

# DriveThru: a Document Extraction Platform and Benchmark Datasets for Indonesian Local Language Archives

Mohammad Rifqi Farhansyah<sup>1\*</sup>, Muhammad Zuhdi Fikri Johari<sup>2\*</sup>, Afinzaki Amiral<sup>3</sup>,

Ayu Purwarianti<sup>1</sup>, Kumara Ari Yuana<sup>2</sup>, Derry Tanti Wijaya<sup>4†</sup>

Institut Teknologi Bandung<sup>1</sup> Universitas Amikom Yogyakarta<sup>2</sup>

Universitas Dian Nuswantoro<sup>3</sup> Boston University<sup>4</sup>

wijaya@bu.edu

## Abstract

Indonesia is one of the most diverse countries linguistically. However, despite this linguistic diversity, Indonesian languages remain underrepresented in Natural Language Processing (NLP) research and technologies. In the past two years, several efforts have been conducted to construct NLP resources for Indonesian languages. However, most of these efforts have been focused on creating manual resources thus difficult to scale to more languages. Although many Indonesian languages do not have a web presence, locally there are resources that document these languages well in printed forms such as books, magazines, and newspapers. Digitizing these existing resources will enable scaling of Indonesian language resource construction to many more languages. In this paper, we propose an alternative method of creating datasets by digitizing documents, which have not previously been used to build digital language resources in Indonesia. DriveThru is a platform for extracting document content utilizing Optical Character Recognition (OCR) techniques in its system to provide language resource building with less manual effort and cost. This paper also studies the utility of current state-of-the-art LLM for post-OCR correction to show the capability of increasing the character accuracy rate (CAR) and word accuracy rate (WAR) compared to off-the-shelf OCR. The platform is available online at <https://ocrdt.ragambahasa.id/> and the benchmark dataset, evaluation script, and the models are available at our GitHub repository<sup>1</sup>. We also provide a short (~1-minute) screencast of our system on YouTube: <https://youtu.be/q5uJOHKcBsg>

## 1 Introduction

Indonesia is known as one of the world's most populated countries with a population exceeding 270

million. Spreading over 17 thousand islands, Indonesia is also one of the most diverse countries in the world with over 1,300 ethnic groups speaking over 700 local languages (Eberhard et al., 2024). In terms of the number of speakers, the top 20 of Indonesian languages are spoken by over 1 million people *each*. Despite this linguistic diversity, Indonesian languages remain underrepresented in Natural Language Processing (NLP) research and technologies (Aji et al., 2022). Thus, very little of NLP's significant progress in the past few years have found its way to applications for these languages.

To spur the development of research and technologies for Indonesian languages, in the past two years several efforts have been carried out to construct NLP resources for Indonesian languages (Cahyawijaya et al., 2023a; Winata et al., 2023; Cahyawijaya et al., 2023b). However, these efforts have been focused on a small number of Indonesian languages (top 10 languages in terms of existing Web presence). The resources are created either manually or translated from English resources using existing machine translation (MT) systems. As the process of hiring annotators and managing annotation is costly and time-consuming, and because existing MT systems are limited in coverage, these efforts are difficult to scale to more languages.

Although many Indonesian languages do not have a web presence, locally there are resources that document these languages well in printed forms such as books: textbooks, grammar books, dictionaries, story books, etc., magazines, and newspapers. Digitizing these existing resources will enable scaling of Indonesian language resource construction to many more languages. In addition, digitizing existing resources such as books can alleviate some of the drawbacks of prior works. Firstly, a book must have passed through quality assurance stages before being published, thus, the requirement of recruiting native speakers as a dataset cu-

---

\* Contributed equally

† Corresponding author

<sup>1</sup><https://github.com/ragambahasa>

rator can be altered since the works will be mostly focused on collecting books and identifying languages of these books instead of creating resources from scratch. Secondly, the time and cost for constructing resources can be reduced since it takes less time and costs to digitize books than creating resources from scratch. Several books are available online and published openly by the Indonesian National Library or the Indonesian government itself.

In this study, we propose **DriveThru** platform, an alternative system that digitizes documents to assist Indonesian NLP researchers in their language resource collection. In this work, aside from using an off-the-shelf OCR system, TesseractOCR, we also benchmark LLMs for post-OCR error correction in local Indonesian languages, which occurs when off-the-shelf OCR systems are unable to recognize certain characters. We explore OCR and post-OCR error correction four low-resource Indonesian local languages: Javanese (jav), Sundanese (sun), Minangkabau (min), and Balinese (ban) written in latin scripts. This work demonstrates our approach for collecting underrepresented language resources through document extraction i.e., digitization of printed documents written in these languages.

## 2 Related Work

The development of a language resource dataset via document extraction encounters significant challenges including the presence of noisy data, incorrect character recognition, and frequent occurrences of hallucinations. Several studies on Finnish language encountered challenges when they had to digitize old documents with previously unseen fonts ([Drobac et al., 2017](#); [Koistinen et al., 2017](#)).

Prior works have attempted to address these issues by implementing post-processing of the OCR outputs. Many prior works involve manual corrections of the OCR outputs. For example, to develop language resources for the Bodo language, one of the community languages spoken in India ([Narzary et al., 2022](#)), the project utilizes Google Docs for annotators to manually correct OCR outputs. Another work ([Clematide et al., 2016](#)) developed a crowd-correction platform called Kokos to improve the quality of OCR outputs by engaging volunteers to correct digitized yearbooks written in German and French. Although manual corrections are simple, it can be time-consuming and costly as it requires recruiting and training many native speakers

of the language to be involved to produce numerous datasets.

Other prior works have attempted to overcome some of the shortcomings of these prior works by implementing an automation process for OCR post-correction. When there is no or limited dataset available for a language, using an OCR system's output to post-correct another OCR system's output by comparing the two can be an alternative. This has been done for post-OCR text correction in romanized Sanskrit ([Krishna et al., 2018](#)). Other work ([Poncelas et al., 2020](#)) has created a tool for correcting common errors in the Tesseract OCR engine for an English chapter book. These approaches however, can only be conducted by people who know how to program because the tool interface is in command-line format.

In addition, large language models (LLMs) have recently been employed to conduct automatic OCR post-correction. Pre-trained models such as ByT5 ([Löfgren and Dannélls, 2024](#)), which operates at the character level, have been utilized for Swedish. Furthermore, Google Vision AI toolkit combined with LSTM has been employed for endangered languages such as Ainu, Griko, and Yakka ([Rijhwani et al., 2020](#)). Additionally, a fully unsupervised character-based sequence-to-sequence NMT model has been applied for error correction in English and Finnish ([Duong et al., 2021](#)). State-of-the-art systems for OCR post-correction involves generative LLMs and the use of prompt-based approaches using LLAMA2 to successfully reduce character error rate for English newspapers ([Thomas et al., 2024](#)).

Several of these studies have also integrated OCR with web applications. However, many of these applications are trained in high-resource languages ([Cassidy, 2016](#)) such as English. Additionally, some projects are no longer maintained ([Reynaert, 2014](#)), rendering them inaccessible ([Weerasinghe et al., 2008](#)). In this work, we develop a platform called DriveThru for low-resource language document extraction and study the effectiveness of automatic OCR post-corrections for these languages.

## 3 System Description

DriveThru platform is inspired by the concept of fast-food restaurant drive-thru services, which allow customers to order food without leaving their vehicle, entering the restaurant, and ordering from

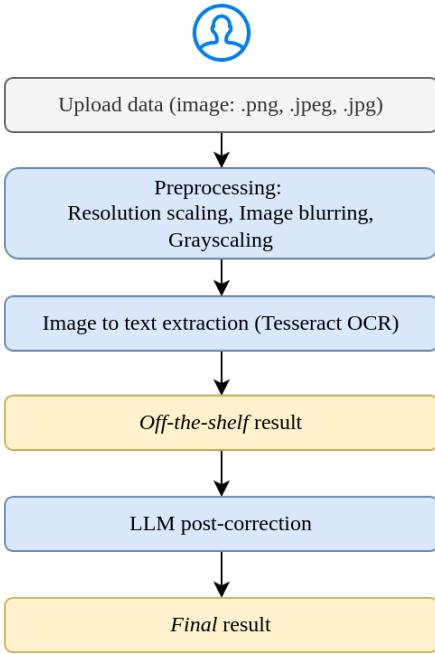


Figure 1: The figure shows how the platform’s extraction mechanisms work behind the scenes after users upload their images. The first step involves preprocessing the images. Then, Tesseract OCR extracts the text, producing the initial output. Finally, a language model will perform text correction to get the final results of document extraction.

a waitress. We use this philosophy in our system so that users do not need to create an account or log in to the apps; instead, they simply upload their document in the form of images (e.g., scans of printed documents). Finally, they can get the extracted text after the entire process takes place inside the DriveThru platform. The system workflow is shown in Figure 1.

DriveThru accepts several image formats including: `.png`, `.jpg`, or `.jpeg`. The maximum image uploaded in one cycle is 5 images. Any input surpassing this limit will be rejected by our system. The screen capture of DriveThru’s user interface is provided in Figure 3 in the Appendix.

### 3.1 Vocabulary Dataset

To construct one of our post-OCR correction model (i.e., the LLM’s *few-shot prompting* approach where we prompt LLM to correct an OCR output, providing potentially relevant words from dictionaries), we utilize a **word dictionary** that consists of paired vocabulary entries in bahasa Indonesia (i.e., Indonesian) and the Indonesian low-resource local language. While most of these entries are single words, some include multi-word expressions. These word pairs are extracted from dictionary books of the low-resource language (Table 1).

Language	Number of Pairs
Sunda	10831
Jawa	14680
Minang	12503
Bali	45120

Table 1: Number of vocabulary pairs for each language used in the training dataset.

### 3.2 Similar Words

The previously collected vocabulary dataset is instrumental in identifying the most similar words for each token in the OCR output to be corrected. We use the Longest Common Substring (LCS) algorithm that determines the longest sequence of shared characters between two words to measure their similarity effectively.

Following the similarity computation, we undertake a selection phase to refine the list of similar words for each token in the input text (i.e., the OCR output to be corrected). This selection process is structured as follows:

1. **Similarity Assessment:** Each token in the input text is compared to every word in the word dictionary using the LCS algorithm. Words that achieve a similarity score above a specified threshold are identified for further consideration.
2. **Relevance Filtering:** Words that display excessive similarity (more than  $K$  entries) are removed from the list, ensuring that only the most relevant matches are retained for analysis.
3. **Optimized Selection:** To maintain efficiency and manage prompt length, the number of similar word pairs is capped at a maximum of 10. When the number of relevant pairs exceeds this limit, a random sampling method is used to finalize the selection.

This systematic approach not only improves the precision of our post-OCR corrections but also guarantees that the process remains both efficient and scalable.

### 3.3 Benchmark Dataset

To evaluate our OCR post-correction, we leverage books, manuscripts, and magazines obtained from the National Library Of Indonesia<sup>2</sup> and the Indonesian Ministry of Education, Culture, Research, and

<sup>2</sup><https://www.perpusnas.go.id>

Technology (MoECRT) repository websites<sup>3</sup>. Documents obtained from those websites are following license states in Appendix A. The document titles that we use to evaluate our OCR post-correction approach are listed in the Table 3 in the Appendix.

### 3.4 Preprocessing

The DriveThru workflow starts with preprocessing of the uploaded image file, see Algorithm 1. We use the OpenCV<sup>4</sup> (cv2) python library in this work and begin by scaling the image resolutions times 1024 pixels if the uploaded image width is less than 1024 pixels. This is to make the processed image resolutions larger than 1024 pixels square and ensure that the image’s content does not disappear during the next preprocessing step. We use *cv2.INTER\_CUBIC* modules since it’s the only modules that produce high-quality and sharply focused images in our case.

---

#### Algorithm 1 Image resolution scaling

---

**Require:**  $w \leftarrow 1024$

```

 $W \leftarrow w$ 
if  $W < 1024$  then
     $ratio = 1024/W$ 
     $image = cv2.resize(image,$ 
         $fx = ratio, fy = ratio,$ 
         $interpolation = \backslash$ 
         $cv2.INTER_CUBIC)$ 
end if
 $gray = cv2.cvtColor(image,$ 
     $cv2.COLOR_BGR2GRAY)$ 
 $blur = cv2.GaussianBlur(gray, (3, 3), 0)$ 
 $thresh = cv2.threshold(blur, 0, 255,$ 
     $cv2.THRESH_BINARY_INV + \backslash$ 
     $cv2.THRESH_OTSU)[1]$ 
 $invert = 255 - thresh$ 

```

---

In addition, we apply the conversion of image to grayscale using *cv2.COLOR\_BGR2GRAY* module to prevent the OCR engines from being affected by color distractions. The next preprocessing step involves a blurring effect on the image using *cv2.GaussianBlur*, which is essential for eliminating any background noise. To remove noise from the image, we apply a threshold using *cv2.threshold* to retain the densely packed pixels and discard the sparse ones. Finally, we invert the image colors from black-to-white to white-

to-black by subtracting the threshold value from the 255 RGB value.

### 3.5 Model

#### OCR Engines (TesseractOCR)

The image-to-text process is conducted using TesseractOCR<sup>5</sup> without any fine-tuning of the base model. To achieve the most accurate extraction results compared to the ground truth, adjustments are made by configuring the OCR Engine mode (*-oem*) to 3 (default settings based on engine availability) and the Page Segmentation Mode (*-psm*) to 6. The language setting is left at its default, English, as the low-resource language documents we are extracting are written in Latin scripts.

To integrate TesseractOCR with a Python-based web application, we utilized PyTesseract<sup>6</sup> as an interface for the TesseractOCR within the Python programming environment. The output from the off-the-shelf OCR is subsequently processed through LLM post-correction using 3 distinct LLMs, which will be detailed in the following sections.

#### LLaMA 3

Llama 3<sup>7</sup> or Meta Llama 3 is a successor of Llama 2 (Touvron et al., 2023) which is a group of open-source pre-trained and instruction-tuned generative text models made by Meta AI (AI@Meta, 2024). It was released in two parameter sizes, 8B and 70B, both in pre-trained and instruction-tuned types. Llama 3 was pre-trained with over 15 trillion tokens from public data. The fine-tuning used public instruction datasets and more than 10 million human-annotated examples. We choose Llama 3 as it has the best accuracy as well as performance among existing LLM model that is pre-trained on English-centric data. This project uses **Meta-Llama-3-70B-Instruct** model for our post-OCR correction model. This model is employed in both zero-shot and few-shot prompting scenarios to correct misspelled words in Indonesian local languages. By integrating this advanced model, we effectively address spelling errors in local languages, demonstrating the versatility and robustness of our approach.

#### GPT-4

GPT-4, the successor to GPT-3.5, is a large-scale and multimodal model capable of handling both text and image inputs and generating text outputs

---

<sup>5</sup><https://github.com/tesseract-ocr>

<sup>6</sup><https://github.com/h/PyTesseract>

<sup>7</sup><https://github.com/meta-llama/Llama3>

---

<sup>3</sup><https://repository.kemdikbud.go.id>

<sup>4</sup><https://github.com/opencv/opencv>

(OpenAI, 2024). It surpasses human performance in various professional and academic assessments. This large language model excels in bug improvement and reinforces foundational knowledge from previous models and most state-of-the-art systems. GPT-4 demonstrates superior English-language performance compared to other large language models, including its predecessor GPT-3.5, and also performs well in low-resource languages such as Latvian, Welsh, and Swahili. However, GPT-4 retains some limitations similar to previous models, including the occurrence of hallucinations, a limited context window, and an inability to learn from previous events. We employ GPT-4 for post-correction of OCR outputs due to its pre-training with the Indonesian language. Similar to the previous model, this model is utilized in both zero-shot and few-shot prompting contexts to correct misspelled words in Indonesian indigenous languages.

### Post-Correction Approach

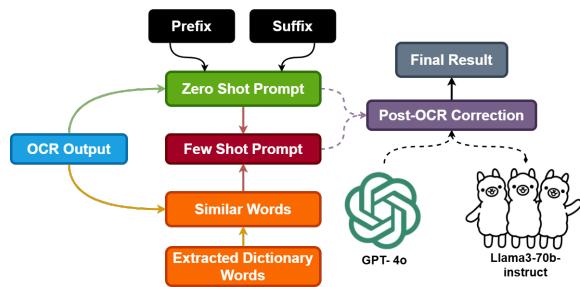


Figure 2: The figure shows our post-OCR correction flow with 2 approach: few-shot and zero-shot prompting.

As outlined above, this project employs two advanced techniques for post-OCR correction: zero-shot and few-shot prompting. These techniques are implemented as follows (Figure 2):

- Vocabulary Dataset Collection:** As described in Section 3.1, we collect vocabulary datasets for each local language. These datasets are essential for identifying the most relevant similar words for each token in the input text i.e., the OCR output to be corrected.
- Similar Words Formation:** We form a list of similar words by comparing the words in the dictionary with each token in the input text, focusing on those with the highest similarity scores. For a detailed explanation of the similarity calculation mechanism, refer to Section 3.2.

**3. Prompt Construction:** After obtaining the relevant data, we construct a prompt that includes a prefix (i.e., "Fix the grammar of the following text"), suffix (i.e., "The following are potentially similar words from the dictionary"), and the input text (i.e., OCR output) itself. In zero-shot prompting, the prefix and input text are directly input into the Large Language Model (LLM) as an instruction to execute. In contrast, for few-shot prompting, the prompt is enhanced with the suffix that contains additional hints in the form of a list of word pairs from the input text and similar vocabulary from the dictionary.

**4. Result Generation:** The final output is generated based on the responses provided by the LLM. This output incorporates the corrections suggested by the model based on the input and the prompt.

## 4 Results and Discussion

DriveThru was developed to assist future NLP researchers, students, scholars, data scientists, organizations, language enthusiasts, and other entities with their language resource collection tasks. We aim to make the system's interface as simple as possible, even if no instructions are provided in beforehand.

Using DriveThru is straightforward. As shown in Figure 3 in the Appendix, users can upload files by simply dragging and dropping them from their file managers into the designated area, or if users prefer to browse their file through the "browse files" dialog boxes it can be achieved by clicking on the drag-and-drop area. The platform allows for the upload of up to five files simultaneously. If a file is mistakenly uploaded, it can be removed by clicking the cross icon below the drag-and-drop area, or by selecting the light gray "Clear" button to remove all entries. To proceed with the uploaded files, users can click the red "Proceed" button.

### Evaluation

The off-the-shelf (OTS) TesseractOCR engine is considered capable of recognizing the majority of the provided images, though some hallucinations occur. As shown in Table 5 in the Appendix, in terms of word counts, OTS Tesseract produces a higher total word count than the ground truth (GT), with a difference of 2,699 words. According to the CAR in Table 2 and WAR scores in Table 6

CAR					
Language	OTS	Llama3 (ZS)	Llama3 (FS)	GPT-4 (ZS)	GPT-4 (FS)
1 Balinese	0.943	0.917	0.919	0.893	0.914
2 Javanese	-0.993	0.970	0.956	0.965	0.965
3 Sundanese	0.911	0.738	0.168	-0.368	-0.699
4 Minangkabau	0.958	0.942	0.942	0.924	0.926
avg (%)	0.45475	<b>0.892</b>	0.746	0.603	0.526

Table 2: Percentage of Character Accuracy Rates (CAR) on different extraction techniques. The off-the-shelf (OTS) column means there are no additional steps to repair the extracted text, while the others perform additional post-correction steps involving LLMs. Overall, Llama3 with a Zero-shot approach outperforms other LLMs in post-correction OCR.

in the Appendix, OTS Tesseract demonstrates the highest accuracy. However, it struggles with detecting images containing Javanese document archives, resulting in the lowest score among the other languages.

Applying zero-shot learning for post-correction OCR improves the average word and character accuracy of Llama3 models. Even yet, it appears that the Balinese, Sundanese, and Minangkabau scores do not differ significantly from the OTS scores. In contrast, for Javanese, there was a significant improvement from below zero percent to above 50 percent, leading to fewer hallucinations than before.

Even if the scores for some approaches (either zero-shot or few-shot) are better than the OTS, it may not be sufficient. This is because in our human annotation file, all entries are written as they appear, regardless of whether they are correct or not by *lexical rules*. Post-OCR correction using zero-shot or few-shot techniques can achieve more by fixing punctuation, hyphenated words, removing unclear parts that were misinterpreted by OCR, and more. However, if the OCR output is severely distorted, post-OCR correction still faces significant challenges.

From the text above, further details can be illustrated through qualitative examples in Table 4 in the Appendix:

1. **Example 1** in Table 4 shows that post-OCR correction can resolve the hyphenated word problem when scanning a document using OCR. For instance, "ndu- weni" is replaced by "nduweni", "ba- nget" is replaced by "banget", and "ma- mah" is replaced by "mamah".

2. **Example 2** in Table 4 demonstrates that post-OCR correction can remove unclear parts from the image that have been scanned with OCR if they are not detected as part of the language vocabulary. It removes "Bataan ceeiaalteleea Saati: St meinen, aa" because the post-OCR correction detects that

this string is merely an unclear scanned part from the image and is not included in the language vocabulary.

3. **Example 3** in Table 4 illustrates that post-OCR correction can also solve problems related to punctuation usage. The post-OCR correction can remove unclear parts from the text and add a period at the end of the sentence.

## Limitations

The RagamBahasa platform has not yet been integrated with a post-processing model due to a lack of independent computing resources for the project. Currently, the models are temporarily deployed on high-performance computing resources of an academic institution, which are shared with other research projects. We have not applied enhancements to the OCR engine to reduce character misinterpretation during text recognition. While it is adequate for extracting Indonesian local languages in Latin script, improvements are needed for reading regional scripts (e.g., Javanese, Balinese, Sundanese, etc.). Our benchmarks have only been demonstrated on languages classified as Institutional-Stable by Ethnologue. Despite being low-resource languages, scanned language resources are relatively accessible online. This effort should be expanded to include languages classified as endangered, such as Betawi, Acehnese, and Wolio, with more diverse capturing techniques. Regardless of the limitations of our tools, this study meets our requirements for building a language resource database of Indonesian local languages in the future.

In addition, as illustrated in Example 4 from Table 4, when the OCR output is highly distorted or unclear, it becomes exceedingly difficult to accurately identify the actual text, even with advanced LLMs using zero-shot or few-shot prompting. This limitation underscores the challenge of dealing with severely degraded OCR input and highlights

the need for further improvements in both OCR technology and post-processing techniques to handle such cases more effectively.

## Ethics Statement

The primary objective of this project is to utilize scanned language resources of Indonesian local languages available online in the Indonesian government archives repository. This initiative is also funded by the same governmental entity to ensure alignment between the research objectives and the funding body. All datasets and source code used in this study are publicly accessible, with copyright and licensing details duly specified.

## Acknowledgments

We are grateful for the financial support and technical assistance on this research provided by the Indonesian Ministry of Education, Culture, Research, and Technology (MoECRT) and the Indonesia Endowment Fund for Education (LPDP) through the ACE Open Research program, part of the US-Indonesia collaboration program. We would like to thank Boston University for providing computing facilities through Shared Computing Cluster (SCC) and Monash University Indonesia for the essential to the success of this project.

## References

AI@Meta. 2024. [Llama 3 model card](#).

Alham Fikri Aji, Genta Indra Winata, Fajri Koto, Samuel Cahyawijaya, Ade Romadhony, Rahmad Mahendra, Kemal Kurniawan, David Moeljadi, Radityo Eko Prasojo, Timothy Baldwin, Jey Han Lau, and Sebastian Ruder. 2022. [One country, 700+ languages: NLP challenges for underrepresented languages and dialects in Indonesia](#). In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7226–7249, Dublin, Ireland. Association for Computational Linguistics.

Samuel Cahyawijaya, Holy Lovenia, Alham Fikri Aji, Genta Winata, Bryan Wilie, Fajri Koto, Rahmad Mahendra, Christian Wibisono, Ade Romadhony, Karissa Vincentio, Jennifer Santoso, David Moeljadi, Cahya Wirawan, Frederikus Hudi, Muhammad Satrio Wicaksono, Ivan Parmonangan, Ika Alfina, Ilham Firdausi Putra, Samsul Rahmadani, Yulianti Oenang, Ali Septiandri, James Jaya, Kaustubh Dhole, Arie Suryani, Rifki Afina Putri, Dan Su, Keith Stevens, Made Nindyatama Nityasya, Muhammad Adilazuarda, Ryan Hadiwijaya, Ryandito Diandaru, Tiezheng Yu, Vito Ghifari, Wenliang Dai, Yan Xu,

Dyah Damapusita, Haryo Wibowo, Cuk Tho, Ichwanul Karo Karo, Tirana Fatyanosa, Ziwei Ji, Graham Neubig, Timothy Baldwin, Sebastian Ruder, Pascale Fung, Herry Sujaini, Sakriani Sakti, and Ayu Purwarianti. 2023a. [NusaCrowd: Open source initiative for Indonesian NLP resources](#). In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 13745–13818, Toronto, Canada. Association for Computational Linguistics.

Samuel Cahyawijaya, Holy Lovenia, Fajri Koto, Dea Adhistha, Emmanuel Dave, Sarah Oktavianti, Salsabil Akbar, Jhonson Lee, Nur Shadieq, Tjeng Wawan Cenggoro, Hanung Linuwih, Bryan Wilie, Galih Muridan, Genta Winata, David Moeljadi, Alham Fikri Aji, Ayu Purwarianti, and Pascale Fung. 2023b. [NusaWrites: Constructing high-quality corpora for underrepresented and extremely low-resource languages](#). In *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 921–945, Nusa Dua, Bali. Association for Computational Linguistics.

Steve Cassidy. 2016. [Publishing the trove newspaper corpus](#). In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 4520–4525, Portorož, Slovenia. European Language Resources Association (ELRA).

Simon Clematide, Lenz Furrer, and Martin Volk. 2016. [Crowdsourcing an OCR gold standard for a German and French heritage corpus](#). In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 975–982, Portorož, Slovenia. European Language Resources Association (ELRA).

Senka Drobac, Pekka Kauppinen, and Krister Lindén. 2017. [Ocr and post-correction of historical finnish texts](#). In *Proceedings of the 21st Nordic Conference on Computational Linguistics*, page 70–76, Gothenburg, Sweden. Association for Computational Linguistics.

Quan Duong, Mika Hämäläinen, and Simon Hengchen. 2021. [An unsupervised method for ocr post-correction and spelling normalisation for finnish](#). In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, page 240–248, Reykjavík, Iceland (Online). Linköping University Electronic Press, Sweden.

David M. Eberhard, Gary F. Simons, and Charles D. Fennig. 2024. [Ethnologue: Languages of the World](#), twenty-seventh edition. SIL International, Dallas, Texas.

Mika Koistinen, Kimmo Kettunen, and Tuula Pääkkönen. 2017. [Improving optical character recognition of finnish historical newspapers with a combination of fraktur & antiqua models and image preprocessing](#). In *Proceedings of the 21st Nordic Conference on*

- Computational Linguistics*, page 277–283, Gothenburg, Sweden. Association for Computational Linguistics.
- Amrith Krishna, Bodhisattwa P. Majumder, Rajesh Bhat, and Pawan Goyal. 2018. Upcycle your ocr: Reusing ocrs for post-ocr text correction in romanised sanskrit. In *Proceedings of the 22nd Conference on Computational Natural Language Learning*, page 345–355, Brussels, Belgium. Association for Computational Linguistics.
- Viktoria Löfgren and Dana Dannélls. 2024. Post-ocr correction of digitized swedish newspapers with byt5. In *Proceedings of the 8th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature (LaTeCH-CLfL 2024)*, page 237–242, St. Julians, Malta. Association for Computational Linguistics.
- Sanjib Narzary, Maharaj Brahma, Mwnthai Narzary, Gwmsrang Muchahary, Pranav Kumar Singh, Apurbalal Senapati, Sukumar Nandi, and Bidisha Som. 2022. Generating monolingual dataset for low resource language Bodo from old books using Google keep. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 6563–6570, Marseille, France. European Language Resources Association.
- OpenAI. 2024. Gpt-4 technical report.
- Alberto Poncelet, Mohammad Aboomar, Jan Buts, James Hadley, and Andy Way. 2020. A tool for facilitating ocr postediting in historical documents. In *Proceedings of LT4HALA 2020 - 1st Workshop on Language Technologies for Historical and Ancient Languages*, page 47–51, Marseille, France. European Language Resources Association (ELRA).
- Martin Reynaert. 2014. TICCLops: Text-induced corpus clean-up as online processing system. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: System Demonstrations*, pages 52–56, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.
- Shruti Rijhwani, Antonios Anastasopoulos, and Graham Neubig. 2020. Ocr post correction for endangered language texts. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, page 5931–5942, Online. Association for Computational Linguistics.
- Alan Thomas, Robert Gaizauskas, and Haiping Lu. 2024. Leveraging llms for post-ocr correction of historical newspapers. In *Proceedings of the Third Workshop on Language Technologies for Historical and Ancient Languages (LT4HALA) @ LREC-COLING-2024*, page 116–121, Torino, Italia. ELRA and ICCL.
- Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaee, Nikolay Bashlykov, Soumya Batra, Prajwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madijan Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models.
- Ruvan Weerasinghe, Asanka Wasala, Dulip Herath, and Viraj Welgama. 2008. NLP applications of Sinhala: TTS & OCR. In *Proceedings of the Third International Joint Conference on Natural Language Processing: Volume-II*.
- Genta Indra Winata, Alham Fikri Aji, Samuel Cahyawijaya, Rahmad Mahendra, Fajri Koto, Ade Romadhony, Kemal Kurniawan, David Moeljadi, Radityo Eko Prasojo, Pascale Fung, Timothy Baldwin, Jey Han Lau, Rico Sennrich, and Sebastian Ruder. 2023. NusaX: Multilingual parallel sentiment dataset for 10 Indonesian local languages. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 815–834, Dubrovnik, Croatia. Association for Computational Linguistics.

## A Copyright and License

In addition to following the author's license, we follow the Indonesian MoECRT repository policy<sup>8</sup> which states "...pengguna yang menggunakan sumber informasi dari Repozitori Institusi Kemendikbudristek harus menuliskan atribut sumber yang digunakan dan tidak digunakan untuk tujuan komersial." This means that anyone who uses resources from the repositories must acknowledge the source and not commercialize the work when disseminating it.

Align with the consideration, the content produced by DriveThru is licensed under CC-BY NC 4.0, which means that people can use our platform to create dataset they need without commercializing or disrespecting the authors' work.

---

<sup>8</sup><https://repositori.kemendikbud.go.id/information.html>

#	Language	ISO	Source Title	Genre	Total of Image
1	Balinese	ban	I Bagus Caratan	Children	26
2	Balinese	ban	Leak Pemoroan	Short Story	24
3	Balinese	ban	Ejaan Bahasa Daerah Bali yang Disempurnakan 1974	Learning Book	13
4	Balinese	ban	Majalah Suara Saking Bali Edisi VII	Magazine	12
5	Balinese	ban	Paparikan Lawe	Learning Book	10
6	Balinese	ban	Pedoman Umum Ejaan Bahasa Bali Dengan Huruf Latin	Learning Book	15
7	Javanese	jav	Panjebar Semangat	Magazine	100
8	Sundanese	sun	Carita ti Carita	Short story	5
9	Sundanese	sun	Hayam Gecok Ngeunah	Short story	5
10	Sundanese	sun	Mangle	Magazine	25
11	Sundanese	sun	Wawacan Rengganis	Story Book	16
12	Sundanese	sun	Wawacan Sejarah Anbia	Story book	10
13	Sundanese	sun	Raja Neger Jeung Bangsa Arab	Story book	17
14	Sundanese	sun	Lain Eta	Story book	12
15	Sundanese	sun	Istri kasasar	Story book	10
16	Minangkabau	min	Kaba Bujang Paman Dan Kaba Rambun Pamenan	Novel	55
17	Minangkabau	min	Kaba Kambang Luari Sutan Pangaduan	Novel	45
<b>Total</b>					<b>400</b>

Table 3: A list of document titles from four different languages with their respective genres. We collected 335 images from a total of 12 books spanning various genres in language resource archives to use for benchmark datasets.

Image	OCR Output	Human Annotation	Post-OCR Correction
	Sing unik maneh, Bekecot pranyata ndu weni untu kang uakehhh banget lo. Adhik-adhik ngerti pira jumlahé? Pranyata untune Bekecot ana 14.175. Untu iki cilik-cilik banget lan wujude kaya parut. Untu iki bisa ma-mah gegodhongan kang akeh banget jroning sewengi. : :	Sing unik maneh, Bekecot pranyata ndu weni untu kang akeh banget lo. Adhik-adhik ngerti pira jumlahé? Pranyata untune Bekecot ana 14.175. Untu iki cilik-cilik banget lan wujude kaya parut. Untu iki bisa ma-mah gegodhongan kang akeh banget jroning sewengi.	Sing unik maneh, Bekecot pranyata ndu weni untu kang akeh banget lo. Adhik-adhik ngerti pira jumlahé? Pranyata untune Bekecot ana 14.175. Untu iki cilik-cilik banget lan wujude kaya parut. Untu iki bisa ma-mah gegodhongan kang akeh banget jroning sewengi.
	Bataan ceeaalteleea Saati: St meinen, aa Terekel Salim naek kena kai, rak ngadon sare. Teu wanieun sare di handap, sieun disampeurkeun sato galak, didodoho eukeur sare.	Terekel Salim naek kena kai, rak ngadon sare. Teu wanieun sare di handap, sieun disampeurkeun sato galak, didodoho eukeur sare.	Terekel Salim naek kena kai, rak ngadon sare. Teu wanieun sare di handap, sieun disampeurkeun sato galak, didodoho eukeur sare.
	.eRe ; Naskah Supersemar jing tangane Pak Harto we	Naskah Supersemar ing tangane Pak Harto	Naskah Supersemar ing tangane Pak Harto.
	tn anit ORI ETS .2.Rtee Deira ty cATE SS PEA LIA AROS TA TOS Bie > 2% NB fe ? 4 oh ry fr any, ct sted ered Sue, ek Acnta, Fil ee na) al Ne la eet tO tee a! aD OP TI PE HOOT Eas	Pangudarasa	Anit ORI ETS. 2. Rtee Deira ty cATE SS PEA LIA AROS TA TOS Bie > 2% NB fe? 4 oh ry fr any, ct sted ered Sue, ek Acnta, Fil ee na) al Ne la eet tO tee a! aD OP TI PE HOOT Eas.

Table 4: Comparison of OCR Output, Human Annotation, and Post-OCR Correction with Few-Shot Approach in GPT-4o

Languages				
	Balinese	Javanese	Sundanese	Minangkabau
<b>GT</b>	16138	12300	18558	30368
<b>OTS</b>	16207	14471	18771	30614
<b>Llama3 (FS)</b>	15955	12897	18524	30490
<b>Llama3 (ZS)</b>	16034	11979	18513	30389
<b>GPT-4 (FS)</b>	16010	13775	18641	30232
<b>GPT-4 (FS)</b>	15941	13123	18606	29785

Table 5: Number of tokens retrieved from Indonesian local language archives extraction, comparing between the number of ground-truth (GT), off-the-shelf (OTS) TesseractOCR, Llama3 and GPT-4 both with Few-shot (FS) and Zero-shot (ZS), respectively.

WAR					
	Language	OTS	Llama3 (ZS)	Llama3 (FS)	GPT-4 (ZS)
1	Balinese	0.777	0.808	0.818	0.795
2	Javanese	-4.04	0.532	-0.966	-2.080
3	Sundanese	0.777	0.903	0.872	0.760
4	Minangkabau	0.866	0.806	0.779	0.879
	<b>avg (%)</b>	-0.405	<b>0.762</b>	0.375	0.088
					-0.147

Table 6: Percentage of Word Accuracy Rates (WAR) on different extraction techniques. The off-the-shelf (OTS) column means there are no additional steps to repair the extracted text, while the others perform additional post-correction steps involving LLMs. Overall, Llama3 with a Zero-shot approach performs well compared to the other LLM in all languages.



### OCR-Drive Thru

OCR-Drive Thru (DT) merupakan sebuah alat yang dapat digunakan oleh siapa saja untuk kebutuhan ekstraksi teks dari sebuah (atau lebih) gambar dengan tingkat akurasi yang tinggi. Hasil produksi dari OCR-DT dilisensikan di bawah lisensi Creative Commons CC-BY-NC-SA 4.0. Sebelum menggunakan anda dapat membaca instruksi yang telah kami tuliskan di bawah ini.

1. OCR-DT hanya menerima masukan berupa file gambar dengan format: .png, .jpg, .jpeg.
2. OCR-DT membatasi max. 5 makro yang dapat diproses dalam satu kali tekan.
3. OCR-DT bersifat membantu proses ekstraksi teks secara otomatis, hal ini dapat mengurangi beban kerja dibanding menuliskan teks dari sebuah gambar secara manual.
4. Harap jangan memasukkan informasi/dokumen yang bersifat sensitif pada OCR-DT, atau tanggung jawab ada pada pengguna.

Drop your files here or click to upload

Screenshot23.png x Screenshot24.png x

Clear Proceed

#1: Screenshot23.png

tusing ja keweh mapan ia anak sugih. Mlajah ilmu listrik jadwalne duang minggu cepok. Sawireh Pak Agus nawang Kak Badung teken Nang Lotok tusing taen masrekenan mesuang pipis anggon mayyah maguru, sabilang jadwal tagina pipis nyang satak tali rupiah. Lenan teken Nang Lotok ada masi ane ngamiliuin ajaka patpat, makejeng totongan soroh anak suba tengah tuwu.

#2: Screenshot24.png

Kasuwen-suwen Nang Lotok Cs. marasa kerud, mapan sabilang lathian setatai gurune nagih pipis. Diapinke Kak Badung tusing keweh mapan ia anak sugih. Nang Lotok mare nyidhang nyemak strum rosas tali watt ngelaut suud mlajah, alasane tusing ngelah pipis. Apa buin Pak Agus setatai nagih pipis pamelin lengis binisin. Sasubane Nang Lotok suud mlajah nyemak strum, Pak Agus biin ngeka-ngeka daya, kenken baan apang nyidhang nyuang pipisne Nang Lotok teken Kak Badung, mapan tusing nyuk nugtugang mlajah nyemak strum listrik. Pak Agus suba kadung turunan molaksana corah, mula bakal tusing suud- suud ngeka daya kenken baan nguluk-nguluk anak, kenken baan apang nyidhang mapikolih arta, keto masih mapan tunben maan murid loyar pesan mesuang pipis. Sawatara duang minggu Pak Agus sagetang biin ngenah kumah Kak Badung sambilanga makecak-kecal ngortang ibane bisa narik barang-barang sunira marupa kadutan, akit, permata mirah delima, lan sekancanin soca duwen anake di suniara. Kak Badung anak kabenneng demen teken barang-barang buka keto, beh jog nyantep sajan dot mlajah narik barang ane orang teken Pak Agus.

Figure 3: Screen capture of DriveThru application interface, It shows the instruction to be considered below the application title, it also shows how the application previews two OCR outputs of the uploaded files below the drag-and-drop area