IMAGE TO TEXT INTERACTIVE SYSTEM FOR PDF DOCUMENTS

A Project Report Submitted

in Partial Fulfilment of the Requirements

for the Degree of

Bachelor of Technology

in

Computer Science and Engineering

by

Ashutosh Rai

(Roll No. 2020BCS0020)

to

DEPARTEMENT OF COMPUTER SCIENCE AND ENGINEERING

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY KOTTAYAM-686635, INDIA

April 2023

DECLARATION

I, **Ashutosh Rai** (**Roll No: 2020BCS0020**), hereby declare that, this report entitled **"Image to Text Interactive System for PDF Documents"** submitted to Indian Institute of Information Technology Kottayam towards partial requirement of **Bachelor of Technology(Hon)** in **Computer Science and Engineering** is an original work carried out by me under the supervision of **Dr. Selvi C** and has not formed the basis for the award of any degree or diploma, in this or any other institution or university. I have sincerely tried to uphold the academic ethics and honesty. Whenever an external information or statement or result is used then, that have been duly acknowledged and cited.

Kottayam-686635 **Ashutosh Rai**

April 2023

CERTIFICATE

This is to certify that the work contained in this project report entitled **"Image to Text Interactive System for PDF Documents"** submitted by **Ashutosh Rai** (**Roll No: 2020BCS0020**) to Indian Institute of Information Technology Kottayam towards partial requirement of **Bachelor of Technology(Hon)** in **Indian Institute of Information Technology Kottayam** has been carried out by him under my supervision and that it has not been submitted elsewhere for the award of any degree.

Kottayam-686635 (Dr. Selvi C)

April 2023 Project Supervisor

ABSTRACT

This project report presents an Image to text Interactive System that extracts textual information from semi-structured and unstructured documents, including images of literature and invoice documents. For literature documents, the system utilizes layout parsing, OCR (Optical Character Recognition), and interactive Q/A (Question Answering) system. For invoice documents, an OCR-free approach using a DONUT model is used for VDU (Visual Document Understanding) to extract textual information. This approach overcomes the limitations of OCR-based methods, such as high computational costs, inflexibility on languages or types of documents, and OCR error propagation. The system also incorporates natural language understanding to provide relevant responses to user queries based on the extracted information. Document images, such as commercial invoices, receipts, and literature documents, are ubiquitous in modern working environments. Extracting useful information from these documents is essential for industry and has been a challenging topic for researchers. This project addresses this challenge by providing a powerful tool for efficient and accurate information extraction from various types of documents. The system can be applied in a variety of real-world scenarios, including document classification, information extraction, and visual question answering. This report provides a detailed description of the system architecture, including the workflow diagram. The experimental setup and evaluation results are also presented to demonstrate the effectiveness of the system. Additionally, partial output screenshots are provided to illustrate the system's capabilities. Finally, future work is discussed to improve the system's performance and expand its functionality.

Contents

List of Figures

Chapter 1

Introduction

1.1 Introduction

This chapter provides an introduction to the problem of document understanding and information extraction, and discusses the motivation behind the development of our Image to text Interactive System while outlining the objectives and scope of this project. In today's digital age, vast amounts of information are generated and stored in various formats, including semi-structured and unstructured documents such as images. These documents, such as commercial invoices, receipts, and literature, contain valuable information that can be useful for businesses and individuals. However, extracting relevant information from these documents can be a challenging and time-consuming task. This project aims to address this challenge by developing an Image to Text Interactive System that can extract textual information from images of literature and invoice documents. The system makes use of advanced techniques such as OCR, layout parsing, and VDU to overcome the

limitations of traditional OCR based methods, such as high computational costs, inflexibility on languages or types of documents, and OCR error propagation in the case of invoice documents. In addition, Current approaches require training and data requirements of LLMs (Large Language Models). LLMs are pre-trained on large amounts of text data, which can make them computationally expensive and time-consuming to train. Additionally, LLMs may not perform as well on domain- specific or low-resource languages compared to OCR-based or VDU-based methods. Furthermore, while LLMs have shown impressive performance on various language tasks, they may not be optimized for the specific task of extracting information from document images, where a holistic understanding of the layout and structure of the document is required. In contrast, the OCR-based and VDU-based approaches used in this project are specifically tailored for this task, making them more effective at extracting information from document images. Furthermore, the system incorporates natural language understanding to provide relevant responses to user queries based on the extracted information. This project provides a powerful tool for efficient and accurate information extraction from various types of documents and has the potential to be applied in a variety of real-world scenarios.

Figure 1.1 provides the basics of document question answering system.

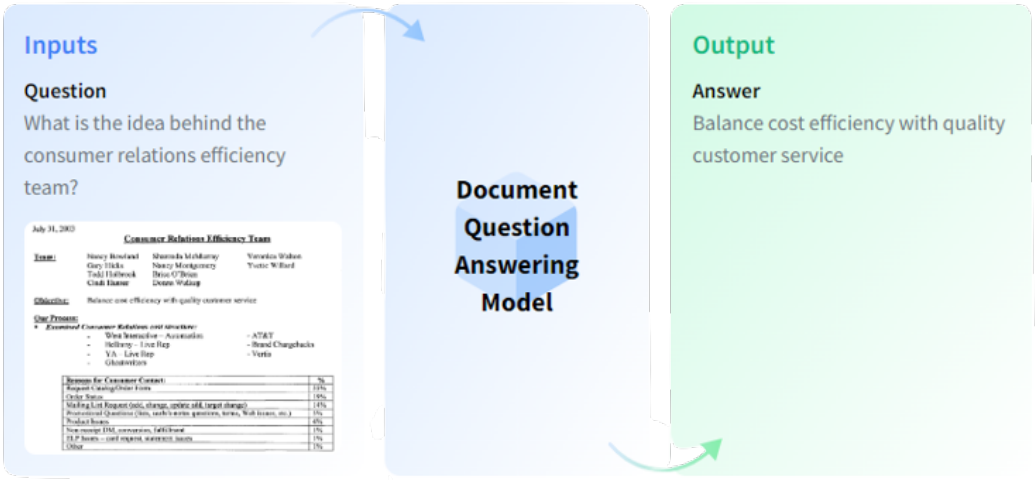


Figure 1.1: Document Question Answering

1.2 Background and Motivation

The ability to extract information from images has become increasingly important in recent years due to the vast amount of unstructured data that is generated and stored in this format. However, traditional OCR based methods for extracting text from images have limitations, such as high computational costs, inflexibility on languages or types of documents, and OCR error propagation. As a result, researchers have turned to alternative approaches such as VDU to overcome these limitations. Document Understanding and Q/A have also become increasingly important in recent years due to the need for efficient and accurate information extraction from various types of documents. Document Understanding involves analysing document structures and layouts, while Document Q/A involves answering questions on document images. These tasks have numerous applications in real-world scenarios, such

as information extraction for business processes, and information retrieval for legal or medical documents. Motivated by the need for a more efficient and accurate method of information extraction from document images, this project aims to develop an Image to Text Interactive System that can extract textual information from images of literature and invoice documents. The system utilizes advanced techniques such as OCR, layout parsing, and VDU to overcome the limitations of traditional OCR-based methods. The system also incorporates NLU (Natural Language Understanding) to provide relevant responses to user queries based on the extracted information. Moreover, the proposed system is motivated by the need for a more efficient and user-friendly approach to information extraction. Manually extracting information from document images is a time-consuming and error-prone task. By developing an interactive system that can extract information from images with minimal user input, the proposed system can save time and reduce errors in various real-world scenarios, such as data entry, record-keeping, and invoice processing.

Chapter 2

Related Work and System Design

2.1 Literature Review

Abnar et al. [3] presents a paper that highlights the advantages and limitations of their proposed incremental reading approach to document-level question answering. The system integrates a retrieval module to filter out irrelevant information, resulting in a more accurate and efficient approach to answer questions. The authors demonstrated that their proposed system outperformed traditional approaches in terms of accuracy and efficiency. However, the evaluation of the system was limited to only two large-scale datasets, which may impact the generalizability of the results. Moreover, the performance of the system could be affected by the complexity of the document and question types.

Cheng et al's paper [4] highlights the advantages and disadvantages of their

proposed model for document-level question answering. Their model achieved superior performance on the SQuAD dataset compared to existing state-of-the-art models, emphasizing the importance of modeling dependencies between sentences in a document. The use of distantly supervised data also allowed for a larger training dataset, which contributed to the improved model performance. However, the paper primarily focuses on document-level question answering for a single dataset, and the generalizability of the model to other datasets and types of documents is not thoroughly evaluated. Additionally, the proposed model requires significant computation resources and training time, limiting its scalability and practical use in real-world applications. Finally, the evaluation metrics used in the paper do not consider the diversity of answer types, potentially underestimating the model's performance in handling complex questions and documents.

Sun et al [8] proposed an approach to improve machine reading comprehension by leveraging general reading strategies. The paper highlights the advantages of the proposed approach, which achieved state-of-the-art performance on the SQuAD dataset and demonstrated effectiveness in handling both short and long passages. The approach requires pretraining on large amounts of data, which may limit its practicality in real-world applications where data may be scarce. Additionally, the general reading strategies used in the approach are based on human intuition and may not be comprehensive or applicable to all types of documents. Furthermore, the evaluation of the approach primarily focuses on the SQuAD dataset and does not thoroughly evaluate its generalizability to other datasets or types of documents.

[5] presents BERT, a state-of-the-art language model that uses bidirectional

transformers and pre-training on large amounts of data to achieve high performance on a wide range of natural language processing (NLP) tasks, including document-level question answering. The pre-training approach employed in BERT allows the model to learn contextualized representations of words and phrases, which is critical for understanding the meaning of text in different contexts. BERT is capable of handling long documents and capturing the relationships between different parts of the document, making it suitable for document-level question answering tasks. However, the pre-training process for BERT is computationally expensive and requires a significant amount of training data, limiting its accessibility to researchers with limited computing resources and data. Additionally, BERT requires fine-tuning on specific tasks to achieve optimal performance, which can be time-consuming and may necessitate task-specific modifications to the model architecture. BERT may also struggle with out-of-vocabulary words or rare words that do not appear in the pre-training data, thereby limiting its ability to handle certain types of documents or questions.

2.2 System Architecture

2.2.1 Workflow Diagram

Figure 2.1 gives the internal working of the proposed Image to Text Question Answering System.

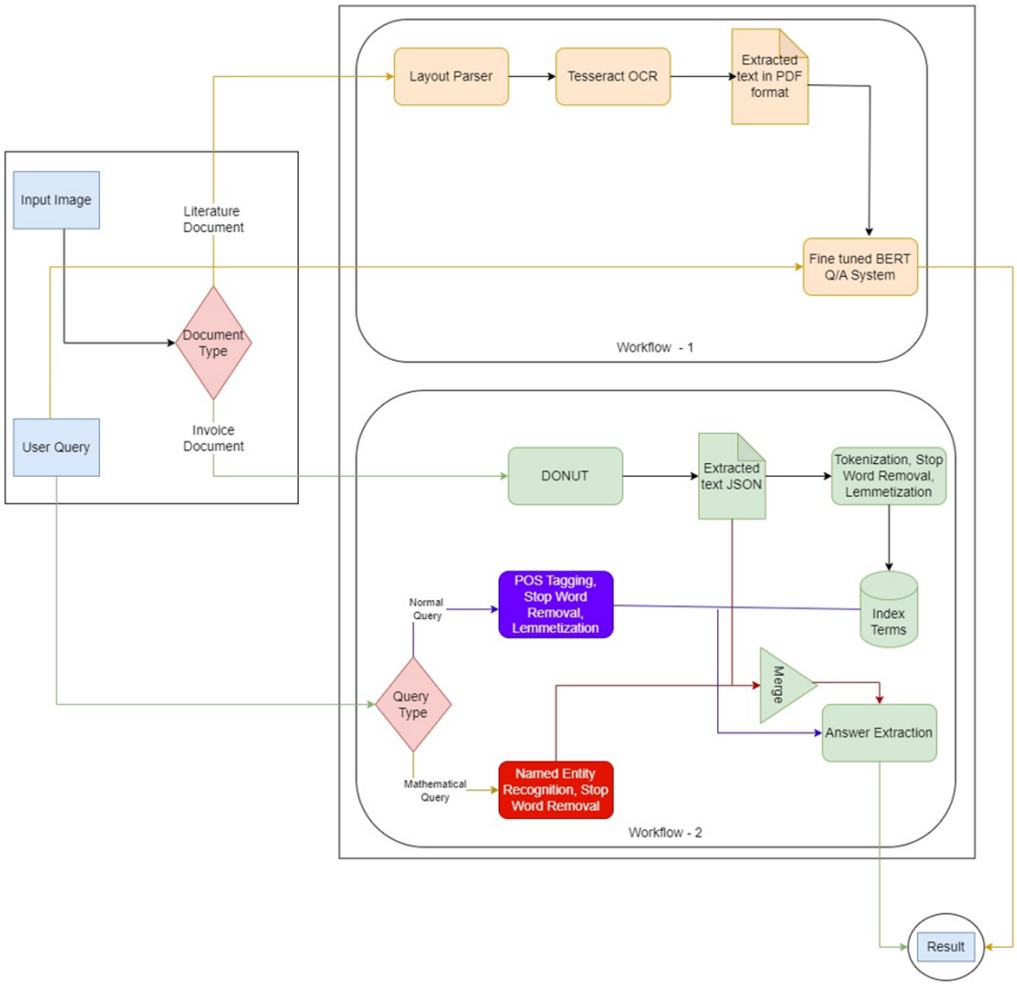


Figure 2.1: Workflow Diagram

2.2.2 Methodology

- **User interface:** The system is designed with a user friendly GUI (Graphical User Interface) that allows the user to input the image of the document along with a query. The user can also select the type of document being input, either "literature" or "invoice," which determines the method of text extraction used.

- **Text extraction for literature documents:** If the input image is of a literature document, the first step is to pass the image through a layout parser [7]. This parser selects the ROIs (regions of interest) in the image that contain the textual information. This helps when the document contains images, graphs, and tables. The title and paragraph regions are then chosen, and the image is passed through the Tesseract OCR engine, which produces the text. The output is then exported as a PDF and fed to the question-answering system along with the user query. The system returns the answer to the query using the OCR output as context, which is displayed to the user through the GUI. This is a zero-shot method of question answering.

- **Text extraction for invoice documents:** If the user initially inputs an invoice document, it is passed through the DONUT [6] model, a transformer-based model that does not rely on OCR. The model outputs the invoice's textual information in the form of a JSON, for example: Organization name: 'ABC', Grand total: '30'. The user query is then processed using a combination of NLU (Natural Language Understanding) techniques to extract the semantic meaning. Appropriate

operations are then performed on the data available through the JSON extracted from the invoice earlier. The output is then displayed to the user through the GUI.

- The methodology used in this project is designed to be efficient and accurate, using advanced techniques such as OCR, layout parsing, and VDU. By incorporating NLU, the system can be tailored to provide relevant responses to user queries based on the extracted information. Overall, this methodology provides a powerful tool for information extraction from various types of documents and has the potential to be applied in a variety of real-world scenarios.

2.2.3 Anatomy of Key Components

- **Layout Parser:** The application of NLP often times requires us to extract texts from input documents as prerequisites. The problem is, sometimes extra work needs to be performed to extract texts from the input documents because they normally come in PDF, JPEG, or PNG format. And this is where we usually use an OCR Engine. It helps us convert written texts in an image or scanned document into machine-readable text data. However, there is one caveat that needs to be addressed before extracting texts with OCR. Sometimes the input document consists of not only a series of texts, but also a title, an image, and a table. If the user wants to extract only the texts from paragraphs in an input document. It would require omitting the texts in the table, title, and image region. This is where we leverage Layout Parser to

categorize each section of our input document before we feed it to an OCR. Layout Parser provides a wide range of pre-trained deep learning models for detecting the layout of a document. Layout Parser has been trained on datasets like: PubLayNet, HJDataset, PrimaLayout, Newspaper Navigator, and TableBank.

- **Q/A System:** A BERT-based Q/A model is a powerful natural language processing system that can answer questions posed in natural language. The system is designed to process large amounts of text data and learn the relationships between words and phrases. During pre-training, the model is trained to predict the missing word in a sentence given the context surrounding it. This allows the model to develop an understanding of the relationships between words and the meanings they convey in different contexts. Once the model is pre-trained, it can be fine-tuned for specific tasks such as Q/A. Fine-tuning involves training the model on a smaller dataset that is specific to the task at hand. In the case of question answering, the model is trained to answer questions based on a given passage of text. To do this, the model first reads the passage and then uses attention mechanisms to identify the most relevant parts of the passage that are likely to contain the answer. It then generates a representation of the question and uses it to query the relevant parts of the passage for an answer. The model generates a probability distribution over all possible answers, and the most probable answer is selected as the output. Additionally, the model can also provide a confidence score indicating how confident it is in its answer. This is particularly useful in situations where the model is not

completely certain about the answer and wants to convey its level of confidence to the user. In the proposed system, the BERT-based Q/A model is used to provide accurate and fast answers to user questions based on a given passage of text.

- **DONUT:** DONUT is a deep learning model that takes an image as an input and encodes it into a sequence of tokens using a Swin Transformer, a type of neural network that can process images in a patch-based manner. The tokenized image is then passed through a BART decoder model, which is a type of neural network that can decode the tokens into an output sequence, typically in the form of a data structure like JSON. During inference, the model takes in prompts or questions related to the image and generate answers in the same architecture. The model is pre-trained on multilingual datasets, it can work with inputs in multiple languages. The DONUT model allows for efficient and accurate processing of invoice document images and its associated information in the project, making it a valuable component.

2.3 Scope and Limitations

Scope: The scope of this project is to develop an interactive image-totext system as a web application that can extract textual information from images of literature and invoice documents. The system is designed to handle various formats of input images and provide accurate and relevant responses to user queries based on the extracted information. The project will focus on implementing advanced techniques such as OCR, layout parsing, and VDU to ensure accurate and efficient extraction of information from document images. OCR will be used to recognize and extract text from images, while layout parsing will help in identifying ROI in the image that contain the textual information. VDU will be implemented to extract relevant information from invoice documents without relying on OCR. In addition, the system will incorporate NLU to enable user queries to be interpreted and answered in a contextually relevant manner. The objective is to provide a user-friendly interface that can be easily used by non-technical users, allowing them to input an image and query, and get relevant information in a fast and efficient manner.

Limitations:

1. **Image Quality:** The system's performance is heavily dependent on the quality of the input image. Poor image quality, such as low resolution or blurry images, may result in inaccurate text extraction and ultimately affect the system's ability to provide accurate responses to user queries.

2. **Language Support and Handwritten Text:** The system may face challenges in accurately extracting handwritten text due to variations in handwriting styles and legibility. Additionally, while the model is trained on multilingual data, its performance may vary depending on the language and the complexity of the text in that language.

3. **Limitations of OCR and VDU:** The OCR and VDU techniques used in the system have their own limitations. OCR may struggle to accurately recognize certain fonts, text sizes, or styles, which can impact the accuracy of the extracted text. Similarly, the VDU model may not be able to accurately identify and extract certain types of information from the input image.

4. **Limited Training Data:** The system's performance is directly proportional to the amount and quality of training data available. As a result, limited training data may impact the system's ability to accurately extract text and provide relevant responses to user queries.

Chapter 3

Experimental Results and Conclusions

3.1 Visual Results

- **Layout Parser:**

  Input:

  Figure 3.1 represents the sample input image given to the Layout parser.

Yet another way to organize design patterns is according to how they reference each other in their "Related Patterns" sections. Figure 1.1 depicts these relationships graphically.

Clearly there are many ways to organize design patterns. Having multiple ways of thinking about patterns will deepen your insight into what they do, how they compare, and when to apply them.

1.6 How Design Patterns Solve Design Problems

Design patterns solve many of the day-to-day problems object-oriented designers face, and in many different ways. Here are several of these problems and how design patterns solve them.

Finding Appropriate Objects

Object-oriented programs are made up of objects. An **object** packages both data and the procedures that operate on that data. The procedures are typically called **methods** or **operations**. An object performs an operation when it receives a **request** (or **message**) from a **client**.

Requests are the *only* way to get an object to execute an operation. Operations are the *only* way to change an object's internal data. Because of these restrictions, the object's internal state is said to be **encapsulated**; it cannot be accessed directly, and its representation is invisible from outside the object.

The hard part about object-oriented design is decomposing a system into objects. The task is difficult because many factors come into play: encapsulation, granularity, dependency, flexibility, performance, evolution, reusability, and on and on. They all influence the decomposition, often in conflicting ways.

Object-oriented design methodologies favor many different approaches. You can write a problem statement, single out the nouns and verbs, and create corresponding classes and operations. Or you can focus on the collaborations and responsibilities in your system. Or you can model the real world and translate the objects found during analysis into design. There will always be disagreement on which approach is best.

Many objects in a design come from the analysis model. But object-oriented designs often end up with classes that have no counterparts in the real world. Some of these are low-level classes like arrays. Others are much higher-level. For example, the Composite (163) pattern introduces an abstraction for treating objects uniformly that doesn't have a physical counterpart. Strict modeling of the real world leads to a system that reflects today's realities but not necessarily tomorrow's. The abstractions that emerge during design are key to making a design flexible.

Figure 3.1: Input for Layout Parser

Identified ROIs:

Figure 3.2 shows the ROI identified by Layout Parser.

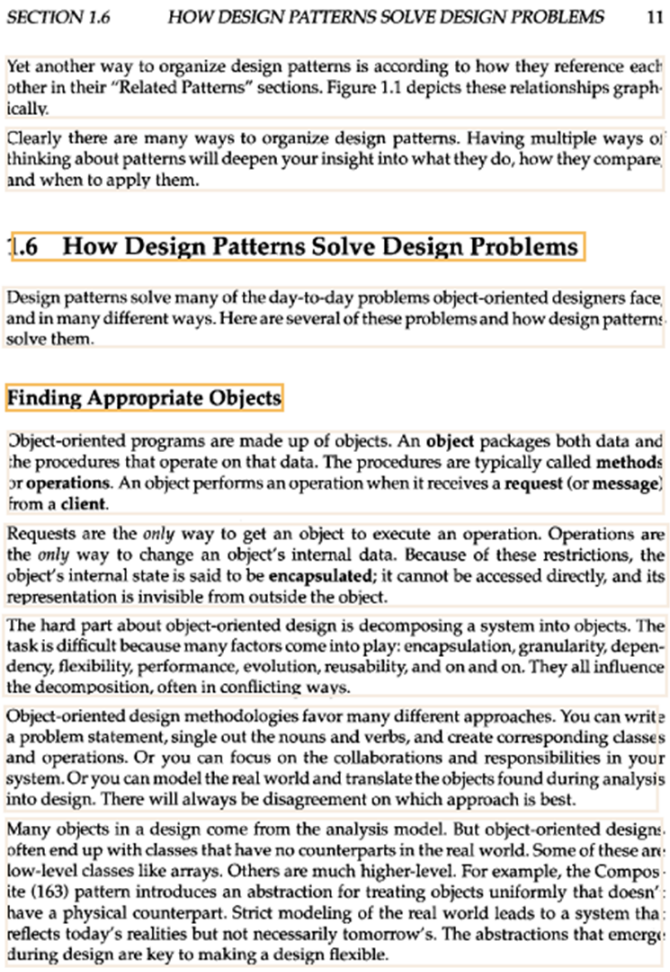


Figure 3.2: Identified ROIs

Tesseract Output:

Figure 3.3 shows the output obtained from Tesseract OCR.

```
for txt in text_blocks.get_texts():
    print(txt, end='\n---\n')
```

```
Object-oriented design methodologies favor many different approaches. You can writ
a problem statement, single out the nouns and verbs, and create corresponding classe
and operations. Or you can focus on the collaborations and responsibilities in you
system. Or you can model the eal world and translate the objects found during analysis
into design. There will always be disagreement on which approach is best

---
Many objects in a design come from the analysis model, But object-oriented designs
often end up with classes that have no counterparts inthe eal world, Some ofthese are
low-level classes ike arrays. Others are much higher-level. or example, the Compos-
ite (163) pattern introduces an abstraction for treating objects uniformly that doesn't
have a physical counterpart, Strict modeling ofthe real world leads to a system that
reflects today's realities but not necessarily tomorrow s. The abstractions that emerge
during design are key to making a design flexible.

---
'The hard part about object-oriented design is decomposing a system into objects. The
taskis difficult because many factors come into play: encapsulation, granularity, depen-
dency, flexibility, performance, evolution, reusability, and onand on. They all influence
the decomposition, often in conflicting ways.

---
Requests are the only way to get an object to execute an operation. Operations are
the only way to change an object's internal data. Because of these restrictions, the
object's internal state is said to be encapsulated; it cannot be accessed directly, and its
representation is invisible from outside the object.

---
Design patterns solve many of the day-to-day problems object-oriented designers face,
and in many different ways. Hereare several ofthese problemsand how design patterns
solve them.

---
'Yet another way to organize design patterns is according to how they reference each
other in their "Related Patterns" sections. Figure 1.1 depicts these relationships graph-
ically.

---
'Object-oriented programs are made up of objects. An object packages both data and
the procedures that operate on that data. The procedures are typically called methods
'or operations. An object performs an operation when it receives a request (or message)
from a client.

---
Clearly there are many ways to organize design patterns. Having multiple ways of
thinking about patterns will deepen your insight into what they do, how they compare,
and when to apply them,

---
```

Figure 3.3: Tesseract Output

- **DONUT:**

  Input:

  Figure 3.4 represents sample input to the DONUT model.

Figure 3.4: Input for DONUT

Output:

Prediction: 'total': '9.30', 'date': '26/11/2017', 'company': 'SANYU STATIONERY SHOP', 'address': 'NO. 31G&33G, JALAN SETIA INDAH X,U13/X 40170 SETIA ALAM'

Reference: 'total': '9.30', 'date': '26/11/2017', 'company': 'SANYU STATIONERY SHOP', 'address': 'NO. 31G&33G, JALAN SETIA INDAH X ,U13/X 40170 SETIA ALAM'

Input:

Figure 3.5 represents sample input to the DONUT model.

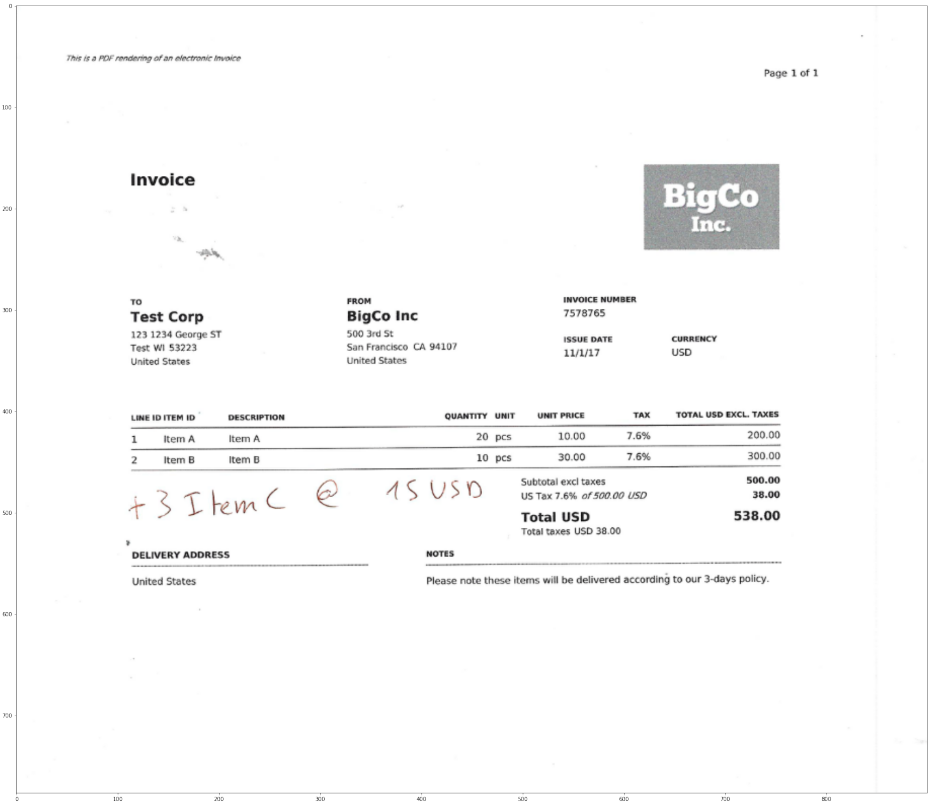


Figure 3.5: Input for DONUT

Output:

Figure 3.6 shows the output JSON obtained from DONUT on passing the sample invoice.

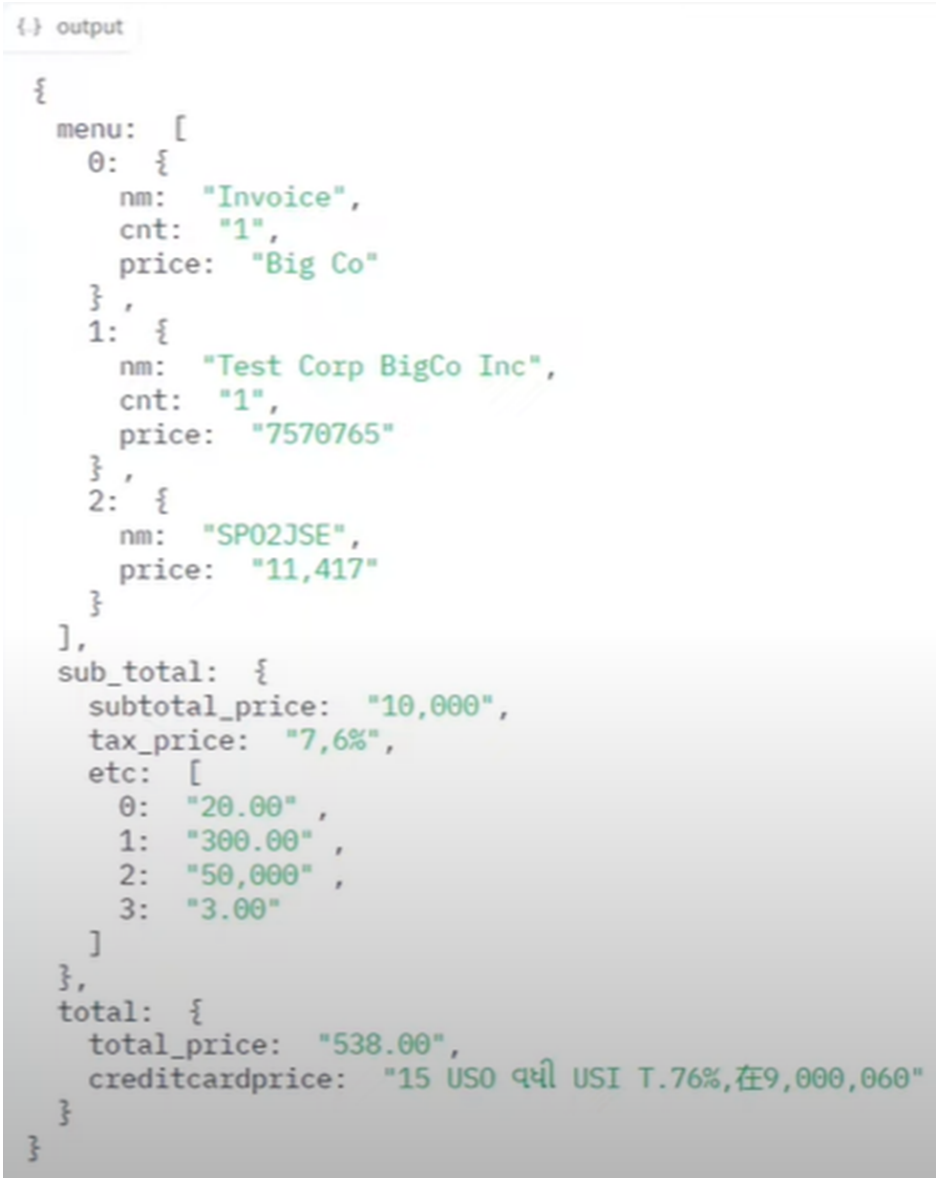
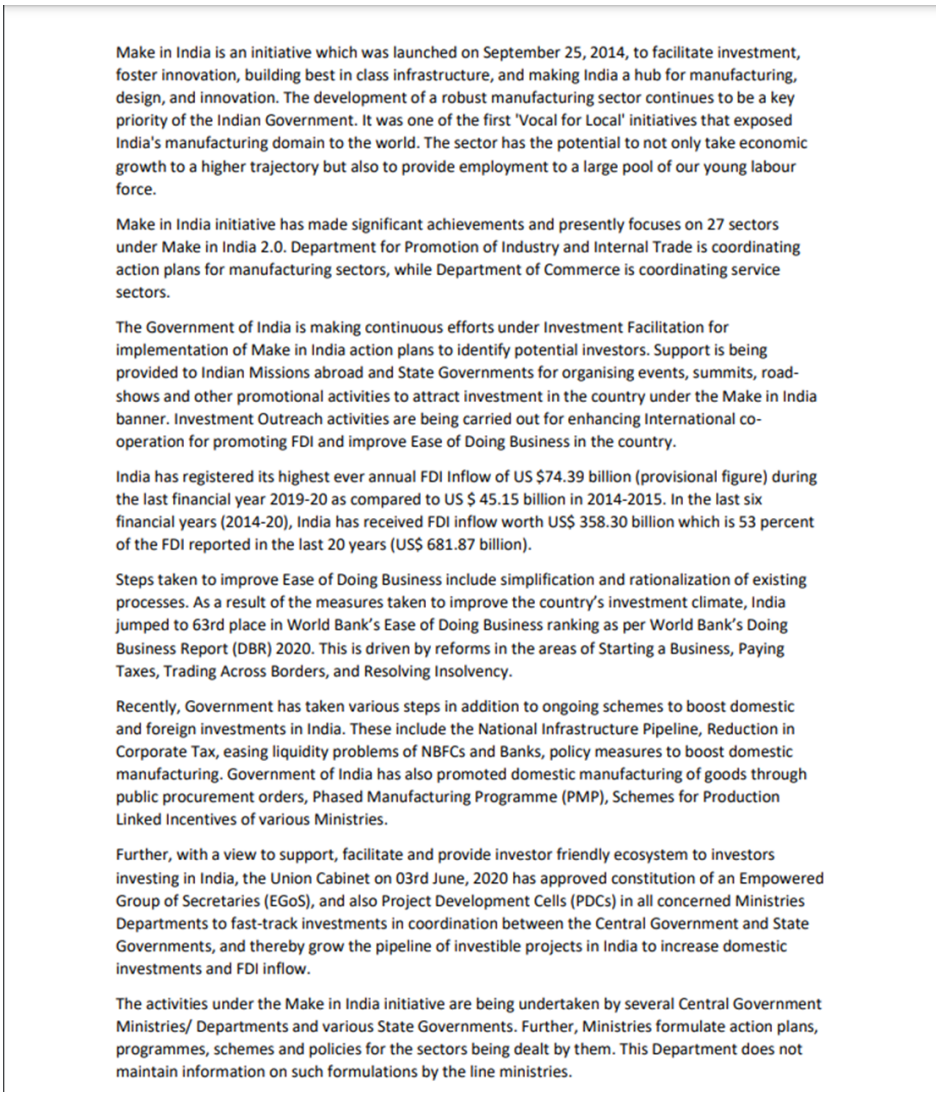


Figure 3.6: Output JSON from DONUT

- **Q/A System:**

  Input:

  Figure 3.7 represents the sample input to the Q/A System.



Make in India is an initiative which was launched on September 25, 2014, to facilitate investment, foster innovation, building best in class infrastructure, and making India a hub for manufacturing, design, and innovation. The development of a robust manufacturing sector continues to be a key priority of the Indian Government. It was one of the first 'Vocal for Local' initiatives that exposed India's manufacturing domain to the world. The sector has the potential to not only take economic growth to a higher trajectory but also to provide employment to a large pool of our young labour force.

Make in India initiative has made significant achievements and presently focuses on 27 sectors under Make in India 2.0. Department for Promotion of Industry and Internal Trade is coordinating action plans for manufacturing sectors, while Department of Commerce is coordinating service sectors.

The Government of India is making continuous efforts under Investment Facilitation for implementation of Make in India action plans to identify potential investors. Support is being provided to Indian Missions abroad and State Governments for organising events, summits, road-shows and other promotional activities to attract investment in the country under the Make in India banner. Investment Outreach activities are being carried out for enhancing International co-operation for promoting FDI and improve Ease of Doing Business in the country.

India has registered its highest ever annual FDI Inflow of US $74.39 billion (provisional figure) during the last financial year 2019-20 as compared to US $ 45.15 billion in 2014-2015. In the last six financial years (2014-20), India has received FDI inflow worth US$ 358.30 billion which is 53 percent of the FDI reported in the last 20 years (US$ 681.87 billion).

Steps taken to improve Ease of Doing Business include simplification and rationalization of existing processes. As a result of the measures taken to improve the country's investment climate, India jumped to 63rd place in World Bank's Ease of Doing Business ranking as per World Bank's Doing Business Report (DBR) 2020. This is driven by reforms in the areas of Starting a Business, Paying Taxes, Trading Across Borders, and Resolving Insolvency.

Recently, Government has taken various steps in addition to ongoing schemes to boost domestic and foreign investments in India. These include the National Infrastructure Pipeline, Reduction in Corporate Tax, easing liquidity problems of NBFCs and Banks, policy measures to boost domestic manufacturing. Government of India has also promoted domestic manufacturing of goods through public procurement orders, Phased Manufacturing Programme (PMP), Schemes for Production Linked Incentives of various Ministries.

Further, with a view to support, facilitate and provide investor friendly ecosystem to investors investing in India, the Union Cabinet on 03rd June, 2020 has approved constitution of an Empowered Group of Secretaries (EGoS), and also Project Development Cells (PDCs) in all concerned Ministries Departments to fast-track investments in coordination between the Central Government and State Governments, and thereby grow the pipeline of investible projects in India to increase domestic investments and FDI inflow.

The activities under the Make in India initiative are being undertaken by several Central Government Ministries/ Departments and various State Governments. Further, Ministries formulate action plans, programmes, schemes and policies for the sectors being dealt by them. This Department does not maintain information on such formulations by the line ministries.

Figure 3.7: Input for Q/A System

Output:

Figure 3.8 shows the output of the Q/A system.

```
[ ]  question_fdi = "What were the various steps taken by the government to boost domestic and foreign investment in India?"

[ ]  max_score = 0;
     final_answer = ""
     new_df = expand_split_sentences(fdi)
     for new_context in new_df:
       #new_paragrapgh = new_paragrapgh + answer_question(question, answer_text)
       ans, score = answer_question(question_fdi, new_context)
       if score > max_score:
         max_score = score
         final_answer = ans
     print(final_answer)
     print(max_score)
```

```
national infrastructure pipeline , reduction in corporate tax , easing liquid ##ity problems of n ##bf ##cs and banks , policy measures to boost domestic manufacturing
6.569849491119385
```

Figure 3.8: Output from Q/A System

Text: national infrastructure pipeline , reduction in corporate tax , easing liquid ##ity problems of n ##bf ##cs and banks , policy measures to boost domestic manufacturing

6.569849491119385

3.2 Performance Measures

- **Tesseract OCR:** The decision to use Tesseract OCR was based on the results of a benchmarking test conducted on a custom-made dataset. The dataset included three categories, namely web page screenshots with various texts, photos with different handwriting styles, and receipts, invoices, and scanned contracts collected randomly from the internet. This diverse dataset allowed for a comprehensive evaluation of Tesseract OCR's performance, ensuring that the chosen system could handle various document types and structures. Furthermore, Tesseract OCR was chosen for its ability to support multiple languages and its ease of integration with other technologies such as natural language processing and machine learning. This versatility makes it a reliable and efficient OCR tool suitable for a range of applications and industries. In conclusion, Tesseract OCR is a popular choice among researchers and developers due to its accuracy, speed, and versatility. It is a reliable and efficient tool suitable for various applications, and its support for multiple languages and integration with other technologies make it a valuable asset for organizations seeking to extract information from different types of documents accurately and efficiently.

  Figure 3.9 shows the comparison of Tesseract with other OCRs. [2]

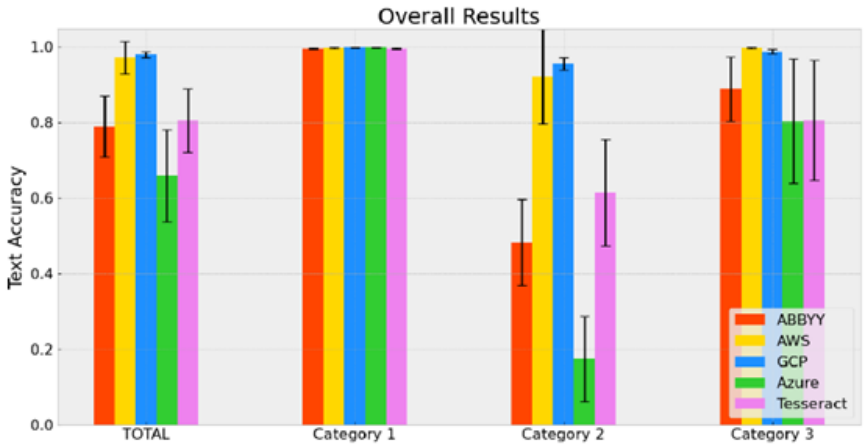


Figure 3.9: Performance Measure of Tesseract OCR

- Additionally, we can calculate the character error rate (CER) and word error rate (WER) by selecting a few sample sentences and words from the input presented in the Visual Results section [1]: Consider the following paragraph identified as ROI in the example:

  Reference (Ground Truth):

  Figure 3.10 is a sample chosen from the identified ROIs for calculation of WER and CER

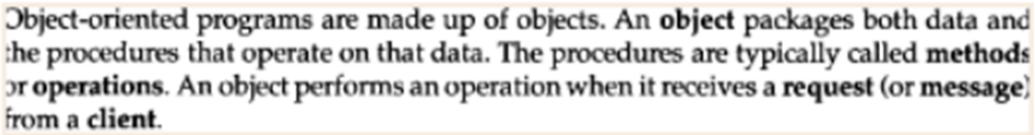

Object-oriented programs are made up of objects. An **object** packages both data and the procedures that operate on that data. The procedures are typically called **methods** or **operations**. An object performs an operation when it receives a **request** (or **message**) from a **client**.

Figure 3.10: Sample for Calculation of CER (Character Error Rate) and WER (Word Error Rate)

OCR Output:

Figure 3.11 Output from OCR for the chosen sample.

```
---
‘Object-oriented programs are made up of objects. An object packages both data and
the procedures that operate on that data. The procedures are typically called methods
‘or operations. An object performs an operation when it receives a request (or message)
from a client.

---
```

Figure 3.11: OCR Output

$$WER = \frac{S_w + D_w + I_w}{N_w} \tag{3.1}$$

Where,

S_w = Number of Substitutions

D_w = Number of Deletions

I_w = Number of Insertions

N_w = Number of characters in reference text (aka ground truth)

Based on Equation 3.1 the calculated WER is 0.

Similarly, for CER:

$$CER = \frac{S + D + I}{N} \tag{3.2}$$

Based on Equation 3.2 the calculated CER is 0.

3.3 Conclusion

In conclusion, the proposed system has demonstrated the effectiveness of leveraging advanced techniques such as NLU, OCR, layout parsing, and VDU to improve the performance of document-level Q/A systems. However, it currently supports only two types of documents, literature and invoice. As part of future work, there is a need for further research to enhance the system's accuracy and speed, such as incorporating techniques like semantic parsing and paraphrasing to handle ambiguous or vague queries. Additionally, the system could be extended to handle more complex question types like multi-hop reasoning and temporal reasoning. The system could also be improved by incorporating feedback mechanisms to enhance its accuracy over time, and by evaluating its performance on larger and more diverse datasets to test its scalability and robustness in real-world scenarios. Furthermore, support for other document types like resumes, emails, memos, reports, and letters, as well as additional languages, could be added in future work. Overall, the proposed system has great potential in advancing the field of Q/A and its applications across various industries, especially in organizations seeking to extract relevant information from large amounts of documents efficiently and accurately. With continued development and research, the proposed system could become an invaluable tool for document analysis, providing insights and answers that can significantly improve decision making processes.

Bibliography

[1] OCR in 2023: Benchmarking Text Extraction/Capture Accuracy — research.aimultiple.com. https://research.aimultiple.com/ocr-accuracy. [Accessed 18-Apr-2023].

[2] What is Document Question Answering? - Hugging Face — huggingface.co. https://huggingface.co/tasks/document-question-answering. [Accessed 18-Apr-2023].

[3] Samira Abnar, Tania Bedrax-Weiss, Tom Kwiatkowski, and William W Cohen. Incremental reading for question answering. *arXiv preprint arXiv:1901.04936*, 2019.

[4] Hao Cheng, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Probabilistic assumptions matter: Improved models for distantly-supervised document-level question answering. *arXiv preprint arXiv:2005.01898*, 2020.

[5] Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186, 2019.

[6] Geewook Kim, Teakgyu Hong, Moonbin Yim, JeongYeon Nam, Jinyoung Park, Jinyeong Yim, Wonseok Hwang, Sangdoo Yun, Dongyoon Han, and Seunghyun Park. Ocr-free document understanding transformer. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXVIII*, pages 498–517. Springer, 2022.

[7] Zejiang Shen, Ruochen Zhang, Melissa Dell, Benjamin Lee, Jacob Carlson, and Weining Li. Layoutparser: A unified toolkit for deep learning based document image analysis. In *Proceedings of the International Conference on Document Analysis and Recognition*, 2021.

[8] Kai Sun, Dian Yu, Dong Yu, and Claire Cardie. Improving machine reading comprehension with general reading strategies. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2633–2643, 2019.