UNIVERSITÀ DI PISA

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

Master's Degree in Artificial Intelligence and Data Engineering

Plūma

Scripta Volant, Digital Manent

Group:

Daniel Namaki Ghaneh

Emanuele Respino

Lorenzo Vittori

Plūma GitHub Repository: https://github.com/NamaWho/pluma

ACADEMIC YEAR 2023/2024

Abstract

This paper introduces Plūma, an innovative solution designed to address the challenge of managing and digitizing handwritten notes in business and educational contexts. In an age where digitization is key to operational efficiency and information management, Plūma transforms handwritten notes into digital formats with high accuracy and ease. The introduction outlines the need for such a solution, highlighting benefits like improved accessibility, simplified sharing, secure archiving, and enhanced operational efficiency.

The application workflow is detailed, showcasing features such as notes upload, text recognition via the advanced Gemini LLM model, text conversion into various digital formats, and an AI-driven note enhancement feature that enriches the digitized content by correcting semantic errors and expanding on transcribed concepts. The paper discusses the underlying technology, including the Gemini API, which leverages neural network techniques for high-precision handwriting recognition.

Performance metrics demonstrate the superior accuracy of Plūma's model over traditional OCR systems, attributed to its adaptability to different writing styles and context-aware text processing. The application's limitations are addressed, including file size constraints and the maximum number of pages that can be transcribed simultaneously, with options for customization to optimize performance.

The conclusion emphasizes the significant advancements Plūma brings to handwriting recognition and digitization. Future developments include real-time processing, support for additional formats, user interface enhancements, cloud integration, customization options, and integration with other productivity tools. Plūma is poised to revolutionize the management and utilization of handwritten notes, offering a robust solution for businesses, educational institutions, and professionals.

Contents

Chapter 1

Introduction

Plūma aims to solve one of the most common yet overlooked challenges in business and education: managing and digitizing handwritten notes. In an era where digitization is essential for operational efficiency, information management, and sustainability, Plūma offers an innovative solution to transform handwritten notes into digital formats in a simple and accurate way.

1.1 The Need for Digitalization

In the modern business context, efficiency and speed in information management are crucial. Handwritten notes, while useful and often indispensable during meetings, brainstorming sessions, and training sessions, pose a challenge when it comes to archiving, searching, and sharing. Converting these notes into digital formats greatly facilitates these operations, improving accessibility and productivity.

1.1.1 Benefits of Digital Conversion

Converting handwritten notes into digital formats offers numerous advantages:

- **Search and accessibility:** Digitized documents can be easily searched and retrieved, saving valuable time.
- **Simplified sharing:** Digital files can be instantly shared with colleagues and collaborators, regardless of their geographic location.
- **Secure archiving:** Digitization reduces the risk of information loss due to physical damage or misplacement of notes.
- **Operational efficiency:** Improves operational efficiency by automating the transcription and note archiving process.

Chapter 2

Application

The application is designed to manage the entire process of converting handwritten notes into various digital formats and possibly, on user request, provide an enhanced version of those notes. Here is a detailed description of the application workflow.

2.1 Workflow

Notes Upload

Users can upload handwritten notes using an intuitive user interface or by specifying the PDF file path. When the upload is complete, a preview is displayed on the screen.

Figure 2.1: Main page of the application

Text Recognition

The core of the program comprises Gemini LLM, an advanced multi-modal artificial intelligence model, employing a deep neural network. This network not only analyzes images to identify characters and words but also considers the contextual elements of

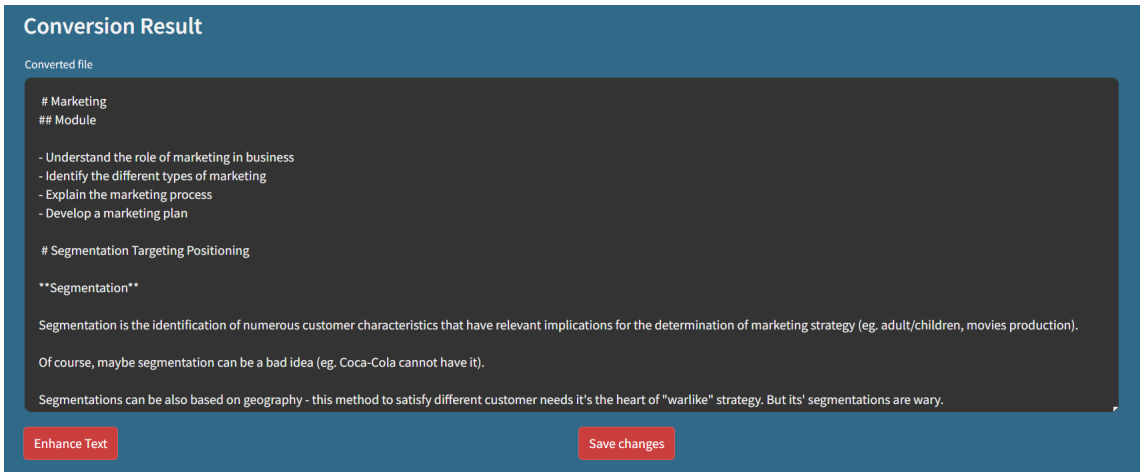

Figure 2.2: Results are displayed allowing the user to modify them

the notes, thereby significantly enhancing its precision and accuracy. This flexibility allows the model to correctly reconstruct words and phrases, despite grammatical errors and mismatching characters conversion.

Text Conversion

Once recognized, the text is converted into the desired format specified by the user. Supported formats include:

- **Plain text (.txt):** A format readable by any text editor.
- **Markdown (.md):** A lightweight markup language with plain text formatting syntax.
- **LaTeX (.tex):** A high-quality typesetting system, for academic and professional purposes.

Saving and Exporting

The converted text can be saved locally, facilitating sharing and accessibility from any device.

2.2 Note Enhancement feature

The application introduces a feature that leverages advanced artificial intelligence to not only digitize handwritten notes but also enhance them. This feature searches the web for relevant information on the topic, corrects semantic errors, and expands on superficially transcribed concepts. This capability significantly enriches the quality and depth of the notes, making them more comprehensive and valuable. The

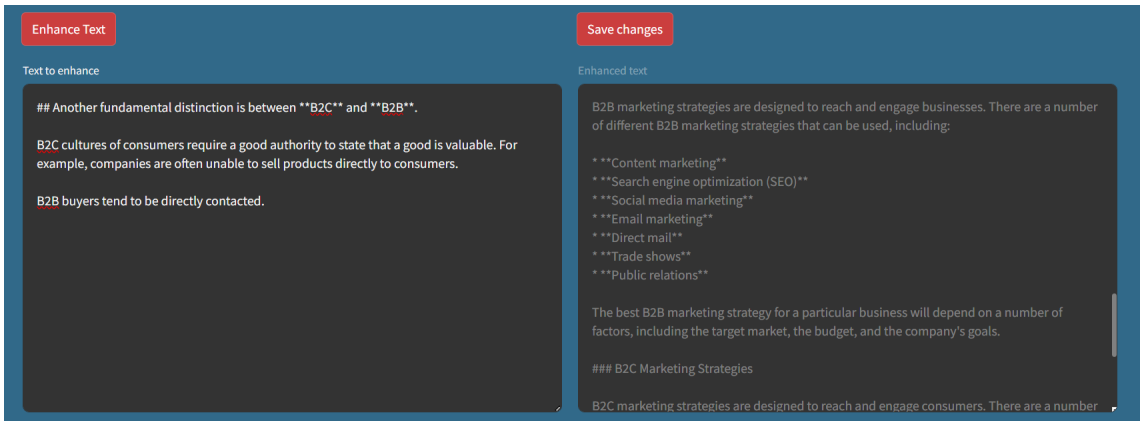

Figure 2.3: User can specify text sections to improve its content

enhancement process is designed to be user-friendly and highly effective. The interface is structured with two text boxes: the first is an input box where users enter the original part of the notes they want to improve, while the second box displays the suggested corrected and enhanced note from the Gemini LLM model.

- **User input:** The user inputs the section of notes they want to improve in the first box, presses the enhancement button, and in the second box, the suggested text improvement provided by the Gemini LLM model appears.
- **Text recognition:** The application processes the handwritten notes, converting them into digital text using the specialized AI model.
- **Web search and information retrieval:** The system searches the web for additional information related to the topics identified in the notes. This involves using natural language processing (NLP) techniques to understand the context and extract relevant information.
- **Semantic correction:** The system AI corrects any semantic errors in the transcribed text, ensuring that the notes are grammatically correct and coherent.
- **Concept expansion:** The application expands on the concepts that are superficially transcribed, adding depth and detail to the notes. This involves integrating the information retrieved from the web and contextualizing it within the existing notes.

2.3 Application limits

The application is designed with efficiency and flexibility in mind, allowing for the seamless handling of text and data:

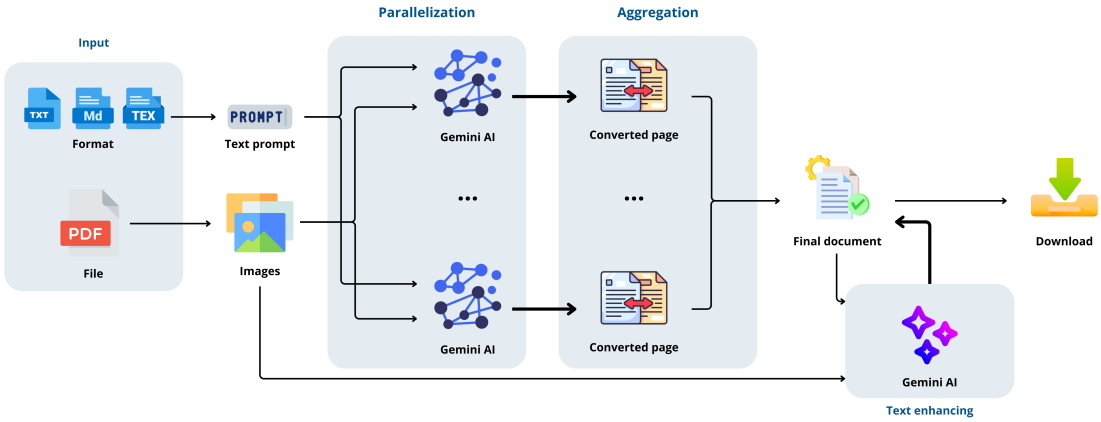

Figure 2.4: Overall Plūma application workflow

- **Resource usage:** The maximum file size that can be uploaded is 200MB. This limit is predefined by the Streamlit application on which the developed interface is based, to prevent excessive resource usage. However, this does not preclude the possibility of increasing the limit. Additionally, using Python allows for the management of larger files, constrained only by the resources available on the machine running the application. Handling large files may impact the application's performance.

- **Transcription limits and time:** Regarding the maximum text length of the uploaded file, the application is designed to require the **separate transcription of each page**. This allows for the **parallelization** of transcription, minimizing the total execution time. Gemini has a maximum response limit of 8,192 tokens. Given an average of 0.75 words per token in English and an average of 4 characters per token, each response can handle approximately **6,144 words** or **32,768 characters**. This also sets the maximum limit of words and characters that can be obtained from each page, including text formatting. To avoid performance issues, we have set the default maximum number of pages that can be transcribed simultaneously to 10. However, this limit can be adjusted to maximize the file conversion speed.

- **Tables and images handling:** The model used by the application can extract text from images, tables, and charts, attempting to recreate the original formatting as closely as possible, accordingly to the required file format.

2.4 Gemini API

Gemini API[1] is designed to convert handwritten notes into digital format with high accuracy and efficiency. It leverages advanced neural network techniques to recognize and transcribe handwritten text. This section delves into the neural net-

work architecture behind Gemini and explains why it performs exceptionally well in this task.

The Gemini API leverages powerful machine learning techniques to convert your handwritten notes into digital text. While the specific details of the underlying neural network are not publicly available, we can discuss the general principles that contribute to its effectiveness.

- **Extracting Features from Images**: Similar to how we recognize shapes in images, Gemini employs advanced algorithms to identify patterns and strokes within your handwritten notes. This initial processing helps the system understand the fundamental building blocks of the text.
- **Modeling Sequential Information**: Handwriting involves characters arranged in a specific order. Gemini utilizes a special type of neural network called a Recurrent Neural Network (RNN)[2] to analyze this sequential nature. RNNs are adept at understanding the relationships between characters, allowing Gemini to decipher the flow of your writing.
- **Decoding and Refining the Output**: Once Gemini extracts features and analyzes their order, it employs a technique called Connectionist Temporal Classification (CTC) to convert the processed information into actual text. CTC excels at aligning the predicted characters with the handwritten sequence, ensuring an accurate representation of your notes.

2.4.1 Key Factors Contributing to Accuracy

- **Extensive Training**: Gemini is likely trained on a massive dataset of handwritten text, encompassing various writing styles and qualities. This comprehensive training allows the system to adapt to different handwriting variations and improve its recognition accuracy over time.
- **Data Augmentation Techniques**: To further enhance its robustness, Gemini might utilize data augmentation techniques. This involves artificially creating variations of the training data, such as rotations or slight distortions, helping the system handle real-world variations in handwritten notes.

2.4.2 Gemini Pro Vision

Gemini Pro Vision[3] is an advanced model within the Gemini suite, specifically designed for the intricate task of handwriting recognition. This model employs a sophisticated combination of convolutional neural networks (CNNs) and recurrent neural networks (RNNs) to achieve high accuracy in interpreting handwritten text. The CNNs are adept at extracting detailed visual features from the input images, identifying individual characters and their nuances. Meanwhile, the RNNs, particularly those with Long Short-Term Memory (LSTM) units, excel in understanding

the sequential nature of handwriting, capturing the contextual flow and structure of the text. This dual approach enables Gemini Pro Vision to not only recognize isolated characters but also to comprehend the overall context of the notes, leading to a significant enhancement in both precision and accuracy. The model's ability to generalize across various handwriting styles and conditions makes it an invaluable tool for digitalizing handwritten documents with a high degree of reliability.

Chapter 3

Analysis and Comparison

Handwriting recognition poses a significant challenge compared to printed text recognition. Traditional OCR models, such as Tesseract [4], are optimized for printed and structured text but struggle with the variability and complexity of handwriting. In this context, the model used in this application offers an innovative and specialized approach to address these challenges.

3.1 Traditional OCR Models

Traditional OCR [5] models are highly optimized for printed text recognition. However, they present several limitations when applied to handwriting recognition:

- **Dependence on regular structures:** Work best with text that follows a regular and predictable structure.
- **Limited adaptability:** Struggle with significant variations in handwriting and require highly specific datasets to function correctly.
- **Standard preprocessing:** Use standard preprocessing techniques that are not always sufficient to improve the quality of handwritten text.

3.2 Application Neural Model

The model implemented in the application is specifically designed for handwriting recognition, with the following key features:

- **Deep neural network:** The application is based on a deep neural network trained on a vast dataset of handwritten notes, enabling the model to recognize a wide range of writing styles.
- **A dynamic approach:** The specific model used is a transformer model, extremely dynamic and adaptable to the different style in handwriting.

- **Advanced preprocessing:** Includes preprocessing techniques which significantly improve the quality of extracted text.
- **Context integration:** Uses post-processing algorithms that consider the context of words and sentences to improve recognition accuracy.

3.3 Comparison table

FEATURE	PLŪMA APPLICATION	GENERIC TRANSCRIPTION APPLICATION
MODEL	DEEP-LEARNING MULTI-MODAL NEURAL NETWORK	OCR NEURAL NETWORK
CHARACTERS RECOGNITION	✓	✓
CONTENT FORMATTING	✓	✓
CONTEXT RECOGNITION	✓	✗
GRAMMATICAL ERROR CORRECTION	✓	✗
CONTENT ENHANCEMENT	✓	✗
HANDWRITING FLEXIBILITY	✓	✗

Figure 3.1: Features comparison between Plūma Model and Traditional OCR Models

3.4 Performance Metrics

3.4.1 Cosine

To evaluate the performance of standards OCR and our model in extracting handwritten text from a PDF file containing notes, both models were assessed using **Term Frequency-Inverse document Frequency (TF-IDF)** to transform texts into vectors. Subsequently, cosine similarity is computed between these vectors to establish the similarity against the actual text of the handwritten notes. The results indicated that our model achieved an **accuracy of 84%**, whereas the OCR model achieved 56%. This demonstrates Neural Network-based models achieve superior performance in accurately recognizing and extracting handwritten text compared to traditional OCR technologies.

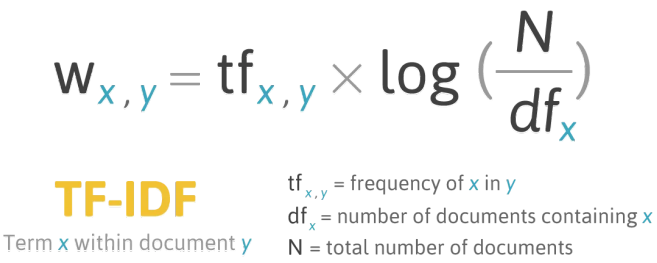

Figure 3.2: Term Frequency - Inverse Document Frequency technique explained [6]

3.4.2 Jaccard Similarity

The accuracy of text extraction was evaluated using Jaccard similarity too, which measures the similarity between sets of words by comparing the intersection and union of words between two texts. In this evaluation, we compared our model and the OCR model outputs to the original handwritten notes. Our model achieved a Jaccard similarity score of **accuracy of 42%**, indicating a stronger resemblance to the original text compared to the OCR model, which scored **accuracy of 21%**. These scores reflect the challenge of preserving word order in handwritten notes, contributing to lower overall similarity metrics.

$$\text{Jaccard index:} \quad J(A,B) = \frac{|A \cap B|}{|A \cup B|} \quad \text{and} \quad 0 \leqslant J(A,B) \leqslant 1$$

$$\text{Jaccard distance:} \quad d_J(A,B) = 1 - J(A,B) = 1 - \frac{|A \cap B|}{|A \cup B|} \quad \text{and} \quad 0 \leqslant d_J(A,B) \leqslant 1$$

Figure 3.3: Jaccard Similarity technique explained [7]

Chapter 4

Conclusion

4.1 Results

The results obtained from the first test, which aimed to analyze the accuracy of words within the text, show a significant increase in the accuracy of the application's model compared to traditional OCR models. This is likely due to the model's superior ability and flexibility to adapt to different writing styles within the same text, as well as the processing of the detected text based on the context-provided semantics, allowing for the correction of errors during the text detection phase.

4.2 Use Cases

Here are examples of how the application is used in different scenarios:

- **Businesses and offices:** Plūma is a valuable tool for businesses and offices, allowing the digitization of meeting notes, brainstorming session notes, and other handwritten documents. This not only improves document management but also facilitates information sharing among employees.
- **Educational institutions:** Schools and universities can use Plūma to digitize students' handwritten assignments, lecture notes, and exam papers. This streamlines the assessment process and makes it easier to archive and retrieve educational materials.
- **Professionals:** Freelancers and professionals can use Plūma to digitize their personal notes, project ideas, and client meeting notes. This ensures that important information is always at their fingertips and easily shareable with clients and collaborators.

4.3 Future developments

The development of the application does not stop here. The first step should be the application deployment as a web-server application, which could be accessible publicly. Then, many other improvements could be included, such as:

- **Real-time processing:** Optimizing the application for real-time note digitization, enabling users to capture and convert handwritten notes instantly during lectures or meetings. This can involve integrating advanced image processing algorithms and utilizing high-performance computing resources to ensure seamless and fast conversion.
- **Additional formats:** Supporting more digital input formats (e.g., PNG, JPG) and output formats (e.g., DOCX, HTML). This enhancement will make the application more versatile, allowing users to import notes from various sources and export digitized content in formats suitable for different use cases such as editing in word processors, web publishing, and archival.
- **User interface improvements:** Refining the user interface to enhance user experience. This could include customizable themes, intuitive navigation, and additional features such as note organization tools, annotation options, and interactive tutorials.
- **Cloud integration:** Incorporating cloud storage solutions for automatic backup and synchronization of digitized notes across multiple devices. This would ensure that users have access to their notes anytime, anywhere, and facilitate collaboration by allowing shared access to digitized content.
- **Customization options:** Allowing users to customize the digitization process to suit their specific needs. This might include options for different recognition modes (e.g., cursive, block letters) and the ability to define custom templates for note digitization (e.g. different styles).
- **Integration with other tools:** Enabling seamless integration with other productivity tools such as calendar apps, task managers, and educational platforms. This would allow users to easily incorporate their digitized notes into their daily workflows and enhance productivity.

4.4 Conclusion

Plūma represents a significant advancement in the field of handwriting recognition and note digitization. With its specialized AI model, advanced preprocessing techniques, and innovative features like AI-powered note enhancement, it offers a comprehensive solution for converting handwritten notes into valuable digital assets. Whether for businesses, educational institutions, or professionals, Plūma is poised to revolutionize the way handwritten notes are managed and utilized.

Bibliography

[1] "Gemini," Jul. 2024, [Online; accessed 16. Jul. 2024]. [Online]. Available: https://deepmind.google/technologies/gemini

[2] Shelf, "Why Recurrent Neural Networks (RNNs) Dominate Sequential Data Analysis," *Shelf*, Mar. 2024. [Online]. Available: https://shelf.io/blog/recurrent-neural-networks

[3] "Gemini Pro," Jul. 2024, [Online; accessed 16. Jul. 2024]. [Online]. Available: https://deepmind.google/technologies/gemini/pro

[4] "pytesseract," Jul. 2024, [Online; accessed 15. Jul. 2024]. [Online]. Available: https://pypi.org/project/pytesseract

[5] "Understand OCR Model," Jul. 2024, [Online; accessed 15. Jul. 2024]. [Online]. Available: https://help.zoho.com/portal/en/kb/creator/developer-guide/microservices/ai-modeler/articles/understand-ocr-model#OCR_model_prerequisites

[6] T. Mei, "Demystify TF-IDF in Indexing and Ranking - Ted Mei - Medium," *Medium*, Dec. 2021. [Online]. Available: https://ted-mei.medium.com/demystify-tf-idf-in-indexing-and-ranking-5c3ae88c3fa0

[7] F. Vidal, "Similarity Distances for Natural Language Processing," *Medium*, Jan. 2022. [Online]. Available: https://flavien-vidal.medium.com/similarity-distances-for-natural-language-processing-16f63cd5ba55