December 16, 2023
Multimodal Natural Language Inference over Legal Agreements

Long Legal Document Entity Extraction with Hierarchical-based Representation

Capstone Project Final Report

Ning Kang (nk3024), Xindi Deng (xd2287), Yihan Chen (yc4170),
Yuhui Wang (yw3937), Yuxin Lin (yl5129)
Mentor: Vladimir A. Kobzar

1 Introduction

1.1 Problem Statement

Owing to the intricate and lengthy nature, the reading comprehension of legal documents raises great challenges for both human and AI models. Despite significant advancements in AI, its application in law remains limited, particularly the analysis of the Merger and Acquisition Agreements (M&A agreements), which often requires extensive time and expertise for comprehensive understanding.

1.2 Motivation

Our project seeks to address these limitations by developing a model tailored to help the comprehension of M&A agreements. Leveraging the structured dataset of MarkupMnA, where contracts are annotated with structured highlights and systematically broken into manageable sections, we aim to create a streamlined approach for understanding and extracting critical deal point text.

1.3 Overall Approach

Our primary focus was on enhancing the model's comprehension of lengthy legal documents through the implementation of section-based segmentations.

Our initiatives commenced with an exhaustive exploration of the MarkupMnA and MAUD datasets, offering invaluable insights into their structural intricacies. We developed a data visualization tool that presents hierarchical information from agreements in a browser. This tool not only facilitates a comprehensive understanding of the dataset's key structural components and addresses corner cases but also lays the foundation for the efficient utilization of hierarchical information in the construction of further training pipelines.

We introduced two methods for integrating hierarchical information: section-based sliding windows and section-based training sets. In the first method, we maintained the input dataset structure while adjusting the sliding window construct to capture text spans only from the same section. The second method focused on providing the model with pre-segmented agreements. Both methods surpassed the benchmark, confirming the viability of incorporating hierarchical information in the extraction and comprehension of long texts.

Our project signifies a substantial advancement in unlocking the potential of AI in the legal domain, particularly within the nuanced context of agreements. This work not only contributes to the evolution of AI models in the legal sector but also establishes a crucial framework for enhancing the comprehension of complex documents, addressing a pertinent need in the legal and business realms.

2 Background

2.1 Legal Domain Downstream Task

Integrating AI models in the legal sector faces considerable challenges, particularly owing to the intricate and extensive nature of legal documents. While the intersection of legal domain and NLP is still in development, we could identify some relevant previous works in document segmentation and the long legal text that we could build upon.

Rissanen Data Analysis framework [1] could evaluate the utility of segmentation in various downstream tasks, including generation of sub questions before answering questions, analysis of rationales and explanations, and investigation of the importance of different parts of speech.

Lawformer [2] is a pretrained model targeting long legal documents in Chinese. While the model’s primary focus lies in criminal and civil cases, differing from our emphasis on M&A agreements, the model gives insights into both dealing with long documents and formalizing downstream tasks in the legal domain. To overcome the problem of input length limitation of mainstream PLMs (BERT, RoBERTa, etc), Lawformer employs a combination of three types of attention mechanism: sliding window attention, dilated sliding window attention, and the global attention. The paper then further evaluated Lawformer on 4 downstream tasks: legal judgment prediction, similar case retrieval, legal reading comprehension, and legal question answering.

2.2 MAUD

Wang, Scardigli, et al.(2023) [3] proposed there are two main downstream tasks in the long legal documents comprehension: entity extraction and reading comprehension. The reading comprehension task, which is the paper's main focus, refers to interpreting the meaning of the key legal clauses (deal point). To achieve this goal, the model is expected to predict the correct answer to the multiple choice reading comprehension questions based on the key texts of the legal contracts. The paper introduced a dataset called Merger Agreement Understanding Dataset (MAUD) for training, which consists of 152 English M&A agreement contracts and 47,457 annotations by law students and experienced lawyers working for over 10,000 hours. The paper also assessed the performance of several NLP benchmark models in the area of legal text understanding with MAUD.

The entity extraction task, on the other hand, is to extract the key clauses used for the reading comprehension task from the full contract. The task is introduced in the Appendix A.12 of the paper. For the extraction task, the authors proposed another dataset of the 152 contracts in MAUD in a different format, which we will introduce detailedly in a later section, and formalized the task again as a Q&A structure. The paper classified the key clauses into 22 deal point types. The Extractive Q&A model aims to find the spans of the key clauses from the full contract by deal point type. A RoBERTa-based model was fine-tuned and achieved an Area Under the Precision-Recall curve (AUPR) of 19.7%, which indicates there is still much space for advancement.

2.3 MarkupMnA

MarkupMnA (Rao et al., 2023) [4] is designed to aid in the segmentation and understanding of M&A agreements. With the leverage of the data in HyperText Markup Language (HTML) format, the authors built a dataset to annotate the section titles for the separate text. This approach addressed the challenges associated with both the absence of computationally efficient hierarchical document representation and the constraints in handling lengthy multimodal inputs.

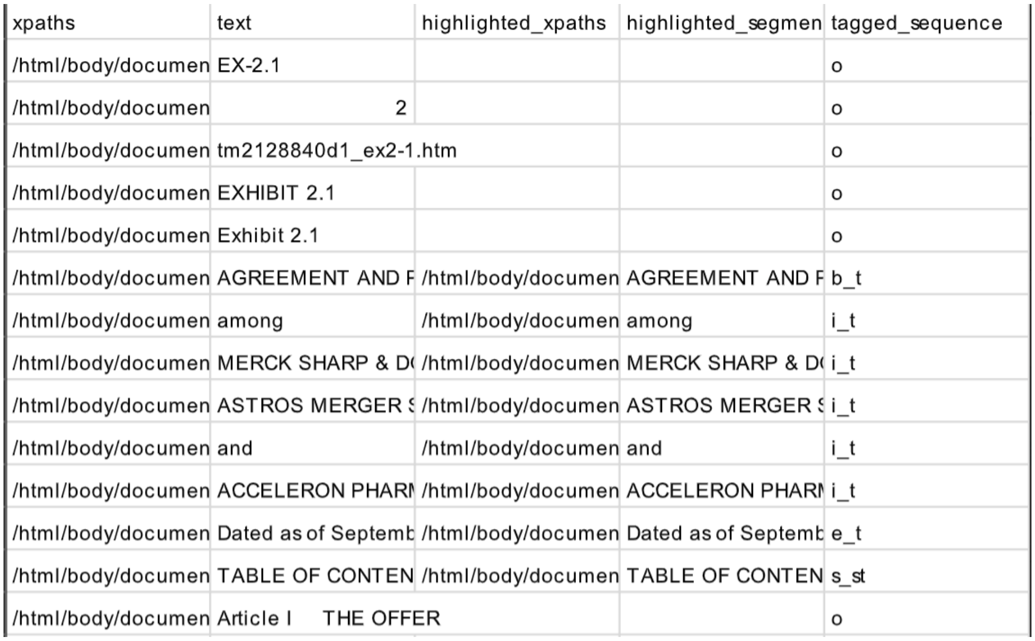
Even though the paper does not provide the experiment result from using MarkupMnA in downstream tasks to prove its utility, it provides us with a good dataset and an alternative idea, which can be used to improve the performance of the entity extraction task in MAUD.

Based on the above investigation and recognizing that MAUD covered the same contracts as MarkupMnA, we determined to proceed with the deal point text extraction task as the main objective of our project with leveraging the hierarchical HTML data for improvement. This decision is rooted in the significance of this task within the legal domain and its potential for real-world applications. By focusing on enhancing the segmentation utility in the MAUD Extraction task, we aim to demonstrate the practical value and efficacy of our model in addressing challenges in the comprehension of M&A agreements.

3 Data Overview

3.1. MarkupMnA

The MarkupMnA dataset consists of 151 M&A agreements, initially formatted in HTML and converted to and saved as CSV files. The authors developed a labeling tool that allows annotators to browse the agreements in a web browser and manually highlight nodes corresponding to page numbers and section titles. Using the Chrome extension, the annotated website can be exported as JSON files, subsequently converted to CSV files with BEIOS labels using a custom Python script. The CSV files keep the information of text, XPath of the text, text highlighted by annotators, XPath of the highlighted text, and the BEIOS labels, as shown in Figure 1.



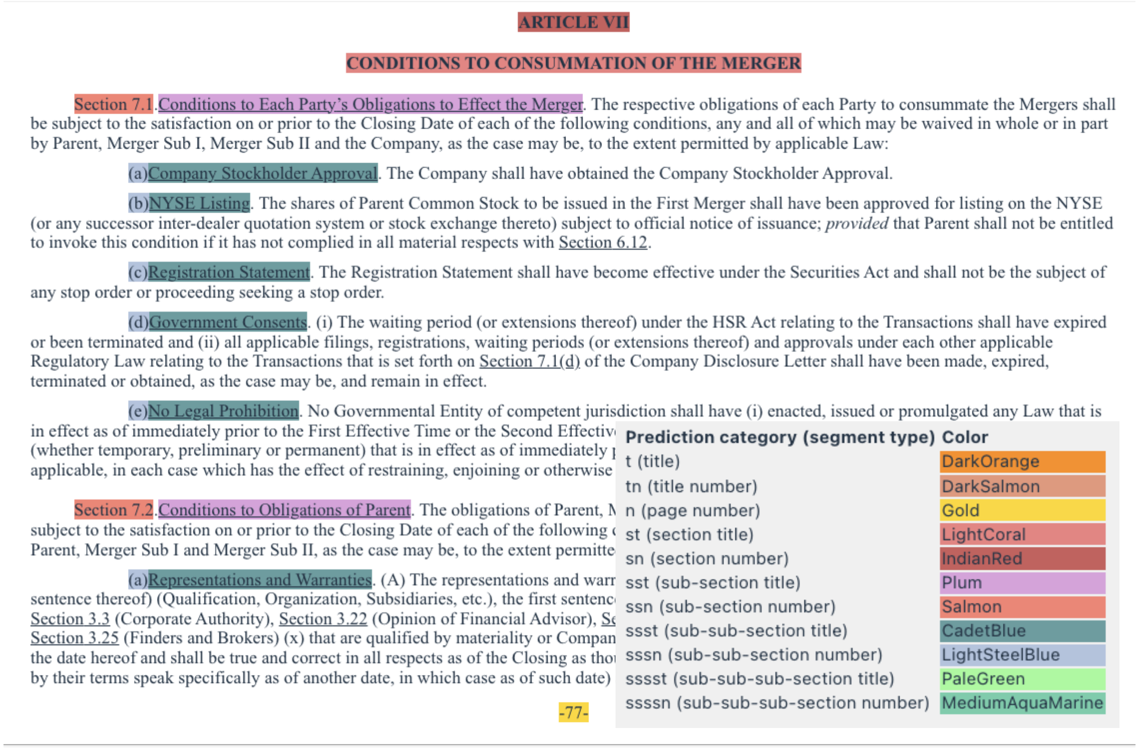
xpaths	text	highlighted_xpaths	highlighted_segmen	tagged_sequence
/html/body/documen	EX-2.1			o
/html/body/documen	2			o
/html/body/documen	tm2128840d1_ex2-1.htm			o
/html/body/documen	EXHIBIT 2.1			o
/html/body/documen	Exhibit 2.1			o
/html/body/documen	AGREEMENT AND F	/html/body/documen	AGREEMENT AND F	b_t
/html/body/documen	among	/html/body/documen	among	i_t
/html/body/documen	MERCK SHARP & D(	/html/body/documen	MERCK SHARP & D(	i_t
/html/body/documen	ASTROS MERGER S	/html/body/documen	ASTROS MERGER S	i_t
/html/body/documen	and	/html/body/documen	and	i_t
/html/body/documen	ACCELERON PHARM	/html/body/documen	ACCELERON PHARM	i_t
/html/body/documen	Dated as of Septemb	/html/body/documen	Dated as of Septemb	e_t
/html/body/documen	TABLE OF CONTEN	/html/body/documen	TABLE OF CONTEN	s_st
/html/body/documen	Article I THE OFFER			o

Figure 1. Example of CSV-formatted Agreements in MarkupMnA

For BEIOS labeling, each token is labeled in the format of *<BEIOS Tagging>_<Prediction Category>*. Since an entity can be separated into multiple tokens in the dataset, *BEIOS Tagging* schema specifies the relative position of the token in the entity: B for ‘beginning’, I for ‘inside’, O for ‘outside’, E for ‘end’, and S for ‘singleton’. *Prediction Category* represents the segment type, including section title, sub-section title, section title number, page number, etc.

Closer observations from the dataset reveal that while the content in the text is generally the same as the content in highlighted text for the same row, it is not entirely identical. In some cases, it is true that only part of the text needs to be annotated, while in other cases, discrepancies appear to be due to manually highlighting errors.

With this insight, we developed a visualization tool, based on MarkupMnA model inference visualizer, to display the highlighted text and differentiate the section hierarchy with color. Our visualizer could be explored at https://xd2287.github.io/MarkupMnA-Colornizer-GroudTruth/. The visualizer, as shown in Figure 2, enables us to check the ground truth of the highlighted text, detect the errors, and correct before training.



ARTICLE VII

CONDITIONS TO CONSUMMATION OF THE MERGER

Section 7.1.Conditions to Each Party's Obligations to Effect the Merger. The respective obligations of each Party to consummate the Mergers shall be subject to the satisfaction on or prior to the Closing Date of each of the following conditions, any and all of which may be waived in whole or in part by Parent, Merger Sub I, Merger Sub II and the Company, as the case may be, to the extent permitted by applicable Law:

(a)Company Stockholder Approval. The Company shall have obtained the Company Stockholder Approval.

(b)NYSE Listing. The shares of Parent Common Stock to be issued in the First Merger shall have been approved for listing on the NYSE (or any successor inter-dealer quotation system or stock exchange thereto) subject to official notice of issuance; *provided* that Parent shall not be entitled to invoke this condition if it has not complied in all material respects with Section 6.12.

(c)Registration Statement. The Registration Statement shall have become effective under the Securities Act and shall not be the subject of any stop order or proceeding seeking a stop order.

(d)Government Consents. (i) The waiting period (or extensions thereof) under the HSR Act relating to the Transactions shall have expired or been terminated and (ii) all applicable filings, registrations, waiting periods (or extensions thereof) and approvals under each other applicable Regulatory Law relating to the Transactions that is set forth on Section 7.1(d) of the Company Disclosure Letter shall have been made, expired, terminated or obtained, as the case may be, and remain in effect.

(e)No Legal Prohibition. No Governmental Entity of competent jurisdiction shall have (i) enacted, issued or promulgated any Law that is in effect as of immediately prior to the First Effective Time or the Second Effectiv
(whether temporary, preliminary or permanent) that is in effect as of immediately p
applicable, in each case which has the effect of restraining, enjoining or otherwise

Section 7.2.Conditions to Obligations of Parent. The obligations of Parent, M
subject to the satisfaction on or prior to the Closing Date of each of the following c
Parent, Merger Sub I and Merger Sub II, as the case may be, to the extent permitte

(a)Representations and Warranties. (A) The representations and warr
sentence thereof) (Qualification, Organization, Subsidiaries, etc.), the first sentenc
Section 3.3 (Corporate Authority), Section 3.22 (Opinion of Financial Advisor), Se
Section 3.25 (Finders and Brokers) (x) that are qualified by materiality or Compan
the date hereof and shall be true and correct in all respects as of the Closing as tho
by their terms speak specifically as of another date, in which case as of such date)

-77-

Prediction category (segment type)	Color
t (title)	DarkOrange
tn (title number)	DarkSalmon
n (page number)	Gold
st (section title)	LightCoral
sn (section number)	IndianRed
sst (sub-section title)	Plum
ssn (sub-section number)	Salmon
ssst (sub-sub-section title)	CadetBlue
sssn (sub-sub-section number)	LightSteelBlue
sssst (sub-sub-sub-section title)	PaleGreen
ssssn (sub-sub-sub-section number)	MediumAquaMarine

Figure 2. Example of Highlighted Contract 126 with our Visualizer

3.2. MAUD

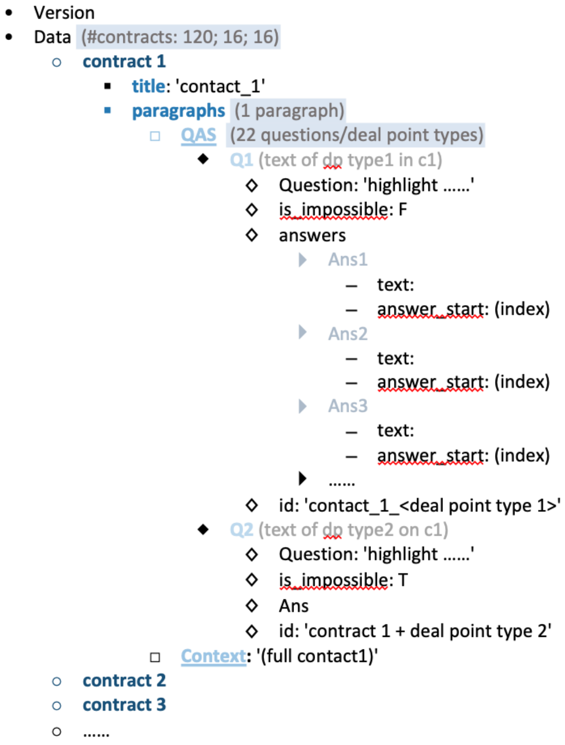
The MAUD extraction dataset encompasses the same M&A agreements as MarkupMnA, while in the format of SQuADv2 JSON. It comprises the plain text of the contracts, deal point type questions, and answer spans to those questions. Figure 3 illustrates the concrete hierarchical structure of the JSON file:

For each contract, there are two components: '*title*' and '*paragraphs*'. The '*title*' is formatted as 'contract_<*id*>', while '*paragraphs*' is a list consisting of one dictionary of keys '*qas*' and '*context*'. The '*context*' keeps the plain text of the full contract, and '*qas*' is a list of 22 dictionaries, corresponding to the 22 deal point types. Each element in '*qas*' has four keys:

- '*question*': used to specify the deal point type being addressed, formulated as "Highlight the parts of the text (if any) related to <*Deal Point Type*> that should be reviewed as a lawyer."
- '*is_impossible*': when true, indicates there is no relevant text for the <*Deal Point Type*>

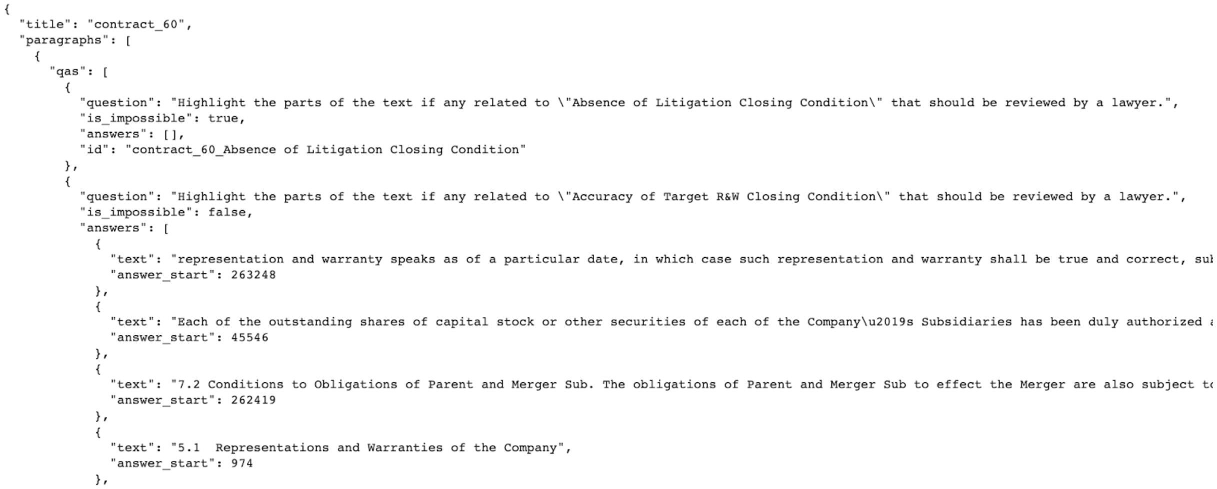
and the corresponding ‘*answers*’ list is empty;

- ‘*answers*’: a list of dictionaries, with each dictionary element referring to a span of text related to the investigated deal point type;
    - ‘*text*’: relevant text span of <*Deal Point Type*>;
    - ‘*answer_start*’: the starting index of ‘*text*’ found in the context;
- ‘*id*’: in the format of ‘<*title*>_<*Deal Point Type*>’.



```
• Version
• Data (#contracts: 120; 16; 16)
    ○ contract 1
        ▪ title: 'contact_1'
        ▪ paragraphs (1 paragraph)
            □ QAS (22 questions/deal point types)
                ◆ Q1 (text of dp type1 in c1)
                    ◇ Question: 'highlight ......'
                    ◇ is_impossible: F
                    ◇ answers
                        ▸ Ans1
                            – text:
                            – answer_start: (index)
                        ▸ Ans2
                            – text:
                            – answer_start: (index)
                        ▸ Ans3
                            – text:
                            – answer_start: (index)
                        ▸ ......
                    ◇ id: 'contact_1_<deal point type 1>'
                ◆ Q2 (text of dp type2 on c1)
                    ◇ Question: 'highlight ......'
                    ◇ is_impossible: T
                    ◇ Ans
                    ◇ id: 'contract 1 + deal point type 2'
            □ Context: '(full contact1)'
    ○ contract 2
    ○ contract 3
    ○ ......
```

Figure 3. Structure of MAUD Files



```
{
  "title": "contract_60",
  "paragraphs": [
    {
      "qas": [
        {
          "question": "Highlight the parts of the text if any related to \"Absence of Litigation Closing Condition\" that should be reviewed by a lawyer.",
          "is_impossible": true,
          "answers": [],
          "id": "contract_60_Absence of Litigation Closing Condition"
        },
        {
          "question": "Highlight the parts of the text if any related to \"Accuracy of Target R&W Closing Condition\" that should be reviewed by a lawyer.",
          "is_impossible": false,
          "answers": [
            {
              "text": "representation and warranty speaks as of a particular date, in which case such representation and warranty shall be true and correct, su
              "answer_start": 263248
            },
            {
              "text": "Each of the outstanding shares of capital stock or other securities of each of the Company\u2019s Subsidiaries has been duly authorized
              "answer_start": 45546
            },
            {
              "text": "7.2 Conditions to Obligations of Parent and Merger Sub. The obligations of Parent and Merger Sub to effect the Merger are also subject t
              "answer_start": 262419
            },
            {
              "text": "5.1  Representations and Warranties of the Company",
              "answer_start": 974
            },
```

Figure 4. An Example of MAUD where *is_impossible* = *False*

Upon further investigation, we observed that the average length of the contracts is 50000 in word and 330000 in character. Additionally, we found the number of answer spans varied for deal point types. For example, the second deal point type ('Accuracy of Target R&W Closing Condition') has the highest number of text spans for almost all contracts, while 'Breach of No Shop' seems to have the least. The dataset was divided in an 80-10-10 ratio, specifically with 120 contracts in the training set, 16 contracts in the validation set, and 16 contracts in the test set. We adhered to the same split for our model.

4 Modeling and Methods

The main objective of our project is to integrate the hierarchical representation of the contracts provided by MarkupMnA into the deal point extraction task in MAUD, aiming to improve the performance. Our work is built on the pretrained RoBERTa model in MAUD.

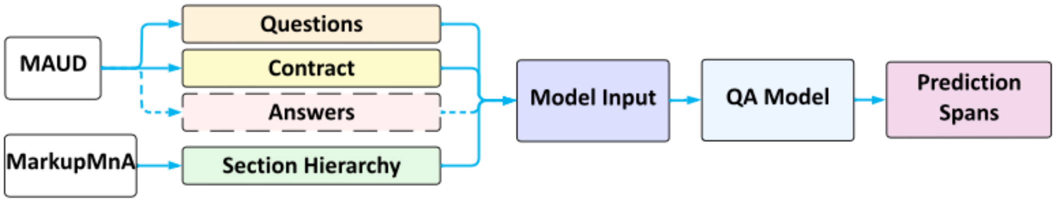


Figure 5. Workflow Diagram.

4.1 Segmentation

To incorporate the hierarchical information, we initiated segmenting the full contracts based on the BEIOS labels in MarkupMnA. With the consideration of length, we chose to segment at the subsection level and utilize section numbers as segment points. Specifically, we collected all tokens labeled '*s_ssn*' or '*e_ssn*' into a list. Given that the text of a section number might appear multiple times in the full contract, for accurate identification of its working location in the plain text, we also captured the next line in the CSV file after the '*s_ssn*' or '*e_ssn*' label as another list. Then, we concatenated the corresponding tokens from the two lists using regular expressions and found the start index of every combined expression in the full contract. These start indices were recorded as the point for segmentation.

4.2 Method 1: Section-Based Sliding Window

Owing to the lengthy nature of the contract in our dataset, the context length always exceeds the model input limitation when using the full contract as the context. As shown in Figure 6 below, with the SQuAD data processor in the `transformers` package, the full contract will be divided into multiple overlapped sliding windows of a fixed length.

The MAUD paper followed this original approach for generating sliding windows using the official `transformers` package. However, since contracts in our dataset consist of relatively independent sections, applying the method depicted in Figure 6 results in sliding windows spanning different sections. This may compromise contextual coherence, posing a challenge to the model's understanding.

To overcome the original defects, we proposed the section-based sliding windows, which incorporated section hierarchy information in the generation of sliding windows. We customized the `transformers` package, and took the list of section start indices as a parameter in sliding window generation. As illustrated in Figure 7, the sliding window length and stride length are adjusted with the section split points. If the current sliding window covers a section split point, the split point will be considered as the end of the current sliding window and the start of the next sliding window. This method allows us to represent the contract with fewer sliding windows than the original approach, without windows crossing section boundaries.

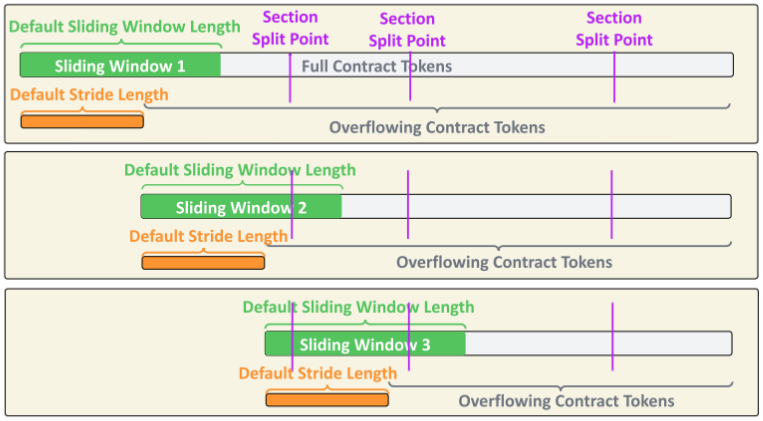


Figure 6. Original Sliding Windows in MAUD

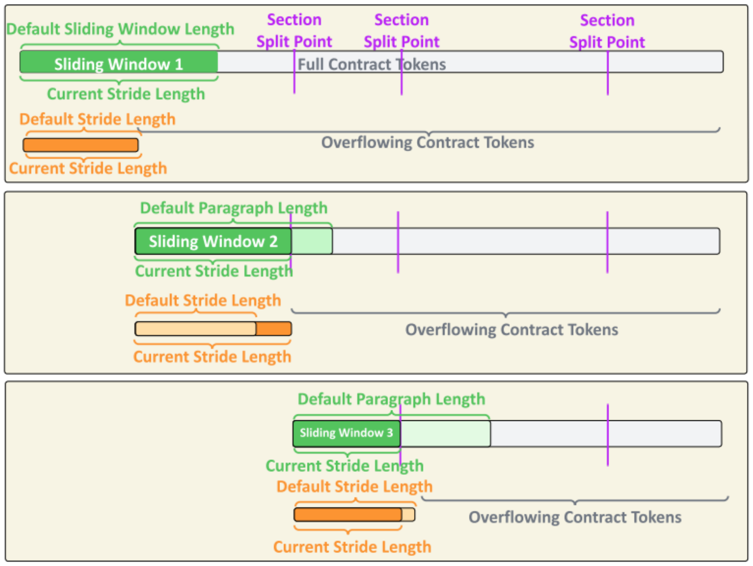
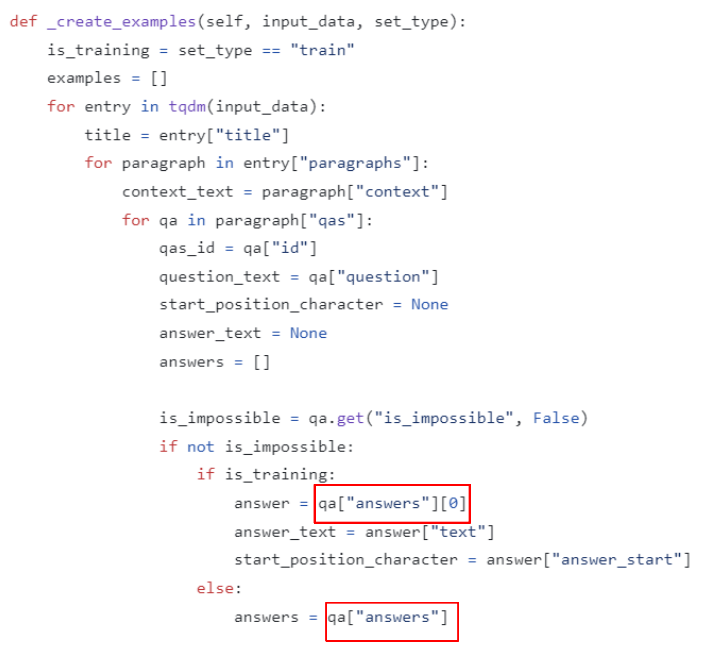


Figure 7. Our Proposed Section-Based Sliding Windows

4.3 Method 2: Section-Based Training Set

The original model from the MAUD paper used the data processing functions from Hugging Face. However, we notice that when processing the raw training dataset, only the first answer span for a particular question-contract pair was extracted. The remaining answer spans were left unused and unexplored. As shown in Figure 8, `answer = qa["answers"][0]` represents the extraction of the first answer span in the training mode, whereas `answers = qa["answers"]` indicates the extraction of all the answer spans in the evaluation mode. It is common to use a single ground truth answer during model training. This makes it easier to train the model because the model would have a clear target to optimize. However, in our task, the lengthy contract introduces numerous different answers, where this traditional data processing method might not suit. We understood the concerns of using multiple answers as ground truth, which would complicate the training process because it's unclear how to weigh different answers if they conflict. To address this issue, we decided to make a slight adjustment to the training dataset based on the hierarchical information.

In Section 4.1, we introduced how to obtain the start index of each section from the MarkupMnA dataset. In this method, we used the start index lists to split each contract into several sections and reconstructed the JSON format dataset with their corresponding answers to be fed into the model. Question-contract pairs were converted to question-section pairs as shown in Figure 9. The location of all the answer spans in a contract were converted to their local location in the sections. For each section, we kept the questions the same and defined a function to add the answer spans, whose start indices between the start index and the end index of the section, to the answer list of the section. Given the update of the training set, we also modified some details for the model, such as the random sampler. In this case, the same data processing method can extract one answer from each question-section pair, and more answer spans can be used. Therefore, the model can better harness the potential of the dataset. We used the section-based training set and fine-tuned the pretrained `roberta-base` model from Hugging Face on four A100 GPUs. Each training epoch took around 80 minutes. Given the limited resources and time, we did not use the full dataset and fine-tuned the model thoroughly until the convergence of the loss.



```python
def _create_examples(self, input_data, set_type):
    is_training = set_type == "train"
    examples = []
    for entry in tqdm(input_data):
        title = entry["title"]
        for paragraph in entry["paragraphs"]:
            context_text = paragraph["context"]
            for qa in paragraph["qas"]:
                qas_id = qa["id"]
                question_text = qa["question"]
                start_position_character = None
                answer_text = None
                answers = []

                is_impossible = qa.get("is_impossible", False)
                if not is_impossible:
                    if is_training:
                        answer = qa["answers"][0]
                        answer_text = answer["text"]
                        start_position_character = answer["answer_start"]
                    else:
                        answers = qa["answers"]
```

Figure 8. Part of Source Code from SQuAD Data Processor

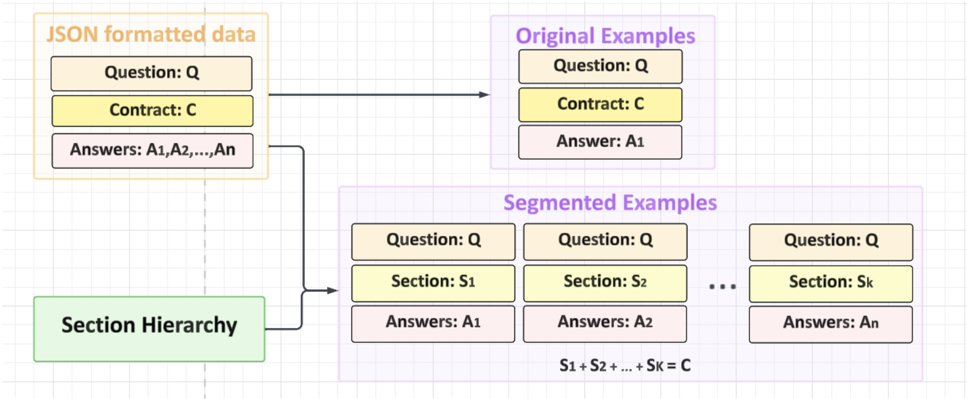


Figure 9. Comparison of the Original Training Set and our Section-Based Training Set

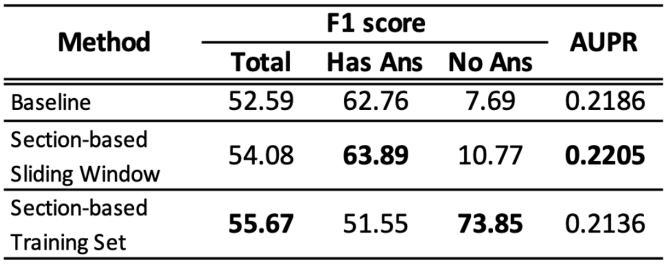
5 Results

We evaluate our methods by the two metrics used in MAUD for comparison: F1-score and AUPR. These two metrics both capture the effect of recall and precision, and are suitable for evaluating the imbalanced dataset.

Our Section-based Sliding Window approach in Method 1 consistently surpasses the original method across various scenarios, manifesting a substantial enhancement in the F1-score. The aggregate F1 score witnesses an increase from 52.59 to 54.08, with the F1 score for question-contract pairs containing answers rising from 62.76 to 63.89 and the F1 score for instances without answers escalating from 7.69 to 10.77. Moreover, this approach demonstrates superior AUPR performance compared to the baseline, exhibiting an increase of 0.19% from

21.86% to 22.05%. This accentuates its proficiency in capturing intricate relationships within the data.

Furthermore, the fine-tuned model, trained using Method 2 and a section-based training set, exhibits a noteworthy improvement in F1 scores, specifically for question-contract pairs lacking corresponding answers, soaring from an initial F1 score of 7.69 to a remarkable 73.85. This delineates valuable insights into its targeted capabilities in addressing specific scenarios.



Method	**F1 score**			**AUPR**
	Total	**Has Ans**	**No Ans**	
Baseline	52.59	62.76	7.69	0.2186
Section-based Sliding Window	54.08	**63.89**	10.77	**0.2205**
Section-based Training Set	**55.67**	51.55	**73.85**	0.2136

Table 1. F1-Score and AUPR

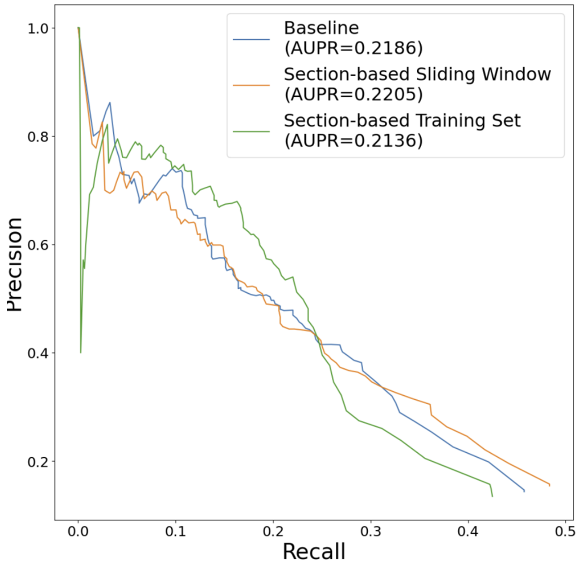


Figure 10. Precision and Recall Curve

6 Conclusion

Our project focused on integrating AI into the legal sector, specifically addressing challenges posed by the lengthy nature of legal documents, primarily the M&A agreements. Motivated by the limited use of AI in legal contexts, we leveraged the MarkupMnA and MAUD datasets to streamline the comprehension of complex agreements. Throughout our project, we conducted thorough data exploration with the devised data visualization tool and introduced innovative approaches, such as the Section-based Sliding Window method and the Section-based Training Set method, resulting in notable improvements in F1-score or AUPR in diverse scenarios with mere fine-tuning.

7 Future Work

In discussing the integration of structural information derived from the MarkupMnA model, alternative methods can be considered to achieve similar functionality. One such approach involves the incorporation of a novel embedding for section titles into the model, analogous to position embedding but designed to encapsulate hierarchical section title information. This section title embedding can serve as a mechanism for the model to discern the global structure of the document and facilitate an detection of the specific section's importance within the entire contract. The inclusion of this embedding has the potential to enhance model performance by providing insights into the contextual relevance of the current text.

Furthermore, an exploration of alternative data processing functions is warranted to address a limitation in the model's current configuration, specifically its tendency to solely consider the first answer for each question within a given context. A modification to accommodate all answers could be devised, enabling the model to leverage a more comprehensive dataset in the Question and Answering context. This enhancement has the potential to yield improvements in F1 score or AUPR, as it enables the utilization of additional information and a more exhaustive exploration of the dataset's potential.

8 Ethical Considerations

Datasets we used are accessible freely online: MAUD from *The Atticus Project* and MarkupMnA from *Zenodo*. Since our focus was on merger and acquisition agreements, which excludes personal data collection, we believe there are no ethical concerns regarding the data.

Reference

[1] Perez, Ethan, Douwe Kiela, and Kyunghyun Cho. "Rissanen data analysis: Examining dataset characteristics via description length." *International Conference on Machine Learning*. PMLR, 2021.
[2] Xiao, Chaojun, et al. "Lawformer: A pre-trained language model for Chinese Legal Long Documents." *AI Open* 2 (2021): 79-84.
[3] Steven H. Wang, Antoine Scardigli, et al. “MAUD: An Expert-Annotated Legal NLP Dataset for Merger Agreement Understanding”. arXiv:2301.00876 (2023).
[4] Sukrit Rao, Pranab Islam, et al. “MARKUPMNA: Markup-Based Segmentation of M&A Agreements”. 2023.