Layout-Aware Information Extraction for Document-Grounded Dialogue: Dataset, Method and Demonstration

Zhenyu Zhang[*]
Institute of Information Engineering,
Chinese Academy of Sciences
Beijing, China
zhangzhenyu1996@iie.ac.cn

Bowen Yu[*]
DAMO Academy, Alibaba Group
Institute of Information Engineering,
Chinese Academy of Sciences
Beijing, China
yubowen.ybw@alibaba-inc.com

Haiyang Yu
DAMO Academy, Alibaba Group
Beijing, China
yifei.yhy@alibaba-inc.com

Tingwen Liu[†]
Institute of Information Engineering,
Chinese Academy of Sciences
School of Cyber Security, University
of Chinese Academy of Sciences
Beijing, China
liutingwen@iie.ac.cn

Cheng Fu
Jingyang Li
DAMO Academy, Alibaba Group
Beijing, China
fucheng.fuc@alibaba-inc.com
qiwei.ljy@alibaba-inc.com

Chengguang Tang
Jian Sun
Yongbin Li
DAMO Academy, Alibaba Group
Beijing, China
chengguang.tcg@alibaba-inc.com
jian.sun@alibaba-inc.com
shuide.lyb@alibaba-inc.com

ABSTRACT

Building document-grounded dialogue systems have received growing interest as documents convey a wealth of human knowledge and commonly exist in enterprises. Wherein, how to comprehend and retrieve information from documents is a challenging research problem. Previous work ignores the visual property of documents and treats them as plain text, resulting in incomplete modality. In this paper, we propose a Layout-aware document-level Information Extraction dataset, LIE, to facilitate the study of extracting both structural and semantic knowledge from visually rich documents (VRDs), so as to generate accurate responses in dialogue systems. LIE contains 62k annotations of three extraction tasks from 4,061 pages in product and official documents, becoming the largest VRD-based information extraction dataset to the best of our knowledge. We also develop benchmark methods that extend the token-based language model to consider layout features like humans. Empirical results show that layout is critical for VRD-based extraction, and system demonstration also verifies that the extracted knowledge can help locate the answers that users care about.

CCS CONCEPTS

• **Computing methodologies → Information extraction**; **Discourse, dialogue and pragmatics**;

KEYWORDS

document-grounded dialogue, visually-rich document, information extraction

ACM Reference Format:
Zhenyu Zhang, Bowen Yu, Haiyang Yu, Tingwen Liu, Cheng Fu, Jingyang Li, Chengguang Tang, Jian Sun, and Yongbin Li. 2022. Layout-Aware Information Extraction for Document-Grounded Dialogue: Dataset, Method and Demonstration. In *Proceedings of the 30th ACM International Conference on Multimedia (MM '22), Oct. 10–14, 2022, Lisboa, Portugal.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3503161.3548765

1 INTRODUCTION

Dialogue systems are playing an increasingly important role in various business applications [12, 13, 31]. Currently, enterprises and organizations often own abundant business documents, which have the potential to solve user queries, such as product documentation and policy guidance. However, users still prefer to efficiently obtain the desired answer through interactive dialogue, rather than inefficient and inconvenient retrieval in lengthy documents. Therefore, building machine-assisted agents based on massive documents at low cost is a topic of great concern in industry [2, 5, 6, 42].

Towards this goal, the document-grounded dialogue (doc2dial) system is usually decomposed into two sub-tasks: knowledge extraction and response generation [10]. Specifically, knowledge extraction aims to identify the most relevant knowledge in the associated document, and response generation focuses on generating a fluent response. Wherein, the first step is the key point that drives doc2dial different from other dialogue tasks. However, in this part, previous works typically assume the input to be textual strings [4], while many real-world business documents are scanned or digital-born (e.g., images of invoices, forms in PDF format). *When converting these visually rich documents (VRDs) into snippets of plain text, a lot of layout information is discarded, which is exactly the indicative*

[*] Both authors contributed equally to this research.
[†] Corresponding author.

MM '22, October 10–14, 2022, Lisboa, Portugal.

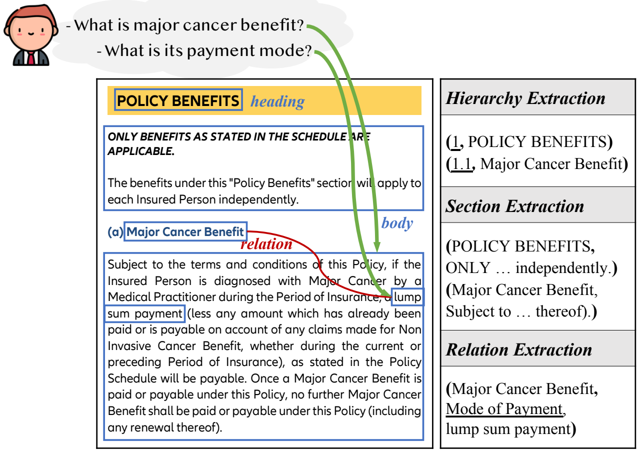
ACM ISBN 978-1-4503-9203-7/22/10…$15.00
https://doi.org/10.1145/3503161.3548765

signal to help humans quickly understand the document and search for some interested knowledge.

Based on an inductive analysis of a large number of real-world queries in business doc2dial systems, we conclude that the answers that users care about, as shown in Figure 1, can be roughly summarized into two kinds of knowledge in the document: (1) *coarse-grained description knowledge* at section level, (2) *fine-grained fact knowledge* within a section. Besides, to accurately locate the section related to the query for better finding relevant knowledge, the system also needs to be aware of the *hierarchy knowledge* of a document. Recently, utilizing layout information to help extract information from VRDs has attracted significant interest, and many layout-aware datasets are proposed. For example, CORD [24] and SROIE [15] aim to extract values for pre-defined keys from receipts, FUNSD [17] and EPHOIE [28] focus on the automatic extraction of key-value pairs from form-like documents. *However, the task forms defined in these datasets are within the scope of span extraction and ignore the extraction of fact knowledge.*

To overcome the above limitations, we construct a new Layout-aware dataset for document-level Information Extraction, LIE. Unlike previous text-only or form-like datasets, LIE is built on multi-page VRDs, requiring systems not only to extract specific text spans but also the relations among them. We provide the original file and all tokens with spatial positions (i.e., layout feature) for each document, then release three information extraction (IE) tasks: *hierarchy extraction*, *section extraction* and *relation extraction*, where the first one is to structuralize the multi-page document, and the last two aims to collect specific knowledge described above.

We also propose benchmark methods for the new dataset. Inspired by the mechanism that humans understand VRDs through both text semantics and document layout, we extend popular token-based language models with layout features. Similar to the 1D position embeddings in transformer-based models, we introduce 2D layout embeddings to capture the spatial relationship among tokens within a document. Then the layout-aware model is firstly pre-trained on unlabeled data with two self-supervised tasks, masked layout-language model and potential heading selection, to better learn the newly introduced layout embeddings, then fine-tuned on task-specific labeled data. Experimental results suggest that injecting layout could improve the IE performance significantly, and the pre-training process accelerates the convergence of models on downstream tasks. Moreover, our model has already gone into production in a world-leading cloud computing platform, we also demonstrate the workflow of doc2dial service and how layout-aware extraction plays a vital role at the end of the paper.


- What is major cancer benefit?
- What is its payment mode?
POLICY BENEFITS
heading
body
(a) Major Cancer Benefit
relation
Hierarchy Extraction
(1, POLICY BENEFITS)
(1.1, Major Cancer Benefit)
Section Extraction
(POLICY BENEFITS, ONLY … independently.)
(Major Cancer Benefit, Subject to … thereof).)
Relation Extraction
(Major Cancer Benefit, Mode of Payment, lump sum payment)

ONLY BENEFITS AS STATED IN THE SCHEDULE ARE APPLICABLE.

The benefits under this "Policy Benefits" section will apply to each Insured Person independently.

Subject to the terms and conditions of this Policy, if the Insured Person is diagnosed with Major Cancer by a Medical Practitioner during the Period of Insurance, a lump sum payment (less any amount which has already been paid or is payable on account of any claims made for Non Invasive Cancer Benefit, whether during the current or preceding Period of Insurance), as stated in the Policy Schedule will be payable. Once a Major Cancer Benefit is paid or payable under this Policy, no further Major Cancer Benefit shall be paid or payable under this Policy (including any renewal thereof).

Figure 1: The illustration of doc2dial, VRD, and the proposed IE tasks (from left to right). Best viewed in color.

2 RELATED WORK

Document-based Information Extraction. Most work in early phase IE focuses on the sentence-level. Nowadays, many researchers extend the scope of extraction beyond isolated sentences [14, 41, 44]. Yao et al. [36] introduced a cross-sentence relation extraction dataset, DocRED, based on Wikipedia. Yu et al. [38] proposed DialogRE to support the prediction of relations in multi-turn dialogues. The length of these documents is equivalent to DocRED, which is actually the paragraph-level extraction. SciREX [16] releases the task of identifying the main results of a scientific article. However, the source document provides more layout information than a sentence or paragraph. Most existing IE datasets drop these valuable features in the process of text reading. Recently, many datasets for IE from VRDs are proposed. CORD [24] proposes the task to label each word in receipts to the right field, wherein 30 fields under 4 categories are defined. SROIE [15] aims to extract values from each receipt for 4 pre-defined keys. Towards extracting keys from documents, FUNSD [17] is the most widely-used dataset and constructed on various business forms with 4 keys. EPHOIE [28] is constructed based on the images of examination paper heads, and 5 kinds of key-value pairs need to be extracted. What is most different about our dataset is that these only support span extraction and their documents are form-like with few words. Our work focuses more on *enabling machines to handle the same multimodal input as humans when they read real-world documents* and *extracting various structural and semantic knowledge from VRDs so as to drive better doc2dial systems.*

Layout-aware Language Model. Humans perceive the document through many aspects, such as language, layout, and vision. Based on the powerful modeling ability of Transformer, LayoutLM [35] and LayoutLMv2 [34] are successively proposed to learn the multi-modality interaction and achieve great success. Following the paradigm, StructuralLM [19] leveraged cells and layout to make the model aware of which words are from the same cell. StrucTexT [22] extracted semantic features from different levels and modalities. DocFormer [1] incorporated multi-modality self-attention with shared spatial embeddings. Besides, Wei et al. [30] combined the power of language models and graph neural networks for the extraction from VRDs. Zhang et al. [40] presented an end-to-end network to bridge text reading and IE for document understanding. The motivation behind these studies is similar to ours, but they pay too much attention to layout and ignore the in-depth semantics. Our work aims to *go beyond surface modeling and understand the document logical hierarchy from a semantic perspective.*

Document-grounded Dialogue. The task of reading documents and responding to queries has been the trigger of recent research advances [2, 6, 31]. Recently, the doc2dial dataset [11] is proposed to facilitate the frontier exploration and landing application of document-grounded dialogue systems. It contains 4,793 goal-oriented dialogues and a total of 488 associated grounding documents from 4 domains for social welfare. The proof-of-concept doc2dial framework [10] and subsequent attempts [4, 9, 18] are following the knowledge extraction - response generation paradigm. However, their data pre-processing includes a well-designed text reading step, which draws the government document into the form of plain text. In this paper, we pay more attention to the step of knowledge extraction and argue that *the multimodal features of business documents should not be lightly dismissed.*

3 DATASET

With the development of learning-based algorithms, a comprehensive benchmark conducted for specific tasks is a prerequisite to motivate advanced works. Compared with previous IE datasets, LIE is formed with several desirable properties: (i) It is proposed for the extraction from multi-page VRDs, which brings new opportunities and challenges than previous text-only or form-like settings. (ii) It pays more attention to the extraction of relational fact knowledge, while prior work only focuses on extracting salient spans. (iii) It needs to understand and parse the hierarchical structure of documents, which is able to facilitate kinds of document understanding tasks and follow-up document-grounded applications.

3.1 Task Formulation

To better extract valuable knowledge from VRDs and support the downstream doc2dial systems, we formulate three IE tasks:

Hierarchy Extraction (HE). It aims to extract document logical hierarchy and generate table-of-content (TOC) by organizing section headings in order. An implementable goal is to detect *section headings* with their *global levels.* In Figure 1, “POLICY BENEFITS” is the first section heading with level 1, so the TOC number is 1, “Major Cancer Benefit” is a subsection heading with level 2 and number 1.1. Typically, headings at the same level share the same font formats, meaning that the non-text signals, including type, size, and indentation, are vital for the task.

Section Extraction (SE). It could be regarded as an extension of key-values extraction in previous form-like datasets [17], where the heading of a section is key, and the corresponding body is value. For VRDs, bodies are usually below or to the right of headings and sometimes indented. Figure 1 shows concrete examples. When mounting the (*heading*, *body*) pairs on TOC, we can easily transform a document into its hierarchical tree, which is beneficial to many downstream applications [27].

Relation Extraction (RE). It needs to detect *two entities* and identify their *relations* from VRDs, as shown in Figure 1. When converting VRDs into plain text, two entities may become far away, but actually, they may be very close or even aligned in the document. In this task, relations belong to a pre-defined set, so it is suitable for obtaining structured knowledge that we care about [29, 37].

3.2 Data Collection and Annotation

To construct LIE, we choose two representative VRDs in the real world, i.e., *product documents* and *official documents.* Specifically, we download Chinese product documents in PDF format from portal websites of the insurance sector and collect official documents issued by government departments from web search engines. Different from previous VRD-based datasets, these documents contain more textual fragments, which is more conducive to evaluating the language understanding ability. Furthermore, the semantic information is expressed not only through the text in each fragment but also how the fragments organized, so it is still important to perceive the document layout.

Table 1: Layout-aware IE datasets comparison. #Doc. Length refers to the average page numbers.

Datasets	#Pages	#Doc. Length	Fact Extraction?
CORD (2019)	1,000	1	✗
SROIE (2019)	973	1	✗
FUNSD (2019)	199	1	✗
EPHOIE (2021)	1,494	1	✗
LIE (our)	**4,061**	**10.15**	✓

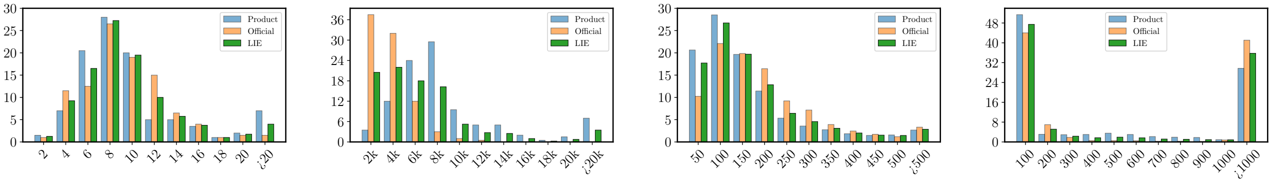
Next, we utilize PDFPlumber[1] to extract tokens with bounding boxes automatically. Here, a document page is considered as a 2D coordinate system with the *top-left origin.* Therefore, in the parsing result, each word is aligned with a unique quadruplet (x_0, y_0, x_1, y_1), where (x_0, y_0) corresponds to the position of the upper left in bounding box, and (x_1, y_1) represents the lower right. We also provide the PDF file so that downstream models have the opportunity to access rich original information, such as visual features.

The data annotation process is carried out by crowdsourcing. Annotators are provided with PDF files and asked to fill the slots by copying text spans (e.g., headings) or generating (i.e., TOC) numbers according to the annotation specification. From the practical application perspective, we summarize 18 and 15 relations for product documents and official documents, respectively, as pre-defined relation schemes (see also Table 6 in Appendix) All the (20) annotators have linguistic knowledge, are instructed with formal annotation principles, and pass trial annotation. To ensure the labeling quality, each instance is labeled by at least two annotators. If the two annotators have disagreements on an instance, it will be assigned to a third annotator. The annotation results are also randomly checked. If the accuracy is lower than 95% (measured in the document-level), all results of the annotator will be reviewed. We ensure that all the annotators are fairly compensated by market price according to their workload. Finally, we develop heuristic rules to align the parsing results with annotations, and about 2% of the annotation instances cannot be aligned. We discard them and blame them for the parsing error (see also Appendix for more details).

3.3 Data Statistics and Analysis

Using the annotation procedure mentioned above, we build a dataset of 4,061 fully annotated pages from 400 documents (200 product and official documents, respectively). All documents have accompanied layout features. Note that these documents totally come from more

[1] https://github.com/jsvine/pdfplumber



(a) document length (pages).

(b) document length (words).

(c) section length (words).

(d) argument distance (words).

Figure 2: Distribution of document length, section length, and argument distance in LIE, where the y-axis is in percent.

than 150 organizations, and documents from one organization usually have more than one layout format, *making the dataset diverse enough and robust to practical applications.*

Table 1 provides the statistics of LIE and some representative VRDs-based IE datasets. One can observe that LIE is significantly larger than all these datasets and is the only dataset that supports extracting triplet knowledge from consecutive pages. In contrast, previous datasets are built based on single-page documents and focus to extract unary spans. We hope the large-scale LIE dataset with complete input modality and task formulation could drive VRDs-based IE from form understanding forward to accurate document understanding. We analyze LIE in detail and plot the data distribution in Figure 2 to support the follow-up study. Figure 2(a) and 2(b) depict the distribution of pages and words in a document, respectively, and Figure 2(c) show the distribution of section length. Obviously, the documents have a large number of pages and words, and even one section may exceed 500 words. The distance distribution between two entity arguments in relation extraction is drawn in Figure 2(d), an intuitive observation is that about half of the arguments are far away in text sequence, or even not in the same section. By and large, all the above are beyond the focus of most existing IE benchmarks and pose new challenges to document understanding and extraction modules.

4 METHOD

Different from plain text, there is rich semantic knowledge hidden under the textual format and layout structure in VRDs, so traditional token-based language models are not directly applicable. In this section, we incorporate 2D layout features into transformer-based models and further propose specific solutions for the IE tasks.

4.1 Base Architecture

Figure 3 shows the architecture of our document encoder. Similar to LayoutLMv2 [34], we built the document encoder in a layout-aware manner, which accepts information from both text and layout. Beyond the 1D position embedding that models the sequential knowledge, we introduce 2D layout embedding to capture the spatial position in documents. We normalize and discretize all 2D coordinates to integers in the range [0,1000] and lookup *four* layout embeddings from *two* embedding tables, to embed x-axis and y-axis features separately. In the end, the textual embedding is added with segment, position, and layout embeddings to get the ultimate input of the transformer encoder. Based on the self-attention mechanism, embedding 2D knowledge into language models will better align layout features with the semantic representation. Finally, the output contextual representation of the transformer can be utilized for the following task-specific layers.

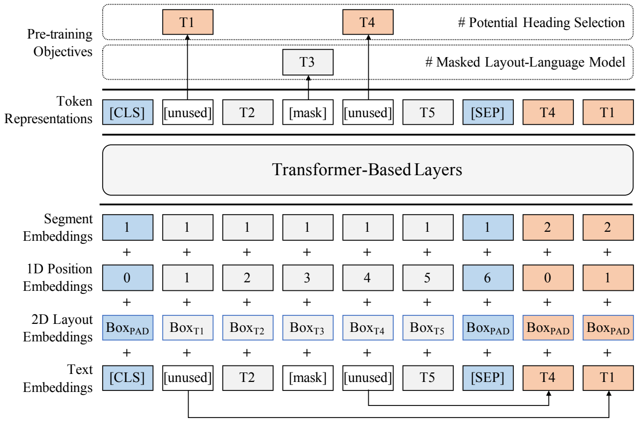


Figure 3: Diagram of the layout-aware language model and pre-training strategies, where 2D layout embeddings are integrated into input layer and potential head selection is proposed to learn document structure. T1 and T4 refer to the selected potential headings in the pre-training stage, we simplify them into one token for a good visualization.

4.2 Sequence Labeling Solution

To quantitatively evaluate the challenges of LIE, we propose a set of solutions for specific tasks whose design philosophy is trying to ensure simplicity. Here, we unify all the tasks into a sequence labeling framework with different tagging schemes.

Hierarchy Extraction (HE). In LIE, there are usually multiple articles within a document. Here, we divide the headings in one document into four levels: article headlines (`L0`) and section headings at different hierarchies (`L1/2/3`). The tag set is defined as {`B-H`, `E-H`, `O`}, in which `B` and `E` mark the beginning and end of a heading, `H` ∈ {`L0`,`L1`,`L2`,`L3`} encodes its level. `O` represents the other tag, indicating that the word is independent of the extracted results. To determine the TOC of documents, we decode each heading span by way of boundary nearest matching [43] and number them in order according to their levels and positions.

Section Extraction (SE). Similarly, we label the spans of section headings and bodies with the tag set {`B-H`, `B-B`, `E-H`, `E-B`, `O`}. In the decoding process, headings and bodies are combined into a section based on the nearest principle.

Relation Extraction (RE). The labeling model is quite more complex than that in section extraction as it needs to extract two entities, match them, and determine their relations. We design a tripart tagging scheme {B-S, E-S, B-O-Rel, E-O-Rel, O}, where S and O denote the subject and object entities, Rel$\in \mathcal{R}$ denotes the relation from pre-defined set. Thus, the total number of tags is $2 \times |\mathcal{R}| + 3$. During decoding, we first detect the subject entity, then the object entity with its involved relation, and finally pair them according to the nearest principle.

4.3 Pre-Training Strategy

Inspired by the great success of pre-trained language models, we pre-train the base encoder, to optimize the newly introduced parameters (i.e., layout embedding tables) and transfer token-based models to the visually rich dataset, with the following self-supervised objectives. Here, $100k$ additional pages from $10k$ in-domain documents ($5k$ product and official documents, respectively) are collected and parsed as the pre-training corpus.

Masked Layout-Language Model (MLLM). Similar to the famous masked language model task in BERT [8], the objective of MLLM is to recover the masked text token based on its text context and whole layout clues. The layout embeddings remain unchanged for masked token, which means that the model knows each masked token's location on the page, thereby equipping the token-based model with layout awareness.

Potential Heading Selection (PHS). To help the model learn structural and semantic knowledge among text fragments, we propose the PHS task. Specifically, we randomly select 15% of the training instances and develop heuristic rules to find text fragments with significant visual features (e.g., large font size) as potential headings. Next, these fragments are masked in the original sequence and spliced to the end of the sequence after random scrambling. Similarly, only text embeddings are masked and layout embeddings are retained. During pre-training, the goal is to select the most appropriate fragment for each masked position. We calculate the cross-entropy loss in optimization. The key intuition behind this is that *an eye-catching text fragment is usually a potential heading*, which semantically dominates the body of a section. In this way, the model could learn better with multi-modality clues and prompt the understanding of long documents.

5 EXPERIMENTS

To maximize the reusability of LIE, we provide an official partition to split annotated documents into train, dev, and test sets in the ratio of 6:2:2, and report the statistics in Table 2. In this section, we conduct comprehensive experiments to evaluate the new benchmark, and also discuss some possible future directions for VRDs-based IE.

5.1 Experiment Setup

Implementation Details. We develop our model based on Transformers [32] and employ BERT [8] as backbone, with the official *bert-base-chinese*[2] model. The input sequence is trimmed to a maximum length of 512. The size of layout embedding is 768. The learning rates of fine-tuning are $1e^{-5}$ (chosen from $1e^{-3}$ to $1e^{-6}$). We optimize our model with Adam and run it on one 32G Tesla V100 GPU for 30 epochs, which takes 12/12 hours for the pre-training, and 1/0.5, 1/0.5, 1/0.5 hours for the fine-tuning of content extract, section extraction, relation extraction in product/official documents respectively. All hyper-parameters are tuned on the dev set.

Table 2: Statistics of the LIE train/dev/test datasets.

		Hierarchy Extraction	Section Extraction	Relation Extraction
Train	*Product*	7,590	6,561	2,034
	Official	2,830	2,390	2,081
Dev	*Product*	2,482	2,183	655
	Official	1,068	892	770
Test	*Product*	2,368	2,065	644
	Official	1,031	861	717

We intuitively name the layout-aware method *LayoutBERT*. To access the effects of pre-training strategies, we implement two advanced methods, *LayoutBERT (w/ MLLM)* and *LayoutBERT (w/ MLLM, PHS)*. Besides, we also report the results of *LayoutLMv2* [34], a representative state-of-the-art model on previous layout-aware IE datasets, as a strong baseline. It employs ResNeXt-FPN [33] to generate image embeddings and concatenates them with text embeddings. For fair comparison, *LayoutLMv2* is also pre-trained in the same dataset with *LayoutBERT.*

Evaluation Metrics. Following popular choices, we adopt *F1* score, the harmonic mean of *precision* and *recall*, for evaluation. Formally, precision and recall are defined as $P = S_m/|\mathcal{P}|$, $R = S_m/|\mathcal{G}|$, where S_m is the matching score of all predicted results, $|\mathcal{P}|$ and $|\mathcal{G}|$ are the size of prediction set and ground truth set, respectively. For *hierarchy extraction* and *relation extraction*, $S_m = \sum_i^{|\mathcal{P}|} m(p_i)$, where $m(p_i)$ is set to 1 if the i-th predicted result matches the ground truth strictly, otherwise it is 0. For *section extraction*, we calculate S_m, as the evaluation in Open IE systems, by computing the similarity between each predicted fact in $\mathcal{P}$ and each ground truth fact in $\mathcal{G}$, then find the optimal matching to maximize the sum of matched similarities by solving a linear assignment problem [26]. In the procedure, the similarity between two facts is defined as $s(p_i, g_j) = \sum_{l=1}^{2} \mathcal{M}(p_i^x, g_j^x)/2$, where p_i^x and g_j^x denote the x-th element (i.e., heading or body) of tuple p_i and g_j, $\mathcal{M}(\cdot,\cdot)$ denotes the gestalt pattern matching measure [25] for two strings.

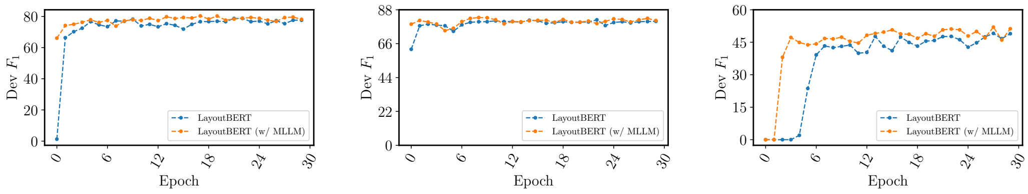
5.2 Results

Comparing the performance of different models in Table 3, the first conclusion we draw is that layout-aware models outperform vanilla token-based models in almost all the tasks, which demonstrates the effectiveness of incorporating layout features with pre-trained language models. Secondly, the process of in-domain pre-training further improves all of the download IE performance. Through further analysis, we also find that pre-training dramatically accelerates the convergence in all downstream tasks (see Figure 4). Thirdly, LayoutLMv2 fails in this task and only achieves comparable results as BERT. Further study suggests that for such text-centered VRDs, the low-resolution image introduced by LayoutLMv2 becomes a

[2] https://huggingface.co/bert-base-chinese

Table 3: Performance of different models on LIE (%). *Product, Official* and *Average* refer to product documents, official documents and average results, respectively. The improvement over baseline is significant (p-value < 0.05).

Models	Hierarchy Extraction			Section Extraction			Relation Extraction		
	Product	*Official*	*Average*	*Product*	*Official*	*Average*	*Product*	*Official*	*Average*
BERT [8]	75.9	71.2	73.6	80.2	70.5	75.4	52.3	71.9	62.1
LayoutBERT	76.8	73.1	75.0	82.8	71.5	77.2	52.1	71.1	61.6
LayoutBERT (w/ MLLM)	77.8	75.9	76.9	84.2	72.8	78.5	53.9	75.8	64.9
LayoutBERT (w/ MLLM, PHS)	**78.3**	**76.5**	**77.4**	**85.1**	**73.2**	**79.2**	**54.2**	**76.7**	**65.5**
LayoutLMv2 [34]	75.4	72.6	74.0	80.9	70.1	75.5	51.9	70.3	61.1



(a) Hierarchy extraction.

(b) Section Extraction.

(c) Relation Extraction.

Figure 4: Convergence processes of layout-aware models on LIE (product documents) with and without in-domain pre-training. To minimize model variables, we compare LayoutBERT (w/ MLLM) and LayoutBERT here.

Table 4: Performance w.r.t. input positional features on hierarchy extraction.

	Product	*Official*	*Average*
LayoutBERT (w/ MLLM)	77.8	75.9	76.9
w/o 1D embedding	49.5	57.4	53.5
w/o 2D embedding	77.0	73.6	75.3

noise interference rather than an information source. Therefore, designing a more reasonable multi-modality integration method is also a point that needs to be studied in the future.

When looking at the results vertically, one can find that: (1) Section extraction and relation extraction hold the highest and lowest performance, respectively. (2) The performance gaps between these two kinds of documents should not be overlooked. We roughly attribute them to the difference of annotated data, including the number of tag sets and training samples.

5.3 Analysis

To understand the dataset and method more deeply, we carry out the model performance with respect to different aspects:

Performance w.r.t. Input Features. Table 4 summarizes the ablation studies when removing different input features. The results show that both 1D position embedding and 2D layout embedding contribute to the final performance, especially when we remove the 1D position embedding, the performance tends to collapse. We believe that sequential information provides great help for document modeling, because there are many text fragments in the document and the 1D position reflects the reading order. Besides, the model is hard to converge if text embedding is discarded. It is contrary to the previous observation on form understanding dataset [3], which again proves that the new dataset requires a powerful natural language understanding ability.

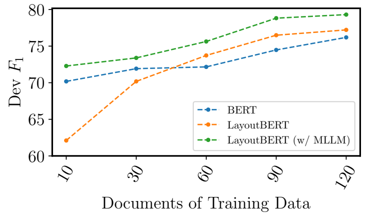
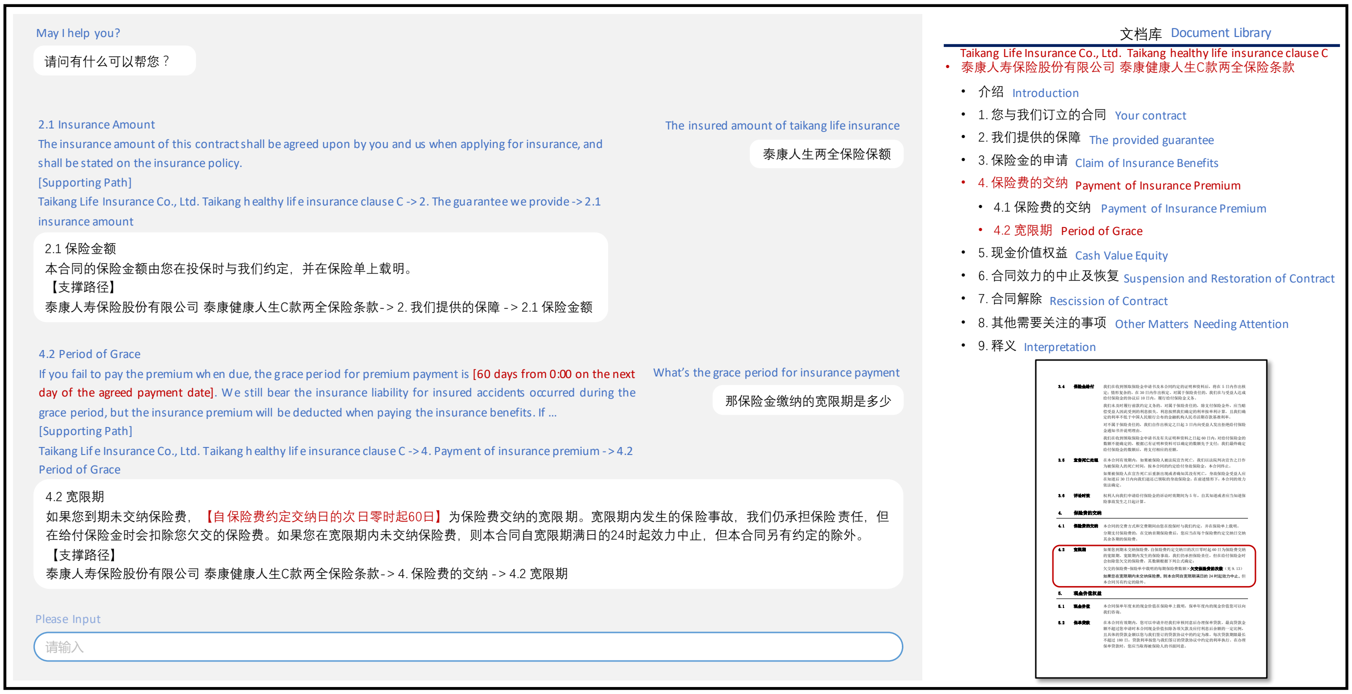


Figure 5: Performance w.r.t. training data on hierarchy extraction (product documents).

Performance w.r.t. Training Data. We randomly sample {10, 30, 60, 90, 120} documents from the training set to explore the performance with different amounts of training data. Figure 5 shows that the performance gain of LayoutBERT relative to BERT increases first and then decreases. We analyze that the newly introduced parameters cannot be fully optimized with limited training data. When the data increases to enough, BERT is also capable of capturing some semantic information related to decision-making. Unsurprisingly, the pre-training stage reduces the sensitivity of training data and brings stable improvements.

Performance w.r.t. Domain Transfer. For the exploration of domain transfer, we present evaluation results in Table 5. An interesting observation is that although the model is only trained on product documents, it can still transfer some knowledge to official



May I help you?

请问有什么可以帮您？

The insured amount of taikang life insurance

泰康人生两全保险保额

2.1 Insurance Amount
The insurance amount of this contract shall be agreed upon by you and us when applying for insurance, and shall be stated on the insurance policy.
[Supporting Path]
Taikang Life Insurance Co., Ltd. Taikang healthy life insurance clause C -> 2. The guarantee we provide -> 2.1 insurance amount

2.1 保险金额
本合同的保险金额由您在投保时与我们约定，并在保险单上载明。
【支撑路径】
泰康人寿保险股份有限公司 泰康健康人生C款两全保险条款-> 2. 我们提供的保障 -> 2.1 保险金额

What’s the grace period for insurance payment

那保险金缴纳的宽限期是多少

4.2 Period of Grace
If you fail to pay the premium when due, the grace period for premium payment is [60 days from 0:00 on the next day of the agreed payment date]. We still bear the insurance liability for insured accidents occurred during the grace period, but the insurance premium will be deducted when paying the insurance benefits. If ...
[Supporting Path]
Taikang Life Insurance Co., Ltd. Taikang healthy life insurance clause C -> 4. Payment of insurance premium -> 4.2 Period of Grace

4.2 宽限期
如果您到期未交纳保险费，【自保险费约定交纳日的次日零时起60日】为保险费交纳的宽限期。宽限期内发生的保险事故，我们仍承担保险责任，但在给付保险金时会扣除您欠交的保险费。如果您在宽限期内未交纳保险费，则本合同自宽限期满日的24时起效力中止，但本合同另有约定的除外。
【支撑路径】
泰康人寿保险股份有限公司 泰康健康人生C款两全保险条款-> 4. 保险费的交纳 -> 4.2 宽限期

文档库 Document Library

Taikang Life Insurance Co., Ltd. Taikang healthy life insurance clause C
- 泰康人寿保险股份有限公司 泰康健康人生C款两全保险条款
  - 介绍 Introduction
  - 1. 您与我们订立的合同 Your contract
  - 2. 我们提供的保障 The provided guarantee
  - 3. 保险金的申请 Claim of Insurance Benefits
  - 4. 保险费的交纳 Payment of Insurance Premium
    - 4.1 保险费的交纳 Payment of Insurance Premium
    - 4.2 宽限期 Period of Grace
  - 5. 现金价值权益 Cash Value Equity
  - 6. 合同效力的中止及恢复 Suspension and Restoration of Contract
  - 7. 合同解除 Rescission of Contract
  - 8. 其他需要关注的事项 Other Matters Needing Attention
  - 9. 释义 Interpretation

Figure 6: Demonstration of the online doc2dial service. The left part is the interactive dialogue interface and the right shows current supporting path in relevant document, which is changes dynamically as user utterance change (The deployed system is in Chinese, and we add English annotations here for better understanding).

Table 5: Performance w.r.t. domain transfer on section extraction. Row/column denote the train/test domains.

	Product	*Official*	*Average*
Product	85.1	9.8	47.5
Official	30.5	73.2	51.9
All (Product+Official)	83.0	68.1	75.6

documents, and vice versa. Moreover, if documents from other domains are introduced in the training phase, it causes some negative impacts but reach a high average performance on the whole. This verifies that LayoutBERT captures common layout and format invariance among different documents and transfer to other domains for document understanding. Besides, introducing more data is a possible solution to further improve performance, even if new data comes from different fields or styles.

5.4 Further Discussions

We investigate the model outputs on dev set and find that there are some typical errors for current models: (1) In *hierarchy extraction*, the predicted results miss some section headings. For example, the model outputs the first and third section headings but misses the second heading, resulting in a continuous error after the second heading if we explicitly generate TOC number. It might be useful to introduce auxiliary training objectives to model headings with same format. (2) In *section extraction*, errors are mainly lie in the prediction of long section bodies (especially in official documents). Note that many bodies appear between two headings, so considering TOC of the document or replacing base encoder with a long-form language model (e.g., BigBird [39]) may be possible solutions. (3) In *relation extraction*, the model cannot work very well if two entities are far apart, which is the new challenge posed by LIE and accounts for a large proportion. In extreme cases, the head and tail entities are located at the document’s beginning and end, respectively. We believe that introducing graph neural networks based on document structure and layout to explicitly enhance the correlation between entities will be a direction worthy of exploration.

6 DEMONSTRATION

Figure 6 shows the system demonstration of our doc2dial service in a real-world cloud computing platform, which is powered by the layout-aware IE model, a business document library, and a series of online dialogue-oriented algorithms [7, 20, 23]. Following the doc2dial framework [10], we perform knowledge extraction with the above three IE tasks in advance. In online service, given an utterance like "The insured amount of taikang life insurance", the doc2dial system first retrieves relevant document from *Document Library*, and then returns a section-level coarse-grained response with supporting path, i.e., the location of this section in the document. These two capabilities benefit from the off-line *section extraction* and *hierarchy extraction* processes, respectively. In the next round, the user asked for "grace period", the system locates in section 4.2 by performing similar workflow, and highlights "60 days from 0:00 on the next day of the agreed payment date" as a fine-grained response, since the span was extracted from relation extraction in advance and it may be the information that the user is most concerned about. To ensure the integrity and professionalism of response, we reserve all text of the answer section.

7 CONCLUSION

Nowadays, the digitalization and intellectualization of various industries are accelerating, we believe that mining and organizing valuable knowledge from massive data to enable downstream applications of enterprises is an important track for the industrial application of AI technology. In this paper, we release a systematic novel layout-aware IE dataset and method for document-grounded dialogue systems, and highlight some possible future directions for VRD-based IE. Experimental results suggest the effectiveness of incorporating pre-trained language models with layout features, and the demonstration confirms the importance of both structural and semantic knowledge in the doc2dial system.

ACKNOWLEDGMENTS

We would like to thank the anonymous reviewers for their insightful comments and constructive suggestions. This work is supported by the National Key Research and Development Program of China (grant No.2021YFB3100600), the Strategic Priority Research Program of Chinese Academy of Sciences (grant No.XDC02040400) and the Youth Innovation Promotion Association of Chinese Academy of Sciences (grant No.2021153).

REFERENCES

[1] Srikar Appalaraju, Bhavan Jasani, Bhargava Urala Kota, Yusheng Xie, and R Manmatha. Docformer: End-to-end transformer for document understanding. In *ICCV*, 2021.

[2] Jon Ander Campos, Arantxa Otegi, Aitor Soroa, Jan Milan Deriu, Mark Cieliebak, and Eneko Agirre. Doqa-accessing domain-specific faqs via conversational qa. In *ACL*, 2020.

[3] Rongyu Cao and Ping Luo. Extracting zero-shot structured information from form-like documents: Pretraining with keys and triggers. In *AAAI*, 2021.

[4] Xi Chen, Faner Lin, Yeju Zhou, Kaixin Ma, Jonathan Francis, Eric Nyberg, and Alessandro Oltramari. Building goal-oriented document-grounded dialogue systems. In *DialDoc*, 2021.

[5] Xiuyi Chen, Fandong Meng, Peng Li, Feilong Chen, Shuang Xu, Bo Xu, and Jie Zhou. Bridging the gap between prior and posterior knowledge selection for knowledge-grounded dialogue generation. In *EMNLP*, 2020.

[6] Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wen-tau Yih, Yejin Choi, Percy Liang, and Luke Zettlemoyer. Quac: Question answering in context. In *EMNLP*, 2018.

[7] Yinpei Dai, Hangyu Li, Chengguang Tang, Yongbin Li, Jian Sun, and Xiaodan Zhu. Learning low-resource end-to-end goal-oriented dialog for fast and reliable system deployment. In *ACL*, 2020.

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, 2019.

[9] Kshitij Fadnis, Pankaj Dhoolia, Li Zhu, Q Vera Liao, Steven Ross, Nathaniel Mills, Sachindra Joshi, and Luis Lastras. Doc2bot: Document grounded bot framework. In *AAAI*, 2021.

[10] Song Feng, Kshitij Fadnis, Q Vera Liao, and Luis A Lastras. Doc2dial: a framework for dialogue composition grounded in documents. In *AAAI*, 2020.

[11] Song Feng, Hui Wan, Chulaka Gunasekara, Siva Patel, Sachindra Joshi, and Luis Lastras. doc2dial: A goal-oriented document-grounded dialogue dataset. In *EMNLP*, 2020.

[12] Wanwei He, Yinpei Dai, Min Yang, Jian Sun, Fei Huang, Luo Si, and Yongbin Li. Unified dialog model pre-training for task-oriented dialog understanding and generation. In *SIGIR*, 2022.

[13] Wanwei He, Yinpei Dai, Yinhe Zheng, Yuchuan Wu, Zheng Cao, Dermot Liu, Peng Jiang, Min Yang, Fei Huang, Luo Si, Jian Sun, and Yongbin Li. Galaxy: A generative pre-trained model for task-oriented dialog with semi-supervised learning and explicit policy injection. In *AAAI*, 2022.

[14] Iris Hendrickx, Su Nam Kim, Zornitsa Kozareva, Preslav Nakov, Diarmuid Ó Séaghdha, Sebastian Padó, Marco Pennacchiotti, Lorenza Romano, and Stan Szpakowicz. Semeval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In *SemEval*, 2010.

[15] Zheng Huang, Kai Chen, Jianhua He, Xiang Bai, Dimosthenis Karatzas, Shijian Lu, and CV Jawahar. Icdar2019 competition on scanned receipt ocr and information extraction. In *ICDAR*, 2019.

[16] Sarthak Jain, Madeleine van Zuylen, Hannaneh Hajishirzi, and Iz Beltagy. Scirex: A challenge dataset for document-level information extraction. In *ACL*, 2020.

[17] Guillaume Jaume, Hazim Kemal Ekenel, and Jean-Philippe Thiran. Funsd: A dataset for form understanding in noisy scanned documents. In *ICDAR: Workshops*, 2019.

[18] Sopan Khosla, Justin Lovelace, Ritam Dutt, and Adithya Pratapa. Team jars: Dialdoc subtask 1-improved knowledge identification with supervised out-of-domain pretraining. In *DialDoc*, 2021.

[19] Chenliang Li, Bin Bi, Ming Yan, Wei Wang, Songfang Huang, Fei Huang, and Luo Si. Structurallm: Structural pre-training for form understanding. In *ACL*, 2021.

[20] Jinpeng Li, Yingce Xia, Rui Yan, Hongda Sun, Dongyan Zhao, and Tie-Yan Liu. Stylized dialogue generation with multi-pass dual learning. In *NeurIPS*, 2021.

[21] Shuangjie Li, Wei He, Yabing Shi, Wenbin Jiang, Haijin Liang, Ye Jiang, Yang Zhang, Yajuan Lyu, and Yong Zhu. Duie: A large-scale chinese dataset for information extraction. In *NLPCC*, 2019.

[22] Yulin Li, Yuxi Qian, Yuechen Yu, Xiameng Qin, Chengquan Zhang, Yan Liu, Kun Yao, Junyu Han, Jingtuo Liu, and Errui Ding. Structext: Structured text understanding with multi-modal transformers. In *ACM MM*, 2021.

[23] Che Liu, Rui Wang, Jinghua Liu, Jian Sun, Fei Huang, and Luo Si. Dialoguecse: Dialogue-based contrastive learning of sentence embeddings. In *EMNLP*, 2021.

[24] Seunghyun Park, Seung Shin, Bado Lee, Junyeop Lee, Jaeheung Surh, Minjoon Seo, and Hwalsuk Lee. Cord: A consolidated receipt dataset for post-ocr parsing. In *NeurIPS: Workshop*, 2019.

[25] John W Ratcliff and David E Metzener. Pattern matching: The gestalt approach. *Dr Dobbs Journal*, 13(7):46, 1988.

[26] Mingming Sun, Xu Li, Xin Wang, Miao Fan, Yue Feng, and Ping Li. Logician: A unified end-to-end neural approach for open-domain information extraction. In *WSDM*, 2018.

[27] Hui Wan, Song Feng, Chulaka Gunasekara, Siva Sankalp Patel, Sachindra Joshi, and Luis Lastras. Does structure matter? encoding documents for machine reading comprehension. In *NAACL*, 2021.

[28] Jiapeng Wang, Chongyu Liu, Lianwen Jin, Guozhi Tang, Jiaxin Zhang, Shuaitao Zhang, Qianying Wang, Yaqiang Wu, and Mingxiang Cai. Towards robust visual information extraction in real world: New dataset and novel solution. In *AAAI*, 2021.

[29] Yucheng Wang, Bowen Yu, Yueyang Zhang, Tingwen Liu, Hongsong Zhu, and Limin Sun. Tplinker: Single-stage joint extraction of entities and relations through token pair linking. In *COLING*, 2020.

[30] Mengxi Wei, Yifan He, and Qiong Zhang. Robust layout-aware ie for visually rich documents with pre-trained language models. In *SIGIR*, 2020.

[31] Zhongyu Wei, Qianlong Liu, Baolin Peng, Huaixiao Tou, Ting Chen, Xuan-Jing Huang, Kam-Fai Wong, and Xiang Dai. Task-oriented dialogue system for automatic diagnosis. In *ACL*, 2018.

[32] Thomas Wolf, Julien Chaumond, Lysandre Debut, Victor Sanh, Clement Delangue, Anthony Moi, Pierric Cistac, Morgan Funtowicz, Joe Davison, Sam Shleifer, et al. Transformers: State-of-the-art natural language processing. In *EMNLP: Demo*, 2020.

[33] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *CVPR*, 2017.

[34] Yang Xu, Yiheng Xu, Tengchao Lv, Lei Cui, Furu Wei, Guoxin Wang, Yijuan Lu, Dinei Florencio, Cha Zhang, Wanxiang Che, et al. Layoutlmv2: Multi-modal pre-training for visually-rich document understanding. In *ACL*, 2021.

[35] Yiheng Xu, Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, and Ming Zhou. Layoutlm: Pre-training of text and layout for document image understanding. In *KDD*, 2020.

[36] Yuan Yao, Deming Ye, Peng Li, Xu Han, Yankai Lin, Zhenghao Liu, Zhiyuan Liu, Lixin Huang, Jie Zhou, and Maosong Sun. Docred: A large-scale document-level relation extraction dataset. In *ACL*, 2019.

[37] Bowen Yu, Zhenyu Zhang, Xiaobo Shu, Tingwen Liu, Yubin Wang, Bin Wang, and Sujian Li. Joint extraction of entities and relations based on a novel decomposition strategy. In *ECAI*, 2020.

[38] Dian Yu, Kai Sun, Claire Cardie, and Dong Yu. Dialogue-based relation extraction. In *ACL*, 2020.

[39] Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and Amr Ahmed. Big bird: Transformers for longer sequences. In *NeurIPS*, 2020.

[40] Peng Zhang, Yunlu Xu, Zhanzhan Cheng, Shiliang Pu, Jing Lu, Liang Qiao, Yi Niu, and Fei Wu. Trie: End-to-end text reading and information extraction for

document understanding. In *ACM MM*, 2020.
[41] Zhenyu Zhang, Bowen Yu, Xiaobo Shu, Tingwen Liu, Hengzhu Tang, Wang Yubin, and Li Guo. Document-level relation extraction with dual-tier heterogeneous graph. In *COLING*, 2020.
[42] Zhenyu Zhang, Bowen Yu, Xiaobo Shu, and Tingwen Liu. Na-aware machine reading comprehension for document-level relation extraction. In *ECML/PKDD*, 2021.
[43] Suncong Zheng, Feng Wang, Hongyun Bao, Yuexing Hao, Peng Zhou, and Bo Xu. Joint extraction of entities and relations based on a novel tagging scheme. In *ACL*, 2017.
[44] Shaowen Zhou, Bowen Yu, Aixin Sun, Cheng Long, Jingyang Li, Jian Sun, and Li Yongbin. A survey on neural open information extraction: Current status and future directions. In *IJCAI*, 2022.

A MORE DETAILS ABOUT DATASET

To the best of our knowledge, LIE is the first document understanding dataset for text-centered VRDs. We believe it will bring practical doc2dial systems and benefit the VRD-based IE community because the existing datasets only focus on form understanding and span extraction. And it potentially has broad impacts since the tackled issues also widely exist in tasks of other areas. Here, we attach some details about the construction of LIE that are not mentioned in the text to help understand its value.

Data Collection.[3] We download product documents from the *Insurance Association of China* website[4]. To collect official documents, we first search for words such as "`policy documents`", then obtain the PDF documents from the results on the "`gov.cn`" websites, and finally develop some heuristic rules to remove these unsuitable documents (e.g., blanks tables). As introduced in the main text, LIE contains 18 and 15 relations for product and official documents. We list all the relations in Table 6, which are designed under the practical situation.

Data Annotation. In the crowdsourcing process, we offer annotators $14 for each document ($7 for the hierarchy and section extraction, $7 for relation extraction), which are the prices given by a third-party professional appraisal agency. Specially, for relation extraction, annotators are also asked to provide the table-of-contents number of the section where the tail entity is located, which should be strictly consistent with the annotations of hierarchy extraction, to help find the entities and indicate information.

Annotation Alignment. To align the artificial annotations with automatic parsing results, we observe that PDFPlumber usually outputs spaces at the end of lines, even in a continuous paragraph. To handle this issue, we remove these spaces according to annotations with some heuristic rules. Inevitably, some annotated fields cannot be found in parsing results. In that cases, we try to match the start and end substrings of annotated field and locate text segment with the minimum edit distance from the annotation.

Annotation Speed. To investigate the annotation speed, we employ three experts (Ph.D. students in NLP field) to annotate LIE. For the extraction of TOC and sections, it takes about *10 mins* to annotate 100 triplets (*TOC number, heading, body*). The speed of relation extraction is much slower, that is *1 hour* for 100 triples (*subject, relation, object*). As a comparison, we also ask them to annotate DuIE [21], a recent sentence-level relation extraction dataset, and it takes only *15 mins* for 100 triplets[5]. It is expected since document-level extraction requires an understanding of the whole document to annotate entities and their relationships that usually span beyond sentences or even paragraphs.

Table 6: Predefined schemas of relation extraction.

Relation Type (*Product*)	Number	Ratio (%)
Insurance Liability	554	16.62
Approved Time Limit	336	10.08
Beneficiary	283	8.49
Mode of Payment	227	6.81
Insurance Period	214	6.42
Company	200	6.01
Extinctive Prescription	189	5.67
Claim Time Limit	177	5.31
Refund Time Limit for Termination	176	5.28
Period of Payment	170	5.11
Notification Time Limit for Accident	155	4.65
Waiting Period	144	4.32
Age at Issue	106	3.18
Hesitation Period	106	3.18
Grace Period	98	2.94
Recovery Time Limit	85	2.55
Loan Time Limit	57	1.71
Loan Ceiling	56	1.68
Total	3,333	100

Relation Type (*Official*)	Number	Ratio (%)
Carbon Copy Recipients	750	21.02
Quote	650	18.22
Main Recipients	491	13.76
Documents Number	426	11.94
Appendix	261	7.32
Issuing Agency	223	6.25
Issuing Date	195	5.47
Release Agency	169	4.74
Release Date	169	4.74
Effective Date	85	2.38
Expiration Date	66	1.85
Contact Number	55	1.54
Period of Validity	19	0.53
Fax Number	5	0.14
Email Address	4	0.11
Total	3,568	100

[3] We focus on Chinese documents in this work and will expand to other languages in the future.
[4] http://www.iachina.cn/
[5] For fair comparison, we utilize predefined schemes with the same size here.