their sizes, and represent every paragraph block as a sequence of font identifiers. To match input dimensions among training samples, we apply left padding with a maximum length of 1 000. We then feed the entire sequence to a simple 128-cell LSTM [4] network to monitor the loss. The choice of the model is purely to capture sequential information within fonts that can be used to identify the label of the paragraphs.

3 Multimodal and Sequential Models

We now go beyond unimodal models by showing how all three modalities can be combined into a single late-fusion multimodal model, and how block sequence information can be captured.

[1] https://github.com/kermitt2/pdfalto

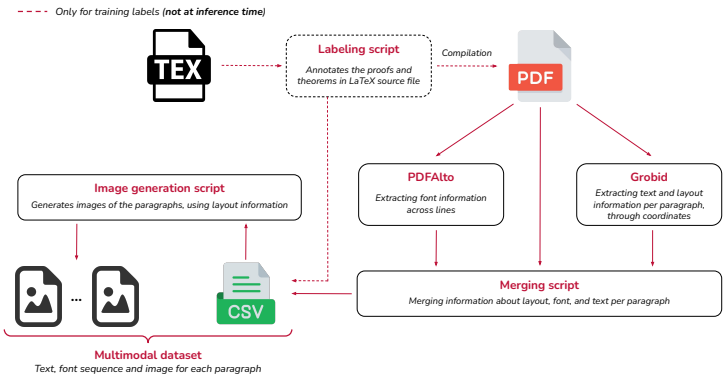

Figure 3: Dataset preparation pipeline

The main multimodal model we use is a cross-modal attention model, inspired by ViLBERT's attention mechanism [9]. We show its full architecture in Figure 2.

In addition to modalities, considering the sequencing of the blocks, i.e., the order in which they appear in the document, allows us to determine with greater confidence the class of each block.

We propose two approaches to do this: First, using a simple linear-chain order-one Conditional Random Fields model (CRFs) [6]. Second, we introduce a novel transformer-based BERT-like encoder architecture (also more efficient for our task) to process multimodal features, using a sliding window (SW) of size $k = 16$. We also investigate the impact of long sequential relationships by employing interleaving architecture found in Hierarchical Attention Transformers (HATs) [2]. The architecture is modified to be adapted in a multimodal setting such as ours.

The CRF and SW models use the following features, on top of frozen unimodal or multimodal model: unimodal text, vision, and font models respectively bring 768, 1280, and 128 features; the multimodal approach includes 1280 joint features; we incorporate four additional geometrical features to describe block positions: normalized page number, indicating a block's page relative to the total pages; normalized horizontal and vertical distances from the block's bounding box corners; and a binary feature indicating if a block and its predecessor are on the same page.

In order to determine whether long-distance dependencies are also useful to capture for our task, we also implement HATs, relying on the same Sliding Window transformer encoder architecture used as as a segment-wise encoder. We then expanded it to learn about connections between different context windows (using cross segment encoder) taking only the Multimodal [CLS] token of every segment. Out of the versions proposed in the original HAT paper [2], we tested the best-performing one, i.e., with interleaving layers.

4 Dataset and Setup

We use Grobid[2] [8], which is the state of the art for information extraction from scholarly documents to parse a PDF document and interpret it into a succession of paragraph blocks.

[2]https://github.com/kermitt2/grobid

Our dataset, encompassing all arXiv papers (around 1.7 million papers) up to May 2020, was acquired via arXiv's bulk data access on Amazon S3. We developed an annotation script to pinpoint theorem-like environments and proofs within these documents, leveraging LaTeX sources. This involved crafting a LaTeX package to instrument commands such as `\newtheorem` for precise identification in the compiled PDFs (≈ 460k papers). See Figure 3. We filtered articles from the dataset to only keep those in English, for which LaTeX source is available (according to arXiv's policy, all those that have been produced using LaTeX), that were compilable on a modern LaTeX distribution, that contained at least a theorem or a proof environment, and for which none of the tools (our ground-truth annotation package, Grobid for extraction of blocks, pdfalto for line-by-line font sequences, bitmap image rendering for CNN's) failed to produce a valid output. This resulted in a final dataset of ≈ 197k papers. We stress that LaTeX sources are only used to produce ground-truth annotations, they are not required at inference time. Grobid sometimes fails to extract correct paragraphs, i.e., some of the paragraphs identified by Grobid overlap blocks of different category (say, *basic* and *theorem*). We label such paragraphs as *overlap*, exclusively used for such outliers.

Our validation set comprises approximately 500 000 paragraph blocks from 3 682 randomly selected PDF articles. The remaining articles formed the training dataset, used entirely for pretraining our language model after filtering potential personal information such as author names and institutions from Grobid extractions to minimize privacy concerns. Training involved dividing the dataset into batches of 1 000 PDF articles, incrementally fitting classifiers on these batches until convergence, without exceeding a few dozen batches. Post-training, classifiers' weights were frozen for integration into the multimodal classifier, subsequently employed as feature extractors for the sequential approaches. The dataset is heavily imbalanced, with the number of paragraphs labeled as *basic*: 314 501, *proof*: 125 524, *theorem*: 85 801, and *overlap*: 3 470.

All experiments were run on a supercomputer with access at any point to 4 NVIDIA (V100 or A100) GPUs. We estimate to 8 000 GPU hours the computational cost of the entire prototyping, hyperparameter tuning, training, validation, and evaluation pipeline.

5 Experimental Results

We now report experimental results on the *basic–theorem–proof* classification problem, first comparing representative unimodal classifiers, with and without the article paragraphs fed to the sequential approach, followed by the multimodal classifier. We then delve into more specific details of every unimodal classifier.

We are interested in two main performance metrics: *accuracy* measures the raw accuracy of the classifier on the validation dataset (disjoint with the training dataset); and (unweighted arithmetic) *mean* F_1*-measure* of the *basic*, *theorem*, and *proof* classes, which summarizes the precision and recall over each class assigning the same weight to every class. As *basic* is the most common class in the dataset, a *dummy* classifier that would always predict the *basic* class would have an accuracy of 59.41%; but its recall would be 100% on *basic* and 0% on the other classes, while its precision would be 59.41% on *basic* and 0% on the other classes, resulting in a mean F_1 of $\frac{1}{3} \times \frac{2 \times 59.41\%}{59.41\% + 100\%} \approx 24.85\%$. This gives an important comparison point

Table 1: Overall performance comparison (accuracy and mean F_1 over the three classes *basic*, *theorem*, and *proof*) of individual modality models and multimodal model, with and without the sequential approach; for each model, the number of batches (1 000 PDF documents, roughly 200k samples) it was trained on is indicated (here + indicates additional batches on which further training of sequential paragraph model)

Modality	Model chosen	Seq. approach	#Batches	#Params (M)	Accuracy (%)	Mean F_1 (%)
Dummy	always predicts *basic*	—	—	—	59.41	24.85
Top-*k* first word	use only first word	—	—	—	52.84	44.20
Line-based [13]	Bert (fine-tuned)	—	—	110	57.31	55.71
Font	LSTM 128 cells	-	11	2	64.93	45.48
		CRF	11+8	2	71.50	64.51
		SW Transformer	11+8	2	76.22	71.77
Vision	EfficientNetV2m_avg	-	9	53	69.44	60.33
		CRF	9+8	53	74.63	70.82
		SW Transformer	9+8	65	79.59	77.66
Text	Pretrained RoBERTa-like	-	20	124	76.45	72.33
		CRF	20+8	124	83.10	80.99
		SW Transformer	20+8	129	87.50	86.67
Multimodal	Cross-modal attention	-	2	185	78.50	75.37
		CRF	2+8	185	84.39	82.91
		SW Transformer	2+8	198	**87.81**	**87.18**
		HAT	2+8	232	87.52	86.58

for all other methods; accuracy measures how well the classifier works on the actual unbalanced data, while mean F_1 favors methods performing well to identify all three classes.

Drawing inspiration from two related works, albeit applied in slightly different settings, we evaluate two straightforward baselines: (1) Top-*k* first words: This method, which echoes the approach used in [3] focusing on the first paragraph of marked environments, constructs a vocabulary of the top-*k* unique words for each class. Labels are assigned based on the first word of a text and whether it matches any word within the class-specific vocabulary. For instance, if the first word is within {*theorem, lemma, proposition, definition*}, the text is labeled as a theorem. (2) Text classifier from [13]: We reuse the text classifier that was fine-tuned in [13], which processes text lines (not paragraphs) extracted from pdfalto. Note this classifier does not identify the *overlap* class.

The results we obtain for the different modalities, with and without the use of either CRF or sliding-window block sequence model, are shown in Table 1. The following lessons can be drawn:

(1) This is a hard task, as the best performance reached is 88% for accuracy and 87% for mean F_1. Indeed, it can be hard even to a human to determine whether a block is part of a proof or theorem environment, especially in the middle of it, so it is unsurprising that we cannot reach near-perfect results.

(2) Looking at unimodal models: the font-based model performs rather poorly, though still beating (at least in terms of mean F_1) the three baselines; the text-based model is the best performing one, suggesting textual clues impact more than visual ones for this task.

(3) The multimodal model outperforms every unimodal model, though the margin with the text model is somewhat low.

(4) Including the Sequential model (both CRF, SW transformer, or HAT) greatly increases the performance of every unimodal or multimodal model, by 5 to 10 points of accuracy or mean F_1. The importance of the use of an approach modeling block sequences is thus clear. Long-distance dependencies captured by HATs do not seem to matter.

Acknowledgments

This work was funded in part by the French government under management of Agence Nationale de la Recherche as part of the "Investissements d'avenir" program, reference ANR-19-P3IA-0001 (PRAIRIE 3IA Institute). This work was also made possible through HPC resources of IDRIS granted under allocation 2020-AD011012097 made by GENCI (Jean Zay supercomputer).

References

[1] Lukas Blecher, Guillem Cucurull, Thomas Scialom, and Robert Stojnic. 2023. Nougat: Neural optical understanding for academic documents. *arXiv preprint arXiv:2308.13418* (2023).

[2] Ilias Chalkidis, Xiang Dai, Manos Fergadiotis, Prodromos Malakasiotis, and Desmond Elliott. 2022. An exploration of hierarchical attention transformers for efficient long document classification. *arXiv preprint arXiv:2210.05529* (2022).

[3] Deyan Ginev and Bruce R. Miller. 2020. Scientific Statement Classification over arXiv.org. In *LREC*.

[4] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997).

[5] Hossein Hosseini, Baicen Xiao, Mayoore Jaiswal, and Radha Poovendran. 2017. On the limitation of convolutional neural networks in recognizing negative images. In *ICMLA*.

[6] John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *ICML*.

[7] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv:1907.11692* (2019).

[8] Patrice Lopez. 2009. GROBID: Combining Automatic Bibliographic Data Recognition and Term Extraction for Scholarship Publications. In *ECDL*, Vol. 5714. 473–474.

[9] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. ViLBERT: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *Advances in neural information processing systems* 32 (2019).

[10] Shrey Mishra. 2024. *Multimodal Extraction of Proofs and Theorems from the Scientific Literature.* Ph. D. Dissertation. Université Paris Sciences & Lettres.

[11] Shrey Mishra, Yacine Brihmouche, Theo Delemazure, Antoine Gauquier, and Pierre Senellart. 2024. First steps in building a knowledge base of mathematical results. In *Proceedings of the Fourth Workshop on Scholarly Document Processing (SDP 2024)*. 165–174.

[12] Shrey Mishra, Antoine Gauquier, and Pierre Senellart. 2024. Modular Multimodal Machine Learning for Extraction of Theorems and Proofs in Long Scientific Documents (Extended Version). *arXiv:2307.09047* (2024).

[13] Shrey Mishra, Lucas Pluvinage, and Pierre Senellart. 2021. Towards extraction of theorems and proofs in scholarly articles. In *DocEng*.

[14] Mingxing Tan and Quoc Le. 2019. EfficientNet: Rethinking model scaling for convolutional neural networks. In *ICML*.

[15] Hugo Touvron, Andrea Vedaldi, Matthijs Douze, and Hervé Jégou. 2019. Fixing the train-test resolution discrepancy. *Advances in neural information processing systems* 32 (2019).