ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

DEPARTMENT OF COMPUTER SCIENCE
AND ENGINEERING

ARTIFICIAL INTELLIGENCE

MASTER THESIS

in

Natural Language Processing

ENHANCING DOCUMENT PARSING AND QUESTION ANSWERING THROUGH OPTIMIZED TABLE PARSING

CANDIDATE

Marta Stella

SUPERVISOR

Prof. Paolo Torroni

CO-SUPERVISORS

Nicolò Donati , PhD

Dr.-Ing. Christoph Alt

Dr. Sebastian Arnold

Academic year 2023-2024

Session 1st

Abstract

The dissertation investigates the significant impact of table parsing on enhancing the accuracy and efficiency of document parsing and question answering systems. This research is motivated by the practical challenges experienced during an internship at *Bayer*, where the necessity for enhanced parsing techniques became clearly evident. By integrating advanced parsing techniques with Natural Language Processing models, the research addresses the challenges of extracting and interpreting information from various types of documents, with a particular emphasis on tables.

A central aspect of this work is the impact of table parsing within the document parsing and question answering processes and the evaluation of the proposed optimizations through experiments and assessments by human experts. These evaluations measure the impact of the optimizations, implemented through table parsing, on parsing quality and the question answering system. They highlight the system's ability to accurately parse documents and generate pertinent and relevant responses to queries, underscoring the crucial role of precision in document parsing for effective question answering.

The research findings demonstrate a substantial improvement in document parsing and question answering capabilities as a result of the optimized table parsing techniques. The dissertation details the advantages and limitations of different parsing methods, proposing solutions that enhance the performance of the document question answering system.

Table parsing is shown to be essential for improving the system's ability to comprehend complex queries and documents, leading to more accurate and efficient information retrieval.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In the rapidly evolving domain of Artificial Intelligence, the field of Natural Language Processing has witnessed significant advancements in recent years. This progress is particularly notable in the domains of document parsing and question answering; effectively managing and extracting meaningful insights from the rapidly expanding volume of both unstructured and semi-structured data is becoming increasingly essential. This necessity underscores the requirement for advanced methodologies capable of navigating, extracting, and interpreting information across a wide range of document formats. The complexity of these challenges extends beyond mere technical issues, impacting the way information is accessed and utilized, especially given the vast amount of data awaiting analysis and interpretation.

The goal of this dissertation is to explore the potential for improving document parsing and question answering, with a specific focus on extracting and processing tabular data. The objective is not only to provide improvements to current systems, but also to enhance the performance and capabilities of both document parsing and document question answering processes. By analyzing the current methodologies, this dissertation examines the potentialities within tabular data extraction and processing to harness the power of unstructured and semi-structured data in an increasingly information-driven world.

1.1 Document Parsing for Question Answering

Document parsing is a fundamental step in the process of transforming unstructured or semi-structured text into a structured format that can be utilized by question answering systems. It involves various techniques and tools that analyze the content of documents, identify key elements, and convert them into a format suitable for further processing. The significance of accurate and efficient document parsing cannot be overstated, as it directly impacts the quality of the information retrieved and the answers generated by the question answering system.

The aim is to investigate the role of document parsing in the context of question answering platforms. By examining different parsing approaches, the goal is to highlight the critical aspects that contribute to the successful extraction of relevant information from documents, assessing the strengths and weaknesses of various parsing tools, including their ability to handle diverse document structures and formats.

1.2 Extracting Information from Tables

Tables are a common and important feature in many documents, providing structured data that is often essential for understanding the overall content. Despite their significance, tables present unique challenges for parsing systems due to their diverse formats and the complex relationships between their elements. Effective table parsing is crucial for extracting accurate information and improving the overall performance of document parsing systems.

Tables encapsulate key information in a concise format, but their structural complexity can lead to significant parsing difficulties. Rows and columns may span multiple cells, contain nested tables, or include a variety of data types and formats. Addressing these challenges requires specialized techniques that go beyond standard text parsing methods.

1.2.1 Improving Table Parsing for Enhanced Information Extraction

The optimization of tabular data extraction is crucial for enhancing the quality and accuracy of retrieved information, which, in turn, significantly improves outcomes in question answering tasks. The examination includes a variety of tools and custom models developed to deal with the challenges table parsing presents, evaluating their contributions to the document parsing framework as a whole.

The implemented approach involves the application of parsing tools, namely *Azure AI Document Intelligence* (formerly known as *Form Recognizer*) and *Unstructured Core Library/API Services*. Additionally, the incorporation of rule-based systems, designed specifically for distinct types of documents, plays a central role as well. The analysis delves into the complexities of these tools and models, exploring their effectiveness in handling diverse tabular structures and optimizing the extraction process to investigate their impact on the accuracy and completeness of extracted data.

1.3 Motivation and Research Questions

The motivation behind this dissertation stems from the practical challenges encountered while working on document parsing in an industry setting. During an internship at *Bayer*, involvement in the *Decision Science and Language AI* team included contributions to a significant project. This experience provided firsthand insights into the difficulties associated with extracting meaningful information from documents, particularly those containing intricate tables or presenting complex layouts. It highlighted the critical need for improved parsing techniques to enhance the effectiveness of document parsing and question answering tools.

1.3.1 Bayer's Project - Internal Generative AI Platform

The project focuses on Bayer's Internal Generative AI Operating System, developed to support various use cases across the value chain within a secure environment. Created by a multidisciplinary team of Bayer experts, the platform aims to utilize the potential of Generative AI within the organization. It is designed to enable broad access to Generative AI, enhancing innovation and productivity.

One of platform's key features is its integration of advanced AI capabilities, including ChatGPT and open-source language models, with access to both internal and external data sources. This integration improves knowledge access and content creation while ensuring compliance with data privacy and security standards, including IT strategies, cybersecurity, legal, data privacy, and responsible AI protocols.

The platform supports various functionalities, such as handling documents, audio, images, and video, and it can be customized for specific use cases. Additionally, the User Interface and API access facilitate the scalable adoption of Generative AI, enabling users and teams to utilize AI-driven insights and automation in their respective areas.

This dissertation focuses on experiments conducted on parsing and document question answering processes to investigate their performance and explore potential enhancements. The project served as a foundation for this research, aiming to demonstrate possibilities for enhancing existing functionalities, particularly in document parsing and the subsequent question answering process. This involved a detailed analysis of the current system's performance in these areas and the development of strategies and approaches to improve efficiency and effectiveness. These enhancements aimed to optimize document processing and understanding, thereby enriching the Generative AI's contribution to decision-making processes.

1.3.2 The Role of Parsing and Question Answering Tools

Document parsing plays a crucial role within this research, as it enables the processing and understanding of various file types, including PDFs, images, Word documents, and PowerPoint presentations. This allows the efficient extraction of valuable information from different types of documents. The document question answering tool leverages document parsing by allowing users to query parsed documents and receive accurate, context-aware answers. This capability significantly enhances knowledge discovery and content creation processes by providing quick access to relevant information embedded within complex documents.

Together, these capabilities highlight the pivotal role of parsing and question answering tools in enabling comprehensive document analysis and facilitating informed decision-making.

1.3.3 Research Questions

This dissertation delves into several key research questions that are crucial for understanding the impact of table parsing on document parsing and question answering systems:

- *How does the accuracy of table parsing affect the overall performance of document parsing systems?*
  This question explores the direct impact of precise table extraction on the integrity and utility of the parsed document content, assessing how well different parsing techniques handle complex tabular data.

- *What improvements can be achieved in question answering results by integrating optimized table parsing techniques?*
  This question points to the enhancements in question answering performance related to the inclusion of accurate table parsing, evaluating the contribution of tabular data to the overall quality of answers.

- *How do different parsing tools and models compare in terms of their effectiveness and efficiency in handling tabular data?*
  This question involves a comparative analysis of various parsing tools, assessing their performance and robustness in processing different types of documents.

1.3.4 Experimental Approach

To address the outlined research questions, a set of experiments is designed to evaluate both the parsing approach of the involved parsers and the results of the document question answering process post-parsing. This evaluation encompasses a range of diverse documents such as PubMedQA, Master Batch Record, and Standard Operating Procedure documents, chosen for their diversity and relevance, providing a robust benchmark for assessing the proposed parsing techniques.

1.4 Summary

In summary, this dissertation aims to demonstrate that the integration of optimized table parsing with general document parsing procedures can lead to significant improvements in the performance of document question answering systems. Through a series of experiments and analyses, evidence will be provided to support this hypothesis, along with insights into the best practices for achieving accurate and efficient document parsing.

The comprehensive approach involves evaluating parsing techniques using various documents, measuring their performance during both parsing and question answering tasks with standard metrics, and conducting case studies to validate the findings in real-world scenarios.

Chapter 2

Background

The processes of parsing and document analysis represent the basis for understanding and extracting meaningful information from both unstructured and semi-structured data. These tasks involve breaking down textual information into structured representations for further analysis. Document analysis is a subset of parsing, and it focuses on understanding the structure, content, and context of documents, thereby enabling processes such as information retrieval and natural language processing and understanding.

Parsing techniques, ranging from traditional methods to deep learning approaches, play a central role in many different applications, not least in document question answering systems. These systems leverage the structured data derived from parsing to generate precise answers to user queries, thereby demonstrating the importance of parsing in real-world applications.

This chapter explores the key concepts of parsing and its applications, focusing on document analysis and information retrieval. It provides an overview of various parsing techniques and introduces the processes of document question answering and evaluation.

2.1 Understanding Parsing in NLP

Natural Language Processing is a domain within Artificial Intelligence that focuses on the interaction between human and machine languages. It can be described as the process by which machines extract information from natural language inputs and then generate natural language outputs. Natural Language Processing can be categorized into two main components, i.e., *Natural Language Understanding* or Linguistics, which involves linguistic analysis to comprehend the text, and *Natural Language Generation*, which implements the task of text generation [10].

The main aspects of Natural Language Processing include information retrieval, which involves the storage, search, and retrieval of information from textual documents, machine translation, which is related to the automatic translation of one human language into another, and language analysis, which is related to parsing the input sentences to construct syntactic trees and perform semantic analysis [6]. Together, these components, outlined in Figure 2.1, enable machines to process and understand human language.

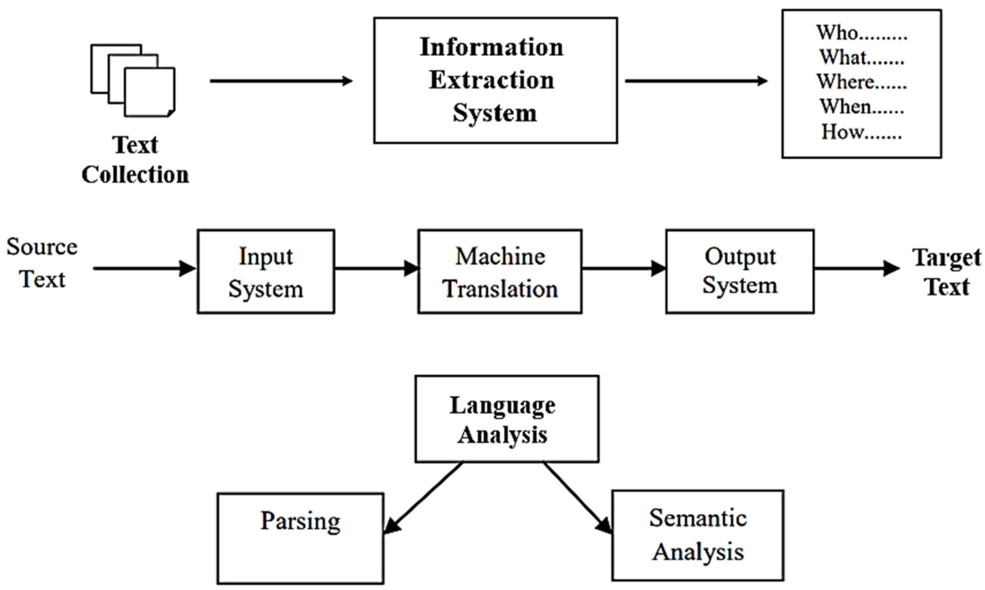


Figure 2.1: Information Extraction, Machine Translation, Language Analysis [6]

Parsing is a fundamental task within Natural Language Processing, bridging the aforementioned gap between human language and machine comprehension. It involves the analysis of text to understand the grammatical structure and the relationships between words, with the aim of enabling machines to reproduce human language with a certain degree of accuracy. There are two main types of parsing:

- **Syntactic Parsing**: This type focuses on the grammatical structure of sentences, identifying parts of speech and the syntactic roles of words.
- **Semantic Parsing**: This type aims to understand the meaning of entities within the sentence beyond their syntax, by interpreting the context.

The application of parsing techniques extends beyond traditional linguistic analysis. The following section delves into the topic of document and table parsing. Through these processes, the goal is to explore and examine techniques and methods implemented to handle the complexities of unstructured textual information and tabular data formats.

2.2 Document and Table Parsing

Document parsing is a fundamental task within the field of Natural Language Processing. It involves the systematic examination of text documents with the aim of extracting relevant information, relationships, and entities. Its importance derives from the ability to dissect both unstructured and semi-structured textual contents. Therefore, it can be defined as the process of extracting structured data from unstructured documents, identifying relevant information, and structuring and organizing it into a format that is both usable and readable.
Document parsing finds its applications in many different domains, highlighting its role in performing information extraction, data mining, and natural language understanding. A **data parser** is a tool that analyzes the information presented in a specific format, taking a large set of data and breaking it down

into smaller and meaningful components. Parsers are used in all high-level programming languages and can be integrated into existing systems to automate workflow in an efficient way. Typically, a document parser employs Optical Character Recognition (OCR) to analyze documents, while advanced parsers may also adopt machine learning approaches [20]. The parser should be capable of distinguishing between various document sections, such as headings, paragraphs, narrative text, titles, and tables, enabling the extraction of specific data and insights from each of them.

The process previously described is known as *Data Parsing*, which is usually performed using two main approaches [11]:

- **Rule-based approach**: In this method, the user defines a template of the document to be used as a reference for extracting information. The drawback of this procedure is that it is necessary to provide a document with the same structure as the template, since even slight differences could prevent the method from working properly.

- **Model-based approach**: This method is mostly used for extracting data from unstructured documents and relies on machine learning and natural language processing. These models are trained using a large range of unstructured documents to enhance their ability to recognize fields and extract data.

In practice, document parsers combine both rule-based and model-based approaches, enabling them to efficiently process different types of document formats, regardless of their layout, and to effectively extract data.

Document parsing faces numerous challenges, ranging from the complexity of document structures to the need for data accuracy and validation post-parsing. The adaptability of parsing tools to different document formats is necessary for achieving accurate data extraction. However, ensuring data accuracy can be particularly challenging, especially with handwritten documents or those formatted in unconventional ways, requiring strong and robust recognition capabilities. After parsing, the validation of extracted data is essential to eliminate unreliable information, especially when dealing with sensitive processes. Scalability is also critical, because parsing solutions need to handle growing volumes of documents in an efficient way to meet business demands. Last but not least, the flexibility to adapt to changing requirements over time is necessary to avoid constant workflow re-automation efforts [19].

Beyond these challenges, there are additional complexities in document parsing technologies, including the intrinsic difficulties of achieving 100% data extraction accuracy across many different templates and document formats, the challenge of debugging complex AI systems, and the limited support for multiple languages. All of the mentioned factors collectively contribute to making it difficult to efficiently parse documents.

In this context, table parsing is a critical aspect of document parsing, involving the extraction of structured data from tables within documents. Tables often hold crucial information, posing challenges for extraction, thus emphasizing the importance of accurately recognizing table structures. Table parsing, like document parsing, presents numerous challenges due to the diverse formats in which tables can be presented. The challenges illustrated in Figure 2.2 exemplify the difficulties in recognizing table structures. These include irregular structures, such as merged cells or nested tables, which make it difficult to extract information. Additionally, the presence of missing data in empty cells, variability of column headers, and the need to understand the context surrounding the table within the document, make the parsing an even more complicated and demanding process [17].

As discussed thus far, effectively addressing both document and table parsing presents significant complexity but is also necessary for reliable information extraction. The *Azure Form Recognizer* and *Unstructured* parsers, which are explored in the following section, play an important role in this field of study.

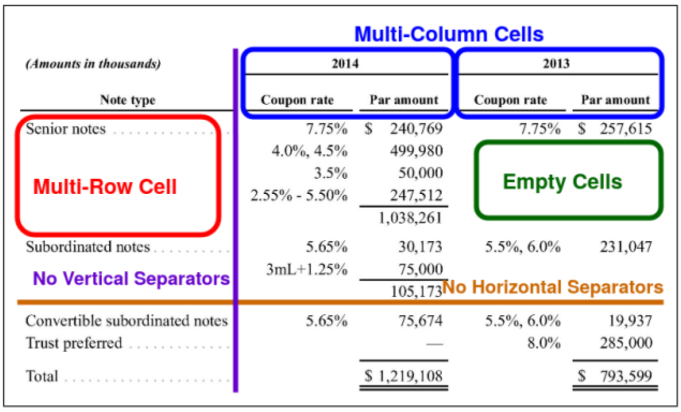


Figure 2.2: Challenges in Table Structure Recognition Task [17]

2.3 Tools for Document and Table Parsing

In the domain of document and table parsing, two state-of-the-art tools have emerged as crucial in transforming unstructured data into organized and valuable information: *Azure AI Document Intelligence* (formerly *Form Recognizer*) and *Unstructured Core Library/API Services*. These tools represent the combination of advanced machine learning models and algorithms to interpret and manage different document formats. Both are designed to automate data extraction and processing, enhance and enrich data-driven strategies, and facilitate document search tasks, marking a significant improvement in document management and offering businesses new ways to use their information that were not available before.

Azure AI Document Intelligence, formerly known as Form Recognizer, is a cloud-based service within Azure AI. It simplifies the creation of intelligent document processing solutions by using machine learning models to extract text, key-value pairs, and tables from different types of documents, including forms, invoices, and receipts [1]. Its key features include:

- **Document Understanding**: Ability to comprehend and extract structured data from various document formats.
- **Pre-built models**: Availability of pre-built models that simplify the extraction of common information from documents.

- **Integration**: Seamless integration with other Azure services and tools.
- **Advanced analytics**: Provision of advanced analytics functionalities, such as sentiment analysis and entity recognition.

Unstructured offers a comprehensive suite of API services and open-source solutions, including the Unstructured Core Library, which further enhance its capabilities. The library has the aim to simplify the preprocessing of documents for downstream tasks [22]. Key features include precise document extraction, extensive file support, and robust core preprocessing functionalities such as *partitioning* and *cleaning*. The partitioning functions within Unstructured enable the extraction of structured content from unstructured documents by breaking them into *'Elements'* such as Title, Narrative Text, Table, List Item, and more. This segmentation process allows for the selection of content that is specifically relevant for the application requirements. Depending on the file type of the source document, Unstructured determines the appropriate partitioning function to apply, ensuring efficient and relevant data extraction.

In this scenario, both Azure Form Recognizer and Unstructured serve as document parsers but they differ in their capabilities and approaches. Azure Form Recognizer is usually customized for extracting data from documents with predefined layouts, as it employs a machine learning approach. On the other hand, Unstructured is designed to extract information from semi-structured and unstructured documents without relying on templates. However, both offer flexibility and customization options to handle diverse formats and structures within analyzed documents.
Despite their differences, they share the same main goal of facilitating comprehensive document understanding and analysis, thus helping in the implementation of the question answering process over documents. Indeed, Azure Form Recognizer and Unstructured can significantly streamline the extraction

and processing of data from documents, which is essential for generating answers that are accurate and contextually relevant.
As previously mentioned, parsers transform unstructured data into a structured format that can be then easily queried. This means that when questions are posed regarding a specific document, the tools have already organized the data in a way that makes it easier the task of finding the correct answer. The text is supposed to be formatted and structured for analysis by Large Language Models, enabling them to provide accurate answers [14]. This is particularly beneficial for complex question answering scenarios, allowing for a more reliable system that is capable of handling a variety of document types and queries, with the aim of providing insightful responses. This foundation is essential for the subsequent exploration into the document question answering process, where structured data translates into insights through advanced querying and analysis techniques.

2.4 Document Question Answering

Document question answering can be defined as the process of extracting relevant and pertinent information from a given document or set of documents to answer a user-provided question. This multi-stage process involves different steps, starting with document parsing, followed by question understanding, document retrieval and comprehension, and answer generation.
The diagram in Figure 2.3 illustrates the various stages of the document question answering process.
The process starts with **document parsing**, a crucial phase that involves breaking down the documents into more manageable units, typically into sentences or words, using techniques like tokenization and sentence segmentation. This step also involves cleaning and preprocessing the text to remove any irrelevant information such as stop words, punctuation, and special characters. Additionally, stemming and lemmatization may be applied to reduce words to

their root form, ensuring that different forms of the same word are treated as identical.

The next stage is **question understanding**, a process that leverages natural language processing techniques to extract the semantic and syntactic structure of the question. Techniques such as Named Entity Recognition, Part-of-Speech Tagging, and Dependency Parsing are used to identify key entities, understand the grammatical role of each word, and determine the relationships between different words in the question, respectively. This step of the process also involves keywords extraction, which highlights the important words or phrases that will guide the successive steps.

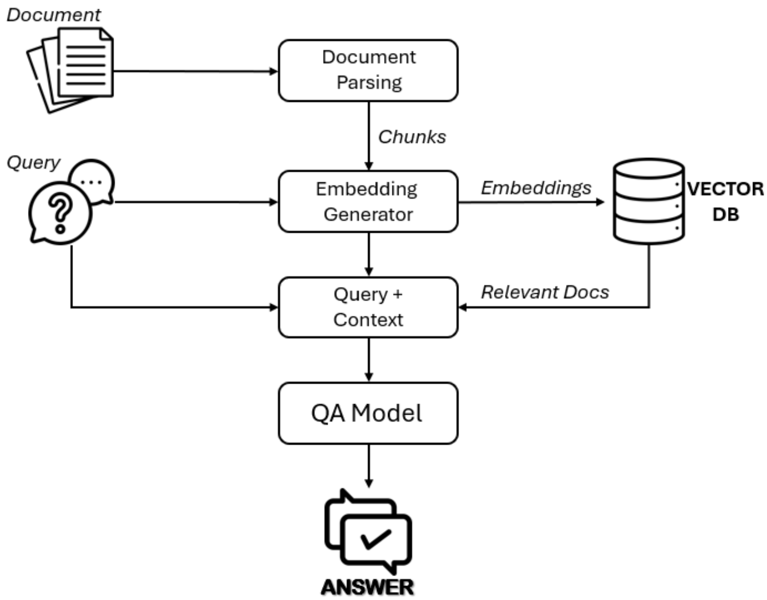


Figure 2.3: Diagram Illustrating the Document Question Answering Process

Following question understanding, the system proceeds to **document retrieval**. This step involves identifying and retrieving documents that are most likely to contain the answer to the question. Techniques such as Keyword Matching and Vector Space Retrieval are utilized. Keyword Matching identifies documents that contain the extracted keywords, while Vector Space Retrieval represents both the question and the documents as vectors in a high-dimensional space. More advanced techniques such as Latent Semantic Indexing or Topic

Modelling may also be employed. These approaches account for the semantic relationships between words and can thus retrieve documents that are conceptually related to the question, even if they don't contain the exact keywords.

The retrieved documents then undergo a **comprehension** stage. In this step, the system identifies sentences or chunks of text that are likely to contain the desired answer. This includes techniques like text chunking, which groups contiguous pieces of text together based on their Part-of-Speech tags, and coreference resolution, which links pronouns and other referring expressions to their appropriate entities. Machine learning models may also be used to score and rank the retrieved text chunks based on their relevance to the question.

The **final answer generation** step involves extracting or synthesizing an answer based on the identified text chunks. This could include simply extracting text from the document or more complex processes like answer synthesis, where the system combines information from multiple text chunks or rephrases the information to create a more concise and coherent answer.

Thus, the described process begins with parsing, after which the system must understand the question and extract keywords from it. Once the question is comprehended, the system retrieves relevant documents from a database. These retrieved documents are then analysed to identify sentences or chunks of text that likely contain the desired answer. This underscores the importance of correct document parsing; without it, the question answering system would be unable to determine the most appropriate response to the user's query.

The following section examines the evaluation of parsing and question answering techniques, assessing their effectiveness in extracting data and generating accurate responses based on parsed information, thereby providing a comprehensive assessment of natural language understanding systems.

2.5 Parsing and Question Answering Evaluation

Delving into the evaluation of the parsing process and the document question answering process within natural language understanding systems reveals crucial insights into their efficiency and performance. This involves comparing techniques and approaches, and identifying weaknesses in the involved applications.

2.5.1 Evaluating Parsing Approaches

The evaluation of parsing quality in the field of Natural Language Processing involves various approaches and metrics, reflecting the complexity of the parsing process. Parsing accuracy is one of the fundamental metrics used to determine the effectiveness of parsing algorithms. Additionally, precision, recall, and F1-score are among the quantitative measures implemented for evaluating the parsing results. Furthermore, considering the previously mentioned metrics, speed must be also considered as an important factor, particularly for applications that require real-time or large-scale parsing. The goal is to ensure that the parsing process is efficient and scalable.
When it comes to parsing using Azure Form Recognizer, its parsing quality is based on several metrics [1]:

- **Accuracy**: It ensures that the parsed information faithfully represents the content of the original document.
- **Confidence Scores**: Azure Form Recognizer assigns *confidence scores* to each extracted element, which can be used to evaluate the reliability of the parsing procedure. This value ranges from 0 to 1, where a value closer to 1 indicates higher confidence.
- **Error Rate**: The frequency of errors in extraction, including missed and wrongly identified information.

- **Processing Time**: A critical performance metric, reflecting the efficiency of the process.

Similarly, the evaluation of the quality of parsing using Unstructured is assessed through different factors [22]:

- **Extraction Quality**: The accuracy of the extracted data in relation to the content of the original document.
- **Detection Class Probabilities**: When Unstructured analyzes a document, it assigns each element a class based on what it believes the element represents. The *detection class probability* value is a measure of how confident the model is in its classification decision for each element. This value ranges from 0 to 1, where a value closer to 1 indicates a higher confidence.
- **Flexibility**: The ability to parse a wide range of file formats, reflecting the adaptability of the tool to varying data sources.
- **Processing Time**: As mentioned earlier, this metric is used to understand and evaluate the efficiency of the process.

By comprehensively evaluating parsing quality using the aforementioned metrics, it is possible to gain insights into the performance and capabilities of the involved algorithms. The evaluation step is essential for driving improvements, enhancing the reliability of NLP applications.

2.5.2 Evaluating Document Question Answering Process

Evaluating the document question answering process involves several methods to assess the effectiveness of providing accurate and relevant answers to user queries. When evaluating the document question answering system, both target answers and generated answers are used for comparison. The target answers, or ground truth, represent the correct, reference responses to the

given questions. Human experts typically manually annotate these answers and serve as the gold standard against which the performance is measured. Thus, they provide a benchmark for assessing the accuracy, relevance, and completeness of the system's responses, namely the generated answers.
Some common approaches to perform document question answering evaluation are:

- **Data Collection**: This involves collecting a dataset consisting of documents and corresponding questions, paired with their ground truth answers. The dataset is used as a baseline for evaluating the performance of the system.
- **Evaluation Metrics**: Define relevant evaluation metrics for assessing the performance of the system.
- **Human Evaluation**: Have experts assess the quality of the system's answers by comparing them to the ground truth ones. Through manual inspection and comparison with ground truth answers, experts can provide valuable qualitative feedback on the accuracy, relevance, and comprehensiveness of the system's responses.
- **Automatic Evaluation**: This involves the comparison of the system's answers to the ground truth ones using predefined evaluation metrics. Automated techniques such as BLEU (Bilingual Evaluation Understudy) and ROUGE (Recall-Oriented Understudy for Gisting Evaluation) can be used to evaluate the quality of generated responses.

Implementing a combination of these methods, the evaluation of the document question answering system can be performed to enhance its overall performance. To this end, a variety of automated evaluation metrics are utilized, each offering different insights into the quality of the answers provided by the system.

ROUGE Score - Recall Oriented Understudy for Gisting Evaluation

ROUGE, short for Recall-Oriented Understudy for Gisting Evaluation, comprises several automatic evaluation methods designed to measure the similarity between summaries. It includes a range of measures, such as ROUGE-N, which specifically evaluates the overlap of n-grams between system-generated text and reference text, with a focus on recall. Formally, ROUGE-N assesses n-gram recall by comparing a candidate summary to a set of reference summaries.

The computation of ROUGE-N can be seen in Figure 2.4. In the displayed formula, n represents the length of the n-gram, $gram_n$, and $Count_{match}(gram_n)$ is the maximum number of n-grams co-occurring in a candidate summary and in a set of reference summaries [13].

$$\text{ROUGE-N} = \frac{\sum_{S \in \{ReferemceSummaries\}} \sum_{gram_n \in S} Count_{match}(gram_n)}{\sum_{S \in \{ReferenceSummaries\}} \sum_{gram_n \in S} Count(gram_n)}$$

Figure 2.4: Formula for Computing ROUGE-N Score [13]

ROUGE-L is based on the longest common subsequence, which can be useful for the evaluation of the fluency and coherence of the output provided by the system. For document question answering systems, ROUGE scores provide a measure of how comprehensively the system's answers capture the main contents of the reference text.

BLEU Score - Bilingual Evaluation Understudy

The BLEU score is commonly used for evaluating machine translation. It operates on the principle of n-gram co-occurrence between the generated translation and one or more reference translations.

BLEU assesses the quality of generated text by computing the precision of n-grams, which are continuous sequences of *n* words from the text. BLEU also includes a brevity penalty to avoid favouring translations that are too brief, concise, incomplete, or lacking necessary details. In the context of document question answering, the BLEU score can measure the precision of the system's answers by comparing them with target answers, providing a quantitative measure of lexical and syntactic similarity [15].

The BLEU score is computed using the formula shown in Figure 2.5, where BP represents the *brevity penalty*. The value of the brevity penalty is outlined in Figure 2.6.

$$\text{BLEU} = \text{BP} \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right)$$

Figure 2.5: Formula for Computing BLEU Score [15]

$$\text{BP} = \begin{cases} 1 & \text{if } c > r \\ e^{(1-r/c)} & \text{if } c \leq r \end{cases}$$

Figure 2.6: Brevity Penalty in the Formula for BLEU Score [15]

Cosine Similarity

In the evaluation of document question answering systems, Cosine Similarity is another method to measure the similarity between the generated answers and the target answers. Cosine Similarity is a metric that measures how similar two vectors are by computing the cosine of the angle between them. In the context of text analysis, it is used to assess the similarity between documents or sentences. The value of the Cosine Similarity ranges from -1 to 1, where 1 indicates identical orientation, 0 indicates orthogonality (i.e., no similarity), and -1 indicates opposite orientation. This metric is handy for text because it accounts for the distribution of words rather than their absolute counts, making it invariant to document length.

Given two vectors of attributes, A and B, the Cosine Similarity is presented using a dot product and magnitude, as shown in Figure 2.7 [21].

$$similarity = cos(\theta) = \frac{A.B}{|A||B|}$$

Figure 2.7: Cosine Similarity [21]

Cosine Similarity can be applied using different text representation methods. Each method has its own strengths and use cases, from simple frequency-based approaches to advanced semantic embeddings.

Bag of Words Model. In the Bag of Words model, documents or sentences are represented as vectors where each dimension corresponds to a word from the vocabulary, and the value represents the frequency of the word in the document or text. This model involves the creation of a vocabulary of all the unique words in the text corpus and then the representation of each document or sentence as a vector indicating the count of each word from the vocabulary itself. This metric is used to compute the *lexical similarity* between the vectorized forms of the system's answers and the target answers.

The primary advantage of Bag of Words is its simplicity; by reducing text to a set of word counts, it allows for straightforward implementation and interpretation. While Bag of Words is a simple and effective approach for comparing sentences, it is important to note that it does not account for the semantic relationships or similarities between words, nor does it consider the relevance of words within the text itself. Consequently, it provides a basic measure of textual similarity sufficient for a clear, quantifiable measure of overlap between texts but may fall short in capturing deeper semantic meaning.

Term Frequency-Inverse Document Frequency. The TF-IDF representation enhances the Bag of Words approach by considering not only the frequency of words in a document, i.e., Term Frequency, but also their distribution across the entire corpus, i.e., Inverse Document Frequency. The Term Frequency (TF) measures how often a word appears in a document, while the Inverse Document Frequency (IDF) measures how important a word is by considering its occurrence across multiple documents. The formulas for computing the Term Frequency and the Inverse Document Frequency are outlined in Figure 2.8.

This approach mitigates the problem of common words overshadowing important terms by assigning higher weights to words that are more unique and informative. It results in a weighted scheme that emphasized words that are more meaningful within each document or sentence, creating a substantial representation of the textual content.

$$TF = \frac{\text{number of times the term appears in the document}}{\text{total number of terms in the document}}$$

$$IDF = log(\frac{\text{number of the documents in the corpus}}{\text{number of documents in the corpus contain the term}})$$

$$TF\text{-}IDF = TF * IDF$$

Figure 2.8: Term Frequency - Inverse Document Frequency Formulas [9]

By applying Cosine Similarity to TF-IDF vectors, it is possible to obtain a more informative and rich evaluation of the generated answers, as this approach takes into account both the *lexical presence* and the *significance of words* within the text.

Sentence Transformer. Sentence Transformers, for example *BERT* or *RoBERTa*, represent a significant advancement in the field of Natural Language Processing. The development of Sentence Transformers has introduced the capability to generate semantically rich embeddings for sentences. These models work by encoding sentences into dense, fixed-size vectors and they are trained to create embeddings that place semantically similar sentences in close proximity within the embedding space. When Cosine Similarity is computed between embeddings of the generated answers and the target ones, it captures the *semantic similarity* at a level that goes beyond simple word co-occurrence, meaning at a much deeper level than Bag of Words or TF-IDF representations. This allows for a deeper evaluation of the meaning conveyed by the answers, enabling the assessment of not just lexical similarity but also the underlying intent and context of the sentences. This makes Sentence Transformers particularly effective for tasks requiring an understanding of sentence meaning and context.

Universal Sentence Encoder. The Universal Sentence Encoder is another advanced method for generating embeddings, designed to capture comprehensive semantic information from text. This encoder utilizes deep learning techniques and it is optimized for different natural language understanding tasks, including semantic similarity, text classification, and clustering [4]. Universal Sentence Encoder generates high-dimensional embeddings for sentences that encapsulate complex linguistic patterns and contextual dependencies.

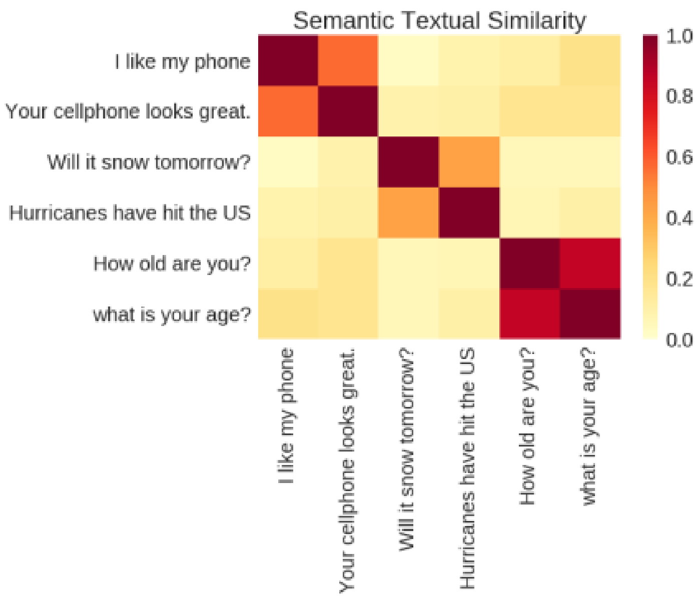


Figure 2.9: Similarity Scores with Universal Sentence Encoder Embeddings [3]

Figure 2.9 shows the sentence similarity scores using embeddings from the Universal Sentence Encoder [3]. By using Cosine Similarity on these embeddings, it is possible to evaluate the similarity between texts at a *semantic level*, discerning nuances in meaning and intent that simpler models might miss. This approach is particularly advantageous for tasks where the phrasing may differ but the underlying meaning is consistent.

By utilizing a variety of metrics, the assessment procedure for document question answering systems becomes comprehensive, multi-dimensional and robust. Each metric offers a unique viewpoint on the quality of the responses, ranging from basic lexical similarity to more complex semantic comprehension. This diverse method of evaluation is crucial in highlighting areas that require improvement, thereby driving the advancement of document question answering systems that are not only more accurate but also contextually sensitive.

2.6 Summary

This chapter focuses on the role of parsing and document analysis in extracting essential information from unstructured and semi-structured data. It emphasizes the significance of various parsing techniques, spanning from traditional methods to deep learning approaches, particularly in the context of document question answering systems.

The discussion begins with an overview of parsing in the field of Natural Language Processing, highlighting syntactic and semantic parsing as fundamental components. Furthermore, it explores document and table parsing, addressing associated challenges and methodologies, including rule-based and model-based approaches for information retrieval from documents.

The introduction of Azure AI Document Intelligence and Unstructured Core

Library/API Services as parsing tools is presented, detailing their functionalities and contributions to data-driven tasks and document search operations. The chapter delves into the process of document question answering, emphasizing the pivotal role of accurate parsing in achieving optimal performance. The chapter concludes with an exploration of evaluation tools and methods for both parsing and document question answering processes. It discusses various metrics utilized to assess the effectiveness of natural language processing systems in comprehending and generating human language.

Chapter 3

Methodology

This chapter delves into the methodology employed for implementing the document parsing process and the subsequent document question answering tasks in the experiments and research conducted. The emphasis is placed on the critical role of the parsing procedure, which is fundamental in the information extraction process. The purpose of the parsing procedure, detailed in the subsequent sections, is to analyze documents with the aim of extracting as much detailed data and information as possible.
The described methodology not only covers the extraction of textual information but it also extends to the task of table parsing, which involves the extraction and understanding of data presented in tabular format within documents.

This approach is implemented with the objective of potentially enhancing the question answering process on processed documents. To investigate this enhancement, subsequent evaluations are conducted on the responses generated during question answering processes, based on the information extracted during parsing. The steps of the evaluation process will be discussed at the end of this chapter.

3.1 High-Level Document Question Answering

Document question answering is a complex task in Natural Language Processing that involves understanding and extracting precise answers from a document based on a given question. This task requires the model to have a deep understanding of the document's content, context, and structure.

One of the methods utilized in document question answering is the *Retrieval-Augmented Generation* model, which combines the strengths of both retrieval-based and generative approaches to provide a more accurate answer.

In the RAG method, the process starts with the retrieval of relevant documents or passages from a large corpus based on the given query by using dense vector retrieval methods to create high-dimensional vector representations of both the query and the documents. The vectors are compared to find the documents that are most similar to the query. Once the relevant documents are retrieved, a generative model is used to create the response.

The RAG model offers several advantages for document question answering. By retrieving relevant documents before generating the response, it can provide a more detailed and pertinent answer than a model that only uses the query as input. Additionally, by including a generative model, it can generate more fluent and human-like responses than those produced by extractive models.

As crucial as methods like RAG are to the document question answering process, their efficacy heavily relies on the quality of the document parsing procedure. Parsing is the first step in the data processing pipeline, and it significantly influences subsequent steps, including the performance of document question answering models.

To delve deeper into the process of document parsing and its crucial role, the next section explores the stages of the data processing pipeline. This section will focus on how parsing methodologies can significantly impact the overall accuracy and efficiency of document understanding.

3.2 Detailed Document Parsing Approach

Document Parsing is the initial step of the data processing pipeline, transforming raw documents into structured data that can be easily interpreted. This process is automatically initiated upon the upload of a file to the system. Specifically, when a user uploads a new file, the uploading process automatically triggers the parsing endpoint to analyze the document using one of the aforementioned tools, namely the Azure Form Recognizer parser and the Unstructured parser, based on the supported file extension.
In order to demonstrate how table parsing can enhance the overall document parsing process by extracting relevant information stored in table formats, the analysis begins by examining the responses of the two implemented parsers, introduced in the previous chapter.

3.2.1 Parsing with Form Recognizer and Unstructured

The behaviour of the **Azure Form Recognizer** parser is defined within the `FormRecognizerParser` class. This class is responsible for uploading and parsing files, indeed utilizing the Form Recognizer parser, which supports the following file extensions: [*".pdf"*, *".bmp"*, *".png"*, *".jpeg"*, *".jpg"*, *".tiff"*]. In this class, the client `DocumentAnalysisClient` is instantiated, providing different methods such as `begin_analyze_document()`, which specifically is designed to analyze text fields and semantic values in a given document. The method returns an instance of `LROPoller` from which it is possible to call the `result()` method to retrieve an `AnalyzeResult` object [1]. This object contains various attributes, including documents, pages, paragraphs, and tables. Consequently, it is possible to leverage the object itself to retrieve data from the input document for understanding its content and layout. The focus is placed on the extracted pages, which are represented as a list of `DocumentPage` objects, and on extracted tables, represented as a list of

`DocumentTable` objects. Each page is processed to create a `Document` element to store information related to it, such as the document identifier, text and additional metadata, like the source and the author related to the uploaded document, and the page number. The text of a given page is retrieved using the `DocumentLine` class, which allows concatenation of textual information line by line. Moreover, the analysis of lines turns to be relevant to detect potential table captions, which typically follow a regular pattern, easily identifiable (e.g., *"Table n. Caption of the table"*).

After creating the `Document` elements, the next step is the analysis of the retrieved list of `DocumentTable` objects. Each object represents a table consisting of table cells arranged in a rectangular layout [1]. Figure 3.1 shows an example of a parsed table in the Form Recognizer `result` object.

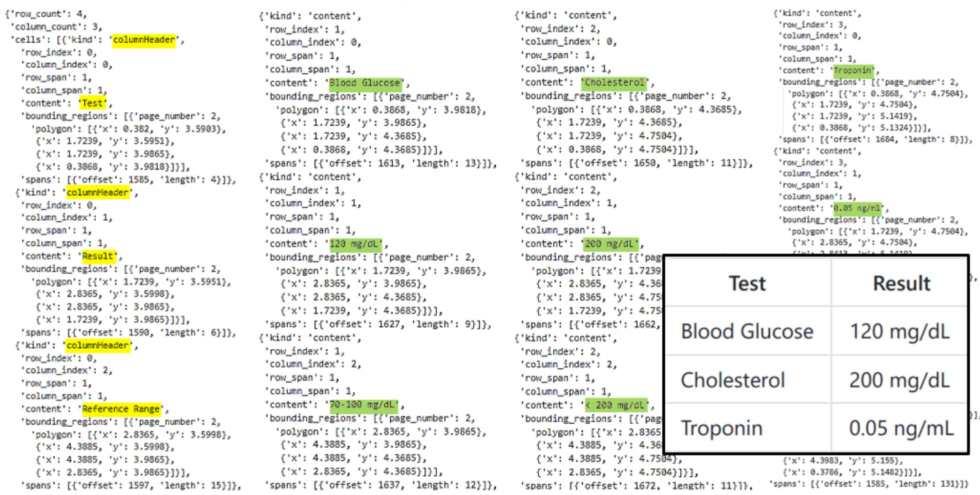


Figure 3.1: Parsed Table Example in Form Recognizer *'result'* Object and Reference Table Snapshot

Among the attributes, one can find the cells contained within the table and, by utilizing the *"kind"* field of cell objects, it is possible to store information about column headers and row cells, while maintaining the association with the former table by the table identifier. Similarly to what has been implemented for the pages of the document, one `DocumentTable` element is created for each table, storing the table caption, the list of column headers, and a dictionary representing the rows. The rows dictionary is a crucial component of the table parsing process. This mentioned dictionary is implemented by utilizing column headers as keys and cell contents as values. This strategy maintains the correlation between rows and columns, guaranteeing that the stored text remains as insightful as possible. It not only preserves the semantics of the table data but also enables effective reconstruction of the table's structure. Indeed, this correlation prevents misinterpretations and mismatches, preserving the context of the data. This approach ensures a comprehensive understanding of the meaning and significance of the information within the table; also, the connection between column names and row contents significantly simplifies querying tasks.

While the textual content of documents is stored as plain text, table content is retained in the **Markdown** format, which is generated using the aforementioned rows dictionary. The creation of a Markdown representation for tables within a document during the parsing process offers numerous advantages. Markdown's simplicity and straightforward syntax make it easy to read, write, and transform tables into this format, and its wide acceptance across multiple platforms enhances the portability of tables, facilitating their sharing and visualization across various platforms. The compatibility with HTML is also beneficial since it enables the conversion when displaying the table on a web page. By using Markdown, the structure and relationship between different data points are preserved, ensuring that the integrity of analyzed data is maintained. Thus, the Markdown format provides a simple and compact way to maintain the table layout, which proves to be particularly beneficial for retrieval purposes.

A very similar reasoning is applied when parsing documents using **Unstructured** as the parser. The `UnstructuredParser` class defines the parser's behavior, which supports the following file extensions: [*".txt"*,*".eml"*,*".html"*, *".md"*, *".json"*, *".jpeg"*, *".jpg"*, *".png"*, *".doc"*, *".docx"*, *".ppt"*, *".pptx"*, *".pdf"*, *".xlsx"*, *".xls"*, *".csv"*]. The parsing process involves making an API

POST request with specified data to customize the response generation. The API endpoint provides several parameters to tailor the processing of documents, with the specific request data outlined as follows:

- **files**: the file to be parsed;
- **strategy**: "`hi_res`";
- **skip_infer_table_types**: "`[]`";
- **pdf_infer_table_structure**: "`true`".

The "`hi_res`" strategy leverages document layout to gain additional information about document elements [22]. The specification of the last two parameters is essential. The default value for "`skip_infer_table_types`" is [*".pdf"*, *".jpg"*, *".png"*, *".heic"*], indicating document types for which table extraction should be skipped [22]. As a result of this parameterized request, table parsing within PDFs is performed in the same manner as for all the other supported file extensions. Specifically, when "`hi_res`" is selected, "`skip_infer_table_types`" is set to empty list, and the parameter "`pdf_infer_table_structure`" is set to `True`, any `Table Elements` extracted from all types of documents will include an additional metadata field, namely *"text_as_html"*, providing the HTML representation of extracted tables [22], as shown in Figure 3.2.

The parsing process relies on retrieved `Element` objects, simplifying the preprocesing of structured and unstructured documents for various tasks and ensuring data from any source is transformed into an easily usable format [22]. The output is a list of document `Element` objects representing different components of the source document; element types can be distinguished by examining the *"type"* parameter. When iterating through the retrieved elements, two main cases are distinguished:

1. if the analysed element *"type"* is equal to *"Table"*, then the table text is stored as an HTML string within a list of tables;

2. in any other case, the retrieved text associated with the current element is concatenated into a string.

The extracted text is then used to generate a `Document` object, as described also for the Form Recognizer parser. Additionally, each HTML string in the list of tables is processed to generate a `DocumentTable` object. As previously mentioned for Form Recognizer, the table content should be converted into **Markdown** format. To achieve this, the HTML string representing the table must be converted into a dictionary of rows. Therefore, the following HTML tags are utilized to reconstruct the table layout:

- <thead>: is used to group header content;
- <th>: defines a header cell;
- <tr>: defines a row and contains one or more <th> or <td> elements;
- <td>: defines a standard data cell.

In so doing, by listing and associating headers and row cells, the rows dictionary needed for conversion into Markdown is created.


```
{'type': 'Table',
 'element_id': '0530e563755cc98493846d0d86255a2c',
 'text': 'Test Result Reference Range Blood Glucose 120 mg/dL 70-100 mg/dL
          Cholesterol 200 mg/dL < 200 mg/dL Troponin 0.05 ng/mL < 0.03 ng/mL',
 'metadata': {'text_as_html': '<table><thead><th>Test</th><th>Result</th><th>Reference Range</th></thead>
                                <tr><td>Blood Glucose</td><td>120 mg/dL</td><td>70-100 mg/dL</td></tr>
                                <tr><td>Cholesterol</td><td>200 mg/dL</td><td>&lt; 200 mg/dL</td></tr>
                                <tr><td>Troponin</td><td>0.05ng/mL</td><td>&lt; 0.03 ng/mL</td></tr></table>',
  'filetype': 'application/pdf',
  'languages': ['eng'],
  'page_number': 2,
  'parent_id': '39c574dd5342ecad8f5e1589362982a3',
  'filename': 'Medical%20Report.pdf'}}
```


Test	Result	Reference Range
Blood Glucose	120 mg/dL	70-100 mg/dL
Cholesterol	200 mg/dL	< 200 mg/dL
Troponin	0.05 ng/mL	< 0.03 ng/mL

Figure 3.2: Parsed Table Example in Unstructured *'result'* Object and Reference Table Snapshot

Until now, the general procedure for parsing files using both Form Recognizer and Unstructured parsers has been outlined.

However, it is also possible to customize their behavior to parse files following a specific template by implementing custom rule-based parsing approaches for the parsers. In doing so, ensure that relevant information contained within those files is neither overlooked nor misunderstood. Consequently, this optimization also ensures that the document question answering process is enhanced.

3.2.2 Custom Rule-Based Parsing Models

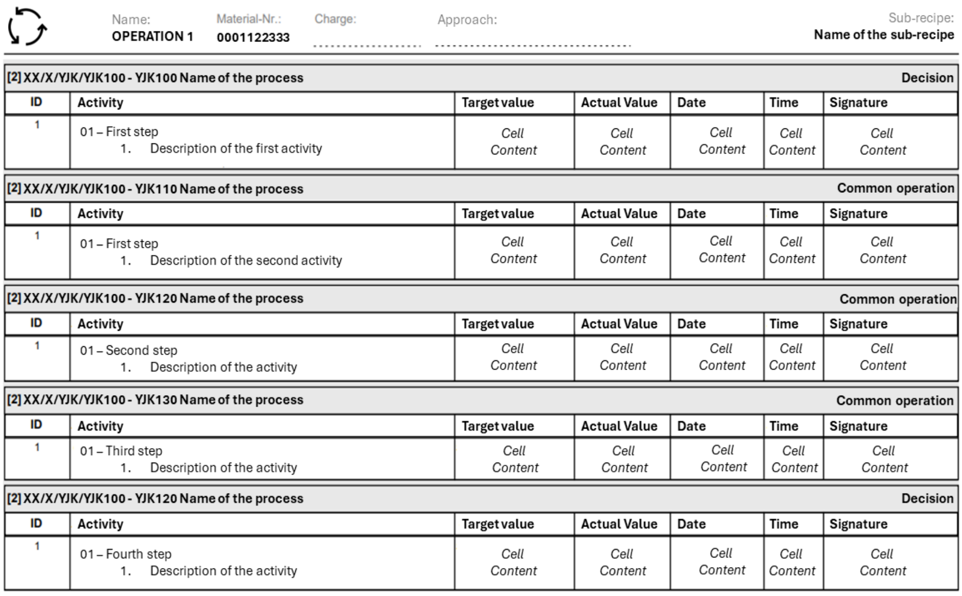
A Rule-Based Custom Model for parsing with Unstructured and Form Recognizer parsers can be described as a tailored approach to retrieve data from documents using predefined rules or patterns. The idea is to define these specific rules or patterns, such as regular expressions, to guide parsing algorithms and improve the overall parsing process. In this analysis, custom models are defined in order to correctly parse Master Batch Record and Standard Operating Procedure documents used for the conducted experiments, which are subsequently employed to perform parsing and document question answering evaluation.

Parsing MBR Documents

A **Master Batch Record** (MBR) is a document that contains the approved ingredients, formulation, and instructions guiding the production of a pharmaceutical product. This document enables manufacturers to follow the necessary regulatory guidelines when manufacturing a product [12].
MBRs are files containing recipes instructing the production line on obtaining a certain product. These documents usually present a highly specific structure that must be analyzed accurately to understand the described procedures in the correct way, as they describe sensitive processes. The provided documents are primarily in PDF format and mainly consist of tables. Since both Unstructured and Form Recognizer support PDF file extension, they can be

parsed using either of these tools.

Figure 3.3 illustrates one possible example of a page from a Master Batch Record. In this specific case, for the documents used in the experiments, on each PDF page, located at the upper right corner, the name of the *subrecipe* that is described by tables within the page itself can be found. This indicates that both the relationship between each table and its corresponding subrecipe, as well as the relationships between tables stored on the same page, must be maintained. These relationships are essential for describing all the steps of each subrecipe accurately. Each table describes one process, which is indicated as table header/caption using a path-like structure (e.g., *XX/XX/YYY/YYY000 - YYY000 Name of the process*) where the last part (in the example, *YYY000*) serves as its identifier. Therefore, when parsing MBRs, it is important not only to preserve the table structure but also its caption, describing the path where the process is stored inside the system, and the subrecipe to which it is associated.


Name: OPERATION 1 Material-Nr.: 0001122333 Charge: Approach: Sub-recipe: Name of the sub-recipe

[2] XX/X/YJK/YJK100 - YJK100 Name of the process						Decision
ID	Activity	Target value	Actual Value	Date	Time	Signature
1	01 – First step 1. Description of the first activity	Cell Content	Cell Content	Cell Content	Cell Content	Cell Content

[2] XX/X/YJK/YJK100 - YJK110 Name of the process						Common operation
ID	Activity	Target value	Actual Value	Date	Time	Signature
1	01 – First step 1. Description of the second activity	Cell Content	Cell Content	Cell Content	Cell Content	Cell Content

[2] XX/X/YJK/YJK100 - YJK120 Name of the process						Common operation
ID	Activity	Target value	Actual Value	Date	Time	Signature
1	01 – Second step 1. Description of the activity	Cell Content	Cell Content	Cell Content	Cell Content	Cell Content

[2] XX/X/YJK/YJK100 - YJK130 Name of the process						Common operation
ID	Activity	Target value	Actual Value	Date	Time	Signature
1	01 – Third step 1. Description of the activity	Cell Content	Cell Content	Cell Content	Cell Content	Cell Content

[2] XX/X/YJK/YJK100 - YJK120 Name of the process						Decision
ID	Activity	Target value	Actual Value	Date	Time	Signature
1	01 – Fourth step 1. Description of the activity	Cell Content	Cell Content	Cell Content	Cell Content	Cell Content

Figure 3.3: MBR Document PDF Page Example: Table Structure Illustration - *Content Redefined*

Custom parsing rules are defined within the **Unstructured** module, where the parsing process is redefined. The subrecipe of the process is identified using a pattern matching approach, with the page number being retained for subsequent association of the MBR table with the corresponding subrecipe. Two regular expressions are established to retain table captions for both standard and MBR tables. Upon matching a MBR table caption, a specific flag is activated, and the caption is also stored into a list. The same process is applied to a standard table, excluding the activation of the flag. This Boolean value is essential, firstly, for storing the second part of the caption for MBR document tables. Indeed, the caption comprises not only the path-like structure previously mentioned but also of a final textual element (e.g., *"Decision"* or *"Common operation"*, as shown in Figure 3.3), which is usually identified by Unstructured as a separate element from the path. Furthermore, the flag is used to check and determine when it is necessary to pair the MBR table with its subrecipe.

After the text has been read and the flag is activated, the table caption is updated with its final part. If the flag is not activated, the analysis of the elements continues. When the current element is a table and the flag is set to `True`, indicating the presence of a MBR table, the association between the table and the subrecipe is carried out using the page number.

Ultimately, a dictionary for storing tables is created, where the key is the caption and the value is a quadruple consisting of the Markdown representation, the subrecipe, the text in HTML format, and the page number.

All the information stored in the aforementioned dictionary is then utilized to create `DocumentTable` objects, each corresponding to one table. Specifically, the subrecipe and the page number enable the creation of relationships between consecutive tables under certain conditions. As mentioned before, tables within the same page, sharing the same subrecipe, denote subsequent steps of the same process. Therefore, the connections between them are crucial.

Prior to creating one `DocumentTable` object, the possible connection with the previously stored table must be checked, by performing the following steps:

1. **Checking for existing tables**: The process starts by checking the presence of tables within the list of tables. If the list is empty, the current one is marked as *"First table"*.

2. **Comparing subrecipes**: If the list is not empty, the subrecipe associated with the current table is compared with the subrecipe of the last inserted one. If they match, the process proceeds to the next step. Otherwise, the relationship is marked as *"No detected relationship - different subrecipe"*.

3. **Matching page numbers**: After checking the subrecipe match, the page number of the current table is compared with the page number of the last inserted one. If the page numbers match, the ID of the last inserted table is stored as the relationship ID. In case of a mismatch, the relationship is marked as *"No detected relationship - different page number"*.

In the case of Unstructured, if a table has more rows than a predefined number, the HTML text is used to create table chunks, which are converted into Markdown format and subsequently used to create embeddings. If the table's row count does not exceed the limit, the Markdown format stored in the aforementioned dictionary is directly used the create the embeddings.

The **Form Recognizer** parser also has specific rules defined for meaningful information storage when parsing MBR documents. Regular expressions are used for pattern matching to identify table captions within both MBR and regular documents. The same logic is applied to subrecipes, which are stored in a list paired with the page number. As implemented with the Unstructured parser, the association between one table and the subrecipe indicated in the page where the table is located is performed using the page number.
As already highlighted for Unstructured, one crucial piece of information that

needs to be stored is the relationship between subsequent tables, as they represent sequential steps within a process delineated by the subrecipe; the goal is to maintain the order and the associations between these tables.
To establish relationships between subsequent tables, the previously outlined steps are executed prior to creating the `DocumentTable` object:

1. **Checking for existing tables**.
2. **Comparing subrecipes**.
3. **Matching page numbers**.

The process ensures the preservation of the order and association among the extracted tables, facilitating downstream analysis and processing based on these established connections.
In all instances, prior to being embedded, the Markdown strings are concatenated with the table captions and subrecipes to maximize the meaningfulness of the text.

Parsing SOP Documents

A **Standard Operating Procedure** (SOP) document provides detailed step-by-step instructions that describe how to carry out a given process [5]. SOPs typically include information such as the purpose of the described procedure, the required materials or equipment to implement it, instructions, and expected outcomes. As for MBRs, SOPs contain mainly tables but they are in DOC or DOCX formats, thus allowing them to be parsed using only Unstructured.
Upon analyzing the selected documents for the experiments, it is evident that tables within SOP documents serve to provide a detailed description of the process being described. Each table depicts one of its steps and may consist of 1 to 3 columns, showcasing the roles involved in the step, its description, and document references. However, the primary focus is on the *"Description"* column as it typically contains all the necessary information to outline the process itself. In fact, below the aforementioned column, all potential conditions

in which the step is implemented are listed and the subsequent approaches to follow are described. One possible example of an SOP document table is shown in Figure 3.4.

3.2 *Subtitle for a short description of the process*

IT tools: Not applicable

Assigned role	Description	Document reference
List of the involved roles	**1)** First step • Condition to be satisfied: • *List of the implemented activities*	*List of references*

Figure 3.4: SOP Document Table Example: Structure Illustration - *Content Redefined*

Just as with MBR documents, the sequence of tables and their relationship are crucial pieces of information in this context. In fact, as previously noted, each table provides the detailed description of a step within the SOP outlined in the analyzed document. Therefore, it is important to preserve this information and leverage it effectively for extracting pertinent details from the document. To correctly store the content of process-description tables, custom rules for the Unstructured parser are defined by setting additional guidelines to those previously described.

First, it is essential to identify within the document where the detailed process description begins. This is achieved through pattern matching, where the text is analyzed to identify the presence of an element of type *"Title"* with textual content equal to *"Detailed process description"*. When this pattern is matched, it indicates that subsequent tables within the document describe the steps of the process; thus, a Boolean flag is activated. Additionally, since all table captions need to be stored, the `parent_id` value for the element of type *"Title"* and textual content *"Detailed process description"* is saved. The `parent_id` may be used to infer where an element resides within the overall hierarchy of a document [22]. This procedure also ensures that tables located

below a section of the document different from the one detailing the process description are parsed as standard tables. For instance, *"Title"* elements may have another *"Title"* element as a parent; these sub-titles can intend to be the captions of the subsequent *"Table"* elements in this specific case, located within the detailed process description section of the document. That's why this value is then used to store all the contents of successive elements of type *"Title"* as table captions, provided that the flag is active.
After saving the captions, the analysis of subsequent elements continues. If the current element is a table and the flag indicating the start of a SOP process is set to `True`, the extraction of the sub-table under the column *"Description"* is implemented, and its Markdown format is computed. As mentioned for MBR documents, the dictionary for storing tables is created, where the key is the caption and the value is, in this case, a tuple consisting of the Markdown representation and the text in HTML format.
Preserving the relationship between successive tables is essential for outlining the process required to achieve the desired outcome in the correct order. When it comes to SOP documents, the table's caption can be used to store associations. For all the other tables within the document, the captions follow a format such as *"Table 1"*, whereas captions for tables representing individual steps of the process are created using the step number and the short description associated with the step itself. Therefore, before creating the `DocumentTable` object, certain steps must be executed to establish the relationships:

1. For each table, the search for a pattern matching the caption is performed using the regular expression "`Table \d+`".

2. If no match is found in the caption, indicating a table representing a step of the process, the caption of the previous table is checked. If the previous table's caption contains a match with the pattern "`Table \d+`", the current one is marked as *"First table"*; otherwise, the table ID of the previous table is used as the relationship identifier.

3. If a match is found in the current table's caption, the relationship identifier is set to *"No detected relationship"*.

These steps involve checking captions and establishing relationships based on the presence or absence of specific patterns, ensuring that the relationship between tables is appropriately identified and handled.
Similar to MBR tables, if a SOP table has more rows than the predefined number, the HTML text is used to create table chunks, converted into Markdown format, which are subsequently used to create embeddings. If the table's row count does not exceed the predefined limit, the Markdown format stored in the aforementioned dictionary is used to create them.
Before embedding, the Markdown strings are combined with the table captions to enhance the overall meaningfulness of the text.
With the detailed document parsing approach now explained, attention turns to the post-parsing procedures. This phase involves integrating the extracted `Document` and `DocumentTable` elements into a vector database for subsequent processing and document question answering.

3.3 Post-Parsing Procedures

Upon completing the parsing process, whether using the Form Recognizer or the Unstructured parser, both `Document` and `DocumentTable` elements are then added to a vector database. In this research, the QDRANT Vector Database is utilized. This database acts as a vector similarity search engine, providing a user-friendly API to store, search, and manage points (i.e., vectors) along with additional payload. QDRANT is well-suited for deploying applications based on the matching of embeddings generated by neural network encoders. The use of vector databases facilitates quicker and more accurate retrieval of unstructured data already represented as vectors. This capability helps in providing users with the most relevant results for their queries [16].

Points in QDRANT are characterized by:

- unique point **ID**;
- **payload**, a JSON object containing metadata;
- **vector**, a high-dimensional representation of the data.

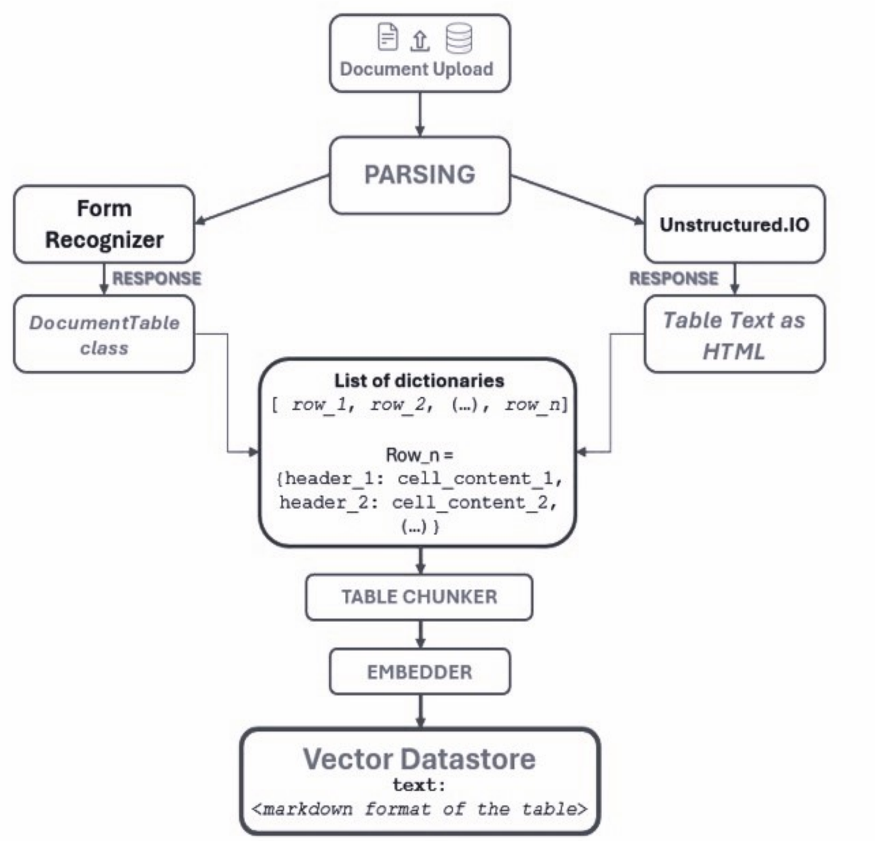


Figure 3.5: Diagram Illustrating Table Parsing within Document Parsing Process and its Subsequent Steps

To insert documents and tables into the vector database, they both need to be **chunked** and **embedded**. Chunking breaks data into smaller pieces since vector databases need data to be in smaller chunks for both storage and retrieval, while embedding converts textual data into fixed-size vectors preserving the semantic context. Consequently, document and tables chunks are created and then passed to the embedders for embedding creation.

The embedding process is necessary for enabling document question answering because, as mentioned before, it transforms textual data into a numerical format. Embeddings capture both semantic and syntactic relationships between words, allowing machines to process textual data effectively [8]. Embedding techniques allow question answering models to process documents with different lengths and also facilitate matching questions to relevant portions of the text, enabling them to provide meaningful answers.
Figure 3.5 presents the diagram illustrating the document parsing process and its subsequent steps.
Provided the `Document` and `DocumentTable` elements are inserted into the data store, the subsequent step involves the actual document question answering procedure, outlined in the following section, which precedes the evaluation of the extracted information.

3.4 Document Question Answering

The document question answering process is a central mechanism for retrieving precise information from specific documents uploaded by users.
The operational workflow of this process is implemented as a sequence of events, starting from a user query and ending with the delivery of the response.

1. **User Query**: The process starts with a user submitting a query, looking for information within a particular document.

2. **Document Analysis**: Upon receiving the user query, the system conducts a comprehensive analysis to comprehend the context and scope of the query. This involves implementing techniques such as natural language processing and document indexing to identify relevant documents.

3. **Information Retrieval**: Subsequent to document analysis, the system employs specific algorithms for information retrieval. This includes

methods like TF-IDF (Term Frequency-Inverse Document Frequency) and Neural Network-based approaches, to extract sections or passages that are relevant to the query from the identified documents.

4. **Answer Generation**: The system creates a concise and accurate response using NLP libraries and machine learning models. This process involves techniques such as summarization, paraphrasing, and models for answer validation.

5. **Response Delivery**: The response is provided to the user.

The outlined process of document question answering plays a critical role in efficiently retrieving precise information from specific documents. Its primary goal is to guarantee that users receive correct responses and enhance their overall knowledge retrieval and acquisition experience.

3.5 Evaluation

In the context of information extraction and retrieval, the efficacy of parsing and question answering systems is crucial.
The evaluation pertains to two components of the research: the document parsing process and the document question answering process. Through detailed analysis, the aim is to uncover strengths, weaknesses, and identify opportunities for improvements.

3.5.1 Evaluating Document Parsing

The evaluation of the parsing process is a complex task. The basis of any robust document analysis platform lies in its ability to accurately parse different document formats. The evaluation framework for the parsing process is multifaceted, considering not only the accuracy and efficiency of data extraction but also the scalability and adaptability of the process to handle documents of diverse complexities and formats.

One of the key metrics for assessing the efficiency of parsing approaches is **parsing time**, which directly impacts the overall throughput of data processing pipeline. Additionally, in the field of document parsing, the challenge of accurately extracting information from different document layouts and format is significant. The output from Form Recognizer is characterized by the **confidence score** which is crucial for the evaluation of the efficacy of the parsing process. Similarly, elements extracted through parsing with Unstructured are characterized by the **detection class probability** field within the metadata.

The confidence score from Form Recognizer plays a crucial role in determining how certain the model is about the output, thereby enabling users to evaluate the reliability of the extracted data. Meanwhile, the detection class probability assigned by Unstructured to each of the retrieved elements quantifies the likelihood that a detected element correctly belongs to a specific classification category assigned by the model.

These metrics, which will be analysed in a subsequent chapter, are useful to the evaluation of the performance and reliability of document parsing approaches and data processing methods. They provide a structured way to measure the success of parsing strategies, ensuring that the system can handle the complexities of document analysis with high accuracy.

To evaluate the effectiveness of these processes, the following section discusses the evaluation framework used for document question answering tasks.

3.5.2 Evaluating Document Question Answering

The process of evaluating the responses generated during the question answering process involves the use of a set of queries for each document that is under analysis. These queries are provided by domain experts, ensuring their relevance and insightfulness.

Each query is also paired with a corresponding target answer, which forms the basis of the Evaluation Framework, built on the comparison between target answers and generated responses.

Following the collection of question-target answer pairs, the next phase of the process involves obtaining answers based on the data collected from the previously described parsing procedure. Upon completion of this phase, the comparison is performed by computing different quality measures. This comparison serves multiple purposes, such as assessing the accuracy of the generated responses and providing valuable insights into the system's strengths and weaknesses. The aim is to ensure that the system's outputs are not only accurate but also align with the expectations of the experts.

The evaluation process is not solely performed by computing quality measures to perform the comparison, but also by conducting human evaluation. In this case, experts are engaged to review the responses generated by the system, bringing their expertise and understanding to evaluate them.

By providing a description of the experimental setup, the next chapter describes the evaluation processes to gain meaningful insights into the efficacy of the document parsing and document question answering processes, thereby enriching the understanding of their performance.

3.6 Summary

The chapter presents the methodologies adopted for document parsing and document question answering processes, emphasizing the role of parsing in extracting detailed and comprehensive data from documents to enhance the efficacy of question answering tasks.

The chapter details the high-level process for document question answering, exploiting the Retrieval-Augmented Generation model that combines retrieval-based and generative approaches to yield accurate answers. Further, it delves into the detailed approach for document and table parsing, describing how the implemented parsers transform raw documents into structured data.

The focus is also posed on rule-based custom models for parsing MBR and SOP documents, and on the post-parsing processes for efficient retrieval during the question answering tasks.
The chapter ends with an evaluation section that describes the methodology for determining the quality of both document parsing and question answering processes.

Chapter 4

Case Studies and Experimental Analysis

This chapter is dedicated to the experiments conducted to evaluate the performance of the question answering process over various documents, including *MBRs*, *SOPs*, and a selection from the *PubMedQA* dataset, all of which have undergone the parsing procedure prior to the question answering process.
The chapter presents an in-depth analysis of the methods used to compute the metrics during the conducted experiments and the subsequent results. The purpose is to assess the effectiveness and the accuracy of the generated answers, as well as the system's ability to extract and synthesize information from complex documents.
The results of the experiments are presented in the form of tables, which provide a clear and concise view of the performances.
In essence, the chapter is crucial in understanding the practical application and implication of the question answering process in real-world scenarios. It offers an overview of the process's strengths and potential areas for improvement. The interpretation and discussion of these results are addressed in the subsequent chapter.

4.1 Experimental Setup

The evaluation of the document question answering system is performed using three different sets of documents: documents selected from the PubMedQA dataset, a MBR document, and SOP documents.

4.1.1 PubMedQA Dataset

PubMedQA is a publicly available dataset specifically designed for biomedical, text-based question answering [7]. It is built on top of *PubMed* abstracts, a comprehensive resource of life science literature, and is thus a reliable and relevant dataset for the evaluation of the document question answering tool. The dataset is unique in its structure and composition. It consists of biomedical research articles, each linked to a question-answer pair, in addition to other information such as the context, namely the section of the document where the answer can be found, labels and meshes, the final decision, namely the *'Yes-No'* answer to the query, and the long answer. Figure 4.1 shows an example of one entry of the PubMedQA dataset.

The questions are designed to reflect the information-seeking behaviour of biomedical researches, and the answers are manually generated by human annotators based on the content of the corresponding document.

For the purpose of this evaluation, a selection of document is made from the PubMedQA dataset; along with the documents, corresponding *'Question-Long Answer'* pairs are also selected, creating the baseline for the evaluation framework.

However, as the PubMedQA dataset provides only one question-answer pair for each article, additional question-answer pairs are added to each document to broaden the scope of the evaluation. These questions were related to both the textual content and the tables present within the documents. The target answers for these additional questions are determined by experts, ensuring their relevance.

```
    "21645374": {
        "QUESTION": "Do mitochondria play a role in remodelling lace plant leaves during programmed cell death?",
        "CONTEXTS": [
            "Programmed cell death (PCD) is the regulated death of cells within an organism. The lace plant (Aponogeton
madagascariensis) produces perforations in its leaves through PCD. The leaves of the plant consist of a latticework of
longitudinal and transverse veins enclosing areoles. PCD occurs in the cells at the center of these areoles and progresses
outwards, stopping approximately five cells from the vasculature. The role of mitochondria during PCD has been recognized
in animals; however, it has been less studied during PCD in plants.",
            "The following paper elucidates the role of mitochondrial dynamics during developmentally regulated PCD in vivo
in A. madagascariensis. A single areole within a window stage leaf (PCD is occurring) was divided into three areas based on
the progression of PCD; cells that will not undergo PCD (NPCD), cells in early stages of PCD (EPCD), and cells in late
stages of PCD (LPCD). Window stage leaves were stained with the mitochondrial dye MitoTracker Red CMXRos and examined.
Mitochondrial dynamics were delineated into four categories (M1-M4) based on characteristics including distribution,
motility, and membrane potential (\u0394\u03a8m). A TUNEL assay showed fragmented nDNA in a gradient over these
mitochondrial stages. Chloroplasts and transvacuolar strands were also examined using live cell imaging. The possible
importance of mitochondrial permeability transition pore (PTP) formation during PCD was indirectly examined via in vivo
cyclosporine A (CsA) treatment. This treatment resulted in lace plant leaves with a significantly lower number of
perforations compared to controls, and that displayed mitochondrial dynamics similar to that of non-PCD cells."
        ],
        "LABELS": [
            "BACKGROUND",
            "RESULTS"
        ],
        "MESHES": [
            "Alismataceae",
            "Apoptosis",
            "Cell Differentiation",
            "Mitochondria",
            "Plant Leaves"
        ],
        "YEAR": "2011",
        "reasoning_required_pred": "yes",
        "reasoning_free_pred": "yes",
        "final_decision": "yes",
        "LONG_ANSWER": "Results depicted mitochondrial dynamics in vivo as PCD progresses within the lace plant, and
highlight the correlation of this organelle with other organelles during developmental PCD. To the best of our knowledge,
this is the first report of mitochondria and chloroplasts moving on transvacuolar strands to form a ring structure
surrounding the nucleus during developmental PCD. Also, for the first time, we have shown the feasibility for the use of
CsA in a whole plant system. Overall, our findings implicate the mitochondria as playing a critical and early role in
developmentally regulated PCD in the lace plant."
    },
```


Figure 4.1: Example of One Entry of the PubMedQA Dataset

After establishing the datasets used for evaluation, the next step is to define the quality measures and metrics applied to assess the performance of the question answering system.

4.2 Quality Measures

The evaluation for PubMedQA and MBR documents is conducted using different metrics: ROUGE score, BLEU score, and Cosine Similarity. Cosine Similarity is computed using various representations, namely Bag of Word representation, TF-IDF representation, Sentence Transformer representation, and Universal Sentence Encoder representation.

4.2.1 ROUGE Score

The ROUGE score is computed by using the `RougeScorer`, a tool that computes different ROUGE scores, each capturing different aspects of the compared texts.

When using:

```
RougeScorer(
['rouge1', 'rouge2', 'rougeL'],
use_stemmer=True)
```

it implies a request for three specific types of ROUGE evaluations:

- **ROUGE-1**: Measures the overlap of unigrams (single words) between the system-generated summary and the reference summary. It is a measure of the presence of individual words, without considering their order or context.
- **ROUGE-2**: Measures the overlap of bigrams (pairs of adjacent words) between the system-generated summary and the reference summary. It is a more stringent metric than the previous one as it considers the order of the words and can capture more information about the coherence of the summary.
- **ROUGE-L**: Measures the longest common subsequence between the system-generated summary and the reference summary. It assesses the longest string of words that appears in both the system-generated summary and the reference summary in the same order. ROUGE-L can naturally capture the sentence-level structure similarity naturally and does not require predefined n-grams.

Using a `stemmer` means that the words are reduced to their base root form before the overlap is computed. This can help in matching words that have the same root but are in different forms (for example, *"running"* and *"ran"* both stem to *"run"*).

Each ROUGE score typically provides three values, namely recall and precision, and the F1-score which is the harmonic mean of precision and recall:

- *Recall*: Measures the proportion of words (or bigrams, subsequences, etc.) from the reference summary that appear in the system-generated summary. A higher recall indicates that more of the reference content is captured by the system-generated summary.

- *Precision*: Measures the proportion of words (or bigrams, subsequences, etc.) from the summary generated by the system that occur in the reference summary. A higher precision means that the content of the system-generated summary is more likely to be relevant to the reference content.

- *F1-score*: Balances precision and recall, providing a single score that weights both equally. The F1-score is particularly useful when comparing systems that might have different trade-offs between precision and recall.

4.2.2 BLEU Score

The computation:

```
bleu_score =
sacrebleu.corpus_bleu([system_output],
                      [references])
```

is an invocation of the `corpus_bleu` function, taken from the `sacrebleu` library, which is used to compute the BLEU score for machine translation quality.

The parameter of the function are the following:

- **System Output**: It is a list containing the translations produced by the machine translation system. Each element in the list is a string representing a translated document or sentence. In this case, it contains the generated answer.

- **References**: This is a list containing the reference translations, which are the human-produced translations that are considered the gold standard. Each element in the list is another list of strings, where each string is a possible reference translation for the corresponding system output. There can be multiple translations for a single piece of system output to account for the variability in acceptable translations. In this case, it is the target answer.

The output of the function is a floating point number ranging from 0 to 100. A higher BLEU score indicates a higher similarity between the system output and the reference text. So, the closer the BLEU score is to 100, the more the generated responses resemble the target ones, indicating better quality.

4.2.3 Bag of Words and TF-IDF Representations

Bag of Words and TF-IDF representations are also used for performing the evaluation of the generated responses. The implemented procedure involves computing the similarity between two text strings, the target answer and the generated answer, by representing the texts in numerical form and then calculating the Cosine Similarity between the numerical representations. As just mentioned, the procedure uses two different methods for text representation, Bag of Words and Term Frequency-Inverse Document Frequency, and then it assesses the similarity using Cosine Similarity.

- **Bag of Words representation by using the CountVectorizer**: The `CountVectorizer` is used to convert the text data into a matrix of token counts, which constitutes the Bag of Word representation. In this representation, each unique word in the text corresponds to a feature (dimension) in the vector, and the value in each dimension is the count of the number of times that word appears in the text. Bag of Words models the presence and frequency of words but ignores the order of words.

- **TF-IDF**: The `TFIDFVectorizer` converts the text into a TF-IDF representation. It is similar to Bag of Words but also accounts for each word's importance in the text corpus. It assigns higher weights to words that are frequent in a document but not across documents.

In both cases, Cosine Similarity is computed, yielding a score ranging from -1 to 1. A score of 1 indicates that the texts are identical, 0 indicates no similarity, and -1 would imply complete dissimilarity. However, with non-negative representations like Bag of Words and TF-IDF, the score will always fall between 0 and 1, reflecting the degree of similarity between the texts.

4.2.4 Sentence Transformer

The aim of the procedure is to compute the semantic similarity between two sentences using a pre-trained transformer model. This is achieved by encoding the sentences into high-dimensional vectors (embeddings) that capture their semantic meaning and then calculating the Cosine Similarity between these embeddings. In this case the `SentenceTransformer` class from the library `sentence_transformers` is used. This library provides an easy-to-use interface for working with sentence embeddings. The `SentenceTransformer` model is initialized using a pre-existing, pre-trained model checkpoint.
Specifically, the involved models are the `bert-base-nli-mean-tokens` and the `xlm-r-distilroberta-base-paraphrase-v1`.

- The `bert-base-nli-mean-tokens` model is an extension of the original BERT model specifically optimized for sentence-level embeddings. It is trained on a large corpus of English data with a focus on natural language inference data and is designed to produce sentence embeddings that can be compared using Cosine Similarity. This model has been fine-tuned in order to produce meaningful sentence representations. The strength of the model lies in its ability to understand and encode the nuances of the English language, making it useful for tasks involving semantic similarity or paraphrase identification.

- The `xlm-r-distilroberta-base-paraphrase-v1` model is derived from the XLM-RoBERTa model, which is a scaled cross-lingual sentence encoder. The model is fine-tuned on paraphrase identification tasks across multiple languages. It leverages knowledge distilled from multilingual training, which allows it to capture semantic meanings across different languages more effectively than monolingual models.

Based on the insights detailed above, the first model is employed for analyzing and comparing responses in English. On the other hand, the second model is utilized for comparing the generated and target answers extracted from the MBR document and some documents selected from the PubMedQA dataset, which are written in German.
The sentences to be compared, namely the target answer and the generated answer, are then converted into a fixed-size embedding vector. The embeddings are created to encode semantically similar sentences into similar vectors. Finally, the Cosine Similarity function is computed with the embeddings of the two sentences, yielding the similarity score.

4.2.5 Universal Sentence Encoder

In this case, the procedure is similar to the previously described one in that it aims to compute the semantic similarity between two sentences, the target answer and the generated answer. However, the key difference lies in the model used to obtain the sentence embeddings. In this procedure, the *Universal Sentence Encoder* from TensorFlow Hub is used instead of the Sentence Transformer library. This model is pre-trained and can convert sentences into high-dimensional embeddings. It is designed to handle a wide range of languages, including German, and is trained to generate meaningful sentence embeddings for various types of input text. As before, the Cosine Similarity between the two sentence embeddings is then computed.
The difference between this approach and the previous one using the Sentence-Transformer library is primarily the model used to generate the embeddings:

- **Universal Sentence Encoder**: This is a model developed by Google that is trained on a variety of data sources and tasks. It is designed to produce embeddings that are useful across a wide range of tasks and languages. These embeddings are generally good at capturing semantic meaning and can be used for tasks like semantic similarity, clustering, and classification.

- **Sentence Transformer**: This uses models specifically fine-tuned for producing embeddings that can be compared with Cosine Similarity. These models are often based on BERT or similar transformer architectures and are fine-tuned on natural language inference data to produce embeddings that reflect sentence meaning.

While these metrics provide a quantitative analysis of the generated responses, they are complemented by human evaluation to ensure a comprehensive evaluation.

4.3 Human Evaluation

Human evaluation plays an important role in the evaluation process, as it is conducted by experts who are enlisted to rate the quality of the generated answers. This kind of evaluation is performed to assess the quality of generated responses to queries related to SOP documents. The aim of this evaluation lies in its ability to provide insights into the effectiveness and relevance of the generated answers, ensuring they meet the standards expected in real-world scenarios.

The results are categorized into distinct grades, each assigned a specific letter and color, creating an easily interpretable framework for assessing the levels of accuracy and detail observed in the responses to the queries. The grades take into account various aspects of the responses, including their precision in addressing the query, the depth of detail provided, and their alignment with the informational needs specified in the SOP documents.

Grades can be described as follows:

- **A (Correct and Detailed)**: This category represents responses that are not only accurate in addressing the query but also include details that enrich the understanding of the SOP content. It indicates that the information extracted is well aligned with the query's intent and offers additional context where necessary.
- **B (Correct but Not Detailed)**: Responses under this classification accurately meet the query's requirements but do not include many details. While the answer to the question is correct, it lacks details that might further clarify the SOP content or provide a deeper understanding.
- **C (Partially Correct)**: This designation is given to responses that only partially address the query. They may contain some elements of accuracy but are incomplete.
- **D (Incorrect or Misinterpreted)**: Responses classified as 'D' are those that misinterpret the query or the SOP content, generating incorrect or irrelevant answers. This implies a discrepancy between the information provided and what is asked, suggesting a misunderstanding of the query or the document's details.
- **NA (No Answer)**: The 'NA' category is used for instances where the query did not yield any response from the SOP document. This could be due to the absence of relevant information within the document or limitations in the parsing process that prevented the extraction of potential answers.

Each of these categories plays an important role in evaluating the effectiveness of the parsing process and the clarity of the SOP documents. By analyzing the distribution of responses across these categories, it is possible to assess the comprehensiveness and accessibility of the information contained within the

SOPs, as well as identify areas for improvement in both document preparation and query formulation.

4.4 Document Question Answering Results

The presented results stem from a comprehensive analysis of the document question answering process applied to PubMedQA, MBR, and SOP documents. For each document, two distinct evaluations are conducted: one incorporating table parsing procedures within the whole parsing process, and another that excludes them. The aim of this analysis is to discern the potential enhancement in query results offered by table parsing, in addition to the baseline parsing procedure. Parsing is conducted using both Form Recognizer and Unstructured parsers for PDFs. However, for SOP documents, which are in Word document format, only the Unstructured parser is used. The efficacy of parsing and the subsequent question answering results related to PubMedQA and MBR documents are assessed using the aforementioned metrics: ROUGE score, BLEU score, Bag of Words and TF-IDF representations, Sentence Transformer, and Universal Sentence Encoder. For the assessment of queries over SOP documents, human evaluation is conducted, wherein experts are enlisted to rate the quality of the generated answers according to the previously described grades.

4.4.1 Results - PubMedQA Documents

Tables 4.1, 4.2, 4.3, and 4.4 present the results derived from parsing five different files selected from the PubMedQA dataset. The first query used in this analysis is extracted directly from the dataset itself, serving as the baseline of the examination. Subsequent queries are determined by experts. These additional queries pertain to both the textual content and the tables within the documents. This approach ensures a deeper evaluation of the parsing procedure to determine the effectiveness of information extraction from the knowledge embedded in both the text and tables of the selected documents.

4.4.2 Results - MBR Document

Tables 4.5, 4.6, 4.7, and 4.8 present the results obtained from the document question answering process performed after parsing a MBR document. The queries, selected by experts, pertain to both the textual content and the tables within the document.

4.4.3 Results - SOP Documents

Table 4.9 presents the results derived from document question answering applied to seven different SOP documents. The queries, selected by experts, pertain to both the textual content and the tables within the documents. These results are categorized according to a legend, shown in Figure 4.2, which is crucial for interpreting the data accurately.

4.5 Summary

The chapter focuses on the evaluation of document question answering process applied to different types of documents, all of which undergo a parsing procedure to prepare the data for analysis. The goal is to assess the accuracy and effectiveness of the generated answers to highlight the strengths of the question answering process and identifying potential areas for improvement. The experimental setup involves the evaluation of question answering system, which includes computing quality measures and performing human assessments. Detailed results demonstrate its performance and offer insights into the document question answering process, deepening the understanding of the system's capabilities and setting the stage for further discussion in the subsequent chapter.

PARSING WITH FORM RECOGNIZER with TABLE PARSING								
	ROUGE SCORE			BLEU SCORE	BoW	TF-IDF	SENTENCE TRANSFORMER	UNIVERSAL SENTENCE ENCODER
	Rouge1	Rouge2	RougeL					
File 1 - 10808977								
QUERY 1	precision=0.351 recall=0.243 fmeasure=0.287	precision=0.027 recall=0.019 fmeasure=0.022	precision=0.189 recall=0.131 fmeasure=0.155	1.35	0.39	0.26	0.78	0.61
QUERY 2	precision=0.192 recall=0.615 fmeasure=0.292	precision=0.102 recall=0.331 fmeasure=0.157	precision=0.126 recall=0.404 fmeasure=0.192	8.41	0.45	0.33	0.85	0.57
QUERY 3	precision=0.692 recall=0.661 fmeasure=0.679	precision=0.361 recall=0.346 fmeasure=0.353	precision=0.521 recall=0.481 fmeasure=0.491	21.72	0.67	0.52	0.86	0.74
QUERY 4	precision=0.518 recall=0.423 fmeasure=0.5	precision=0.207 recall=0.193 fmeasure=0.2	precision=0.463 recall=0.431 fmeasure=0.446	14.81	0.68	0.56	0.82	0.67
QUERY 5	precision=0.476 recall=0.385 fmeasure=0.425	precision=0.097 recall=0.078 fmeasure=0.087	precision=0.309 recall=0.251 fmeasure=0.276	3.04	0.68	0.53	0.81	0.48
File 2 - 11729377								
QUERY 1	precision=0.256 recall=0.428 fmeasure=0.321	precision=0.052 recall=0.087 fmeasure=0.065	precision=0.162 recall=0.271 fmeasure=0.203	2.27	0.55	0.41	0.81	0.54
QUERY 2	precision=0.204 recall=0.638 fmeasure=0.307	precision=0.083 recall=0.263 fmeasure=0.126	precision=0.166 recall=0.517 fmeasure=0.251	6.83	0.65	0.53	0.81	0.72
QUERY 3	precision=0.427 recall=0.616 fmeasure=0.504	precision=0.156 recall=0.226 fmeasure=0.185	precision=0.166 recall=0.239 fmeasure=0.196	13.26	0.78	0.68	0.69	0.71
QUERY 4	precision=0.368 recall=0.588 fmeasure=0.452	precision=0.163 recall=0.262 fmeasure=0.201	precision=0.243 recall=0.388 fmeasure=0.299	10.47	0.76	0.66	0.81	0.71
QUERY 5	precision=0.877 recall=0.428 fmeasure=0.576	precision=0.593 recall=0.288 fmeasure=0.388	precision=0.708 recall=0.346 fmeasure=0.465	17.67	0.77	0.67	0.81	0.78
File 3 - 16418930								
QUERY 1	precision=0.142 recall=0.737 fmeasure=0.235	precision=0.075 recall=0.405 fmeasure=0.127	precision=0.115 recall=0.605 fmeasure=0.193	4.51	0.69	0.56	0.79	0.76
QUERY 2	precision=0.221 recall=0.773 fmeasure=0.318	precision=0.112 recall=0.442 fmeasure=0.179	precision=0.159 recall=0.614 fmeasure=0.252	9.51	0.69	0.58	0.71	0.73
QUERY 3	precision=0.647 recall=0.731 fmeasure=0.687	precision=0.576 recall=0.655 fmeasure=0.613	precision=0.588 recall=0.667 fmeasure=0.625	46.73	0.75	0.65	0.78	0.63
QUERY 4	precision=0.917 recall=0.423 fmeasure=0.579	precision=0.727 recall=0.322 fmeasure=0.445	precision=0.917 recall=0.423 fmeasure=0.579	16.94	0.72	0.61	0.78	0.65
QUERY 5	precision=0.526 recall=0.625 fmeasure=0.571	precision=0.299 recall=0.356 fmeasure=0.325	precision=0.284 recall=0.337 fmeasure=0.308	22.96	0.81	0.74	0.59	0.81
File 4 - 17208539								
QUERY 1	precision=0.115 recall=0.446 fmeasure=0.183	precision=0.032 recall=0.125 fmeasure=0.051	precision=0.083 recall=0.323 fmeasure=0.132	1.56	0.64	0.51	0.71	0.42
QUERY 2	precision=0.601 recall=0.676 fmeasure=0.636	precision=0.327 recall=0.369 fmeasure=0.347	precision=0.344 recall=0.387 fmeasure=0.364	26.01	0.89	0.82	0.93	0.79
QUERY 3	precision=0.598 recall=0.424 fmeasure=0.496	precision=0.268 recall=0.189 fmeasure=0.222	precision=0.307 recall=0.218 fmeasure=0.255	13.61	0.76	0.65	0.78	0.65
QUERY 4	precision=0.421 recall=0.491 fmeasure=0.453	precision=0.206 recall=0.241 fmeasure=0.222	precision=0.333 recall=0.391 fmeasure=0.359	10.73	0.47	0.32	0.84	0.55
QUERY 5	precision=0.631 recall=0.928 fmeasure=0.751	precision=0.438 recall=0.646 fmeasure=0.522	precision=0.591 recall=0.867 fmeasure=0.702	39.42	0.81	0.72	0.97	0.78
File 5 - 26037986								
QUERY 1	precision=0.321 recall=0.568 fmeasure=0.411	precision=0.064 recall=0.115 fmeasure=0.083	precision=0.173 recall=0.307 fmeasure=0.221	5.25	0.59	0.47	0.84	0.72
QUERY 2	precision=0.751 recall=0.714 fmeasure=0.732	precision=0.576 recall=0.548 fmeasure=0.562	precision=0.683 recall=0.651 fmeasure=0.667	51.62	0.85	0.78	0.95	0.78
QUERY 3	precision=0.601 recall=0.911 fmeasure=0.724	precision=0.545 recall=0.829 fmeasure=0.658	precision=0.596 recall=0.903 fmeasure=0.718	53.36	0.92	0.89	0.95	0.94
QUERY 4	precision=0.555 recall=0.555 fmeasure=0.555	precision=0.254 recall=0.254 fmeasure=0.254	precision=0.433 recall=0.433 fmeasure=0.433	20.56	0.78	0.69	0.88	0.66
QUERY 5	precision=0.301 recall=0.568 fmeasure=0.393	precision=0.111 recall=0.191 fmeasure=0.132	precision=0.189 recall=0.358 fmeasure=0.247	10.78	0.55	0.45	0.88	0.71

Table 4.1: **PubMedQA** DocQA Results - *Form Recognizer with Table Parsing*

PARSING WITH FORM RECOGNIZER without TABLE PARSING								
	ROUGE SCORE			BLEU SCORE	BoW	TF-IDF	SENTENCE TRANSFORMER	UNIVERSAL SENTENCE ENCODER
	Rouge1	Rouge2	RougeL					
File 1 - 10808977								
QUERY 1	precision=0.406 recall=0.243 fmeasure=0.304	precision=0.063 recall=0.038 fmeasure=0.047	precision=0.219 recall=0.131 fmeasure=0.164	1.15	0.47	0.33	0.71	0.61
QUERY 2	precision=0.263 recall=0.481 fmeasure=0.341	precision=0.159 recall=0.294 fmeasure=0.207	precision=0.168 recall=0.308 fmeasure=0.218	10.22	0.43	0.31	0.81	0.47
QUERY 3	precision=0.293 recall=0.441 fmeasure=0.353	precision=0.201 recall=0.308 fmeasure=0.242	precision=0.268 recall=0.407 fmeasure=0.323	17.66	0.42	0.28	0.45	0.34
QUERY 4	precision=0.401 recall=0.345 fmeasure=0.371	precision=0.143 recall=0.123 fmeasure=0.132	precision=0.281 recall=0.241 fmeasure=0.259	12.78	0.45	0.31	0.51	0.37
QUERY 5	precision=0.331 recall=0.325 fmeasure=0.329	precision=0.079 recall=0.077 fmeasure=0.078	precision=0.179 recall=0.175 fmeasure=0.177	2.87	0.52	0.38	0.27	0.25
File 2 - 11729377								
QUERY 1	precision=0.231 recall=0.301 fmeasure=0.261	precision=0.033 recall=0.043 fmeasure=0.038	precision=0.132 recall=0.171 fmeasure=0.149	2.53	0.33	0.22	0.66	0.53
QUERY 2	precision=0.143 recall=0.569 fmeasure=0.228	precision=0.056 recall=0.228 fmeasure=0.09	precision=0.108 recall=0.431 fmeasure=0.173	4.85	0.61	0.51	0.75	0.56
QUERY 3	precision=0.549 recall=0.609 fmeasure=0.577	precision=0.231 recall=0.255 fmeasure=0.242	precision=0.242 recall=0.268 fmeasure=0.254	17.22	0.83	0.73	0.63	0.76
QUERY 4	precision=0.343 recall=0.565 fmeasure=0.427	precision=0.101 recall=0.167 fmeasure=0.125	precision=0.214 recall=0.353 fmeasure=0.267	4.14	0.74	0.62	0.81	0.62
QUERY 5	precision=0.464 recall=0.195 fmeasure=0.275	precision=0.145 recall=0.061 fmeasure=0.085	precision=0.339 recall=0.143 fmeasure=0.201	2.97	0.55	0.41	0.46	0.23
File 3 - 16418930								
QUERY 1	precision=0.238 recall=0.816 fmeasure=0.369	precision=0.139 recall=0.486 fmeasure=0.217	precision=0.151 recall=0.526 fmeasure=0.238	8.94	0.75	0.64	0.87	0.71
QUERY 2	precision=0.428 recall=0.682 fmeasure=0.526	precision=0.246 recall=0.395 fmeasure=0.303	precision=0.301 recall=0.477 fmeasure=0.368	21.11	0.67	0.54	0.86	0.71
QUERY 3	precision=0.353 recall=0.461 fmeasure=0.401	precision=0.08 recall=0.105 fmeasure=0.091	precision=0.196 recall=0.256 fmeasure=0.222	4.15	0.51	0.35	0.39	0.44
QUERY 4	precision=0.475 recall=0.731 fmeasure=0.576	precision=0.256 recall=0.401 fmeasure=0.312	precision=0.35 recall=0.538 fmeasure=0.424	17.66	0.71	0.55	0.37	0.62
QUERY 5	precision=0.662 recall=0.271 fmeasure=0.383	precision=0.362 recall=0.147 fmeasure=0.209	precision=0.424 recall=0.173 fmeasure=0.246	4.32	0.77	0.69	0.62	0.69
File 4 - 17208539								
QUERY 1	precision=0.226 recall=0.369 fmeasure=0.281	precision=0.076 recall=0.125 fmeasure=0.095	precision=0.151 recall=0.246 fmeasure=0.187	2.21	0.61	0.44	0.71	0.51
QUERY 2	precision=0.687 recall=0.621 fmeasure=0.652	precision=0.378 recall=0.341 fmeasure=0.358	precision=0.391 recall=0.352 fmeasure=0.37	25.01	0.89	0.826	0.93	0.84
QUERY 3	precision=0.551 recall=0.504 fmeasure=0.529	precision=0.271 recall=0.245 fmeasure=0.257	precision=0.282 recall=0.256 fmeasure=0.269	19.57	0.79	0.61	0.75	0.67
QUERY 4	precision=0.364 recall=0.135 fmeasure=0.197	precision=0.191 recall=0.069 fmeasure=0.101	precision=0.273 recall=0.102 fmeasure=0.148	4.71	0.27	0.17	0.36	0.29
QUERY 5	precision=0.188 recall=0.157 fmeasure=0.171	precision=0.044 recall=0.036 fmeasure=0.041	precision=0.145 recall=0.121 fmeasure=0.131	1.86	0.24	0.14	0.31	0.21
File 5 - 26037986								
QUERY 1	precision=0.243 recall=0.216 fmeasure=0.229	precision=0.0 recall=0.0 fmeasure=0.0	precision=0.141 recall=0.125 fmeasure=0.132	0.81	0.34	0.23	0.66	0.35
QUERY 2	precision=0.805 recall=0.524 fmeasure=0.635	precision=0.451 recall=0.291 fmeasure=0.353	precision=0.634 recall=0.413 fmeasure=0.501	22.43	0.72	0.59	0.98	0.88
QUERY 3	precision=0.294 recall=0.121 fmeasure=0.171	precision=0.216 recall=0.089 fmeasure=0.126	precision=0.216 recall=0.089 fmeasure=0.126	1.07	0.44	0.31	0.52	0.27
QUERY 4	precision=0.281 recall=0.301 fmeasure=0.291	precision=0.048 recall=0.051 fmeasure=0.049	precision=0.187 recall=0.201 fmeasure=0.193	2.34	0.57	0.41	0.34	0.28
QUERY 5	precision=0.389 recall=0.221 fmeasure=0.282	precision=0.132 recall=0.074 fmeasure=0.095	precision=0.296 recall=0.168 fmeasure=0.21	2.02	0.49	0.34	0.36	0.28

Table 4.2: **PubMedQA** DocQA Results - *Form Recognizer without Table Parsing*

PARSING WITH UNSTRUCTURED with TABLE PARSING								
	ROUGE SCORE			BLEU SCORE	BoW	TF-IDF	SENTENCE TRANSFORMER	UNIVERSAL SENTENCE ENCODER
	Rouge1	Rouge2	RougeL					
File 1 - 10808977								
QUERY 1	precision=0.493 recall=0.336 fmeasure=0.399	precision=0.139 recall=0.094 fmeasure=0.112	precision=0.233 recall=0.159 fmeasure=0.189	6.46	0.58	0.43	0.79	0.72
QUERY 2	precision=0.244 recall=0.615 fmeasure=0.351	precision=0.085 recall=0.216 fmeasure=0.121	precision=0.137 recall=0.346 fmeasure=0.197	5.26	0.39	0.29	0.78	0.56
QUERY 3	precision=0.447 recall=0.778 fmeasure=0.567	precision=0.283 recall=0.501 fmeasure=0.361	precision=0.404 recall=0.704 fmeasure=0.513	18.71	0.67	0.53	0.79	0.74
QUERY 4	precision=0.311 recall=0.465 fmeasure=0.372	precision=0.116 recall=0.175 fmeasure=0.141	precision=0.23 recall=0.345 fmeasure=0.276	6.99	0.545	0.39	0.86	0.48
QUERY 5	precision=0.511 recall=0.601 fmeasure=0.552	precision=0.196 recall=0.231 fmeasure=0.212	precision=0.341 recall=0.401 fmeasure=0.368	7.81	0.64	0.55	0.87	0.61
File 2 - 11729377								
QUERY 1	precision=0.278 recall=0.786 fmeasure=0.41	precision=0.127 recall=0.362 fmeasure=0.188	precision=0.146 recall=0.414 fmeasure=0.216	11.84	0.79	0.69	0.72	0.81
QUERY 2	precision=0.177 recall=0.724 fmeasure=0.284	precision=0.089 recall=0.368 fmeasure=0.143	precision=0.143 recall=0.586 fmeasure=0.231	6.61	0.66	0.56	0.72	0.74
QUERY 3	precision=0.631 recall=0.435 fmeasure=0.515	precision=0.223 recall=0.153 fmeasure=0.182	precision=0.337 recall=0.232 fmeasure=0.275	12.31	0.75	0.63	0.91	0.81
QUERY 4	precision=0.922 recall=0.976 fmeasure=0.948	precision=0.854 recall=0.905 fmeasure=0.879	precision=0.911 recall=0.965 fmeasure=0.937	71.02	0.98	0.97	0.98	0.97
QUERY 5	precision=0.404 recall=0.669 fmeasure=0.504	recision=0.251 recall=0.417 fmeasure=0.313	precision=0.291 recall=0.481 fmeasure=0.363	17.38	0.83	0.75	0.91	0.74
File 3 - 16418930								
QUERY 1	precision=0.197 recall=0.658 fmeasure=0.303	precision=0.071 recall=0.243 fmeasure=0.11	precision=0.118 recall=0.395 fmeasure=0.182	3.32	0.57	0.42	0.68	0.66
QUERY 2	precision=0.323 recall=0.704 fmeasure=0.443	precision=0.189 recall=0.419 fmeasure=0.261	precision=0.219 recall=0.477 fmeasure=0.301	13.44	0.64	0.53	0.65	0.71
QUERY 3	precision=0.615 recall=0.411 fmeasure=0.492	precision=0.281 recall=0.184 fmeasure=0.222	precision=0.423 recall=0.282 fmeasure=0.338	8.22	0.49	0.34	0.74	0.61
QUERY 4	precision=0.895 recall=0.654 fmeasure=0.755	precision=0.501 recall=0.361 fmeasure=0.419	precision=0.737 recall=0.538 fmeasure=0.622	12.41	0.85	0.75	0.81	0.73
QUERY 5	precision=0.694 recall=0.419 fmeasure=0.523	precision=0.302 recall=0.182 fmeasure=0.227	precision=0.495 recall=0.299 fmeasure=0.373	10.75	0.77	0.71	0.73	0.69
File 4 - 17208539								
QUERY 1	precision=0.261 recall=0.492 fmeasure=0.341	precision=0.091 recall=0.172 fmeasure=0.118	precision=0.146 recall=0.277 fmeasure=0.191	3.67	0.65	0.51	0.84	0.64
QUERY 2	precision=0.541 recall=0.239 fmeasure=0.332	precision=0.161 recall=0.071 fmeasure=0.098	precision=0.333 recall=0.148 fmeasure=0.205	1.35	0.62	0.51	0.84	0.58
QUERY 3	precision=0.443 recall=0.395 fmeasure=0.418	precision=0.114 recall=0.101 fmeasure=0.107	precision=0.226 recall=0.202 fmeasure=0.213	5.29	0.64	0.54	0.88	0.55
QUERY 4	precision=0.407 recall=0.559 fmeasure=0.471	precision=0.225 recall=0.311 fmeasure=0.261	precision=0.346 recall=0.474 fmeasure=0.401	13.08	0.51	0.36	0.85	0.57
QUERY 5	precision=0.592 recall=0.928 fmeasure=0.723	precision=0.419 recall=0.658 fmeasure=0.512	precision=0.554 recall=0.867 fmeasure=0.676	37.47	0.81	0.72	0.97	0.76
File 5 - 26037986								
QUERY 1	precision=0.202 recall=0.534 fmeasure=0.294	precision=0.043 recall=0.115 fmeasure=0.063	precision=0.099 recall=0.261 fmeasure=0.144	2.73	0.47	0.37	0.78	0.47
QUERY 2	precision=0.264 recall=0.365 fmeasure=0.307	precision=0.128 recall=0.177 fmeasure=0.149	precision=0.195 recall=0.271 fmeasure=0.227	11.18	0.38	0.25	0.75	0.47
QUERY 3	precision=0.864 recall=0.871 fmeasure=0.87	precision=0.774 recall=0.781 fmeasure=0.777	precision=0.864 recall=0.871 fmeasure=0.867	72.87	0.92	0.89	0.98	0.97
QUERY 4	precision=0.555 recall=0.555 fmeasure=0.555	precision=0.254 recall=0.254 fmeasure=0.254	precision=0.433 recall=0.433 fmeasure=0.433	20.56	0.78	0.69	0.87	0.66
QUERY 5	precision=0.301 recall=0.568 fmeasure=0.393	precision=0.101 recall=0.191 fmeasure=0.132	precision=0.189 recall=0.358 fmeasure=0.247	10.78	0.55	0.45	0.88	0.71

Table 4.3: **PubMedQA** DocQA Results - *Unstructured with Table Parsing*

PARSING WITH UNSTRUCTURED without TABLE PARSING								
	ROUGE SCORE			BLEU SCORE	BoW	TF-IDF	SENTENCE TRANSFORMER	UNIVERSAL SENTENCE ENCODER
	Rouge1	Rouge2	RougeL					
File 1 - 10808977								
QUERY 1	precision=0.474 recall=0.421 fmeasure=0.445	precision=0.138 recall=0.123 fmeasure=0.131	precision=0.211 recall=0.187 fmeasure=0.198	7.86	0.62	0.47	0.81	0.74
QUERY 2	precision=0.211 recall=0.615 fmeasure=0.314	precision=0.053 recall=0.157 fmeasure=0.079	precision=0.145 recall=0.423 fmeasure=0.216	2.01	0.44	0.32	0.83	0.54
QUERY 3	precision=0.205 recall=0.631 fmeasure=0.309	precision=0.146 recall=0.461 fmeasure=0.222	precision=0.193 recall=0.592 fmeasure=0.291	9.16	0.49	0.36	0.71	0.52
QUERY 4	precision=0.317 recall=0.448 fmeasure=0.371	precision=0.111 recall=0.158 fmeasure=0.131	precision=0.256 recall=0.362 fmeasure=0.301	6.41	0.55	0.41	0.89	0.56
QUERY 5	precision=0.525 recall=0.525 fmeasure=0.525	precision=0.231 recall=0.231 fmeasure=0.231	precision=0.375 recall=0.375 fmeasure=0.375	6.29	0.57	0.48	0.86	0.59
File 2 - 11729377								
QUERY 1	precision=0.321 recall=0.701 fmeasure=0.439	precision=0.099 recall=0.217 fmeasure=0.136	precision=0.144 recall=0.314 fmeasure=0.197	8.42	0.76	0.64	0.85	0.82
QUERY 2	precision=0.148 recall=0.691 fmeasure=0.245	precision=0.063 recall=0.298 fmeasure=0.105	precision=0.101 recall=0.465 fmeasure=0.165	3.85	0.69	0.56	0.78	0.66
QUERY 3	precision=0.376 recall=0.638 fmeasure=0.473	precision=0.124 recall=0.212 fmeasure=0.157	precision=0.175 recall=0.297 fmeasure=0.22	8.36	0.86	0.77	0.86	0.82
QUERY 4	precision=0.862 recall=0.953 fmeasure=0.905	precision=0.763 recall=0.845 fmeasure=0.802	precision=0.766 recall=0.847 fmeasure=0.804	62.71	0.96	0.93	0.99	0.96
QUERY 5	precision=0.387 recall=0.579 fmeasure=0.464	precision=0.151 recall=0.227 fmeasure=0.182	precision=0.206 recall=0.308 fmeasure=0.247	9.61	0.78	0.66	0.83	0.68
File 3 - 16418930								
QUERY 1	precision=0.188 recall=0.684 fmeasure=0.295	precision=0.073 recall=0.271 fmeasure=0.115	precision=0.116 recall=0.421 fmeasure=0.182	3.13	0.56	0.43	0.66	0.64
QUERY 2	precision=0.361 recall=0.181 fmeasure=0.241	precision=0.238 recall=0.116 fmeasure=0.156	precision=0.318 recall=0.159 fmeasure=0.212	6.46	0.34	0.21	0.21	0.51
QUERY 3	precision=0.75 recall=0.308 fmeasure=0.436	precision=0.2 recall=0.079 fmeasure=0.113	precision=0.562 recall=0.231 fmeasure=0.327	2.04	0.47	0.33	0.81	0.59
QUERY 4	precision=0.909 recall=0.385 fmeasure=0.541	precision=0.801 recall=0.321 fmeasure=0.457	precision=0.909 recall=0.385 fmeasure=0.541	16.79	0.74	0.62	0.55	0.56
QUERY 5	precision=0.531 recall=0.411 fmeasure=0.461	precision=0.195 recall=0.151 fmeasure=0.169	precision=0.308 recall=0.237 fmeasure=0.268	9.91	0.67	0.61	0.72	0.65
File 4 - 17208539								
QUERY 1	precision=0.534 recall=0.477 fmeasure=0.504	precision=0.193 recall=0.172 fmeasure=0.182	precision=0.276 recall=0.246 fmeasure=0.26	8.13	0.72	0.58	0.87	0.64
QUERY 2	precision=0.469 recall=0.429 fmeasure=0.448	precision=0.155 recall=0.142 fmeasure=0.148	precision=0.231 recall=0.211 fmeasure=0.221	5.71	0.81	0.71	0.82	0.65
QUERY 3	precision=0.675 recall=0.323 fmeasure=0.437	precision=0.265 recall=0.126 fmeasure=0.171	precision=0.412 recall=0.197 fmeasure=0.267	3.39	0.74	0.62	0.85	0.59
QUERY 4	precision=0.428 recall=0.661 fmeasure=0.521	precision=0.289 recall=0.448 fmeasure=0.351	precision=0.374 recall=0.576 fmeasure=0.453	11.87	0.52	0.38	0.85	0.61
QUERY 5	precision=0.504 recall=0.687 fmeasure=0.582	precision=0.241 recall=0.329 fmeasure=0.278	precision=0.336 recall=0.458 fmeasure=0.388	24.58	0.68	0.61	0.94	0.75
File 5 - 26037986								
QUERY 1	precision=0.162 recall=0.466 fmeasure=0.241	precision=0.028 recall=0.081 fmeasure=0.041	precision=0.087 recall=0.251 fmeasure=0.129	1.21	0.45	0.35	0.78	0.45
QUERY 2	precision=0.209 recall=0.587 fmeasure=0.308	precision=0.096 recall=0.274 fmeasure=0.143	precision=0.164 recall=0.461 fmeasure=0.242	7.56	0.45	0.31	0.75	0.43
QUERY 3	precision=0.601 recall=0.516 fmeasure=0.428	precision=0.109 recall=0.154 fmeasure=0.128	precision=0.217 recall=0.306 fmeasure=0.254	8.06	0.64	0.52	0.85	0.61
QUERY 4	precision=0.333 recall=0.351 fmeasure=0.341	precision=0.097 recall=0.102 fmeasure=0.099	precision=0.222 recall=0.233 fmeasure=0.228	2.88	0.61	0.47	0.37	0.31
QUERY 5	precision=0.228 recall=0.358 fmeasure=0.279	precision=0.027 recall=0.042 fmeasure=0.033	precision=0.101 recall=0.158 fmeasure=0.123	1.37	0.47	0.33	0.87	0.45

Table 4.4: **PubMedQA** DocQA Results - *Unstructured without Table Parsing*

PARSING WITH FORM RECOGNIZER with TABLE PARSING								
	ROUGE SCORE			BLEU SCORE	BoW	TF-IDF	SENTENCE TRANSFORMER	UNIVERSAL SENTENCE ENCODER
	Rouge1	Rouge2	RougeL					
QUERY 1	precision=0.174 recall=0.571 fmeasure=0.267	precision=0.091 recall=0.333 fmeasure=0.143	precision=0.174 recall=0.571 fmeasure=0.267	3.65	0.26	0.15	0.77	0.51
QUERY 2	precision=0.273 recall=0.751 fmeasure=0.401	precision=0.125 recall=0.364 fmeasure=0.186	precision=0.212 recall=0.583 fmeasure=0.311	3.32	0.31	0.22	0.72	0.43
QUERY 3	precision=0.313 recall=0.569 fmeasure=0.404	precision=0.137 recall=0.251 fmeasure=0.177	precision=0.246 recall=0.446 fmeasure=0.317	9.81	0.37	0.25	0.77	0.68
QUERY 4	precision=0.031 recall=0.111 fmeasure=0.048	precision=0.0 recall=0.0 fmeasure=0.0	precision=0.031 recall=0.111 fmeasure=0.048	0	0	0	0.27	0.27
QUERY 5	precision=0.151 recall=0.176 fmeasure=0.162	precision=0.026 recall=0.031 fmeasure=0.028	precision=0.075 recall=0.088 fmeasure=0.081	1.43	0.21	0.12	0.33	0.75
QUERY 6	precision=0.077 recall=0.038 fmeasure=0.051	precision=0.0 recall=0.0 fmeasure=0.0	precision=0.077 recall=0.038 fmeasure=0.051	0.41	0.08	0.04	0.43	0.48
QUERY 7	precision=0.219 recall=0.205 fmeasure=0.212	precision=0.032 recall=0.031 fmeasure=0.031	precision=0.187 recall=0.176 fmeasure=0.182	2.31	0.33	0.21	0.71	0.61
QUERY 8	precision=0.061 recall=0.083 fmeasure=0.071	precision=0.0 recall=0.0 fmeasure=0.0	precision=0.041 recall=0.055 fmeasure=0.047	1.16	0.05	0.02	0.24	0.11
QUERY 9	precision=0.117 recall=0.251 fmeasure=0.159	precision=0.026 recall=0.057 fmeasure=0.036	precision=0.065 recall=0.139 fmeasure=0.088	0.88	0.21	0.12	0.41	0.56

Table 4.5: **MBR** DocQA Results - *Form Recognizer with Table Parsing*

PARSING WITH FORM RECOGNIZER without TABLE PARSING								
	ROUGE SCORE			BLEU SCORE	BoW	TF-IDF	SENTENCE TRANSFORMER	UNIVERSAL SENTENCE ENCODER
	Rouge1	Rouge2	RougeL					
QUERY 1	precision=0.231 recall=0.428 fmeasure=0.301	precision=0.083 recall=0.167 fmeasure=0.111	precision=0.231 recall=0.428 fmeasure=0.301	3.67	0.25	0.14	0.61	0.51
QUERY 2	precision=0.062 recall=0.083 fmeasure=0.071	precision=0.0 recall=0.0 fmeasure=0.0	precision=0.062 recall=0.083 fmeasure=0.071	0	0.06	0.03	0.58	0.31
QUERY 3	precision=0.085 recall=0.138 fmeasure=0.105	precision=0.019 recall=0.031 fmeasure=0.024	precision=0.057 recall=0.092 fmeasure=0.071	0.71	0.14	0.08	0.51	0.52
QUERY 4	precision=0.034 recall=0.111 fmeasure=0.053	precision=0.0 recall=0.0 fmeasure=0.0	precision=0.034 recall=0.111 fmeasure=0.053	0	0	0	0.27	0.25
QUERY 5	precision=0.161 recall=0.147 fmeasure=0.154	precision=0.067 recall=0.061 fmeasure=0.063	precision=0.129 recall=0.117 fmeasure=0.123	3.78	0.21	0.12	0.31	0.77
QUERY 6	precision=0.116 recall=0.063 fmeasure=0.082	precision=0.0 recall=0.0 fmeasure=0.0	precision=0.071 recall=0.038 fmeasure=0.049	0.42	0.05	0.03	0.33	0.45
QUERY 7	precision=0.024 recall=0.029 fmeasure=0.027	precision=0.0 recall=0.0 fmeasure=0.0	precision=0.024 recall=0.029 fmeasure=0.027	1.31	0.06	0.03	0.36	0.62
QUERY 8	precision=0.038 recall=0.055 fmeasure=0.045	precision=0.0 recall=0.0 fmeasure=0.0	precision=0.038 recall=0.055 fmeasure=0.045	1.24	0	0	0.32	0.24
QUERY 9	precision=0.116 recall=0.222 fmeasure=0.152	precision=0.015 recall=0.028 fmeasure=0.019	precision=0.072 recall=0.139 fmeasure=0.095	0.92	0.18	0.09	0.45	0.57

Table 4.6: **MBR** DocQA Results - *Form Recognizer without Table Parsing*

PARSING WITH UNSTRUCTURED with TABLE PARSING								
	ROUGE SCORE			BLEU SCORE	BoW	TF-IDF	SENTENCE TRANSFORMER	UNIVERSAL SENTENCE ENCODER
	Rouge1	Rouge2	RougeL					
QUERY 1	precision=0.208 recall=0.714 fmeasure=0.332	precision=0.13 recall=0.501 fmeasure=0.207	precision=0.167 recall=0.571 fmeasure=0.258	4.26	0.41	0.27	0.79	0.53
QUERY 2	precision=0.345 recall=0.833 fmeasure=0.488	precision=0.286 recall=0.727 fmeasure=0.411	precision=0.345 recall=0.833 fmeasure=0.488	4.01	0.64	0.53	0.77	0.56
QUERY 3	precision=0.264 recall=0.631 fmeasure=0.373	precision=0.104 recall=0.251 fmeasure=0.147	precision=0.187 recall=0.446 fmeasure=0.263	5.64	0.37	0.25	0.68	0.66
QUERY 4	precision=0.043 recall=0.111 fmeasure=0.062	precision=0.0 recall=0.0 fmeasure=0.0	precision=0.043 recall=0.111 fmeasure=0.062	1.79	0.11	0.06	0.32	0.26
QUERY 5	precision=0.312 recall=0.441 fmeasure=0.366	precision=0.149 recall=0.212 fmeasure=0.175	precision=0.187 recall=0.265 fmeasure=0.219	4.81	0.43	0.31	0.71	0.72
QUERY 6	precision=0.132 recall=0.329 fmeasure=0.188	precision=0.025 recall=0.063 fmeasure=0.036	precision=0.066 recall=0.164 fmeasure=0.094	1.32	0.24	0.16	0.51	0.61
QUERY 7	precision=0.176 recall=0.353 fmeasure=0.235	precision=0.045 recall=0.091 fmeasure=0.061	precision=0.162 recall=0.323 fmeasure=0.216	3.43	0.31	0.19	0.77	0.58
QUERY 8	precision=0.015 recall=0.055 fmeasure=0.024	precision=0.0 recall=0.0 fmeasure=0.0	precision=0.015 recall=0.055 fmeasure=0.024	0.93	0	0	0.49	0.38
QUERY 9	precision=0.122 recall=0.257 fmeasure=0.165	precision=0.0 recall=0.0 fmeasure=0.0	precision=0.067 recall=0.143 fmeasure=0.092	0.94	0.09	0.05	0.42	0.54

Table 4.7: **MBR** DocQA Results - *Unstructured with Table Parsing*

PARSING WITH UNSTRUCTURED without TABLE PARSING								
	ROUGE SCORE			BLEU SCORE	BoW	TF-IDF	SENTENCE TRANSFORMER	UNIVERSAL SENTENCE ENCODER
	Rouge1	Rouge2	RougeL					
QUERY 1	precision=0.033 recall=0.286 fmeasure=0.061	precision=0.017 recall=0.167 fmeasure=0.031	precision=0.033 recall=0.256 fmeasure=0.061	0.58	0.11	0.05	0.51	0.36
QUERY 2	precision=0.015 recall=0.501 fmeasure=0.174	precision=0.053 recall=0.273 fmeasure=0.089	precision=0.105 recall=0.501 fmeasure=0.174	1.96	0.33	0.21	0.61	0.23
QUERY 3	precision=0.309 recall=0.385 fmeasure=0.342	precision=0.062 recall=0.069 fmeasure=0.211	precision=0.211 recall=0.261 fmeasure=0.233	2.98	0.34	0.23	0.65	0.58
QUERY 4	precision=0.0 recall=0.0 fmeasure=0.0	precision=0.0 recall=0.0 fmeasure=0.0	precision=0.0 recall=0.0 fmeasure=0.0	0	0	0	0.26	0.27
QUERY 5	precision=0.193 recall=0.352 fmeasure=0.251	precision=0.049 recall=0.091 fmeasure=0.064	precision=0.145 recall=0.265 fmeasure=0.187	2.33	0.37	0.25	0.71	0.77
QUERY 6	precision=0.139 recall=0.152 fmeasure=0.145	precision=0.035 recall=0.038 fmeasure=0.037	precision=0.081 recall=0.089 fmeasure=0.085	1.71	0.14	0.08	0.61	0.62
QUERY 7	precision=0.0 recall=0.0 fmeasure=0.0	precision=0.0 recall=0.0 fmeasure=0.0	precision=0.0 recall=0.0 fmeasure=0.0	1.09	0	0	0.28	0.58
QUERY 8	precision=0.024 recall=0.139 fmeasure=0.041	precision=0.0 recall=0.0 fmeasure=0.0	precision=0.015 recall=0.083 fmeasure=0.025	0.31	0.02	0.01	0.43	0.26
QUERY 9	precision=0.102 recall=0.222 fmeasure=0.141	precision=0.0 recall=0.0 fmeasure=0.0	precision=0.051 recall=0.111 fmeasure=0.071	0.83	0.07	0.04	0.42	0.55

Table 4.8: **MBR** DocQA Results - *Unstructured without Table Parsing*

	Answers generated including table parsing	Answers generated excluding table parsing
FILE 1 - SOP01		
QUERY 1		
QUERY 2		
QUERY 3		
QUERY 4		
QUERY 5		

	Answers generated including table parsing	Answers generated excluding table parsing
FILE 5 - SOP05		
QUERY 1		
QUERY 2		
QUERY 3		
QUERY 4		
QUERY 5		

	Answers generated including table parsing	Answers generated excluding table parsing
FILE 2 - SOP02		
QUERY 1		
QUERY 2		
QUERY 3		
QUERY 4		
QUERY 5		
QUERY 6		

	Answers generated including table parsing	Answers generated excluding table parsing
FILE 7 - SOP07		
QUERY 1		
QUERY 2		
QUERY 3		
QUERY 4		
QUERY 5		
QUERY 6		

	Answers generated including table parsing	Answers generated excluding table parsing
FILE 3 - SOP03		
QUERY 1		
QUERY 2		
QUERY 3		

	Answers generated including table parsing	Answers generated excluding table parsing
FILE 4 - SOP04		
QUERY 1		
QUERY 2		
QUERY 3		

	Answers generated including table parsing	Answers generated excluding table parsing
FILE 6 - SOP06		
QUERY 1		
QUERY 2		
QUERY 3		
QUERY 4		

Table 4.9: **SOP** DocQA Results - *Unstructured with/without Table Parsing*

LEGEND

A	Correct and Detailed
B	Correct but Not Detailed
C	Partially Correct
D	Incorrect or Misinterpreted
NA	NO ANSWER

Figure 4.2: Legend for interpreting SOPs results

Chapter 5

Discussion of Document Question Answering Results

The chapter delves into the comprehensive analysis and interpretation of the results presented previously. These results are derived from experiments that evaluate the performance of the document question answering process. The analysis includes a comparison between the results obtained with and without performing table parsing during the document parsing process to highlight the impact of this technique.

The aim is to provide a thorough discussion of the results, examining their implications and potential impact on the performance. The objective is not only to understand the effectiveness of the document question answering process but also to highlight its strengths, weaknesses, and areas for improvement. Additionally, the examination seeks to determine whether table parsing can be considered a crucial step in document parsing for the extraction and interpretation of detailed information from documents, especially within complex domains.

5.1 Analysing PubMedQA Documents

To comprehensively evaluate the results of the document question answering process over PubMedQA documents, it is essential to take into account the metrics provided: ROUGE scores, BLEU score, and Cosine Similarity measures using Bag of Words, TF-IDF, Sentence Transformer, and Universal Sentence Encoder embeddings. These metrics offer insights into the quality of the generated answers compared to the target answers, focusing on aspects such as overlap in key terms, semantic similarity, and overall coherence.
The parsing process for PubMedQA documents is performed both using Unstructured and Form Recognizer methods. This comprehensive approach ensures a thorough assessment of the effectiveness of different parsing strategies on the quality of the responses.

5.1.1 Form Recognizer Analysis: Impact of Accurate Table Parsing

This section discusses the results obtained from parsing the PubMedQA documents with the Form Recognizer parser, including the contents of the tables by properly parsing them.

File 1 - 10808977

According to the **ROUGE scores** results across the questions, it is possible to observe varying degrees of overlap, with some questions showing relatively higher precision and recall. This indicates a closer match between generated and target answers.
The **BLEU scores** across the questions vary, indicating differences in the quality of the generated answers. For instance, the third question, as shown in Figure 5.1, presents a notably higher BLEU score, suggesting a closer match to the target answer.
Cosine Similarity results obtained with Sentence Transformer and Universal

Sentence Encoder embeddings generally provide higher similarity scores, suggesting that these embeddings capture semantic similarities more effectively then Bag of Words or TF-IDF representations.

```
Question: What are the values associated to Intention to have a mammogram in 1996?
Target answer: The values associated to the intention to have a mammogram in 1996
are characterized by Percent on schedule of 72 and p-value equal to 0.001.

Generated answer: The percent of schedule for intention to have a mammogram in
1996 (yes vs. no) is 72%, and the p-value associated with it is .001.
```

Figure 5.1: QUERY 3 of File 1 from PubMedQA: Target vs. Generated Answers using Form Recognizer with Table Parsing

File 2 - 11729377

Higher **ROUGE scores** in some answers suggest a better match in terms of covered content, while lower scores in other indicates areas where the generated answers may have missed key details or introduced information not present in the target responses.
The **BLEU scores** across different questions show a range of values, indicating varying levels of quality in generated answers. Higher scores suggest that the generated answers are more closely aligned with the target answers, capturing the most relevant information more accurately.
Higher values in **Cosine Similarity** measures, especially those obtained using Sentence Transformers and Universal Sentence Encoders, suggest a stronger semantic alignment between the generated and target answers. These measures indicate how well the generated answers capture the underlying meaning and context of the target answers, beyond the textual overlap.

File 3 - 16418930

The precision, recall, and F-measure values associated to different **ROUGE scores** suggest that while there is some alignment between the generated and target texts, the extent of this alignment varies significantly across different questions. An example is the first question shown in Figure 5.2, which is characterized by high recall but low precision. This indicates that while most of the target answer is captured by the generated answer, there is also significant additional information that may not be directly relevant.

Question: Landolt C and Snellen e acuity: differences in strabismus amblyopia
Target answer: Using the charts described, there was only a slight overestimation of visual acuity by the Snellen E compared to the Landolt C, even in strabismus amblyopia. Small differences in the lower visual acuity range have to be considered.

Generated answer: There is a difference in visual acuity measurements between Landolt C and Snellen E in the case of strabismic amblyopia. It is noted that with higher Snellen E visual acuity in the lower visual acuity range, there is an increasing difference that is not statistically significant. The E-Haken (Snellen E) visual acuity tends to be slightly higher than the Landolt-Ring (Landolt C) visual acuity, particularly in the lower range, as indicated by the regression line, although this difference is not statistically significant. The correlation coefficient of 0.98 suggests a strong statistical relationship between the logarithmic visual acuity values measured by both methods. Furthermore, it is mentioned that the arrangement of the optotypes in the tests ensures that significant overestimation of Landolt-Ring visual acuity by Snellen E visual acuity in cases of strabismic amblyopia is not a concern, although a small deviation in the lower visual acuity range should be considered. In conclusion, while there is a tendency for Snellen E visual acuity to be slightly higher than Landolt C visual acuity in strabismic amblyopia, especially at lower acuity levels, this difference is not statistically significant and both methods show a strong correlation in visual acuity values.

Figure 5.2: QUERY 1 of File 3 from PubMedQA: Target vs. Generated Answers using Form Recognizer with Table Parsing

Across the questions, the **BLEU scores** vary, with some questions showing relatively high scores, suggesting a closer match to the target answers, and others showing lower scores, indicating less similarity.
For example, the third question, shown in Figure 5.3, presents a notably high BLEU score, suggesting a good quality match between the generated and target answers.

Question: What is the MW LR associated to n = 9 and Visus > 0.1?
Target answer: The value of MW LR related to Visus > 0.1 and n equal to 9 is 0.17 ± 0.3, as shown in the table 5 of the provided document.

Generated answer: The value of MW LR related to Visus that ranges in the interval between 0.32 and 0.1 and n equal to 9 is 0.17 + 0.3 and 0.17 - 0.3.

Figure 5.3: QUERY 3 of File 3 from PubMedQA: Target vs. Generated Answers using Form Recognizer with Table Parsing

Generally, the **Cosine Similarity** scores are relatively high across the different representations, indicating a good degree of semantic alignment between the generated and target texts. The Universal Sentence Encoder embeddings tend to provide high similarity scores, suggesting that they are effective at capturing the semantic content of the answers.

File 4 - 17208539

ROUGE scores across the questions indicate different levels of overlap between the generated answers and the target answers.

Taking as example the first question, shown in Figure 5.4, it has relatively low precision and higher recall. This indicates that the generated answer may include relevant information but also contains additional details not present in the target answer.

Question: Are the long-term results of the transanal pull-through equal to those of the transabdominal pull-through?
Target answer: Our long-term study showed significantly better (2-fold) results regarding the continence score for the abdominal approach compared with the transanal pull-through. The stool pattern and enterocolitis scores were somewhat better for the TERPT group. These findings raise an important issue about the current surgical management of HD; however, more cases will need to be studied before a definitive conclusion can be drawn.

Generated answer: The long-term results of the transanal pull-through (TERPT) and the transabdominal pull-through (ABD) for Hirschsprung's disease seem to depend on several factors, and it's not inherently the method that's the issue. One discussion point is that if the anastomosis is too low, there could be problems with continence, and if retractors are used and the sphincter is stretched, there might be issues with the sphincter as well. This is not necessarily a problem with the transanal pull-through itself, but rather the way the operation is performed. Another point raised is that the traction on the sphincters could be due to both the application of the retractors and the pulling and manipulating of the colon through the anus. Furthermore, if the colon is tethered from above, it could broaden the anorectal angle and make continence more difficult, which is considered a potential problem with the purely transanal pull-through where the ligaments are not released from above. However, it's also noted that in most cases of rectosigmoid aganglionosis, the colon is redundant, and the coloanal anastomosis is not done under tension, so there might not be an opening of the anorectal angle. Therefore, this might not necessarily relate to the continence outcome. In conclusion, the long-term results of the transanal pull-through and the transabdominal pull-through seem to be influenced by several factors related to the specific techniques used during the operation, rather than being inherently equal or unequal.

Figure 5.4: QUERY 1 of File 4 from PubMedQA: Target vs. Generated Answers using Form Recognizer with Table Parsing

Across the questions, **BLEU scores** vary significantly. For example, the fifth question, shown in Figure 5.5, presents a notably high BLEU score, suggesting a high quality of the generated answer in terms of matching the target answer.

Question: What are the different total scores associated to different groups like ABD or TERPT?
Target answer: The total scores associated with the ABD and TERPT groups are divided into different groups. For the ABD group: excellent (0-10) total score is present in 11 cases (52.4%); good (11-20) in 7 (33.3%); fair (21-30) in 3 (14.3%); there were no poor cases (31-40). While, for the TERPT group: excellent (0-10) in 9 cases (45%); good (11-20) in 6 (30%); fair (21-30) in 5 (25%); there were no poor (31-40) cases

Generated answer: The table total scores associated with the ABD and TERPT groups are: For the ABD group: - Excellent (0-10) total score was found in 11 (52.4%) cases - Good (11-20) total score was found in 7 (33.3%) cases - Fair (21-30) total score was found in 3 (14.3%) cases - There were no Poor (31-40) scores - For the TERPT group: - Excellent (0-10) total score was found in 9 (45%) cases - Good (11-20) total score was found in 6 (30%) cases - Fair (21-30) total score was found in 5 (25%) cases - There were no Poor (31-40) scores. Please note that these scores reflect the outcomes of the respective groups in the study mentioned in the document.

Figure 5.5: QUERY 5 of File 4 from PubMedQA: Target vs. Generated Answers using Form Recognizer with Table Parsing

The **Cosine Similarity** scores are relatively high across the different representations, indicating a good degree of semantic alignment between the generated and target texts. The Sentence Transformer and Universal Sentence Encoder embeddings tend to provide high similarity scores.

File 5 - 26037986

Higher **ROUGE scores** in some answers suggest a closer match in terms of content, while lower scores in others indicate discrepancies in capturing or presenting the key information from the target answers. The third question, shown in Figure 5.6, has relatively high precision and recall, indicating a strong alignment between the generated and target texts.

Question: What information do we have about Post-operative destination and 30-day mortality for patients over 70 years?
Target answer: The total number of patients included in the study was 190, which represents 43% of the study population. The 30-day mortality rate is equal to 21%, the 1-year mortality rate to 37%, and the average length of stay was 32 days. For patients who underwent resection without stoma formation, that are 67 representing 35% of the study population, the 30-day mortality rate is 21%, the 1-year mortality rate is 32%, and the average length of stay was 26 days. For patients who underwent resection with stoma formation, that are 37 patients representing 20% of the study population, the 30-day mortality rate is 24%, the 1-year mortality rate is 32%, and the average length of stay was 33 days.

Generated answer: The table includes data on the number of patients, 30-day mortality percentage, 1-year mortality percentage, and length of stay in days for different post-operative destinations. Here is a summary of the information provided: The total number of patients included in the study was 190, which represents 43% of the study population. The 30-day mortality rate for this group was 21%, the 1-year mortality rate was 37%, and the average length of stay was 32 days. For patients who underwent resection without stoma formation (67 patients, representing 35% of the study population), the 30-day mortality rate was 21%, the 1-year mortality rate was 32%, and the average length of stay was 26 days. For patients who underwent resection with stoma formation (37 patients, representing 20% of the study population), the 30-day mortality rate was 24%, the 1-year mortality rate was 32%, and the average length of stay was 33 days. This table provides insights into the outcomes of patients based on their post-operative destination, specifically focusing on mortality rates and length of hospital stay after undergoing emergency laparotomy procedures.

Figure 5.6: QUERY 3 of File 5 from PubMedQA: Target vs. Generated Answers using Form Recognizer with Table Parsing

The **BLEU scores** across the questions show a range of values, reflecting differences in the quality of the generated answers. Higher scores, such as those seen in the second question, shown in Figure 5.7, suggest a good quality match, capturing the essential information accurately.

The **Cosine Similarity** scores are generally relatively high, especially those obtained using Sentence Transformers and Universal Sentence Encoders. This suggests that, semantically, the generated answers capture the essence of the target answers well, even if there are some discrepancies in the specific details.

Question: What is the overall 1-year mortality rates in all ages and particularly in those aged over 70, both during their hospital stay and subsequently in the community?
Target answer: The overall 1-year mortality rate in all ages is 25%, which rises to almost 50% in those aged 70 and older. Specifically, for those aged over 70 years old, the 1-year mortality rate is 37% . This information indicates a significant increase in mortality rates with age, particularly in the older population, both during their hospital stay and subsequently in the community.

Generated answer: The overall 1-year mortality rate in all ages for patients undergoing emergency laparotomy is 25%, which rises to almost 50% in those aged 70 and older. This indicates a significant increase in mortality rates with age, highlighting the need for improved outcomes and care, both during the hospital stay and in the community, particularly for the older patient demographic.

Figure 5.7: QUERY 2 of File 5 from PubMedQA: Target vs. Generated Answers using Form Recognizer with Table Parsing

To summarize, ROUGE scores demonstrate a mixed performance in how well generated answers match target answers, with some responses closely aligning and others significantly deviating by either missing important information or adding useless details. This inconsistency highlights the challenges in accurately capturing essential content. BLEU scores further highlight this point; higher BLEU scores in some responses suggest a good quality match, whereas lower scores in others point to a need for improvement to ensure that generated answers more closely align with the target answers. Cosine Similarity measures show strong semantic similarity, suggesting that despite discrepancies in details or wording, the generated answers capture the essence of the target answers.

5.1.2 Form Recognizer Analysis: Implications of Missing Table Parsing

This section presents the discussion of the results obtained from parsing the PubMedQA documents with the Form Recognizer parser, without properly performing table parsing.

File 1 - 10808977

Across all questions, the **ROUGE scores** indicate a moderate level of overlap between the generated and target answers. The F-measures, which balance precision and recall, are relatively low, meaning that while there is some overlap, significant portions of the target answers are not captured by the generated ones.

The **BLEU scores** vary significantly across questions, with some scores indicating a better match than others. The values suggest that while some answers may capture key elements of the target response, there might be mismatches in terms of exact wording or phrasing.

The **Cosine Similarity** results using the Bag of Words and TF-IDF representations are relatively low across all questions, indicating a moderate similarity between target and generated answers. This suggest that the overall match is not very high even if some keywords are shared. Using Sentence Transformer and Universal Sentence Encoder representations, the results are generally higher than the ones obtained in the previous cases, but overall, they remain quite low.

Question: What is the effectiveness of tailored interventions in increasing mammography use among women?

Target answer: Tailored intervention increased mammography among women aged 40-74 and tailored print interventions resulted in 67% of women having a mammogram in 1996 and 69% in 1997; telephone counseling interventions instead resulted in 71% in 1996 and 72% in 1997 6. Tailored interventions can be effective in promoting mammography use among women.

Generated answer: Tailored interventions seem to have a positive impact on increasing mammography use among women. For instance, a study found that using a stage-matched, tailored intervention increased mammography use among women aged 40-74. Another study indicated that tailored print communication could be effective, although it did not specify the extent of its effectiveness in relation to mammography use. Additionally, a study suggested that tailored messages might make a difference in physician recommendations for mammography. *However, the documents do not provide specific quantitative data regarding the effectiveness of these interventions.*

Figure 5.8: QUERY 2 of File 1 from PubMedQA: Target vs. Generated Answers using Form Recognizer without Table Parsing

For example, the second question, shown in Figure 5.8, presents high Cosine Similarity when using the Sentence Transformer representation. This suggests that semantically, the generated answer is quite similar to the target one. Despite this, upon reviewing the answers, it is noticeable that the generated response lacks specific quantitative data due to improper parsing of tables.

File 2 - 11729377

ROUGE scores are generally low across all questions, indicating lack of overlap between the generated and target answers. This suggests that while the generated responses may capture some of the concepts, they diverge in detail or exact wording from the target responses.

BLEU scores are also generally low, indicating that the generated answers are not closely aligned with the expected target responses in terms of precise language and structure.

The **Cosine Similarity** scores are higher when using Sentence Transformer and Universal Sentence Encoder embeddings, suggesting these representations find a moderate level of semantic similarity. However, the lower scores obtained with the other two representations indicate less similarity at the level of specific word usage and term frequency.

File 3 - 16418930

ROUGE scores across the questions indicate varying levels of overlap. High recall scores suggest that the generated answers tend to include most of the target content but lower precision scores hint at additional information being included.

One example of this is shown in Figure 5.9, where the recall is high but it has very low precision, since additional information are included in the generated answer.

Question: Landolt C and snellen e acuity: differences in strabismus amblyopia
Target answer: Using the charts described, there was only a slight overestimation of visual acuity by the Snellen E compared to the Landolt C, even in strabismus amblyopia. Small differences in the lower visual acuity range have to be considered.

Generated answer: According to the document, the Landolt C and Snellen E acuity tests have been used for assessing visual acuity in strabismus amblyopia. The document suggests that the Landolt C (referred to as Landoltring in the document) acuity tends to be slightly overestimated compared to the Snellen E (referred to as E-Haken in the document) acuity. This overestimation, however, is not significant even in the case of strabismus amblyopia. However, a slight deviation in the lower visual acuity range should be considered. The document presents a correlation coefficient of 0.98, indicating a high degree of correlation between the results of the two tests. It is also mentioned that the Snellen E values were generally slightly higher than the Landolt C values, and this difference increases in the lower range.

Figure 5.9: QUERY 1 of File 3 from PubMedQA: Target vs. Generated Answers using Form Recognizer without Table Parsing

BLEU scores show a range of alignment from low to moderate, with higher scores in questions where the generated answer more closely matched the target in terms of specific details and wording.
Cosine Similarity metrics, particularly those using advanced models like the Sentence Transformer and Universal Sentence Encoder, often show higher scores than Bag of Words or TF-IDF. This means that, even when the exact wording or details differ, the generated responses capture the semantic content of the target ones.

File 4 - 17208539

ROUGE scores vary across questions, with higher scores in questions that don't require the table contents to be correctly parsed. Lower scores are obtained in those questions strictly related to tables within the document, reflecting the failure to capture the specific details when Table Parsing is not properly performed.
BLEU scores also vary, with the second question, shown in Figure 5.10, presenting a high score, suggesting a close match in the structure and information content.

Question: What are the potential factors contributing to compromised continence in the TERPT procedure?
Target answer: The potential factors contributing to compromised continence in the Transanal Endorectal Pull-Through (TERPT) procedure include tethering of the colon from above, which can broaden the anorectal angle and make continence more difficult; the use of retractors during the procedure, which can stretch the sphincter and potentially compromise anal sphincter pressures, leading to continence issues; manipulating and pulling the colon through the anus, which can exert traction on the sphincters beyond the application of retractors, as mentioned by a group of pediatric surgeons who all used the same standard technique. Additionally, the technical aspects of the operation, such as making the anastomosis too low, can also affect continence. It is suggested that the problems with continence are not inherent to the transanal pull-through itself but rather related to the way the operation is performed, indicating the importance of avoiding technical pitfalls.

Generated answer: According to the document, potential factors contributing to compromised continence in the Transanal Endorectal Pull-Through (TERPT) procedure include: 1. The application of retractors and stretching of the sphincter during the procedure. This could potentially cause problems with the sphincter function post-operatively. 2. Manipulation and pulling of the colon through the anus during the procedure can also put traction on the sphincters, which can further contribute to compromised continence. 3. Tethering the colon from above can broaden the anorectal angle, making continence more difficult. This can be a problem with the purely transanal pull-through procedure where the ligaments are not released from above. However, it's important to note that these factors can vary based on the surgical technique used and the individual patient's condition.

Figure 5.10: QUERY 2 of File 4 from PubMedQA: Target vs. Generated Answers using Form Recognizer without Table Parsing

Similar to ROUGE scores, **Cosine Similarity** results are higher for those questions related to the textual content of the document while they are lower when it comes to answering using the information stored within tables.

File 5 - 26037986

High **ROUGE scores**, observed in the second question, indicate a strong overlap suggesting that the generated answer closely matched the target one in terms of specific details. For questions strictly related to table contents, the results are very low.

Also in this case, in the second question, which may be answered even if the information in the document tables is not well processed, the **BLEU score** is quite high. On the other hand, for questions covering table contents, BLEU scores are lower.

Taking again into consideration the second question, shown in Figure 5.11, high **Cosine Similarity** scores suggest a strong semantic match between the generated and the target answers. This implies that despite any difference in wording, the core concepts are similar. As mentioned for the other metrics, answers that should contain table information perform lower results.

Question: What is the overall 1-year mortality rates in all ages and particularly in those aged over 70, both during their hospital stay and subsequently in the community?
Target answer: The overall 1-year mortality rate in all ages is 25%, which rises to almost 50% in those aged 70 and older. Specifically, for those aged over 70 years old, the 1-year mortality rate is 37% . This information indicates a significant increase in mortality rates with age, particularly in the older population, both during their hospital stay and subsequently in the community.

Generated answer: The overall 1-year mortality rate for all ages is 25%. However, this rate increases significantly in older age groups, rising to almost 50% in individuals aged 70 and older. These figures encompass mortality during hospital stay and subsequent community living.

Figure 5.11: QUERY 2 of File 5 from PubMedQA: Target vs. Generated Answers using Form Recognizer without Table Parsing

To sum up, performance varies significantly across questions, depending on whether the requested information is located solely within the document's tables or within its textual content. The analysis reveals strengths in capturing general concepts and weaknesses in detailing specific, accurate information. This variability underscores the importance of enhancing information retrieval and of implementing proper table parsing processes to improve precision, especially in contexts requiring detailed information extraction.

5.1.3 Comparative Insights: Form Recognizer with and without Table Parsing

To evaluate and compare the results obtained with and without performing table parsing, examining the metrics across various questions is useful. Several key observations emerge across the different questions and metrics when comparing the results. These observations highlight the impact of table parsing processes on the accuracy, comprehensiveness, and semantic similarity of generated answers compared to target answers.

Firstly, it is possible to take as an example the first question and the related answers from File 1:

> `QUERY:` What are the values associated to Intention to have a mammogram in 1996?
>
> `EXPECTED ANSWER:` The values associated to the intention to have a mammogram in 1996 are characterized by Percent on schedule of 72 and p-value equal to 0.001.
>
> `GENERATED ANSWER including table parsing:` The percent of schedule for intention to have a mammogram in 1996 (yes vs. no) is 72%, and the p-value associated with it is.001.
>
> `GENERATED ANSWER excluding table parsing:` I'm sorry, but *none of the provided documents contain a table with values associated with the intention to have a mammogram in 1996*. Please ensure that the correct document was uploaded or provide additional documents that might contain this information.

When table parsing is performed, the results show a moderate level of overlap and semantic similarity between the expected and the generated answers. Without proper table parsing, the system is not able to retrieve the requested information from the document.
Another example of this behavior can be seen in the fifth question and the related generated answers from File 1:

> `QUERY:` What are the values related to Reported facilitator during 1995?
>
> `EXPECTED ANSWER:` According to the information stored in the document, in 1995 the 77% of the participants reported a facilitator. Having a deeper look at the document, it is possible to retrieve also information related to 1996 and 1997. Namely, in 1996, 75% of the participants reported a facilitator and in 1997 the 73%.
>
> `GENERATED ANSWER including table parsing:` The values related to Reported facilitator during 1995, as per the table in the uploaded document, are as follows: in 1995, the reported value was 77, and the p-value was .073. These values are from the table in the uploaded document.
>
> `GENERATED ANSWER excluding table parsing:` I'm sorry, but *none of the documents provided contain a table or any values related to a 'Reported facilitator during 1995'*. Please ensure the correct document has been uploaded or provide additional details to aid in the search.

In this case, the response generated including table parsing does not perfectly overlap with the expected one, resulting in moderate ROUGE and BLEU scores. Nonetheless, the high semantic similarity suggests that the content of this generated response closely aligns with the target response.

To highlight the impact of table parsing within the document parsing process, consider the answers to the fifth question from File 2:

`QUERY:` What information are provided related to Surgical complications?

`EXPECTED ANSWER:` The information related to surgical complications are provided in the context of split-liver transplantation (SLT) and living-related liver transplantation (LRT). The table within the document compares the objectives and results of SLT and LRT, and it mentions that liver function parameters and surgical complication rates did not differ significantly between the two groups. It also notes that in the SLT group, the mean cold ischemic time was longer than in the LRT group, and serum values of alanine aminotransferase during the first postoperative week were significantly higher in the SLT group. Biliary complications were more frequent in the living donor group compared with the split group, but this difference was not statistically significant. Additionally, there is no technical reason why biliary complications should be more frequent after living donor liver transplantation.

`GENERATED ANSWER including table parsing:` The table in the uploaded document mentions surgical complications related to liver transplantation. Specifically, it notes that biliary complications were more frequent in the living donor group compared with the split group. However, the statistical analysis showed that this difference was not significant. The document also mentions that there is technically no reason why biliary complications should be more frequent after living donor liver transplantation.

`GENERATED ANSWER excluding table parsing:` I'm sorry, but *the tool did not provide any specific information related to surgical complications in a table within the uploaded document*. It might be possible that the tool was *unable to extract tables* from the document. Please provide more context or check the document manually for the required information.

In this case, the results, characterized by very high ROUGE and BLEU scores as well as high Cosine Similarity scores from performing table parsing, indicate a strong accuracy and overlap with the target response, alongside a significant semantic match.

Analyzing the results from File 3, one relevant example is the third question:

> `QUERY:` What is the MW LR associated to n = 9 and Visus > 0.1?
>
> `EXPECTED ANSWER:` The value of MW LR related to Visus > 0.1 and n equal to 9 is 0.17 ± 0.3, as shown in the table 5 of the provided document.
>
> `GENERATED ANSWER including table parsing:` The value of MW LR related to Visus that ranges in the interval between 0.32 and 0.1 and n equal to 9 is 0.17 + 0.3 and 0.17 - 0.3.
>
> `GENERATED ANSWER excluding table parsing:` The document does *not provide specific information about the MW LR associated with n = 9 and Visus > 0.1*. The document contains some charts and discussions related to Visus measurements with Landoltring and E-Haken, but no specific values or relationships for n = 9 and Visus > 0.1 are explicitly mentioned.

Tab. 5 Mittelwerte gerundet (± SEM) in dekadisch logarithmischen Stufen für LR und EH der schielamblyopen Augen (n = 39) für 4 Visusbereiche und exakte mittlere Differenzen in Visusstufen

Visus	*n*	*MW LR*	*MW EH*	*Diff. EH–LR*
> 1,0	0			
1,0–0,33	13	0,56 ± 0,1	0,59 ± 0,2	0,17
0,32–0,1	9	0,17 ± 0,3	0,22 ± 0,5	1,07
< 0,1	17	0,044 ± 0,4	0,050 ± 0,4	0,58

Figure 5.12: Table within File 3 taken from PubMedQA dataset [2]

The generated answer, after parsing tables within the document, accurately identifies the row of the table containing the requested information, as confirmed by referring to Figure 5.12. This figure depicts the table from File 3 of

the PubMedQA dataset, which contains the required information.
On the other hand, without table parsing, a failure to provide specific information can be observed, indicating a lack of detail extracted from the document. The fourth question over File 4 can also serve as an explicative example for this behavior:

> `QUERY:` What are the values associated to scoring item ABD (n=21)?
>
> `EXPECTED ANSWER:` The values associated with the scoring item ABD (n=21) are listed as follows: total is equal to 11.28 + 7.75; continence is equal to 4.38 + 4.77*; stool pattern is equal to 2.71 + 2.24*; enterocolitis is equal to 4.19 + 2.98. Asterisks (*) next to some values indicate that P is less than .05.
>
> `GENERATED ANSWER including table parsing:` The values associated ABD (n=21) are: Total scoring ranged from 0 to 40: 0 to 10, excellent; 11 to 20, good; 21 to 30, fair; 31 to 40, poor. The total score for ABD was 11.28 ± 7.75.- When the subcategories of stooling scores were examined, the continence score for ABD was 4.38 ± 4.76. The stooling pattern score for ABD was 2.71 ± 2.24.
>
> `GENERATED ANSWER excluding table parsing:` I'm sorry, but *the tool did not find any information related to 'scoring item ABD (n=21)'* in the provided document.

Table 4 A comparison of the different scoring items in the ABD and TERPT groups, as well as in the redo and nonredo groups

Scoring item	Total	Continence	Stool pattern	Enterocolitis
ABD (n = 21)	11.28 ± 7.75	4.38 ± 4.77*	2.71 ± 2.24*	4.19 ± 2.98
TERPT (n = 20)	12.75 ± 8.07	8.00 ± 6.06*	1.58 ± 1.26*	3.18 ± 2.23
Redo (n = 6)	21.16 ± 3.25**	11.33 ± 3.38*	4.50 ± 1.61**	5.33 ± 2.18
Nonredo (n = 35)	10.42 ± 7.31	5.25 ± 5.53	1.75 ± 1.65	3.41 ± 2.65

* $P < .05$.
** $P < .001$.

Figure 5.13: Table within File 4 taken from PubMedQA dataset [18]

When table parsing is employed, the generated answer provides a detailed breakdown of the scores associated with the *'ABD group'*, which closely matches

with the target response. Thus, the metrics indicate a good level of similarity, especially when computing Cosine Similarity using the Sentence Transformer. Figure 5.13 shows the table within File 4 of the PubMedQA dataset from which the information is derived.
Moving to File 5, one last example can be shown:

> `QUERY:` What is the overall 1-year mortality rates in all ages and particularly in those aged over 70, both during their hospital stay and subsequently in the community?
>
> `EXPECTED ANSWER:` The overall 1-year mortality rate in all ages is 25%, which rises to almost 50% in those aged 70 and older. Specifically, for those aged over 70 years old, the 1-year mortality rate is 37%. This information indicates a significant increase in mortality rates with age, particularly in the older population, both during their hospital stay and subsequently in the community.
>
> `GENERATED ANSWER including table parsing:` The overall 1-year mortality rate in all ages for patients undergoing emergency laparotomy is 25%, which rises to almost 50% in those aged 70 and older. This indicates a significant increase in mortality rates with age, highlighting the need for improved outcomes and care, both during the hospital stay and, in the community, particularly for the older patient demographic.
>
> `GENERATED ANSWER excluding table parsing:` The overall 1-year mortality rate for all ages is 25%. However, this rate increases significantly in older age groups, rising to almost 50% in individuals aged 70 and older. These figures encompass mortality during hospital stay and subsequent community living.

In this scenario, the first answer provides specific mortality rates for all ages and for individuals over 70, closely matching the expected answer. Meanwhile, the second answer provides general mortality rates without specific details. This serves as further evidence that including table parsing enhances the information extraction process.

The comparison demonstrates the significant impact of table parsing processes on enhancing information extraction from documents. It clearly shows that table parsing increases the accuracy and specificity of extracted information, as demonstrated by performance metrics. The absence of table parsing results in reduced accuracy and relevance of generated answers, making the question answering system less capable of answering queries related to tables.

5.1.4 Unstructured Parser Analysis: Impact of Accurate Table Parsing

This section discusses the results obtained from parsing the PubMedQA documents with the Unstructured parser, including the contents of the tables by properly and accurately parsing them.

File 1 - 10808977

ROUGE scores vary across questions, with some answers showing good overlap and others indicating room for improvement.

BLEU scores are relatively low, indicating that while some correct information is captured, the phrasing and order may differ significantly from the target answer.

Cosine Similarity scores are relatively higher when using Sentence Transformer and the Universal Sentence Encoder representations, suggesting that despite differences in wording or structure, the generated answers capture a similar semantic space to that of the target answers.

For example, in the third question shown in Figure 5.14, the generated answer closely matches the target answer, providing a detailed breakdown with correct inclusion of percentage and the p-value, matching the target answer's data.

Question: What are the values associated to Intention to have a mammogram in 1996?
Target answer: The values associated to the intention to have a mammogram in 1996 are characterized by Percent on schedule of 72 and p-value equal to 0.001.

Generated answer: The values associated with the intention to have a mammogram in 1996 are as follows: the percent on schedule for those who intended to get a mammogram, it was 77%, and for those who did not intend to, it was 40%; the associated p-value is .001.

Figure 5.14: QUERY 3 of File 1 from PubMedQA: Target vs. Generated Answers using Unstructured with Table Parsing

File 2 - 11729377

The variability in **ROUGE and BLEU scores** across questions highlights the challenge of capturing and reproducing specific details and statistical data. Where the generated answers align closely with the target answers in terms of specific details, for example in the fourth question shown in Figure 5.15, the metrics reflect this with higher scores.

Question: What is the conclusion of the conducted study?
Target answer: The conclusion of the conducted study is that the short- and long-term outcomes after living-related liver transplantation (LRT) and split-liver transplantation (SLT) did not differ significantly. To avoid the risk for the donor in LRT, SLT represents the first-line therapy in pediatric liver transplantation in countries where cadaveric organs are available. LRT provides a solution for urgent cases in which a cadaveric graft cannot be found in time or if the choice of the optimal time point for transplantation is vital.

Generated answer: The conclusion of the conducted study, as per the document, is that the short-term and long-term outcomes after Living Related Transplantation (LRT) and Split Liver Transplantation (SLT) did not significantly differ. However, to avoid the risk for the donor in LRT, SLT represents the first-line therapy in pediatric liver transplantation in countries where cadaveric organs are available. LRT provides a solution for urgent cases in which a cadaveric graft cannot be found in time or if the choice of the optimal time point for transplantation is needed.

Figure 5.15: QUERY 4 of File 2 from PubMedQA: Target vs. Generated Answers using Unstructured with Table Parsing

The BLEU score in this case is *71.02*, which is exceptionally high, reflecting an almost exact match in phrasing and sequence of ideas between the generated and target answers.

High **Cosine Similarity** scores indicate that the generated answers generally capture the correct topics, even if they sometimes miss specific details.

File 3 - 16418930

The **ROUGE scores** across the questions show variable performance in terms of precision, recall, and F-measure. These variations indicate that while the generated answers often capture a significant portion of the content found in the target answers (as suggested by the recall values), they may also include additional information not present in the target or miss key phrases, affecting precision.

One example of this case is shown in Figure 5.16, where the generated answer includes relevant content not precisely matching the target's wording.

Question: Landolt C and snellen e acuity: differences in strabismus amblyopia
Target answer: Using the charts described, there was only a slight overestimation of visual acuity by the Snellen E compared to the Landolt C, even in strabismus amblyopia. Small differences in the lower visual acuity range have to be considered.

Generated answer: According to the document, a study was conducted on 39 patients with amblyopia due to strabismus, using charts with the Snellen E and the Landolt C to assess visual acuity. The results showed that the differences between Landolt C acuity (LR) and Snellen E acuity (SE) were small. The mean decimal values for LR and SE were 0.25 and 0.29 in the entire group and 0.14 and 0.16 for the eyes with strabismus amblyopia. The mean difference between LR and SE was 0.55 lines in the entire group and 0.55 lines for the eyes with strabismus amblyopia, with higher values of SE in both groups. The results of the other groups were similar with only small differences between LR and SE.

Figure 5.16: QUERY 1 of File 3 from PubMedQA: Target vs. Generated Answers using Unstructured with Table Parsing

The **BLEU scores** range from low to moderate across the questions, suggesting that while there is some level of coherence between the generated and target answers, there are also discrepancies in exact wording, order, and structure.

Cosine Similarity scores generally indicate a moderate to high level of semantic similarity between the target answers and the generated ones, which tend to capture the underlying meaning and key concepts of the target responses.

File 4 - 17208539

The **ROUGE scores** exhibit a range of values, indicating variability in the overlap between the generated and target answers, though not always with high precision.

The **BLEU scores** also show variability, with some questions achieving higher scores, indicating a better match in the sequence of words and phrases. For instance, the fifth question shown in Figure 5.17, has a BLEU score of *37.47* suggesting that the generated answer closely matches the phrasing and structure of the target answer.

Question: What are the different total scores associated to different groups like ABD or TERPT?
Target answer: The total scores associated with the ABD and TERPT groups are divided into different groups. For the ABD group: excellent (0-10) total score is present in 11 cases (52.4%); good (11-20) in 7 (33.3%); fair (21-30) in 3 (14.3%); there were no poor cases (31-40). While, for the TERPT group: excellent (0-10) in 9 cases (45%); good (11-20) in 6 (30%); fair (21-30) in 5 (25%); there were no poor (31-40) cases.

Generated answer: According to the table in the document, the total scores associated with the ABD and TERPT groups are as follows: For the ABD group: - Excellent (0-10) total score was found in 11 (52.4%) cases - Good (11-20) total score was found in 7 (33.3%) cases - Fair (21-30) total score was found in 3 (14.3%) cases - There were no Poor (31-40) scores - For the TERPT group: - Excellent (0-10) total score was found in 9 (45%) cases - Good (11-20) total score was found in 6 (30%) cases - Fair (21-30) total score was found in 5 (25%) cases - There were no Poor (31-40) scores. Please note that these scores reflect the outcomes of the respective groups in the study mentioned in the document.

Figure 5.17: QUERY 5 of File 4 from PubMedQA: Target vs. Generated Answers using Unstructured with Table Parsing

Cosine Similarity scores generally indicate moderate to high semantic similarity between the generated and target answers. This suggests that, at a semantic level, the generated answers tend to capture the key concepts of the target answers well.

File 5 - 26037986

The **ROUGE and BLEU scores** show variability, indicating differences in how well the generated answers capture the content and match the sequence of words and phrases of the target responses.

Question: What information do we have about Post-operative destination and 30-day mortality for patients over 70 years?
Target answer: The total number of patients included in the study was 190, which represents 43% of the study population. The 30-day mortality rate is equal to 21%, the 1-year mortality rate to 37%, and the average length of stay was 32 days. For patients who underwent resection without stoma formation, that are 67 representing 35% of the study population, the 30-day mortality rate is 21%, the 1-year mortality rate is 32%, and the average length of stay was 26 days. For patients who underwent resection with stoma formation, that are 37 patients representing 20% of the study population, the 30-day mortality rate is 24%, the 1-year mortality rate is 32%, and the average length of stay was 33 days.

Generated answer: The table includes the following data: The total number of patients included in the study was 190, the 43% of the study population. The 30-day mortality rate for this group was 21%, the 1-year mortality rate was 37%, and the average length of stay was 32 days. For patients who underwent resection without stoma formation (67 patients, 35% of the study population), the 30-day mortality rate was 21%, the 1-year mortality rate was 32%, and the average length of stay was 26 days. For patients who underwent resection with stoma formation (37 patients, 20% of the study population), the 30-day mortality rate was 24%, the 1-year mortality rate was 32%, and the average length of stay was 33 days.

Figure 5.18: QUERY 3 of File 5 from PubMedQA: Target vs. Generated Answers using Unstructured with Table Parsing

Cosine Similarity scores indicate moderate to high semantic similarity between the generated and target answers. For example, the third question shown in Figure 5.18, has very high Cosine Similarity scores (*BoW: 0.92, TF-IDF: 0.89, Sentence Transformer: 0.98, Universal Sentence Encoder: 0.97*), suggesting that the generated response captures the underlying meaning of the target answer well.

In summary, high ROUGE and BLEU scores in certain instances reflect a strong alignment in content, structure, and phrasing. Conversely, lower scores in other questions reveal discrepancies in detail accuracy, wording, and sequencing. Cosine Similarity scores generally show moderate to high semantic alignment across most questions, suggesting that the generated answers successfully capture the essence of the target answers, despite variations in detail or expression. Overall, these results highlight the crucial need for balancing precise detail with semantic coherence in the generated responses.

5.1.5 Unstructured Parser Analysis: Implications of Missing Table Parsing

This section presents the results obtained from parsing the PubMedQA documents with the Unstructured parser, without properly performing table parsing.

File 1 - 10808977

ROUGE scores across the questions vary, suggesting a mix of partial and good alignments depending on the question.

BLEU scores are relatively low, indicating that there might be room for improvement in the precision of the generated answers.

Cosine Similarity scores, especially those from the Sentence Transformer and Universal Sentence Encoder, are generally high, indicating a good semantic similarity between the generated and target answers.

Taking the third question as an example, shown in Figure 5.19, the generated answer provides detailed percentage but misinterpret the specific values related to *"being on schedule for a mammogram in 1996"*. The target response focuses on a specific percentage and p-value, which the generated answer overlooks.

Question: What are the values associated to Intention to have a mammogram in 1996 in one of the table present in the uploaded document with id gpcuc:10808977.pdf?
Target answer: The values associated to the intention to have a mammogram in 1996 are characterized by Percent on schedule of 72 and p-value equal to 0.001.

Generated answer: The values associated with the intention to have a mammogram in 1996 are as follows: - Percent on schedule (those who intended to get a mammogram): 77% - Percent off schedule (those who intended to get a mammogram): 23% - Total (those who intended to get a mammogram): 87% - Percent on schedule (those who did not intend to get a mammogram): 40% - Percent off schedule (those who did not intend to get a mammogram): 60% - Total (those who did not intend to get a mammogram): 13%.

Figure 5.19: QUERY 3 of File 1 from PubMedQA: Target vs. Generated Answers using Unstructured without Table Parsing

File 2 - 11729377

ROUGE and BLEU scores vary across questions, indicating differences in the degree of textual overlap between target and generated responses. The highest scores are observed for the fourth question, shown in Figure 5.20, since it can be answered even without collecting all the information from tables.

Question: What is the conclusion of the conducted study?
Target answer: The conclusion of the conducted study is that the short- and long-term outcomes after living-related liver transplantation (LRT) and split-liver transplantation (SLT) did not differ significantly. To avoid the risk for the donor in LRT, SLT represents the first-line therapy in pediatric liver transplantation in countries where cadaveric organs are available. LRT provides a solution for urgent cases in which a cadaveric graft cannot be found in time or if the choice of the optimal time point for transplantation is vital.

Generated answer: The conclusion of the conducted study as per the document indicates that the short- and long-term outcomes after Living Related Transplantation (LRT) and Split Liver Transplantation (SLT) did not differ significantly. SLT is suggested as the first-line therapy in pediatric liver transplantation in countries where cadaveric organs are available, to avoid the risk to the donor in LRT. LRT, on the other hand, provides a solution for urgent cases in which a cadaveric graft cannot be found in time or if the choice of the optimal time point for transplantation is required.

Figure 5.20: QUERY 4 of File 2 from PubMedQA: Target vs. Generated Answers using Unstructured without Table Parsing

The **Cosine Similarity** scores are generally high across all representations, even when there are differences in the exact wording or details provided.

File 3 - 16418930

ROUGE and BLEU scores vary, reflecting differences in detail and accuracy. BLEU scores range from low to moderate, which suggests that while some generated answers have a reasonable level of precision, others fall short. One example of this case can be seen in the second question, shown in Figure 5.21, for which no answer is provided.

Cosine Similarity scores are moderate, reflecting the discrepancies in specific details and accuracy, as well as the gap in the coverage of the document content by the generated answers.

These moderate scores also highlight the misinterpretations and extraction errors from the document, related to the absence of proper table parsing procedure.

```
Question: What are Lea symbols and how are they used in visual acuity assessment?
Target answer: Lea symbols are used in visual acuity assessment and detection of amblyopia. They are a set
of symbols that are used to assess visual acuity and detect amblyopia in clinical research. They have been
the subject of various studies related to visual acuity assessment.

Generated answer: I'm sorry, but the documents provided do not contain any information regarding Lea
symbols or their use in visual acuity assessment.
```

Figure 5.21: QUERY 2 of File 3 from PubMedQA: Target vs. Generated Answers using Unstructured without Table Parsing

File 4 - 17208539

ROUGE scores indicate a moderate to high level of overlap between the generated and target answers, while **BLEU scores** range from low to moderate suggesting that some generated answers fall short.

Cosine Similarity scores are generally moderate, with the highest scores obtained over questions not related to table contents.

Generally speaking, the generated answers either omit necessary details or incorporate extraneous context not requested by the question.

File 5 - 26037986

The **ROUGE scores** indicate a moderate level of overlap between the generated and target responses for some questions but low overlap for others.

Also the **BLEU scores** range from low to moderate, suggesting that just some generated answers have a reasonable level of precision.

The **Cosine Similarity** scores are generally moderate, presenting higher results for those questions related to the textual content of the document without involving tabular information.

Considering the fifth question as an example, shown in Figure 5.22, the generated answer provides information about *"living donors and grafts"*, which is not relevant to the question, leading to an incorrect response that does not align with the target answer.

Question: How many males and females patients are involved into the presented research and what data are detailed?
Target answer: According to the information shown in the document, there are 223 male patients (50%) and 223 female patients (50%) involved in the presented research. The data detailed include the number of patients according to gender, age, procedure category, consultant surgeon presence, and post-operative destination. Additionally, mortality rates at 30 days and 1 year for each category are highlighted. For instance, the 30-day mortality rate for males was 9%, and the 1-year mortality rate was 22%. For females, the 30-day mortality rate was 13%, and the 1-year mortality rate was 28%.

Generated answer: The research involves 43 living donors, out of which 23 were female and 20 were male. The median age of these donors was 30 years, ranging from 20 to 41 years old. The document also details data related to the grafts used in the study. All grafts from living donors consisted of segments 2 and 3 (left lateral liver lobe) with a median graft weight of 230 g (range 150-360). The grafts in the SLT group comprised 49 left lateral lobes (segments 2 and 3) with a median weight of 310 g (range 200–600). Eighteen percent of the grafts in both groups were small-for-size grafts, whereas significantly more large-for-size grafts were used in the SLT group. The median graft-recipient weight ratio was comparable in both groups: SLT = 3.8 (range 0.74– 9.2) and LRT = 3.2 (0.77-6.9)

Figure 5.22: QUERY 5 of File 5 from PubMedQA: Target vs. Generated Answers using Unstructured without Table Parsing

To sum up, the evaluation of generated answers across various documents highlights a combination of detailed data presentation and inaccuracies, especially in extracting broader contexts and specific details from the documents. While some answers, specifically those that do not include tabular information, align well with target responses, others lack details or present incorrect information.

5.1.6 Comparative Insights: Unstructured Parser with and without Table Parsing

Evaluating and comparing the results obtained with and without performing table parsing provides insights into the effectiveness of incorporating structured data analysis in understanding and summarizing research findings.

First of all, it is possible to take as example the first question and the generated answers from File 1:

> `QUERY:` What are the values associated to Intention to have a mammogram in 1996?
>
> `EXPECTED ANSWER:` The values associated to the intention to have a mammogram in 1996 are characterized by Percent on schedule of 72 and p-value equal to 0.001.
>
> `GENERATED ANSWER including table parsing:` The values associated with the intention to have a mammogram in 1996 are as follows: the percent on schedule for those who intended to get a mammogram, it was 77%, and for those who did not intend to, it was 40%; the associated p-value is .001.
>
> `GENERATED ANSWER excluding table parsing:` The values associated with the intention to have a mammogram in 1996 are as follows: - Percent on schedule (those who intended to get a mammogram): 77% - Percent off schedule (those who intended to get a mammogram): 23% - Total (those who intended to get a mammogram): 87% - Percent on schedule (those who did not intend to get a mammogram): 40% - Percent off schedule (those who did not intend to get a mammogram): 60% - Total (those who did not intend to get a mammogram): 13%.

The highest BLEU score is observed when table parsing is performed, suggesting a strong linguistic match. Moreover, Cosine Similarity scores are high, indicating a good semantic understanding. Without table parsing, the level of linguistic similarity is lower. In the example is noticeable that the second generated answer is not very informative and the content is not well reported. This means that table parsing enhances the accuracy and the semantic understanding of responses that involve specific data points.
Looking at File 2, one big difference can be seen in the fifth question, where, with table parsing, higher BLEU and ROUGE scores are obtained, indicating better linguistic and content alignment. Even if in this case the answers are quite good overall, the precision and linguistic similarity tend to be lower, suggesting that the inclusion of table parsing enhances the accuracy and relevance of the response.
Another relevant example is the second question from File 3:

> `QUERY:` What are Lea symbols and how are they used in visual acuity assessment?
>
> `EXPECTED ANSWER:` Lea symbols are used in visual acuity assessment and detection of amblyopia. They are a set of symbols that are used to assess visual acuity and detect amblyopia in clinical research. They have been the subject of various studies related to visual acuity assessment.
>
> `GENERATED ANSWER including table parsing:` Lea symbols are a set of specialized symbols used for visual acuity assessment and detection of amblyopia. These symbols, designed to be easily identifiable and distinguishable, are used in place of traditional letter or number charts, making them particularly useful in testing the visual acuity of young children or individuals who are unable to read. The assessment process involves the individual identifying or matching the symbols from a distance, allowing

the examiner to determine the smallest size of symbol that the individual can recognize. This information is then used to calculate the individual's visual acuity.

`GENERATED ANSWER excluding table parsing:` I'm sorry, but *the documents provided do not contain any information regarding Lea symbols or their use in visual acuity assessment*.

When table parsing processes are employed, the generated answer offers a comprehensive explanation that aligns with the target responses, even if it provides additional information. On the other hand, with the absence of table parsing, the system incorrectly assert that the document does not contain the required information.
An additional assessment of the relevance of table parsing processes can be seen in the third question:

`QUERY:` What is the MW LR associated to n = 9 and Visus > 0.1?

`EXPECTED ANSWER:` The value of MW LR related to Visus that ranges in the interval between 0.32 and 0.1 and n equal to 9 is 0.17 ± 0.3, as shown in the table 5 of the provided document.

`GENERATED ANSWER including table parsing:` For Visus > 0.1, the closest range we can refer to is 0.32 - 0.1, for which the MW LR is 0.17 ± 0.3

`GENERATED ANSWER excluding table parsing:` The MW LR associated with n = 9 and Visus > 0.1 is 0.56 ± 0.1.

In this case, the second answer provides an incorrect information, highlighting the challenges in extracting specific data points from the text without the structural knowledge provided by tables.
Looking at the results over the File 5, some responses diverge significantly in content. This shift in focus underlines the challenges in extracting specific data without the support of structured data analysis, resulting in less targeted

and comprehensive answers. One example of this behaviour can be seen in the question 3 from File 5:

`QUERY:` What information do we have about Post-operative destination and 30-day mortality for patients over 70 years?

`EXPECTED ANSWER:` The total number of patients included in the study was 190, which represents 43% of the study population. The 30-day mortality rate is equal to 21%, the 1-year mortality rate to 37%, and the average length of stay was 32 days. For patients who underwent resection without stoma formation, that are 67 representing 35% of the study population, the 30-day mortality rate is 21%, the 1-year mortality rate is 32%, and the average length of stay was 26 days. For patients who underwent resection with stoma formation, that are 37 patients representing 20% of the study population, the 30-day mortality rate is 24%, the 1-year mortality rate is 32%, and the average length of stay was 33 days.

`GENERATED ANSWER including table parsing:` The table includes the following data: The total number of patients included in the study was 190, the 43% of the study population. The 30-day mortality rate for this group was 21%, the 1-year mortality rate was 37%, and the average length of stay was 32 days. For patients who underwent resection without stoma formation (67 patients, 35% of the study population), the 30-day mortality rate was 21%, the 1-year mortality rate was 32%, and the average length of stay was 26 days. For patients who underwent resection with stoma formation (37 patients, 20% of the study population), the 30-day mortality rate was 24%, the 1-year mortality rate was 32%, and the average length of stay was 33 days.

`GENERATED ANSWER excluding table parsing:` According to the information for patients aged over 70 years, the 30-day and 1-year mortality rates vary depending on the ASA grade of the patient. Here are the details: - ASA grade I: 2 patients (1%) with 0% 30-day and 1-year mortality. - ASA grade II: 42 patients (22%) with 5% 30-day and 20%

> 1-year mortality. - ASA grade III: 84 patients (44%) with 20% 30-day and 36% 1-year mortality. - ASA grade IV: 60 patients (32%) with 32% 30-day and 52% 1-year mortality. - ASA grade V: 4 patients (2%) with 75% 30-day and 75% 1-year mortality. In the over 70 years old cohort, the formation of a stoma increases the length of stay with no statistical difference (p = 0.113). There is also no statistical difference in 30-day (p = 0.182) or 1-year (p = 0.068) mortality. Moreover, the length of stay is greater in those who get a stoma when compared to those who do not (33 vs. 26 days, p = 0.113).

The response diverges significantly in content, focusing instead on the association between mortality rates and ASA grades, rather than directly linking this information to post-operative destinations or specifically addressing the 30-day mortality for patients over 70 years, as required by the query.

The comparative analysis of document processing with and without table parsing clearly demonstrates the advantages of incorporating structured data analysis. Table parsing significantly improves the precision and relevance of the extracted information. It not only improves the alignment of generated responses with the expected content but also facilitates the extraction of specific details. Conversely, the absence of table parsing processes often results in responses that are less detailed and inaccurate, thereby limiting the system's ability to capture the essential semantic content.

5.1.7 Form Recognizer and Unstructured: Comprehensive Evaluation Across Parsing Scenarios

Based on the results obtained from analyzing five different documents from the PubMedQA dataset using two distinct parsers, with and without implementing table parsing, a comparison of the performance according to the computed metrics is feasible.

Table parsing emerges as a significant process in improving the accuracy and detail of the generated answers. For instance, in the first document, the implementation of table parsing with the Form Recognizer parser resulted in a more detailed extraction of numerical data and study outcomes. This detailed extraction of percentages and findings highlights the value of table parsing in contexts where precision and accuracy in numerical data is crucial.

The Form Recognizer parser, particularly when combined with proper table parsing procedures, demonstrated proficiency in extracting precise numerical information. Conversely, the Unstructured parser, while effective in general information extraction, showed limitations in its ability to extract specific numerical data without the proper analysis of tabular data.

The comparative analysis of the Form Recognizer and Unstructured parsers, especially in the context of table parsing, shows the relevance of advanced data extraction techniques to improve the accuracy of the retrieved information.

5.2 Analysing MBR Document

As with the PubMedQA documents, the evaluation of the MBR document is based on the provided metrics: ROUGE scores, BLEU score, and Cosine Similarity measures using Bag of Words, TF-IDF, Sentence Transformer, and Universal Sentence Encoder.

The parsing process for the MBR document is performed using both Unstructured and Form Recognizer parsers.

5.2.1 Form Recognizer Analysis: Impact of Accurate Table Parsing

The **ROUGE scores** exhibit variability across questions, generally ranging from moderate to low. This suggests that the generated answers may omit key details or not closely match the target answers in terms of wording.

The **BLEU scores** also show variation across questions. In several instances,

the BLEU scores are low, indicating discrepancies between the generated and target responses in terms of information accuracy and presentation.
The **Cosine Similarity** scores for Bag of Words and TF-IDF representations are generally moderate, pointing to a lack of semantic similarity. However, the scores using Sentence Transformer and Universal Sentence Encoder embeddings are higher, suggesting that despite some differences, the generated answers capture a portion of the semantic meaning present in the target ones.
Overall, the system shows a variable performance across different queries, with instances of relatively high semantic similarity and structural coherence.

5.2.2 Form Recognizer Analysis: Implications of Missing Table Parsing

The **ROUGE scores** across the questions reveal a general trend of low precision, recall, and F-measure values. This indicates that the generated answers have limited overlap with the target responses in terms of the exact words and sequences of words.
The **BLEU scores** are generally low across all queries, suggesting that the generated responses might not be structurally or lexically similar to the expected answers.
The **Cosine Similarity** scores significantly vary across different methods of representation. The Sentence Transformer and Universal Sentence Encoder embeddings generally provide higher similarity scores compared to the Bag of Words and TF-IDF representations. This suggest that when considering the overall meaning, the generated answers have some level of semantic alignment with the target responses. Despite that, the scores are generally quite low, suggesting that generated answers lack accuracy, details, and alignment with the target answers.
To sum up, the question answering system struggles to accurately retrieve the required information and often demonstrates a limited understanding of the

document content.

5.2.3 Comparative Insights: Form Recognizer with and without Table Parsing

The results obtained employing table parsing processes generally demonstrate improved precision and recall across most questions, as indicated by the performance of ROUGE scores. This improvement suggests that table parsing contributes to a better alignment between the generated and the target responses, likely due to the more structured extraction of information from tables. Furthermore, the BLEU scores are higher when implementing table parsing, as it may help in generating more coherent and structurally similar answers to the target ones, especially in cases where the required information is stored in tabular format within the document. The Cosine Similarity scores are also generally higher in scenarios where table parsing is applied. This indicates that table parsing enhances the semantic understanding of the document.

By effectively extracting and utilizing structured information from tables, the system can achieve a better understanding and comprehension of the contents. However, there is still room for further improvements, especially in capturing highly specific procedural details.

5.2.4 Unstructured Parser Analysis: Impact of Accurate Table Parsing

ROUGE scores show varying levels of precision, recall, and F-measure across the questions. The scores suggest that while there is some overlap between the generated and target answers, the level of similarity varies significantly. The scores are generally higher for ROUGE-1 compared to ROUGE-2, indicating that the generated responses are more likely to share individual words with the target answers than consecutive word pairs.

The **BLEU scores** are low across all the questions, suggesting that the generated answers may not closely match the structure or the exact wording of the target ones.

The **Cosine Similarity** measures, both in Bag of Words and TF-IDF representations, generally show a low to moderate level of similarity. This might be due to differences in word usage or the inclusion of additional or irrelevant information in the generated responses. On the other hand, similarity scores using Sentence Transformer and Universal Sentence Encoder embeddings show higher scores in some questions.

The higher scores in semantic similarity metrics indicate that the generated answers may still capture the essence or main points of the target answers to a certain extent.

5.2.5 Unstructured Parser Analysis: Implications of Missing Table Parsing

Across the questions, the **ROUGE scores** indicate varying level of overlap. For some of them, the scores are very low or even zero, implying that the generated responses might not capture all the necessary details or may diverge significantly from the expected content.

The **BLEU scores** are also generally low. This suggests that the generated responses may not be structuring information in the same way as the target ones or might be missing key terminology.

The **Cosine Similarity** scores are relatively low for most questions, indicating lack of alignment in the use of specific terms and phrases and difficulties in capturing the semantic meaning of the target answers.

To sum up, while there are instances where the semantic meaning of the answers aligns to some extent, as suggested by the higher scores from Sentence Transformer and Universal Sentence Encoder embeddings, the specific details, terminology, and structure often diverge significantly from the target answers.

5.2.6 Comparative Insights: Unstructured Parser with and without Table Parsing

The integration of table parsing processes generally leads to higher ROUGE scores across most questions, which suggests that it enables the extraction of more precise and relevant information. Similarly, BLEU scores tend to be higher when table parsing is implemented. This improvement highlights the importance of table parsing in accurately capturing details and terminologies that are relevant to the target answers. Additionally, the Cosine Similarity scores are also generally higher in scenarios where table parsing is applied. This demonstrates that the proper extraction of contents from tables not only improves the alignment in terms of specific wording but also enhances the semantic understanding of the document, enabling the generation of answers that are more semantically aligned with the target responses. Table parsing allows for better extraction and organization of specific details, which is particularly important for questions requiring detailed procedural steps, lists of items, or specific quantities.

While the current results obtained including the table parsing processes are quite good, there is still room for further refinement, especially in capturing highly specific procedural details or terminologies.

5.2.7 Comprehensive Evaluation: Form Recognizer vs. Unstructured Parser Across Parsing Scenarios

Based on the results obtained for the question answering process over the MBR document using both Form Recognizer and Unstructured parsers, an analysis and comparison of their performance has been conducted to determine whether including table parsing processes leads to improvement and which parser performs better.

The implementation of table parsing processes generally results in enhanced performance and metrics across both parsers, with an evident improvement in the scores for most questions. This demonstrates that correctly parsing table contents contributes positively to understanding and extracting relevant information from the document.

Between the two parsers, the Unstructured parser seems to perform better overall, even without including table parsing. This suggests that this parser might be better at dealing with non-tabular data or extracting information from the text without relying on structured table data. This is because when the Unstructured parser is unable to recreate the exact structure of tables within the document, it retains their textual content, thus avoiding the loss of information. However, the drawback of this approach is that it may result in storing misunderstood information or details that are not entirely accurate.

For the MBR document, which contains a mix of structured and unstructured data, the Unstructured parser, with the implementation of table parsing, provides a more comprehensive understanding and extraction capability.

5.3 Analysing SOP Documents

The evaluation of various SOP documents focuses on the analysis of *'question-answer'* pairs, which are assessed by human experts. This assessment aims to test the clarity and comprehensiveness of the SOP documents. These evaluations, graded on a scale from **A** (*Correct and Detailed*) to **NA** (*No Answer*), are designed to reflect the real-world application of SOP information retrieval. This includes particularly emphasizing the role of table parsing in enhancing interpretability and extraction of information.

Given that SOPs are in DOCX and DOC formats, parsing is carried out using the Unstructured parser, as the Form Recognizer does not support these file extensions. The analysis aims to provide insights into potential improvements in parsing SOPs, contributing to better-informed and compliant research practices.

File 1 - SOP01

Overall, the analysis of the first file suggests that answers generated by performing table parsing tend to provide more detailed and comprehensive information compared to answers obtained without the implementation of table parsing processes, which often achieve lower ratings. This highlights the value of table parsing in extracting and interpreting detailed information from documents, particularly in complex domains.

File 2 - SOP02

These evaluations underscore the importance of table parsing in extracting accurate knowledge from documents. The discrepancies between answers retrieved with and without performing table parsing processes highlight the potential for misinterpretation or lack of detail when critical data structuring tools are not utilized. This analysis further points out the need for meticulous attention to information stored within tables.

File 3 - SOP03

The performance of the document question answering system over the document parsed with table parsing processes highlights its ability to provide detailed, accurate answers by effectively navigating and interpreting the document's structured content. In contrast, the absence of table parsing processes seems to limit the specificity of the information that can be extracted, as evidenced by the *'Correct but Not Detailed'* or *'No Answer'* evaluations for the responses.

File 4 - SOP04

The insights derived from these results further underscore the significant advantages of implementing table parsing for document analysis, especially in contexts requiring the extraction of specific, detailed information from complex documents. Table parsing not only enhances the accuracy of the extracted information but also the depth and relevance of the responses, proving necessary for accurately interpreting and summarizing structured document content.

File 5 - SOP05

The results obtained highlight that while table parsing significantly enhances the detail and correctness of the responses, its absence can lead to partial, incorrect, or misinterpreted answers. This analysis showcases the need for information extraction techniques, such as table parsing, to ensure accurate, comprehensive, and relevant data retrieval from tricky documents.

File 6 - SOP06

The results underline that both methods can yield correct answers for broadly stated or straightforward questions but table parsing proves necessary for extracting specific, structured information, such as dataset variables or detailed procedural tasks. This analysis emphasizes the importance of advanced document parsing techniques in ensuring comprehensive data retrieval, particularly in contexts requiring detailed understanding of specialized content.

File 7 - SOP07

As mentioned before, table parsing enhances the ability to capture structured details, also in documents describing different processes and role assignments using tabular structures. This analysis emphasizes the importance of advanced document parsing techniques for ensuring accurate, detailed, and contextually appropriate data retrieval.

Throughout the analysis of different documents, the significant impact of table parsing on the accuracy, detail, and contextualization of extracted information is evident.

The evaluation of generated answers revealed key insights:

- **Enhanced Detail and Accuracy:** Table parsing consistently provides more detailed and accurate interpretations of complex documents, particularly in extracting structured information such as mapping between roles and performed tasks, processes and their steps, and specific technical requirements.

- **Contextual Understanding:** The ability to accurately interpret and summarize specific sections, variables, and procedural steps is notably improved with table parsing, underscoring its importance in navigating and understanding detailed document content.

- **Limitations Without Table Parsing:** Without table parsing, responses sometimes lack detail, are partially correct, or in some cases, completely misinterpret the document's content, making the system unable to provide the required information.

- **Consistency Across Documents:** Across various documents, table parsing consistently outperformed non-table parsing in delivering correct and detailed answers, demonstrating its utility in a wide range of document analysis tasks.

5.4 Summary

The discussion of results underscores the transformative impact of table parsing on the efficacy of document question answering systems across a diverse array of document formats. By enabling the systematic extraction of structured data from tables, this enhancement not only streamlines the retrieval process but also ensures a more comprehensive understanding of the content. These advancements are supported by the alignment of linguistic nuances and content specifics, alongside an enriched semantic comprehension.

Both Unstructured and Form Recognizer parsers gather substantial benefits from the incorporation of table parsing methodologies into their parsing procedures. This integration empowers these systems to achieve a higher level of precision in extracting complex numerical data and comprehensive content from different documents, thereby strengthening their overall utility and reliability in real-world applications.

Chapter 6

Discussion of Document Parsing Results

The effort to evaluate document parsing processes delves into the complex domain of data extraction and analysis. This exploration seeks to quantify the accuracy, reliability, and efficiency of information parsed from various document types, involving metrics such as parsing time, confidence scores, and detection class probabilities.

The understanding of these metrics is essential for selecting parsing solutions that meet specific use cases and optimizing workflows for system enhancements. Parsing time affects the data processing pipeline's efficiency, while confidence scores and detection class probabilities provide insights into the precision and reliability of the extracted information. The evaluation process also involves comparing various parsing tools and methods and evaluating their performance across different document formats.

This chapter aims to provide an examination of the results from various document parsing processes, comparing the Form Recognizer and the Unstructured parsers. It seeks to uncover the different performances across different document types and parsing tasks, highlighting the factors that influence document parsing processes.

6.1 Parsing Time

The **parsing time**, as already mentioned in one of the previous chapter, directly impacts the overall throughput of data processing pipeline. When comparing the time required for parsing documents using Form Recognizer and Unstructured, several insights emerge from both theoretical considerations and practical tests.

Form Recognizer is specifically optimized for extracting information from forms and structured documents, leveraging machine learning models that are fine-tuned for understanding specific document formats, such as PDFs. Conversely, Unstructured is capable of processing a wide range of document types and offers multiple strategies for processing. This approach prioritizes flexibility and adaptability over speed optimization for any single document type. However, this flexibility might not always lead to the fastest processing time, especially for documents that require a high level of detail or contain complex layouts. Form Recognizer, with its structured data extraction focus, likely employs specialized OCR and machine learning technologies optimized for quick and efficient parsing.

These theoretical considerations are confirmed by practical tests conducted. For the MBR document, Unstructured took *2 minutes* to parse, whereas Form Recognizer only took *20 seconds*, showcasing Form Recognizer's efficiency in processing structured PDF documents. In processing SOPs (all DOCX or DOC documents), Unstructured processed 7 files in *15 seconds*, demonstrating its capability and efficiency with Word documents, which might not require as complex processing as PDFs. For the PubMedQA PDF documents, Form Recognizer outperformed Unstructured, taking only *1 minute and 25 seconds* to process 5 files, compared to Unstructured's *4 minutes and 25 seconds*, further illustrating Form Recognizer's strength in handling PDF formats efficiently.

Alongside the time required for parsing documents using Form Recognizer and Unstructured, their efficiency can also be evaluated based on confidence scores and detection class probabilities, as discussed in the next section.

6.2 Confidence Scores and Detection Class Probabilities

Accurately extracting information from diverse document layouts and formats represents a significant challenge in the field of document parsing. The output from Form Recognizer is characterized by the *confidence score* values, whereas the output from Unstructured presents the *detection class probability* values, each of them associated with every extracted element during the parsing procedure. These metrics are crucial for evaluating the efficacy and effectiveness of the parsing process. Both the confidence score and detection class probability are numerical values, typically ranging from 0 to 1, that are assigned to each element extracted from the document. These values quantify the model's certainty related to the accuracy of the extracted data. A score closer to 1 indicates high confidence, while a score closer to 0 suggests lower confidence. Thus, they serve as indicators of the potential accuracy of the parsing procedure.

6.2.1 MBR Confidence Score and Detection Class Probability Evaluation

Figure 6.1 displays a histogram of the confidence scores obtained after parsing the MBR document using Form Recognizer as parser. These scores present a wide range of values, ranging from *0.08* to *0.95*, showcasing the variability in the model's certainty about the parsed elements. The average confidence score is below the midpoint of the range, suggesting that, on average, the model has a moderate level of certainty regarding its extracted data. This value is likely influenced by a substantial number of low-confidence scores, which suggests that the model encounters many challenges in parsing a large number of elements with high certainty. This consideration is further supported by the median confidence score of *0.36*, suggesting a skewed distribution with more low-confidence scores than high-confidence ones. The diversity in confidence scores from the MBR document parsing highlights the complexities inherent in automated document extraction, especially when the document has a very complex layout like the MBR.

Following the same reasoning, Figure 6.2 shows the detection class probabilities obtained from parsing with the Unstructured parser. The average detection probability of *0.63* suggests that, on average, the model has a fairly high level of confidence in its classifications. Additionally, the median detection probability of *0.66* suggests a skew towards higher confidence classifications.

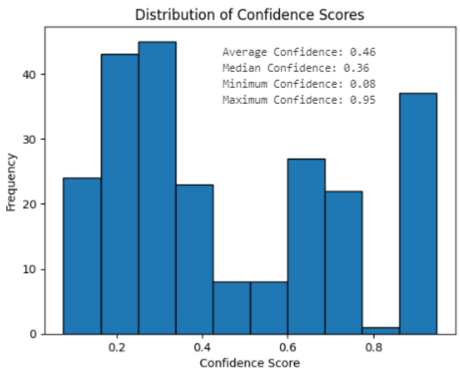


Figure 6.1: Form Recognizer: Confidence Scores Histogram for MBR document

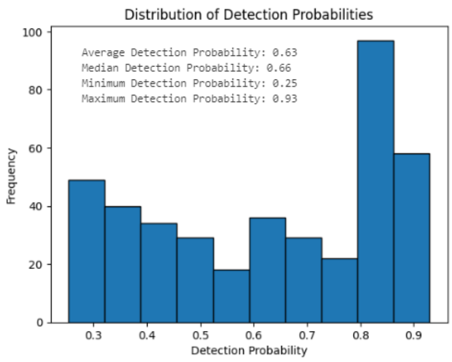


Figure 6.2: Unstructured: Detection Class Probabilities Histogram for MBR document

6.2.2 PubMedQA Confidence Score and Detection Class Probability Evaluation

Figure 6.3 displays a histogram of the confidence scores obtained after parsing the document with identifier **17205839** from the PubMedQA dataset using

Form Recognizer as the parser. The confidence scores present a range of values, from a minimum of *0.07* to a maximum of *0.83*, indicating variability in the model's certainty about the extracted elements. The median confidence score is close to the average (*0.42*), suggesting a relatively balanced distribution of confidence scores but still points to a moderate level of overall certainty.

Figure 6.4 displays a histogram of the confidence scores obtained after parsing the document with identifier **11729377** from the PubMedQA dataset using Form Recognizer as the parser. The variation of scores suggests a mixed degree of clarity in the document's contents as interpreted by the parser. The presence of several scores below *0.3* highlights specific challenges with certain elements.

Figure 6.5 displays a histogram of the confidence scores obtained after parsing the document with identifier **26037986** from the PubMedQA dataset using Form Recognizer as the parser. The median confidence score (*0.27*), being closer to the lower end of the range, emphasizes that a significant number of extractions are on the lower side of the confidence.

Figure 6.6 displays a histogram of the confidence scores obtained after parsing the document with identifier **16418930** from the PubMedQA dataset using Form Recognizer as the parser. The scores indicate that the parser encountered both easily recognizable elements and more challenging ones. This varied difficulty of the document's content is also reflected in the average confidence score (*0.48*), which indicates a mix of both high and low-confidence extractions.

Figure 6.7 displays a histogram of the confidence scores obtained after parsing the document with identifier **10808977** from the PubMedQA dataset using Form Recognizer as the parser. In this case, there is just a small set of values so it is challenging to draw comprehensive conclusions about the overall quality of the process for this specific document.

Following the same approach, it is possible to make some considerations about the detection class probabilities obtained when parsing with Unstructured.
Figure 6.8 displays a histogram of the detection class probabilities obtained after parsing the document with identifier **17208539** from the PubMedQA dataset using Unstructured as the parser. The high median value points to a significant portion of elements being classified with very high confidence, underscoring the model's precision and reliability. According also to the average detection probability, the model is effectively recognizing and classifying the document's content with a high degree of certainty.
Figure 6.9 displays a histogram of the detection class probabilities obtained after parsing the document with identifier **11729377** from the PubMedQA dataset using Unstructured as the parser. The average detection probability of *0.88* indicates that, on average, the model is highly confident in its classifications. Additionally, the median detection probability of *0.93* is even higher than the average, indicating that the majority of the elements are classified with very high confidence.
Figure 6.10 displays a histogram of the detection class probabilities obtained after parsing the document with identifier **26037986** from the PubMedQA dataset using Unstructured as the parser. The average detection probability of *0.86* indicates strong performance in recognizing and classifying the elements within the document. As in the previous case, the median detection probability of *0.92* is higher than the average.
Figure 6.11 displays a histogram of the detection class probabilities obtained after parsing the document with identifier **16418930** from the PubMedQA dataset using Unstructured as the parser. The average and median detection probabilities indicate that the model demonstrates a strong confidence in its classifications across the document, with the majority of elements being classified with very high confidence.
Figure 6.12 displays a histogram of the detection class probabilities obtained after parsing the document with identifier **10808977** from the PubMedQA

dataset using Unstructured as the parser. The results suggest that the model performed well in recognizing and classifying the elements within the document, following the same trends described before.

6.2.3 SOPs Detection Class Probability Evaluation

Since SOP documents are parsed using Unstructured, the evaluation of the parsing performance is made just by looking at the detection class probabilities.

Figure 6.13 displays a histogram of the detection class probabilities obtained after parsing the document **SOP01** using Unstructured as the parser. The probabilities reflect a mix of moderate to very high confidence in the model's classifications. The median detection probability suggests that the majority of the document's elements are classified with a confidence level above *0.76*. This analysis points to a robust performance by the model.

Figure 6.14 displays a histogram of the detection class probabilities obtained after parsing the document **SOP02** using Unstructured as the parser. The presence of lower probabilities alongside higher ones reflects the model's variable performance, possibly due to differences in document quality, content complexity, or specific challenges of certain sections of the text.

Figure 6.15 displays a histogram of the detection class probabilities obtained after parsing the document **SOP03** using Unstructured as the parser. The histogram shows a moderate to high confidence level in the model's classification. The average detection probability is *0.67*, indicating a generally solid level of confidence across the document's elements, while the median detection probability at *0.68* suggests a fairly consistent confidence level throughout. The presence of both lower-end probabilities and very high probabilities indicates variability in the model's certainty across different document sections, possibly reflecting variations in document clarity, complexity, or content specificity.

Figure 6.16 displays a histogram of the detection class probabilities obtained after parsing the document **SOP04** using Unstructured as the parser. The detection class probabilities exhibit a notably high level of confidence across the board, with values ranging from *0.56* to *0.94*. The median detection probability is very high (*0.93*), underscoring that the majority of elements are classified with very high confidence.

Figure 6.17 displays a histogram of the detection class probabilities obtained after parsing the document **SOP05** using Unstructured as the parser. The average detection probability stands at *0.70*, indicating a generally good level of model confidence in its classifications, while the median probability of *0.73* suggests that more than half of the elements were classified with a confidence level above this threshold.

Figure 6.18 displays a histogram of the detection class probabilities obtained after parsing the document **SOP06** using Unstructured as the parser. The average detection probability is *0.82*, reflecting a robust overall confidence across the document's elements. The presence of a few lower probabilities suggests certain areas where the model faces challenges. However, the predominance of high probabilities indicates a generally effective and reliable classification capability by the model.

Figure 6.19 displays a histogram of the detection class probabilities obtained after parsing the document **SOP07** using Unstructured as the parser. The detection class probabilities range from a low of *0.25* to a high of *0.92*, indicating a spectrum of confidence levels in the model's classifications. The average detection probability is *0.62*, which suggests a moderate overall confidence across the document's elements. The median detection probability at *0.68* points to a slightly higher confidence level for more than half of the elements, suggesting variability in the model's certainty with certain sections of the document being classified with more confidence than others.

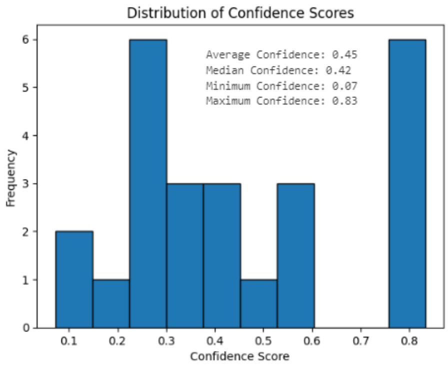


Figure 6.3: Form Recognizer: Confidence Score Histogram for PubMedQA Doc 17205839

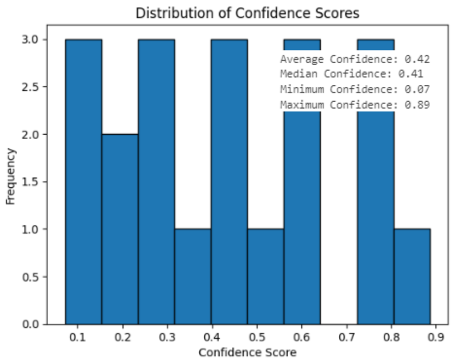


Figure 6.4: Form Recognizer: Confidence Score Histogram for PubMedQA Doc 11729377

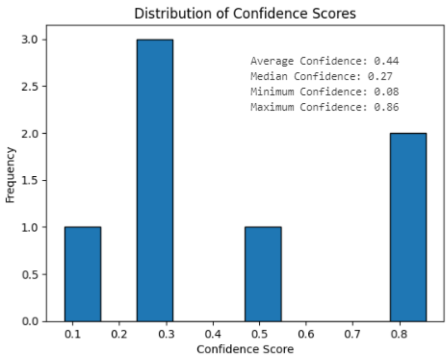


Figure 6.5: Form Recognizer: Confidence Score Histogram for PubMedQA Doc 26037986

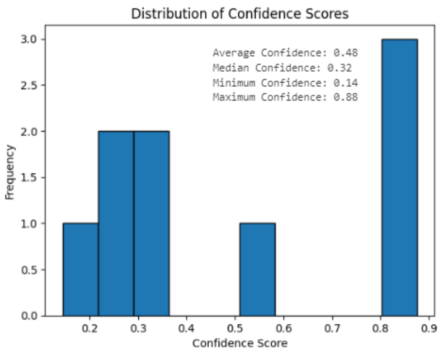


Figure 6.6: Form Recognizer: Confidence Score Histogram for PubMedQA Doc 16418930

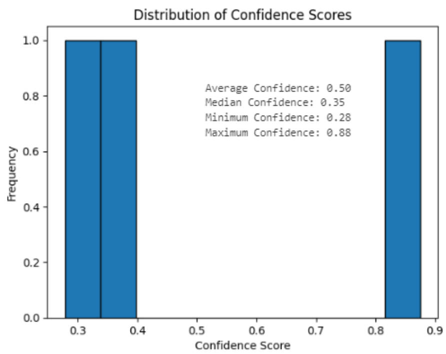


Figure 6.7: Form Recognizer: Confidence Score Histogram for PubMedQA Doc 10808977

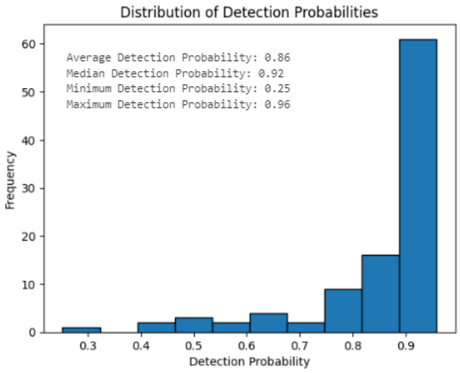


Figure 6.8: Unstructured: Detection Class Probabilities Histogram for PubMedQA Doc 17205839

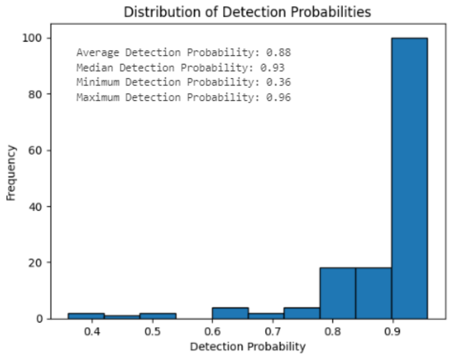


Figure 6.9: Unstructured: Detection Class Probabilities Histogram for PubMedQA Doc 11729377

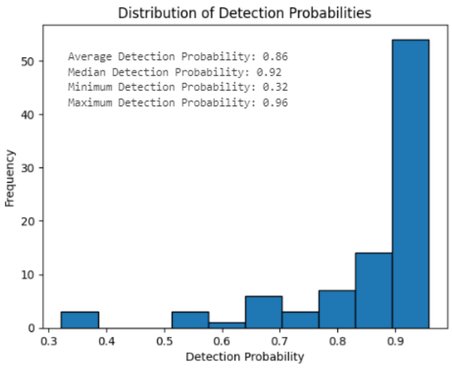


Figure 6.10: Unstructured: Detection Class Probabilities Histogram for PubMedQA Doc 26037986

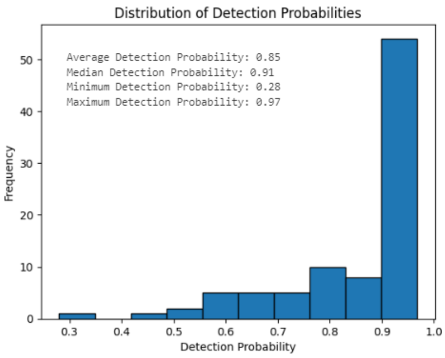


Figure 6.11: Unstructured: Detection Class Probabilities Histogram for PubMedQA Doc 16418930

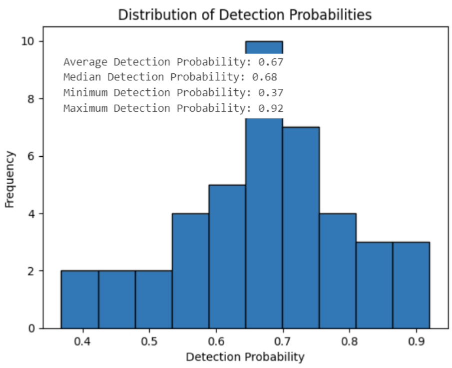


Figure 6.12: Unstructured: Detection Class Probabilities Histogram for PubMedQA Doc 10808977

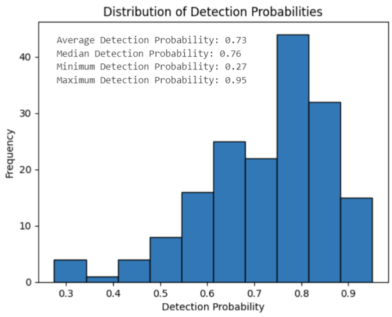


Figure 6.13: Unstructured: Detection Class Probabilities Histogram for SOP01

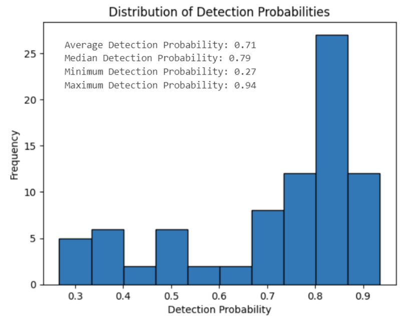


Figure 6.14: Unstructured: Detection Class Probabilities Histogram for SOP02

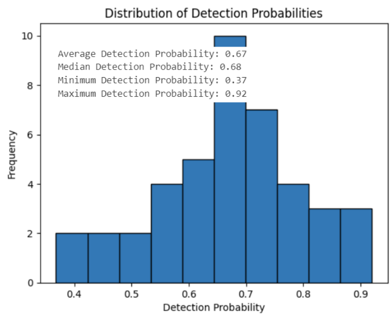


Figure 6.15: Unstructured: Detection Class Probabilities Histogram for SOP03

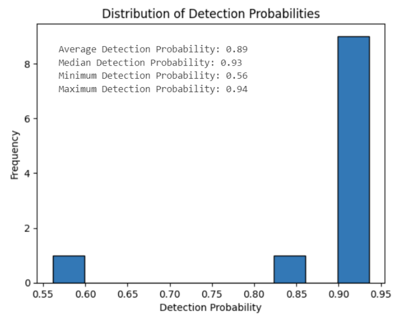


Figure 6.16: Unstructured: Detection Class Probabilities Histogram for SOP04

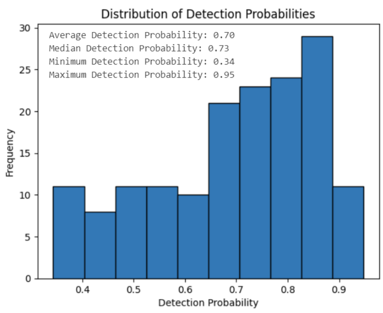


Figure 6.17: Unstructured: Detection Class Probabilities Histogram for SOP05

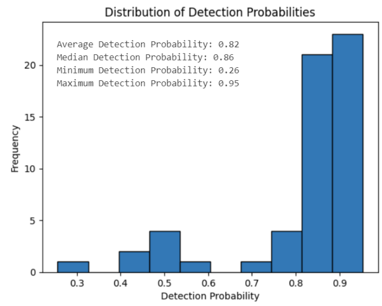


Figure 6.18: Unstructured: Detection Class Probabilities Histogram for SOP06

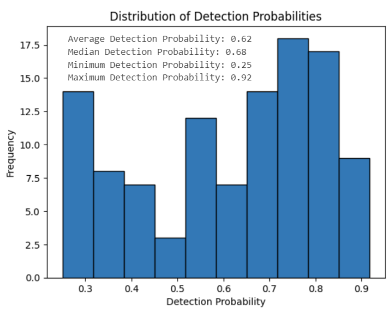


Figure 6.19: Unstructured: Detection Class Probabilities Histogram for SOP07

6.3 Summary

Assessing parsing results is a detailed and essential process to guarantee the accuracy and dependability of extracted information, which is necessary in the domains of data extraction and analysis. This discussion is based on various metrics, such as parsing time, confidence scores, and detection class probabilities.

The evaluation of document parsing results underscores the importance of leveraging comprehensive metrics to assess parsing performance. It facilitates informed decision-making regarding the selection of parsing solutions that best fit specific use cases, optimizing workflows and advancing system capabilities.

Chapter 7

Conclusion

The dissertation delves into the intricate topics of document parsing and question answering, focusing on the impact of table parsing. Through a series of experiments and analyses across various document types and parsing methods, this study aims to uncover the strengths and challenges of the methods and approaches implemented for these experiments and research, setting the stage for future developments. The integration of table parsing has notably enhanced the accuracy of document question answering, emphasizing its essential role in complex document analysis. Challenges in replicating the nuances of human language highlight the need for ongoing refinement in semantic analysis techniques.

The chapter outlines the key findings and contributions of the research, emphasizing the progress and future possibilities in document analysis technologies. The dissertation not only aims to analyze and evaluate the document parsing and question answering processes but also lays the groundwork for potential advancements and developments to improve the capabilities and precision of these systems.

7.1 Key Findings and Contributions

One of the important findings is the marked improvement in document question answering accuracy when table parsing is incorporated into the document analysis process. This enhancement is particularly evident in documents where structured data plays a crucial role in providing information, such as MBR and SOP documents. The ability to accurately parse and interpret table contents significantly enriches the generated responses, making it a key component for any document question answering system dealing with similarly structured documents.

While the systems showed the ability to capture the semantic essence of the target answers, achieving high lexical and structural similarity remains a challenge. This discrepancy highlights the complexity of understanding and reproducing human language nuances, emphasizing the need for further improvements in semantic analysis techniques within document question answering systems.

An additional key finding pertains to the challenges associated with table parsing in documents such as MBRs and SOPs. The research identifies some difficulties such as inconsistent table representation and complex headers, which significantly impact the parsing accuracy. Addressing these challenges, it is possible to propose some guidelines aimed at improving table structure and readability, thereby enhancing parsing efficiency. The guidelines not only contribute to the field by providing a roadmap for better document design but also underscore the nuanced complexities involved in parsing processes.

7.1.1 Impact of Table Parsing on Document Parsing Systems

The accuracy of table parsing significantly influences the overall performance of document parsing systems. Precise table extraction enhances the integrity and utility of the parsed document content by ensuring that critical structured data is accurately captured and interpreted. The experiments demonstrate that documents with complex tabular data, such as MBRs and SOPs, significantly benefit from table parsing. The comparative analysis shows that systems incorporating table parsing techniques handle complex tabular data more effectively, leading to more accurate and reliable parsed documents.

7.1.2 Improvements in Question Answering Results through Optimized Table Parsing

Integrating optimized table parsing techniques leads to substantial improvements in question answering results. The inclusion of accurate table parsing allows the system to extract and utilize structured data more effectively, thereby enhancing the overall quality of the answers. This is particularly evident in the improved performance metrics (ROUGE, BLEU, and Cosine Similarity scores) observed in the experiments. The ability to accurately interpret table contents enriches the responses, providing more detailed and contextually relevant answers. This demonstrates that optimized table parsing is a critical factor in achieving high-quality question answering results.

7.1.3 Comparative Analysis of Parsing Tools and Models

The comparative analysis of various parsing tools and models reveals distinct strengths and limitations. Form Recognizer demonstrated superior efficiency in parsing structured PDF documents, benefiting from its optimization for such formats. On the other hand, Unstructured showcased remarkable flexibility, handling a wide range of document formats with a good level of efficiency.

Form Recognizer stands out as the best choice for parsing PubMedQA documents, excelling in extracting detailed numerical data and study outcomes crucial for biomedical queries, especially when table parsing processes are implemented. In contrast, Unstructured is competent in general information extraction and relies on table parsing for enhanced effectiveness. Table parsing significantly improves precision and relevance in extracting detailed information from PubMedQA and MBR documents, ensuring compliance and consistency in production contexts.

This analysis underscores the importance of selecting the appropriate parsing tool based on the specific requirements of the document type and the desired outcome of the parsing process. The choice of parser can significantly influence the accuracy and efficiency of handling tabular data, making it a crucial consideration in system design.

The results presented in Table 7.1 and Table 7.2 demonstrate the substantial improvements achieved through the implementation of table parsing processes first in the document parsing and, consequently, in the question answering tasks. It is important to note that these results presented in the tables below are based on question answering over PubMedQA documents and MBR document, as SOP documents are evaluated using Human Evaluation.

Metric	Threshold	Unstructured without Table Parsing	Unstructured with Table Parsing	Form Recognizer without Table Parsing	Form Recognizer with Table Parsing
BLEU Score	10	50%	**80%**	20%	**40%**
ROUGE-1	0.7	50%	**70%**	20%	**40%**
ROUGE-2	0.4	45%	**65%**	15%	**35%**
ROUGE-L	0.6	55%	**75%**	25%	**45%**
Cosine Similarity (BoW)	0.85	70%	**80%**	70%	**80%**
Cosine Similarity (TF-IDF)	0.78	70%	**80%**	70%	**80%**
Cosine Similarity (Sentence Transformer)	0.85	70%	**80%**	70%	**80%**
Cosine Similarity (Universal Sentence Encoder)	0.75	60%	**70%**	60%	**70%**

Table 7.1: Percentage of Queries with Scores above Thresholds With and Without Table Parsing

Table 7.1 illustrates the percentage of queries that surpassed a selected threshold for each involved metric, comparing four setups: Unstructured without table parsing vs. Unstructured with table parsing; Form Recognizer without table parsing vs. Form Recognizer with table parsing. The thresholds are selected based on the analysis of typical performance ranges observed in question answering tasks, ensuring they represent a high level of performance.

For instance, when considering a BLEU score threshold of 10, the Unstructured parser improves from 50% to 80% with the use of table parsing, while Form Recognizer sees an improvement from 20% to 40%. Similar improvements are seen across all other metrics, such as ROUGE, and various Cosine Similarity measures.

The results clearly show that the inclusion of table parsing significantly improves the performance across all metrics. These results confirm the hypothesis that table parsing strengthens both syntactic and semantic accuracy in the responses generated during the question answering process over different documents.

Table 7.2 highlights the number of queries that scored 0 for each metric across the four different setups. The data shows a significant reduction in the number of queries with zero scores when table parsing is implemented. This reduction in zero-score queries suggests that table parsing helps in capturing more relevant and pertinent information, thereby enhancing the overall performance and robustness of the document parsing and question answering systems.

Metric	Unstructured without Table Parsing	Unstructured with Table Parsing	Form Recognizer without Table Parsing	Form Recognizer with Table Parsing
BLEU Score	5	**2**	10	**7**
ROUGE-1	4	**1**	9	**6**
ROUGE-2	6	**3**	12	**8**
ROUGE-L	5	**2**	11	**7**
Cosine Similarity (with all representations)	0	**0**	0	**0**

Table 7.2: Number of Zero-Score Queries With and Without Table Parsing

For example, for the BLEU Score metric, the number of zero-score queries decreases from 5 to 2 for Unstructured and from 10 to 7 for Form Recognizer when table parsing is incorporated. This trend is also observed across the ROUGE and Cosine Similarity metrics.

The Cosine Similarity metrics, which include Bag of Words, TF-IDF, Sentence

Transformer, and Universal Sentence Encoder representations, are designed to measure the semantic similarity between the target answers and the generated answers. Unlike other metrics such as BLEU or ROUGE, which can return a zero score if there's no overlap in terms of n-grams between the reference and the candidate text, Cosine Similarity metrics are less likely to produce a zero score unless the compared texts are completely dissimilar or orthogonal in the vector space.
In this context, the fact that there are no zero-scoring queries for Cosine Similarity metrics suggests that the responses generated by the system, even without table parsing, still bear some semantic similarity to the reference text. However, it should be noted that while these metrics don't produce zero scores, the implementation of table parsing still improves the performance, indicating that the generated answers become more semantically aligned with the target answers.

To further explore the intricacies of document parsing, particularly in handling structured data within documents like MBRs and SOPs, the subsequent section delves into the specific challenges and guidelines for effective table parsing.

7.2 Table Parsing: Challenges and Guidelines

The parsing of tables within documents, such as MBR and SOP documents, presents several challenges. These challenges, which include inconsistent table representation, complex headers, multi-line rows, and embedded instructions, require specialized and improved parsing techniques capable of handling varied layouts and accurately extracting and interpreting the intended information. This section provides guidelines to facilitate the parsing process and to improve the creation of tables within these documents, aiming to enhance the accuracy and efficiency of future document parsing systems.

7.2.1 Challenges for MBR and SOP documents

The layout and structure of MBR and SOP documents present unique parsing challenges, particularly when it comes to tables. These challenges include:

- **Inconsistent Representation of Tables**: Tables are represented in different styles across documents, requiring varied parsing strategies.

- **Complex Headers and Multi-line Rows**: Tables often include complex headers and rows that span multiple lines, making the parsing process even more challenging.

- **Embedded Instructions and Conditions**: Tables often contain instructions and conditions in addition to values, requiring context-aware parsing.

- **Identification of Relevant Data**: Distinguishing between table data and separate elements (e.g., headers or footnotes) can be challenging, especially when non-table text is interspersed with table data.

7.2.2 Guidelines

To facilitate table parsing within MBR and SOP documents, the following guidelines can be applied:

- **Consistent Table Structure**: Maintain a consistent table structure across all documents to facilitate identification and parsing of table elements.

- **Clear Delimiters**: Use clear delimiters between different data points within the same row to distinguish between different data points.

- **Simplified Headers and Single-line Rows**: Avoid multi-line rows and complex headers to reduce parsing complexity.

- **Separate Instructions and Conditions**: Avoid embedding instructions or conditions within table cells. Instead, add them as footnotes or in a separate section.

- **Delineate Tables with Borders**: Use borders to visually separate tables from other elements in the document, helping parsing algorithms identify the start and end of the table.

These guidelines provide a guide for creating documents that are easier to parse and analyze, while still conveying all necessary information. Understanding these parsing challenges underscores the need for strategic guidelines to optimize table parsing within MBR and SOP documents. These guidelines not only address current parsing complexities but also lay a foundation for advancing document parsing technologies in various applications.

7.3 Implications for Future Research and Development

The insights acquired from this analysis have several implications for the future development of document question answering and document parsing tasks.

Given the significant impact of table parsing on system accuracy, future research should prioritize the implementation and development of advanced table parsing techniques. This could involve exploring machine learning models specifically designed to understand and interpret complex table structures, thereby enhancing the system's ability to extract and utilize structured data effectively and efficiently. In this context, the guidelines suggested for enhancing table creation within MBR and SOP documents could serve as a valuable reference, preparing for the development of documents that are more compatible with advanced parsing techniques.

It is also important to highlight the necessity of tailoring parser selection to the document's specific needs. Future developments could focus on creating more versatile parsing tools that combine the strengths of existing parsers, offering both the efficiency in handling structured documents seen in Form Recognizer and the flexibility demonstrated by Unstructured.

Improving the semantic understanding capabilities of document question answering systems is essential for bridging the gap between generated responses and target answers. Future efforts could explore the integration of more sophisticated and advanced natural language processing models and techniques, such as transformer-based architectures, to better capture and reproduce the nuances of human language.

7.4 Concluding Remarks

This dissertation has highlighted the significant role of table parsing in enhancing document parsing and question answering processes. By delving into the challenges of table parsing within specific documents such as MBRs and SOPs, and offering targeted guidelines to overcome these challenges, this work contributes to the advancement of document analysis technologies.

The identification of table parsing challenges and the establishment of guidelines not only improves parsing accuracy but also set the stage for future innovations. These insights encourage the development of parsing algorithms tailored to the complexities of table structures, promising more robust and effective document question answering systems.

In conclusion, the findings underscore the crucial importance of table parsing in the broader context of information retrieval and analysis. The progress made in this dissertation sets the stage for future research to further enhance the capabilities of document analysis systems, leveraging structured data to improve information accuracy and relevance. Looking to future developments, continued research and advancements in these areas will be essential to unlock the full power of unstructured data and to implement more intelligent, efficient, and accurate knowledge management systems.

Bibliography

[1] Azure AI Document Intelligence Documentation. URL: https://learn.microsoft.com/en-us/azure/ai-services/document-intelligence/?view=doc-intel-4.0.0. Accessed: February, 2024.

[2] R. Becker and M. Gräf. Landolt-Ring- vs. Snellen-E-Visus: Unterschiede bei Schielamblyopie? [Landolt C and snellen e acuity: differences in strabismus amblyopia?] *Klinische Monatsblätter für Augenheilkunde*, 223(1):24–28, 2006. DOI: 10.1055/s-2005-858613.

[3] D. Cer, Y. Yang, S.-y. Kong, N. Hua, N. Limtiaco, R. S. John, N. Constant, M. Guajardo-Cespedes, S. Yuan, C. Tar, Y.-H. Sung, B. Strope, and R. Kurzweil. Universal Sentence Encoder, 2018. arXiv: 1803.11175 [cs.CL].

[4] D. Cer, Y. Yang, S.-y. Kong, N. Hua, N. Limtiaco, R. John, N. Constant, M. Guajardo-Céspedes, S. Yuan, C. Tar, B. Sung, B. Strope, and R. Kurzweil. TF: Universal Sentence Encoder. https://tfhub.dev/google/universal-sentence-encoder/4, 2018.

[5] N. Ceta. Standard Operating Procedure (SOP). 2023. URL: https://workflowautomation.net/blog/standard-operating-procedure-sop. Accessed: March, 2024.

[6] S. Dhuria. Natural Language Processing: An approach to Parsing and Semantic Analysis, 2015. URL: https://ijniet.org/wp-content/uploads/2015/07/7.pdf.

[7] Q. Jin, B. Dhingra, Z. Liu, W. W. Cohen, and X. Lu. PubMedQA: A Dataset for Biomedical Research Question Answering, 2019. arXiv: 1909.06146 [cs.CL].

[8] B. Kaller. Understanding Embeddings in Natural Language Processing. 2023. URL: https://medium.com/@briankworld/understanding-embeddings-in-natural-language-processing-23506f4a150b. Accessed: March, 2024.

[9] F. Karabiber. TF-IDF — Term Frequency-Inverse Document Frequency. n.d. URL: https://www.learndatasci.com/glossary/tf-idf-term-frequency-inverse-document-frequency/. Accessed: April, 2024.

[10] D. Khurana, A. Koli, K. Khatter, and S. Singh. Natural language processing: state of the art, current trends and challenges. *Multimedia Tools and Applications*, 82(3):3713–3744, July 2022. ISSN: 1573-7721. DOI: 10.1007/s11042-022-13428-4. URL: http://dx.doi.org/10.1007/s11042-022-13428-4.

[11] V. P. Kuruvilla. A Complete Guide to Document Parsing in 2024. 2024. URL: https://nanonets.com/blog/document-parsing/. Accessed: March, 2024.

[12] P. Lemay. Master Batch Record: A Guide to Pharma Compliance Records. 2022. URL: https://tulip.co/blog/master-batch-record-a-guide-to-pharma-compliance-records/. Accessed: March, 2024.

[13] C.-Y. Lin. ROUGE: A Package for Automatic Evaluation of Summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics, July 2004. URL: https://aclanthology.org/W04-1013.

[14] D. Owczarek. Generative Question Answering with Large Language Models. 2023. URL: https://nexocode.com/blog/posts/generative-question-answering-llms/. Accessed: April, 2024.

[15] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. Bleu: a Method for Automatic Evaluation of Machine Translation. In P. Isabelle, E. Charniak, and D. Lin, editors, *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics, July 2002. DOI: 10.3115/1073083.1073135. URL: https://aclanthology.org/P02-1040.

[16] Qdrant Documentation. 2021. URL: https://qdrant.tech/documentation/. Accessed: February, 2024.

[17] S. Raja, A. Mondal, and C. V. Jawahar. Visual Understanding of Complex Table Structures from Document Images. *CoRR*, abs/2111.07129, 2021. arXiv: 2111.07129. URL: https://arxiv.org/abs/2111.07129.

[18] M. I. El-Sawaf, R. A. Drongowski, J. N. Chamberlain, A. G. Coran, and D. H. Teitelbaum. Are the long-term results of the transanal pull-through equal to those of the transabdominal pull-through? A comparison of the 2 approaches for Hirschsprung disease. *Journal of Pediatric Surgery*, 42(1):41–47, 2007. DOI: 10.1016/j.jpedsurg.2006.09.007.

[19] J. Schäfer. Challenges of Workflow Automation with Document Parsing. 2021. URL: https://konfuzio.com/en/document-parsing/#6-herausforderungen-einer-workflow-automatisierung-mit-document-parsing. Accessed: March, 2024.

[20] J. Schäfer. Document Parsing - How to extract valuable data from documents. 2024. URL: https://konfuzio.com/en/document-parsing/. Accessed: March, 2024.

[21] P. Sitikhu, K. Pahi, P. Thapa, and S. Shakya. A Comparison of Semantic Similarity Methods for Maximum Human Interpretability, 2019. arXiv: 1910.09129 [cs.IR].

[22] Unstructured Documentation. URL: https://docs.unstructured.io/welcome. Accessed: February, 2024.

Acknowledgements

I wish to express my deep gratitude to all those who have supported and guided me through the journey of completing this dissertation.
I am particularly thankful to my supervisor, Prof. Paolo Torroni, for his invaluable guidance, insight, and encouragement. His advice and feedback have been essential in shaping the direction and enhancing the quality of this work.
My appreciation extends to my co-supervisor, Nicolò Donati, for his support and dedication.

This work would not have been achievable without the unique opportunity provided by Bayer. I extend my sincere appreciation to the company for allowing me to undertake my internship there, offering a stimulating and supportive work environment.
I wish to acknowledge the exceptional team at Bayer, with whom I had the privilege of working. The collaboration, support, and professionalism from this team have greatly enriched my experience.
A heartfelt thank you goes to my mentors at Bayer, Christoph Alt and Sebastian Arnold. Their guidance has been transformative, providing me with many ideas and significantly enriching my knowledge. Moreover, they have supported my research and imparted to me the invaluable skill of collaborating effectively within a team. Their professional expertise, insightful suggestions, and support have been instrumental in the successful completion of this work.

My deepest gratitude goes to everyone for their invaluable contributions.