Page Stream Segmentation with LLMs: Challenges and Applications in Insurance Document Automation

Hunter Heidenreich[1], Ratish Dalvi[1], Nikhil Verma[1], Yosheb Getachew[1]

[1]Roots Automation, New York, NY

Correspondence: ai@rootsautomation.com

Abstract

Page Stream Segmentation (PSS) is critical for automating document processing in industries like insurance, where unstructured document collections are common. This paper explores the use of large language models (LLMs) for PSS, applying parameter-efficient fine-tuning to real-world insurance data. Our experiments show that LLMs outperform baseline models in page- and stream-level segmentation accuracy. However, stream-level calibration remains challenging, especially for high-stakes applications. We evaluate post-hoc calibration and Monte Carlo dropout, finding limited improvement. Future work will integrate active learning to enhance model calibration and support deployment in practical settings.

1 Introduction

1.1 Background and Motivation

Page stream segmentation (PSS) (Collins-Thompson and Nickolov, 2002) is a critical task in industries like insurance, law, and healthcare, where bulk transmission of unstructured document collections occurs routinely. Documents are often bundled during digitization processes without clear boundaries, which creates inefficiencies and requires manual reorganization. In adversarial settings such as litigation or insurance claims, senders have little incentive to format documents for optimal downstream processing.

Automated PSS is therefore essential for converting bundled collections into discrete, actionable units compatible with an organization's systems. Failure to automate this process can lead to costly delays, misclassification, and poor decision-making, particularly in high-stakes domains.

Research in PSS has been hindered by a lack of publicly available datasets reflecting real-world complexity. Privacy concerns in sectors like healthcare and finance limit access to realistic data (Agin et al., 2015), forcing reliance on synthetic datasets (Mungmeeprued et al., 2022a; Van Heusden et al., 2022), which often fail to capture the variability of actual document streams.

At the same time, large-scale Transformer models have driven advances in document processing tasks. While multimodal models are promising, their increased computational complexity during training and inference must be justified by performance improvements.

Building on Heidenreich et al. (2024), who demonstrated the efficacy of parameter-efficient fine-tuning (PEFT) of unimodal large language models (LLMs) for synthetic PSS, our study extends this framework to real-world insurance data.

1.2 Key Contributions

Our key contributions based on empirical evaluation of real-world insurance data include:

1. **Real-World Evaluation**: We extend Heidenreich et al. (2024) by applying LLMs to insurance data, demonstrating that LLMs outperform XGBoost on both page- and stream-level metrics in real-world PSS tasks. Prior findings that smaller transformer models such as RoBERTa and LayoutLMv3 provided minimal gains over XGBoost motivates the focus on LLMs.

2. **Calibration Assessment**: We assess the calibration of LLM-based models and evaluate post-hoc calibration to mitigate overconfidence, crucial for automation requiring human intervention.

3. **Stream-Level Confidence**: We introduce a stream-level confidence measure based on page-level predictions to determine which streams can be automated versus those requiring human review, analyzed through an accuracy-vs-throughput curve.

Proceedings of the 31st International Conference on Computational Linguistics: Industry Track, pages 305–317
January 19–24, 2025. ©2025 Association for Computational Linguistics

2 Related Work

2.1 Page Stream Segmentation (PSS)

PSS has evolved from rule-based systems to neural models, but generalizing across diverse document types remains challenging (Collins-Thompson and Nickolov, 2002; Daher and Belaïd, 2014). Transformer-based models (Vaswani et al., 2017) are central to NLP and document processing, though their application to PSS is still emerging. Prior work, including Guha et al. (2022) and Mungmeeprued et al. (2022a), primarily used encoder-based models with convolutional layers. However, multimodal models often add complexity without consistently outperforming unimodal approaches (Heidenreich et al., 2024).

In our prior work (Heidenreich et al., 2024), we evaluated diverse baselines, including RoBERTa (text-only; Liu et al., 2019), DiT (vision-only; Li et al., 2022), LiLT (text with layout; Wang et al., 2022), and LayoutLMv3 (text with layout and vision; Huang et al., 2022). While these models slightly outperformed XGBoost, they fell significantly short of LLMs, which showed unmatched segmentation performance. This motivates our exclusive focus on LLMs in this study.

2.2 LLMs for Document Processing

LLMs have shown success in document processing tasks, such as those benchmarked in DocVQA (Mathew et al., 2021), but many evaluations use synthetic or narrowly scoped datasets, which fail to capture the complexity of real-world streams (Van Landeghem et al., 2024). This can obscure model limitations, particularly in tasks like PSS, where document diversity is key.

We address this by evaluating LLMs on a domain-specific insurance dataset, providing insights into their practical performance and limitations, highlighting their real-world applicability.

2.3 Calibration and Confidence

Calibration is essential for ensuring reliable predictions in high-stakes tasks, especially where uncertainty can guide decisions (Mielke et al., 2022; Huang et al., 2023; Kapoor et al., 2024). In binary classification tasks like PSS, proper calibration helps flag uncertain predictions that may require human intervention.

Although we do not propose a novel calibration method, we assess the effects of Monte Carlo (MC) dropout (Gal and Ghahramani, 2016) and logistic regression-based confidence estimation on PSS performance. Our analysis reveals the limitations of these approaches in mitigating overconfidence, highlighting the need for more sophisticated calibration methods in future work (Kapoor et al., 2024). Accurate identification of low-confidence predictions is critical for reliable automation.

3 Page Stream Segmentation (PSS)

3.1 Problem Definition

Given a sequence of N pages, $P = (p_1, p_2, \ldots, p_N)$, the task is to infer the boundaries between documents, resulting in a sequence of M documents, $D = (d_1, d_2, \ldots, d_M)$, where each document d_k is a contiguous subsequence of P. We focus on restoring page-level boundaries in multi-page files, where documents have been bundled into a single stream for transmission.

This task is framed as a binary classification problem. For each page p_i, the model predicts whether it starts a new document ($y_i = 1$) or continues the current document ($y_i = 0$), producing a binary vector $\hat{\mathbf{y}} \in \{0, 1\}^N$. The prediction is based on a local context of adjacent pages:

$$(p_{i-l}, \ldots, p_{i-1}, p_i, p_{i+1}, \ldots, p_{i+r}) \mapsto y_i.$$

We primarily explore a local context setting of $(l = 1, r = 0)$. Additional results for other context settings are shown in Appendix B.

3.2 Evaluation Metrics

We evaluate model performance at both the page and stream levels to capture segmentation accuracy across predictions.

Page-Level Evaluation For page-level evaluation, we use precision, recall, and F1 score to assess boundary prediction accuracy.

Stream-Level Evaluation At the stream level, we evaluate segmentation by comparing predicted segmentations $\mathcal{P}$ to ground truth $\mathcal{G}$. True positives (TP) are documents in both $\mathcal{P}$ and $\mathcal{G}$, false positives (FP) are in $\mathcal{P} \setminus \mathcal{G}$, and false negatives (FN) are in $\mathcal{G} \setminus \mathcal{P}$. We compute precision, recall, and F1 for each stream.

Calibration Metrics In high-stakes environments, well-calibrated predictions are crucial. We use Expected Calibration Error (ECE) for

Dataset	**Real**	**Lang.**	**Online**	**Streams**	**Docs**	**Pages**
Tobacco800 (Doermann, 2019)	×	EN	✓	-	742	1.3k
Spanish Banking (Rusiñol et al., 2014)	✓	ES	×	-	7.2k	69.7k
ITESOFT (Karpinski and Belaïd, 2016)	✓	EN	×	532	2.4k	4.3k
Court Lawsuits (Mota et al., 2020)	✓	PT	×	117	-	3.0k
Archive26k (Wiedemann and Heyer, 2021)	✓	DE	×	120	4.9k	26.9k
A.I. Lab Splitter (Braz et al., 2021)	✓	PT	✓	4.3k	5.5k	31.8k
WooIR (Van Heusden et al., 2022)	✓	NL	✓	229	7.1k	45.0k
TABME (Mungmeeprued et al., 2022b)	×	EN	✓	110.0k	44.8k	122.5k
Title Insurance (Guha et al., 2022)	✓	EN	×	-	30.4k	185.5k
SVic+ (Luz De Araujo et al., 2023)	✓	PT	✓	6.5k	-	339.5k
Internal (ours)	✓	EN	×	7.5k	20.3k	44.7k

Table 1: Overview of datasets for PSS, highlighting data authenticity, language, and accessibility.

average-case calibration and Maximum Calibration Error (MCE) for worst-case calibration (Pakdaman Naeini et al., 2015). These metrics assess both binary predictions and stream-level confidence estimates.

Additionally, we plot accuracy versus throughput at the stream level, reporting area under the curve (AUC) and accuracy/throughput at 90% and 80% confidence thresholds, where an accurate stream is defined as perfectly segmented.

4 Experimental Setup

4.1 Dataset

Public datasets for PSS in English are extremely limited, with only two publicly available datasets—Tobacco800 and TABME. However, these datasets have significant shortcomings. TABME, while larger, is entirely synthetic, constructed by randomly concatenating unrelated documents into artificial streams. This synthetic nature fails to capture the nuanced challenges of real-world PSS tasks, such as domain-specific conventions in concatenations or the presence of structured and unstructured content. Furthermore, both datasets originate from the same source, limiting their collective utility. These factors render existing public datasets misaligned with the realities of insurance document processing.

In contrast, insurance datasets present unique challenges due to their structural and informational diversity. Our proprietary dataset comprises text-dense documents (e.g., health records and contracts), tabular data (e.g., policies and loss runs), scanned letters, emails, and unstructured narratives (e.g., police reports). The data is characterized by domain-specific jargon spanning legal, medical, and insurance contexts, as well as sensitive Personally Identifiable Information (PII). These features make such datasets crucial for evaluating the performance of PSS systems in real-world scenarios. However, privacy regulations and ethical considerations prevent public release of the dataset, even in anonymized form.

While public datasets like TABME are simpler due to their synthetic construction, our dataset reflects the complexity of real-world streams and provides a robust test bed for segmentation tasks. Future work could address this gap by creating synthetic benchmarks that closely mimic real-world data while adhering to strict PII safeguards. Nonetheless, for high-stakes applications like insurance automation, real-world data remains critical for assessing system performance.

Our dataset consists of 7.5k streams, 20.3k documents, and 44.7k pages, aligning with other private datasets like Title Insurance and ITESOFT. It contains authentic English documents, capturing the complexity of the insurance domain. We partition the dataset into four splits: training (60%), validation (10%), calibration (15%), and test (15%).

4.2 Model Architecture

We experiment with two decoder-only LLMs: Phi-3.5-mini (3.8B parameters) (Abdin et al., 2024) and Mistral-7B (7B parameters) (Jiang et al., 2023), chosen for their varying sizes and architecture to test input robustness.

Given the findings of Heidenreich et al. (2024), which demonstrated that smaller transformer models like RoBERTa, LayoutLMv3, and LiLT marginally surpassed XGBoost but significantly underperformed compared to LLMs, we opted not to include these baselines in the current study. This decision enables us to focus on evaluating the unique capabilities of LLMs in PSS while reducing redundancy in experimental comparisons.

4.3 Training

We use Low-Rank Adaptation (LoRA) (Hu et al., 2021) to fine-tune models efficiently, adapting them for PSS while minimizing computational costs.

The fine-tuning process is standardized across models using a consistent prompt format (see Appendix A), ensuring comparability. We incorporate OCR for layout-sensitive text representations (Wang et al., 2023; Li et al., 2024; Bayani, 2024), given the importance of whitespace-based layout in LLMs.

For some models, we introduce stochasticity through Monte Carlo (MC) dropout (Gal and Ghahramani, 2016; Lin et al., 2024), applied to LoRA weights to capture epistemic uncertainty. When doing so, we fix a dropout rate of $p = 0.5$ and denote the variant with a 'MC-' prefix. We also experiment with post-hoc calibration methods to assess their impact on confidence estimates.

4.4 Calibration

To estimate confidence, we track key statistics of the model's output predictions (Huang et al., 2023; Liu et al., 2024), recording the probability of the "1" class, entropy, and log-odds. For models using MC dropout, we compute the mean, standard deviation, min, and max of these quantities across multiple forward passes.

We also calculate the variation ratio (VR), measuring the fraction of predictions disagreeing with the modal class:

$$\text{VR} = 1 - \frac{f_{c=c^*}}{N}, \tag{1}$$

where $f_{c=c^*}$ is the frequency of the modal class across N forward passes.

A logistic regression model is used to recalibrate predictions based on these uncertainty statistics.

At the page level, we define the confidence for each page p_i as:

$$C_i = p_i \cdot \mathbb{I}(p_i > 0.5) + (1 - p_i) \cdot \mathbb{I}(p_i \leq 0.5), \tag{2}$$

where p_i is the calibrated probability for page p_i, and $\mathbb{I}(\cdot)$ is the indicator function. Higher confidence is assigned to more certain predictions.

Stream-level confidence C is then computed as the product of page-level confidences:

$$C = \prod_{i=1}^{N} C_i, \tag{3}$$

where N is the number of pages in the stream. This provides an overall confidence measure for the entire stream.

5 Results

5.1 Model Comparison

Table 2 compares model performance on page- and stream-level segmentation tasks, using $(l = 1, r = 0)$ as context for all models.

XGBoost serves as a baseline, achieving a page-level F1 of 0.902 and stream-level F1 of 0.827. Although reasonable, LLMs outperform XGBoost across all metrics. Mistral shows a 0.5-1.0 F1 point improvement over Phi at both levels.

Recalibration of predictions has minimal impact, as expected. For MC dropout variants, the effect is mixed—Phi shows a slight precision gain at the cost of recall, while Mistral sees reduced recall without significant precision gains.

Further analysis reveals XGBoost struggles with documents containing multiple stamps or misleading page sequences (e.g., original and fax page numbers), whereas LLMs consistently succeed. This highlights LLMs' strength in capturing complex document features. An example instance of this is shown in Figure 1.

5.2 Model Calibration

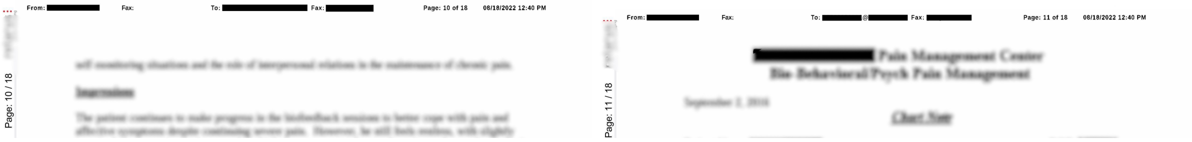
Table 3 shows Expected Calibration Error (ECE) and Maximum Calibration Error (MCE) for each model. Lower values indicate better alignment between predicted probabilities and actual outcomes.

XGBoost exhibits the lowest MCE values at both page and stream levels and shows strong overall calibration. In contrast, all LLMs show higher calibration errors, with post-hoc recalibration improving page-level calibration but not stream-level. MC dropout does not improve calibration and even increases errors, questioning its use for this task given its higher computational cost.

We visualize the reliability of the Mistral model at the page and stream levels in Figure 2. Notably, we observe that Mistral has difficulty accurately expressing low-confidence packages, overestimating the true likelihood of perfect segmentation. After

	Page-Level Metrics			Stream-Level Metrics		
Model	**Prec.**	**Rec.**	**F1**	**Prec.**	**Rec.**	**F1**
XGBoost	0.912	0.893	0.902	0.832	0.827	0.827
Phi	0.935	0.931	0.933	0.864	0.862	0.861
Phi*	0.934	0.933	0.934	0.864	0.863	0.861
MC-Phi	0.941	0.934	0.937	0.875	0.874	0.872
MC-Phi*	0.937	0.939	0.938	0.873	0.874	0.872
Mistral	0.953	0.935	0.944	0.883	0.879	0.879
Mistral*	0.947	**0.946**	**0.947**	0.884	**0.883**	**0.882**
MC-Mistral	**0.954**	0.931	0.943	**0.885**	0.878	0.880
MC-Mistral*	0.948	0.938	0.943	0.883	0.879	0.880

Table 2: Comparison of model performance on page- and stream-level metrics. Asterisk $(^*)$ indicates re-calibrated models. The best value per column is bolded, and the best within each model type is underlined.


From: [redacted] Fax: To: [redacted] Fax: [redacted] Page: 10 of 18 08/18/2022 12:40 PM

Page: 10 / 18

From: [redacted] Fax: To: [redacted] Fax: [redacted] Page: 11 of 18 08/18/2022 12:40 PM

Page: 11 / 18

Figure 1: An example pair of page headers where LLMs correctly identify a split and XGBoost incorrectly predicts continuity. Despite the introduction of a new page header, XGBoost over-relies on the consecutive page labeling. This is a salient feaure, but misleading for some sets of faxed documents.

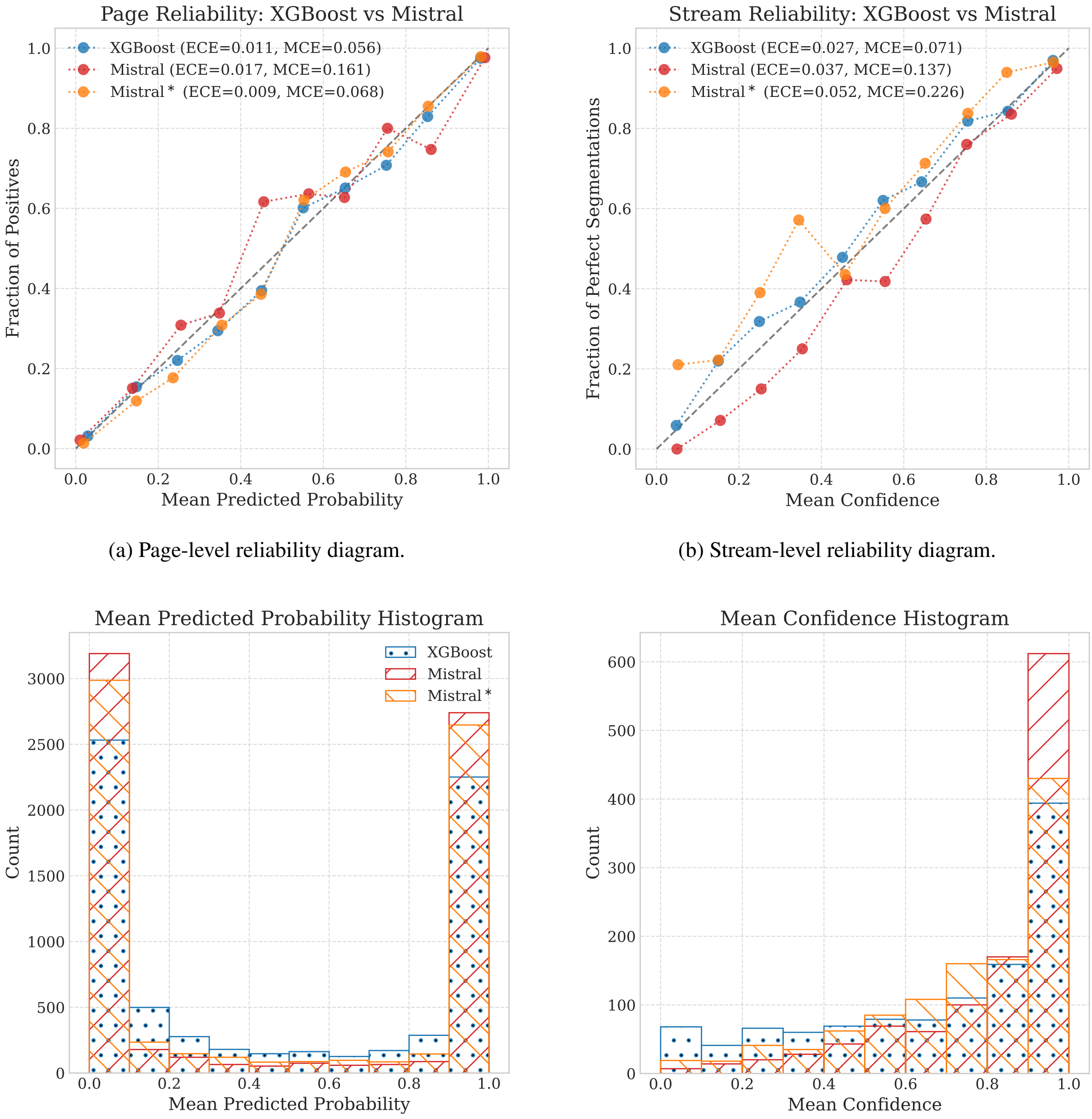
	Page		Stream	
Model	**ECE**	**MCE**	**ECE**	**MCE**
XGBoost	0.011	**0.056**	**0.027**	**0.071**
Phi	0.012	0.103	0.025	0.131
Phi*	0.010	0.101	0.036	0.098
MC-Phi	0.023	0.134	0.049	0.208
MC-Phi*	0.010	0.063	0.055	0.142
Mistral	0.017	0.161	0.037	0.137
Mistral*	**0.009**	0.068	0.052	0.226
MC-Mistral	0.020	0.132	0.042	0.213
MC-Mistral*	0.010	0.129	0.054	0.128

Table 3: Model calibration errors. Asterisk $(^*)$ indicates re-calibrated models. The best values in each column are bolded, and the best within each model type is underlined.

		$C > 0.9$		$C > 0.8$	
Model	**AUC**	**ACC**	**T**	**ACC**	**T**
XGBoost	0.908	**0.97**	0.35	0.93	0.49
Phi	0.931	0.96	0.49	0.94	0.61
Phi*	0.930	**0.97**	0.37	**0.96**	0.54
MC-Phi	0.934	0.94	0.58	0.92	0.71
MC-Phi*	0.933	**0.97**	0.33	0.95	0.51
Mistral	0.938	0.95	0.54	0.93	0.70
Mistral*	**0.939**	0.96	0.38	**0.96**	0.53
MC-Mistral	0.934	0.94	0.54	0.92	0.71
MC-Mistral*	0.937	0.96	0.38	**0.96**	0.49

Table 4: Model stream accuracy versus throughput (T) at confidence levels of 80% and 90%. Area under the curve summarizes each model's curve. Asterisk $(^*)$ indicates re-calibrated models. The best values are bolded, and the best within each model type is underlined.

calibration, Mistral* results in a better calibrated page predictor, but its behavior is shifted towards underestimating the likelihood of stream accuracy.



(a) Page-level reliability diagram.

(b) Stream-level reliability diagram.

(c) Histogram of predicted probabilities.

(d) Histogram of stream confidence scores.

Figure 2: Reliability and confidence comparisons between XGBoost and Mistral models.

5.3 Automation Throughput

Table 4 compares models by stream-level accuracy and throughput at confidence thresholds of 90% ($C > 0.9$) and 80% ($C > 0.8$). The AUC summarizes performance across confidence levels. Throughput reflects the proportion of data processed automatically, while accuracy represents correctness on this subset.

At 90% confidence, XGBoost performs comparably to recalibrated LLMs in accuracy but automates less data. As the threshold lowers to 80%, LLMs maintain higher accuracy over larger volumes, while XGBoost's accuracy drops. This is visualized in Figure 3 for Mistral.

The improved AUC for LLMs indicates better handling of PSS nuances, enabling reliable predictions with less manual intervention, crucial for high-throughput environments.

6 Discussion and Conclusion

In this work, we demonstrated the effectiveness of large language models (LLMs) for Page Stream Segmentation (PSS) in the insurance domain, significantly outperforming traditional models like XGBoost in both page- and stream-level segmentation. However, calibration remains a challenge,

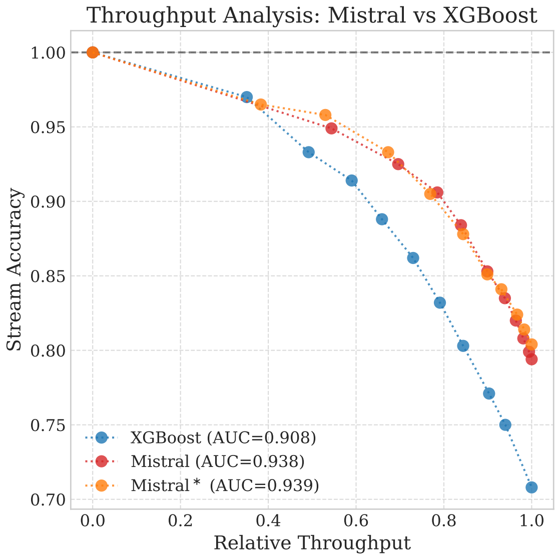


Figure 3: Stream-level accuracy versus throughput plots for Mistral and XGBoost models. For each curve, its automation potential is summarized as the AUC.

particularly in high-stakes scenarios where overconfidence poses operational risks.

A key challenge in PSS research is the lack of publicly available datasets that reflect real-world complexity. Existing datasets, such as TABME, are synthetic and fail to capture the structural diversity and domain-specific jargon found in insurance documents, including health records, policies, and contracts. While our proprietary dataset addresses these gaps, privacy constraints prevent its public release. Future efforts should prioritize developing synthetic benchmarks that emulate real-world data while ensuring strict privacy safeguards, as such benchmarks are critical for advancing PSS systems.

Despite evaluating post-hoc calibration and Monte Carlo dropout, these methods increased model complexity without significantly improving stream-level calibration. This underscores the need for more robust calibration techniques. Future work will explore advanced calibration methods and the integration of active learning, where human feedback iteratively improves model performance and reliability.

Our approach offers a clear path to real-world deployment in document-heavy sectors like insurance. Calibrated confidences can guide human validation, with low-confidence streams prioritized to address model uncertainty. This strategy improves reliability while maintaining scalability for automation in high-stakes environments.

In these domains, ethical considerations are paramount. Misclassifications from overconfident models can lead to costly errors. Ensuring well-calibrated predictions and incorporating human oversight at key decision points will mitigate risks and enable responsible automation.

References

Marah Abdin, Sam Ade Jacobs, Ammar Ahmad Awan, Jyoti Aneja, Ahmed Awadallah, Hany Awadalla, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, Alon Benhaim, Misha Bilenko, Johan Bjorck, Sébastien Bubeck, Qin Cai, Martin Cai, Caio César Teodoro Mendes, Weizhu Chen, Vishrav Chaudhary, Dong Chen, Dongdong Chen, Yen-Chun Chen, Yi-Ling Chen, Parul Chopra, Xiyang Dai, Allie Del Giorno, Gustavo de Rosa, Matthew Dixon, Ronen Eldan, Victor Fragoso, Dan Iter, Mei Gao, Min Gao, Jianfeng Gao, Amit Garg, Abhishek Goswami, Suriya Gunasekar, Emman Haider, Junheng Hao, Russell J. Hewett, Jamie Huynh, Mojan Javaheripi, Xin Jin, Piero Kauffmann, Nikos Karampatziakis, Dongwoo Kim, Mahoud Khademi, Lev Kurilenko, James R. Lee, Yin Tat Lee, Yuanzhi Li, Yunsheng Li, Chen Liang, Lars Liden, Ce Liu, Mengchen Liu, Weishung Liu, Eric Lin, Zeqi Lin, Chong Luo, Piyush Madan, Matt Mazzola, Arindam Mitra, Hardik Modi, Anh Nguyen, Brandon Norick, Barun Patra, Daniel Perez-Becker, Thomas Portet, Reid Pryzant, Heyang Qin, Marko Radmilac, Corby Rosset, Sambudha Roy, Olatunji Ruwase, Olli Saarikivi, Amin Saied, Adil Salim, Michael Santacroce, Shital Shah, Ning Shang, Hiteshi Sharma, Swadheen Shukla, Xia Song, Masahiro Tanaka, Andrea Tupini, Xin Wang, Lijuan Wang, Chunyu Wang, Yu Wang, Rachel Ward, Guanhua Wang, Philipp Witte, Haiping Wu, Michael Wyatt, Bin Xiao, Can Xu, Jiahang Xu, Weijian Xu, Sonali Yadav, Fan Yang, Jianwei Yang, Ziyi Yang, Yifan Yang, Donghan Yu, Lu Yuan, Chengruidong Zhang, Cyril Zhang, Jianwen Zhang, Li Lyna Zhang, Yi Zhang, Yue Zhang, Yunan Zhang, and Xiren Zhou. 2024. Phi-3 Technical Report: A Highly Capable Language Model Locally on Your Phone. *arXiv preprint*. ArXiv:2404.14219 [cs] version: 1.

Onur Agin, Cagdas Ulas, Mehmet Ahat, and Can Bekar. 2015. An approach to the segmentation of multi-page document flow using binary classification. In *Sixth International Conference on Graphic and Image Processing (ICGIP 2014)*, volume 9443, page 944311. International Society for Optics and Photonics, SPIE.

David Bayani. 2024. Testing the Depth of ChatGPT's Comprehension via Cross-Modal Tasks Based on ASCII-Art: GPT3.5's Abilities in Regard to Recognizing and Generating ASCII-Art Are Not Totally Lacking. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 2063–2077, St. Julian's, Malta. Association for Computational Linguistics.

Fabricio Ataides Braz, Nilton Correia da Silva, and Jonathan Alis Salgado Lima. 2021. Leveraging effectiveness and efficiency in page stream deep segmentation. *Engineering Applications of Artificial Intelligence*, 105:104394.

Tianqi Chen and Carlos Guestrin. 2016. XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794.

Kevyn Collins-Thompson and Radoslav Nickolov. 2002. A Clustering-Based Algorithm for Automatic Document Separation. In *Proceedings of the SIGIR 2002*, Tampere, Finland.

Hani Daher and Abdel Belaïd. 2014. Document flow segmentation for business applications. In *Document Recognition and Retrieval XXI*, volume 9021, page 90210G. International Society for Optics and Photonics, SPIE.

David Doermann. 2019. Tobacco 800 dataset (tobacco800). `https://tc11.cvc.uab.es/datasets/Tobacco800_1`. Accessed: 2024-08-07.

Yarin Gal and Zoubin Ghahramani. 2016. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1050–1059, New York, New York, USA. PMLR.

Abhijit Guha, Abdulrahman Alahmadi, Debabrata Samanta, Mohammad Zubair Khan, and Ahmed H. Alahmadi. 2022. A Multi-Modal Approach to Digital Document Stream Segmentation for Title Insurance Domain. *IEEE Access*, 10:11341–11353. Conference Name: IEEE Access.

Daniel Han and Michael Han. Unsloth.

Hunter Heidenreich, Ratish Dalvi, Rohith Mukku, Nikhil Verma, and Neven Pičuljan. 2024. Large Language Models for Page Stream Segmentation. *arXiv preprint*. Version Number: 1.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. LoRA: Low-Rank Adaptation of Large Language Models. *arXiv preprint*. ArXiv:2106.09685 [cs].

Yuheng Huang, Jiayang Song, Zhijie Wang, Shengming Zhao, Huaming Chen, Felix Juefei-Xu, and Lei Ma. 2023. Look Before You Leap: An Exploratory Study of Uncertainty Measurement for Large Language Models. *arXiv preprint*. ArXiv:2307.10236 [cs].

Yupan Huang, Tengchao Lv, Lei Cui, Yutong Lu, and Furu Wei. 2022. Layoutlmv3: Pre-training for document ai with unified text and image masking. In *Proceedings of the 30th ACM International Conference on Multimedia*, MM '22, page 4083–4091, New York, NY, USA. Association for Computing Machinery.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7B. *arXiv preprint*. ArXiv:2310.06825 [cs].

Sanyam Kapoor, Nate Gruver, Manley Roberts, Arka Pal, Samuel Dooley, Micah Goldblum, and Andrew Wilson. 2024. Calibration-tuning: Teaching large language models to know what they don't know. In *Proceedings of the 1st Workshop on Uncertainty-Aware NLP (UncertaiNLP 2024)*, pages 1–14, St Julians, Malta. Association for Computational Linguistics.

Romain Karpinski and Abdel Belaïd. 2016. Combination of Structural and Factual Descriptors for Document Stream Segmentation. In *2016 12th IAPR Workshop on Document Analysis Systems (DAS)*, pages 221–226, Santorini, Greece. IEEE.

Junlong Li, Yiheng Xu, Tengchao Lv, Lei Cui, Cha Zhang, and Furu Wei. 2022. Dit: Self-supervised pre-training for document image transformer. In *Proceedings of the 30th ACM International Conference on Multimedia*, MM '22, page 3530–3539, New York, NY, USA. Association for Computing Machinery.

Weiming Li, Manni Duan, Dong An, and Yan Shao. 2024. Large Language Models Understand Layout. *arXiv preprint*. ArXiv:2407.05750 [cs].

Yang Lin, Xinyu Ma, Xu Chu, Yujie Jin, Zhibang Yang, Yasha Wang, and Hong Mei. 2024. LoRA Dropout as a Sparsity Regularizer for Overfitting Control. *arXiv preprint*. ArXiv:2404.09610 [cs].

Linyu Liu, Yu Pan, Xiaocheng Li, and Guanting Chen. 2024. Uncertainty estimation and quantification for llms: A simple supervised approach. *Preprint*, arXiv:2404.15993.

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. *CoRR*, abs/1907.11692.

Pedro H. Luz De Araujo, Ana Paula G. S. De Almeida, Fabricio Ataides Braz, Nilton Correia Da Silva, Flavio De Barros Vidal, and Teofilo E. De Campos. 2023. Sequence-aware multimodal page classification of Brazilian legal documents. *International Journal on Document Analysis and Recognition (IJDAR)*, 26(1):33–49.

Minesh Mathew, Dimosthenis Karatzas, and C.V. Jawahar. 2021. Docvqa: A dataset for vqa on document images. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 2200–2209.

Sabrina J. Mielke, Arthur Szlam, Emily Dinan, and Y-Lan Boureau. 2022. Reducing Conversational Agents' Overconfidence Through Linguistic Calibration. *Transactions of the Association for Computational Linguistics*, 10:857–872.

Caio Mota, Andressa Lima, André Nascimento, Péricles Miranda, and Rafael de Mello. 2020. Classificação de páginas de petições iniciais utilizando redes neurais convolucionais multimodais. In *Anais do XVII Encontro Nacional de Inteligência Artificial e Computacional*, pages 318–329, Porto Alegre, RS, Brasil. SBC.

Thisanaporn Mungmeeprued, Yuxin Ma, Nisarg Mehta, and Aldo Lipani. 2022a. Tab this folder of documents: page stream segmentation of business documents. In *Proceedings of the 22nd ACM Symposium on Document Engineering*, DocEng '22, pages 1–10, New York, NY, USA. Association for Computing Machinery.

Thisanaporn Mungmeeprued, Yuxin Ma, Nisarg Mehta, and Aldo Lipani. 2022b. Tabme dataset. `https://github.com/aldolipani/TABME`. Accessed: 2024-08-07.

Mahdi Pakdaman Naeini, Gregory Cooper, and Milos Hauskrecht. 2015. Obtaining well calibrated probabilities using bayesian binning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 29(1).

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Marçal Rusiñol, Volkmar Frinken, Dimosthenis Karatzas, Andrew D. Bagdanov, and Josep Lladós. 2014. Multimodal page classification in administrative document image streams. *International Journal on Document Analysis and Recognition (IJDAR)*, 17(4):331–341.

Ruben Van Heusden, Jaap Kamps, and Maarten Marx. 2022. WooIR: A New Open Page Stream Segmentation Dataset. In *Proceedings of the 2022 ACM SIGIR International Conference on Theory of Information Retrieval*, pages 24–33, Madrid Spain. ACM.

Jordy Van Landeghem, Sanket Biswas, Matthew Blaschko, and Marie-Francine Moens. 2024. Beyond Document Page Classification: Design, Datasets, and Challenges. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 2962–2972.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, and Nathan Lambert. TRL: Transformer Reinforcement Learning.

Jiapeng Wang, Lianwen Jin, and Kai Ding. 2022. LiLT: A simple yet effective language-independent layout transformer for structured document understanding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7747–7757, Dublin, Ireland. Association for Computational Linguistics.

Wenjin Wang, Yunhao Li, Yixin Ou, and Yin Zhang. 2023. Layout and Task Aware Instruction Prompt for Zero-shot Document Image Question Answering. *arXiv preprint*. ArXiv:2306.00526 [cs].

Gregor Wiedemann and Gerhard Heyer. 2021. Multimodal page stream segmentation with convolutional neural networks. *Language Resources and Evaluation*, 55(1):127–150.

A Training Details

A.1 Traditional Model

We employed count-based and TF-IDF-based vector representations using scikit-learn (Pedregosa et al., 2011). Individual pages were treated as "documents" for fitting the vector representations. The text was lowercased, and the default word-based tokenization strategy was applied to produce unigrams. Additionally, TF-IDF vectors were L2-normalized as per the scikit-learn default.

An XGBoost (Chen and Guestrin, 2016) classifier was trained on adjacent page pairs to predict true document breaks. Each page was independently vectorized, and the vectors of the preceding and current pages were concatenated as the input to the XGBoost model. We adjusted for class imbalance by scaling with the positive class ratio and used 100 estimators.

A.2 Decoders

We primarily rely on Unsloth (Han and Han) for performance efficient fine-tuning of LLMs. For Mistral-7B, we use the 4-bit quantized version of the instruct v0.3 model[1]. For Phi-3.5-mini, we use the 4-bit quantized version of the instruct model[2]. Models are trained using Hugging Face's TRL library (von Werra et al.) on completion tokens only, ignoring the instructions when backpropagating. The prompt template is shown in Listing 1, where

[1] `https://huggingface.co/unsloth/mistral-7b-instruct-v0.3-bnb-4bit`

[2] `https://huggingface.co/unsloth/Phi-3.5-mini-instruct-bnb-4bit`

```
You are a skilled document reviewer. Given extracted text from pages
of documents, your task is to determine if a page starts a new
document or continues from the previous one. You will be presented
with the text of the current page and the text of the preceding page.

Example:

Prior text:
###
This is the text on the page before the page you are evaluating.
###
Page text:
###
This is the text on the page you are evaluating.
###

Carefully review the text to decide if the current page starts a new
document or continues from the previous one.

Here is the input:

Prior text:
###
{pg_prev}
###
Page text:
###
{pg}
###

Output your prediction as a JSON object. When the page is the start
of a new document, your output should be {"label": 1}. If the page
continues the document from the previous page, your output should be
{"label": 0}. Do not provide any explanation, additional information,
 or punctuation. Simply provide the JSON object.

Does the page start a new document?
```

Listing 1: Instruction prompt used for page stream segmentation. When using a bidirectional context (i.e., $r > 0$), the prompt is modified to feature a pg_next after the page of interest. When including mutliple pages in the context (i.e., $l, r > 1$), pages are separated with a page break sequence.

pg and pg_prev attempt to preserve layout structure in 2D (Wang et al., 2023). When applying MC dropout to models, we fix a dropout rate of $p = 0.5$, and sample a model's output $N = 16$ times for every input page.

All other hyperparameters are summarized in Table 5. We perform all decoder fine-tuning on a single NVIDIA H100 GPU, with LoRA weights in BF16 format.

B Additional Results

B.1 $(l = 1, r = 0)$

Similar to the automation curve displayed for Mistral in Figure 3, we present additional automation curves for Phi (Figure 4, MC-Mistral (Figure 6), and MC-Phi (Figure 5). Overall, the automation curves exhibit similar features, with the Mistral model offering the best accuracy-throughput trade-

Params.	Mistral-7B	Phi-3.5-mini
Peak LR	3×10^{-5}	3×10^{-5}
Batch size	16	16
Weight decay	0.01	0.01
Optimizer	`paged_adamw_8bit`	`paged_adamw_8bit`
Max train epochs	5	5
LR warm-up steps	200	200
LoRA r	16	16
LoRA α	16	16
Sequence length	8192	8192

Table 5: Hyperparameters used for PEFT of decoder-only LLMs.

offs of considered models.

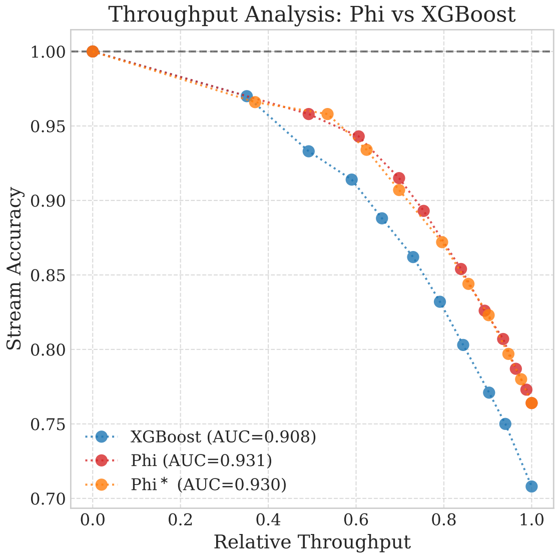


Figure 4: Stream-level accuracy versus throughput plots for Phi and XGBoost models. For each curve, its automation potential is summarized as the AUC.

B.2 $(l = 1, r = 1)$

Under a context setting of $(l = 1, r = 1)$, models observe the page before and after the page of interest when making a classification decision. To support a bidirectional page context, we slightly modify the LLM prompt to include references to a "next page."

Though exhibiting negligible difference with the results presented in the main body, we show page and stream metrics in Table 6, calibration errors in Table 7, and automation-related metrics in Table 8.

B.3 $(l = 2, r = 0)$

We briefly explored extending the left-hand context of models, allowing models to predict the start of a document based on the prior two pages. Between the two prior pages, we insert a page break sequence. We present page and stream metrics in Table 9, calibration errors in Table 10, and automation-related metrics in Table 11.

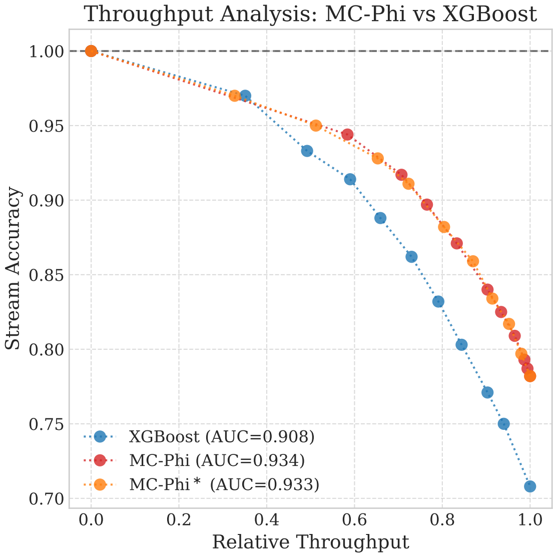


Figure 5: Stream-level accuracy versus throughput plots for MC-Phi and XGBoost models. For each curve, its automation potential is summarized as the AUC.

	Page-Level Metrics			Stream-Level Metrics		
Model	**Prec.**	**Rec.**	**F1**	**Prec.**	**Rec.**	**F1**
Phi	0.935	0.939	0.937	0.868	0.866	0.865
Phi*	0.934	0.942	0.938	0.870	0.869	0.868
Mistral	0.948	0.937	0.943	0.88	0.878	0.877
Mistral*	0.947	0.94	0.943	0.881	0.88	0.879

Table 6: Decoder-only LLM performance under a context setting of $(l = 1, r = 1)$, where a model takes the previous, current, and subsequent page as input to decide if the current page begins a new document.

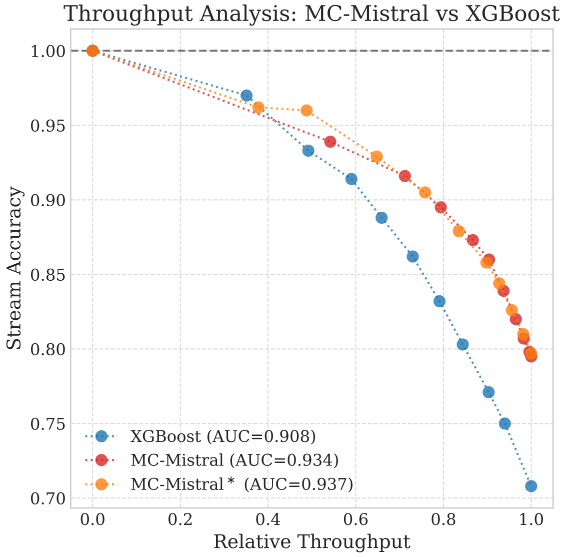


Figure 6: Stream-level accuracy versus throughput plots for MC-Mistral and XGBoost models. For each curve, its automation potential is summarized as the AUC.

		$C > 0.9$		$C > 0.8$	
Model	**AUC**	**ACC**	**T**	**ACC**	**T**
Phi	0.929	0.94	0.53	0.92	0.66
Phi*	0.932	0.97	0.35	0.95	0.54
Mistral	0.931	0.93	0.56	0.91	0.72
Mistral*	0.937	0.98	0.28	0.95	0.47

Table 8: Model automation metrics under a context setting of $(l = 1, r = 1)$.

	Page		Stream	
Model	**ECE**	**MCE**	**ECE**	**MCE**
Phi	0.020	0.093	0.040	0.150
Phi*	0.007	0.114	0.039	0.090
Mistral	0.024	0.197	0.057	0.219
Mistral*	0.010	0.042	0.068	0.174

Table 7: Model calibration errors under a context setting of $(l = 1, r = 1)$.

	Page-Level Metrics			**Stream-Level Metrics**		
Model	**Prec.**	**Rec.**	**F1**	**Prec.**	**Rec.**	**F1**
Ph i	0.953	0.935	0.944	0.883	0.879	0.879
Phi*	0.947	0.946	0.947	0.884	0.883	0.882
Mistral	0.952	0.927	0.939	0.882	0.875	0.877
Mistral*	0.935	0.944	0.939	0.867	0.871	0.867

Table 9: Decoder-only LLM performance under a context setting of $(l = 2, r = 0)$, where a model takes the previous two pages and the current page as input to decide if the current page begins a new document.

	Page		**Stream**	
Model	**ECE**	**MCE**	**ECE**	**MCE**
Phi	0.013	0.135	0.027	0.185
Phi*	0.012	0.085	0.040	0.086
Mistral	0.014	0.149	0.027	0.079
Mistral*	0.007	0.082	0.047	0.107

Table 10: Model calibration errors under a context setting of $(l = 2, r = 0)$.

		$C > 0.9$		$C > 0.8$	
Model	**AUC**	**ACC**	**T**	**ACC**	**T**
Phi	0.934	0.96	0.49	0.93	0.63
Phi*	0.936	0.98	0.35	0.96	0.54
Mistral	0.936	0.96	0.49	0.93	0.65
Mistral*	0.932	0.97	0.34	0.95	0.54

Table 11: Model automation metrics under a context setting of $(l = 2, r = 0)$.