

DuReader_{vis}: A Chinese Dataset for Open-domain Document Visual Question Answering

Le Qi^{*}, Shangwen Lv², Hongyu Li², Jing Liu², Yu Zhang^{1†},
Qiaoqiao She², Hua Wu², Haifeng Wang² and Ting Liu¹

¹Research Center for Social Computing and Information Retrieval,
Harbin Institute of Technology, Harbin, China

²Baidu Inc.

{lqi, zhangyu, tliu}@ir.hit.edu.cn

{lvshangwen, lihongyu04, liujing46, sheqiaoqiao, wu_hua, wanghaifeng}@baidu.com

Abstract

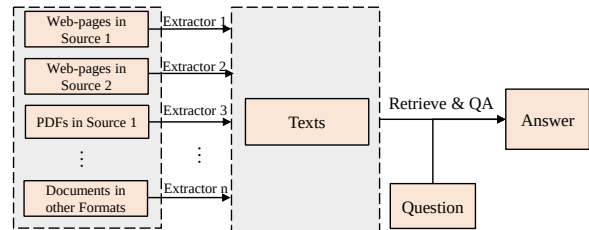
Open-domain question answering has been used in a wide range of applications, such as web search and enterprise search, which usually takes clean texts extracted from various formats of documents (e.g., web pages, PDFs, or Word documents) as the information source. However, designing different text extraction approaches is time-consuming and not scalable. In order to reduce human cost and improve the scalability of QA systems, we propose and study an **Open-domain Document Visual Question Answering** (Open-domain DocVQA) task, which requires answering questions based on a collection of document images directly instead of only document texts, utilizing layouts and visual features additionally. To advance this task, we introduce the first Chinese Open-domain DocVQA dataset called DuReader_{vis}, containing about 15K question-answering pairs and 158K document images from the Baidu search engine. There are three main challenges in DuReader_{vis}: (1) long document understanding, (2) noisy texts, and (3) multi-span answer extraction. The extensive experiments demonstrate that the dataset is challenging. Additionally, we propose a simple approach that incorporates the layout and visual features, and the experimental results show the effectiveness of the proposed approach. The dataset and code will be publicly available at <https://github.com/baidu/DuReader/tree/master/DuReader-vis>.

1 Introduction

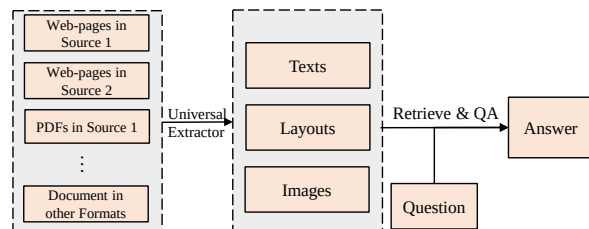
Open-domain Question Answering (Open-domain QA) is a task that requires answering questions based on a collection of document texts. It has been used in a wide range of applications, such as web search (He et al., 2018; Nguyen et al., 2016;

^{*} The work was done when Le Qi was doing internship at Baidu.

[†] Corresponding author.



(a) The procedure in open-domain QA, which first utilizes different content extraction methods to get textual contents.



(b) The procedure in open-domain DocVQA, which utilizes a universal extractor to get textual contents and layout information.

Figure 1: The procedure comparison between open-domain QA and open-domain DocVQA.

Chen et al., 2017a), enterprise QA (Castelli et al., 2020), biomedical QA (Levy et al., 2021), etc.

The typical procedure of an open-domain QA system can be summarized in Figure 1(a). It needs first to design specific text extraction methods for real-world documents in different formats (e.g., PDFs, web pages, scanned documents, etc.), and extract certain text contents from them (e.g. the main body of web pages). Since there is no universal method for text extraction, it is expensive to build a unified QA system that can process documents in different formats as the information source. This greatly limits the scalability of QA systems, where a scalable QA system should process various formats of documents at a low cost, and not be restricted by the document format. In addition, the visual layouts (e.g., font size, list format, and table format) and the visual features (e.g., text color, pictures, and figures) will be lost after text extrac-

tion, which are of great significance to comprehend documents.

To tackle the above limitations, we propose and study an **Open-domain Document Visual Question Answering** (Open-domain DocVQA) task, which takes a collection of document images (converted from real-world documents) as the information source to answer questions, as shown in Figure 1(b). In this task, we apply a universal document extractor (e.g., OCR) to extract all the texts and layouts from the document images and then utilize them along with the visual features to perform the following procedures, including **Document Visual REtriever** (DocVRE) to retrieve relevant document images, and **Document Visual Question Answering** (DocVQA) to extract answers from retrieved document images. The open-domain DocVQA task encourages us to design an open-domain QA system that can be applied to various data sources in a scalable way, leveraging text, layout, and visual information simultaneously.

In open-domain QA, it is intuitive to build the corresponding datasets from the ones for machine reading comprehension (that requires answering questions based on one or a few documents), e.g., Natural Questions Open (Kwiatkowski et al., 2019), SQuAD-open (Chen et al., 2017b). With the development of document intelligence research, several datasets of visual machine reading comprehension (or question answering) have been created, such as VisualMRC (Tanaka et al., 2021), InfographicVQA (Mathew et al., 2021a) and DocVQA¹ (Mathew et al., 2021b). However, the questions are collected in a crowd-sourced way rather than from real-users' information seeking questions which makes them not suitable for **Open-Domain DocVQA** research. Besides, most document images in existing datasets are short documents with simple layouts and few visual features, but we often need to take more complex documents in open-domain scenarios. Furthermore, the answers in existing datasets are mainly short (e.g., entities, numbers, etc.). In contrast, we have longer answers in various formats like paragraphs, lists, and tables in real applications. Except the above limitations, there are very few Chinese datasets to the best of our knowledge.

To deal with the limitations above, we intro-

¹In other literature, DocVQA was used as the terminology referring to visual machine reading comprehension (Mathew et al., 2021b), we follow the task name as the DocVQA stage in our task.

duce DuReader_{vis}, the first Chinese open-domain DocVQA dataset, to promote the studies in Open-Domain DocVQA. Specifically, we collect questions and document images from Baidu Search². The questions are real ones issued by users to the search engine. Besides, the document images are converted from web pages that are easy to obtain with long documents, complex layouts, and rich visual features. In addition, the answers in DuReader_{vis} contain long answers, such as multi-span texts, lists, and tables. In total, DuReader_{vis} contains 14K unique questions, 158K document images, and 15K manually annotated question-answer pairs.

In this paper, we propose a simple approach incorporating text, layout, and visual features and conduct extensive experiments on DuReader_{vis}. The experimental results show that there are three main challenges (Section 5.4.1) in DuReader_{vis}: 1) long document understanding, where the document images are converted from long documents with rich visual features and complex layouts; 2) noisy texts, such as the advertisements and related links in web-pages, increasing the difficulty of understanding the documents; and 3) multi-span answer extraction, where the actual answers could be multi-span texts, lists, and tables. Furthermore, the additional zero-shot study (Section 5.4.2) on real-world documents in different formats (including PDFs, Word documents, and scanned images) demonstrates the scalability of our approach and the good transferability of models trained on DuReader_{vis}.

Our main contributions are as follows:

- We propose and study an open-domain DocVQA task to encourage developing an open-domain QA system that can be applied to various data sources in a scalable way, without the expensive and specific efforts to text extraction.
- We introduce the first Chinese open-domain DocVQA dataset DuReader_{vis} with three main challenges: long document understanding, noisy texts, and multi-span answer extraction.
- We propose a simple baseline method as the open-domain DocVQA baseline, and the gap between the baseline and human performance shows huge room for improvement.

²<https://www.baidu.com>

Dataset	Task	#Query	#Images	Source of Query	Source of Images	Answer Type	Answer-span Type
DocVQA	DocVQA	30K	12K	Crowdsourced	Industry documents	Extractive	Single
VisualMRC	DocVQA	50K	10K	Crowdsourced	Fixed-format webpages	Abstractive	-
InfographicVQA	DocVQA	30K	5.4K	Crowdsourced	Infographics	Extractive, Number Reasoning	Single, Multi
DuReader _{vis}	Open-domain DocVQA	15K	158K	User logs	Open-domain webpages	Extractive	Single, Multi

Table 1: The comparison between DuReader_{vis} and existing DocVQA datasets. # denotes “the number of”.

2 Related Work

2.1 Open-domain Question Answering

Open-domain Question Answering (open-domain QA) is a task of finding answers to the question from a large collection of textual documents. Many datasets of different domains have been proposed, varying from web search (e.g. Natural Questions (Kwiatkowski et al., 2019); SQuAD-open (Chen et al., 2017b); SearchQA (Dunn et al., 2017); MS-MARCO (Nguyen et al., 2016)), enterprise search (Castelli et al., 2020) to biomedical QA about COVID (Levy et al., 2021).

In previous works, a two-stage approach is usually used to solve the task, i.e. a document retrieval stage with BM25 (Chen et al., 2017b) or dense retrieval (Karpukhin et al., 2020a; Qu et al., 2021a), and a document question answering stage with a machine reading comprehension model (Karpukhin et al., 2020a; Mao et al., 2020). However, as mentioned in Section 1, the specific text extraction method makes the real open-domain QA applications hard to be scalable and loses layouts and visual features that may be necessary for document understanding.

2.2 Document Visual Question Answering

Document Visual Question Answering (DocVQA) is a task to answer questions based on a given real-world document image. In DocVQA (Mathew et al., 2021b), document images are collected from the Industry Documents Library, covering different document types like tables, forms, and figures, while the answers are mainly entities and numbers. VisualMRC (Tanaka et al., 2021) is an abstractive DocVQA task, where document images are a small part of a Wikipedia web page. InfographicVQA (Mathew et al., 2021a) focuses on elementary reasoning skills such as counting, sorting, and arithmetic operations. Nevertheless, all the questions in these datasets are not information-seeking questions (Dasigi et al., 2021) from real users but are generated by annotators with known documents, making these datasets unsuitable to be extended as open-domain DocVQA datasets. Be-

sides, these datasets have few document images with long documents and complex layouts, and their answers are mainly short answers such as entities and numbers.

As a comparison, we focus on building a new dataset DuReader_{vis}, which consists of (i) real questions from real-world users; (ii) long document images; (iii) long annotated answers with various answer types and multi-span answers. A detailed comparison between DuReader_{vis} and existing DocVQA datasets is shown in Table 1.

3 DuReader_{vis}

This section defines the task formally, then shows the data collection and annotation process, and finally conducts the statistics and analysis.

3.1 Task Overview

DuReader_{vis} is a Chinese dataset for Open-domain DocVQA. Given a collection of document images \bar{I} as the information source, a system is asked to extract one or multiple text spans from \bar{I} as the answer A of the question Q . The task contains two stages: 1) the Document Visual Retrieval (DocVRE) stage to retrieve relevant document images \hat{I} that may answer the question Q from the whole document image collection \bar{I} ($|\bar{I}| \gg |\hat{I}|$); and 2) the Document Visual Question Answering (DocVQA) stage to extract the answer A from the relevant document image set \hat{I} .

3.2 Data Collection and Annotation

This subsection describes the data collection, the annotation procedure and the quality control during annotations.

3.2.1 Question Collection

We randomly sample 40K queries from the search log of Baidu and then apply a pre-trained question classifier (with precision and recall higher than 92%) to filter out non-question queries, leaving about 18K queries. Then, we ask annotators to further filter out pornography or violence-related questions. Eventually, we hold about 16K questions.

3.2.2 Document Image Collection

After question collection, we need to collect document images for the the DocVRE and DocVQA stages in our task. For the DocVQA stage, we take the whole screenshots³ of the top-4 web pages in the Baidu search results (drop the unavailable web pages) by an open-source tool Puppeteer⁴ for each question in the collected 16K questions as the document images to annotate answers. Then, to build a larger document collection for the DocVRE stage in our task, we randomly sample more document images in the same way through other insensitive queries. Finally, there are about 158K document images in our collection.



解决蓝牙配件与 iOS 设备无法配对的问题?

- 检查蓝牙配件是由兼容 iOS 设备;
- 确认蓝牙配件是否开机;
- 检查蓝牙配件的剩余电量;
- 前往 iOS 设备「设置」-「蓝牙」, 检查是否开启「蓝牙」功能;
- 关机并重新启动 iOS 设备, 并重新关闭「蓝牙」再次打开;

Q: 蓝牙配件连不上iphone怎么办? (What should I do if the Bluetooth accessories cannot be connected to the iphone?)
A: ●检查蓝牙配件是否兼容iOS 设备; (● Check whether the Bluetooth accessories are compatible with iOS devices;)
●确认蓝牙配件是否开机; (● Confirm whether the Bluetooth accessories are turned on;)
.....

Figure 2: An example in DuReader_{vis}. Since the original document image is too large, we only show a part of it and indicate the answer by the red bounding box. The answer is a list in the example.

3.2.3 Answer Annotation and Quality Control

Finally, we annotate answers through the collected 16K questions and their relevant document images. Each annotated sample consists of a question, one

³We do not extract clean texts from web-pages with complex extractors or utilize DOM (Document Object Model) structures as in (Chen et al., 2021) to represent web-pages because we aim to propose a scalable extractor for different document input formats, and we take the web-pages as one kind of data sources to verify the effectiveness.

⁴Available at <https://github.com/puppeteer/puppeteer>.

of its relevant document images, and the corresponding document URL. The annotator must extract the answer text and mark the answer type. The sample will be removed if the text content in the document image does not contain the correct answer. There are three answer types: text, list, and table. If the answer is in the list type or the table type, the annotators must annotate all the list items or the table cells that can answer questions. Finally, after filtering out questions with no annotated answers, we obtain 15K question-answer pairs, with 14K unique questions.

To ensure the data quality, we perform the annotation in an internal annotation platform, where all the annotators and reviewers are formal employees and native speakers. The data samples are divided into packages during annotation, with 1000 samples for each. For a single package, the annotators extract the answers first. Then at least two reviewers check the accuracy of this package by reviewing 100 random samples independently. If the average accuracy is below the threshold (i.e., 93%), the annotators will be asked to revise the answers, until the accuracy is higher than the threshold.

3.3 Statistics and Analysis

In this subsection, we will analyze the statistical features of DuReader_{vis}. DuReader_{vis} has 14K unique questions, and 158K document images, and 15K question-answer pairs in total. We randomly split the samples, and there are 11K, 1.5K, and 2.5K question-answer pairs in the training, development, and test sets. We will provide questions, document images, answers, and document URLs in our dataset and make the dataset public only for research purposes. There is an example of the question-answer pair shown in Figure 2.

3.3.1 Document Images

As shown in Table 2, the average length of textual contents in the document images of DuReader_{vis} is 1968.21, which is significantly longer than DocVQA (182.75), VisualMRC (151.46) and InfographicVQA (217.89). Modeling such a long sequence is a challenging task for many pre-trained language models (e.g., Devlin et al. 2019; Liu et al. 2019) due to limited input length (usually less than 512 tokens), making the first challenge in DuReader_{vis}.

In addition, the document images in DuReader_{vis} come from over 17K random websites, thus are diverse in topics and document layouts. With such

long documents, rich visual features, and complex layout, the noise in the sample will be inevitable, making it the second challenge in $\text{DuReader}_{\text{vis}}$.

3.3.2 Questions and Answers

Existing DocVQA datasets contain mostly factoid questions, with mainly short entities and numbers as answers. In comparison, $\text{DuReader}_{\text{vis}}$ contains both factoid and non-factoid questions. To demonstrate the diversity of question types in $\text{DuReader}_{\text{vis}}$, we randomly check 200 questions and classify their type as factoid or non-factoid, and the results show that there are 43% of the questions are non-factoid. Besides, the answers in $\text{DuReader}_{\text{vis}}$ are more complex. In fact, only 40% of the answers are normal text. There are 25% list answers and 35% table answers, of which the answers are likely to be discontinuous and have to be modeled as multi spans. As shown in Table 2, the average length of the answers in $\text{DuReader}_{\text{vis}}$ is 180.54, undoubtedly longer than the factoid answers in DocVQA (2.43), VisualMRC (9.53), and InfographicVQA (1.60). The long and multi-span answers make it the third challenge in $\text{DuReader}_{\text{vis}}$.

Dataset	Document Images		Answers
	Avg. #Tokens	Avg. Size	Avg. Length
DocVQA	151.46	(2,084, 1,776)	2.43
VisualMRC	182.75	-	9.55
InfographicVQA	217.89	(2,541, 1,181)	1.60
$\text{DuReader}_{\text{vis}}$	1986.21	(4,316, 2,054)	180.54

Table 2: Statistics of DocVQA datasets, where Avg. denotes ‘‘Average’’.

4 Proposed Model

In this section, we propose a simple baseline for $\text{DuReader}_{\text{vis}}$. The approach contains three parts: 1) the Universal Document Extractor to obtain textual contents, layout and visual information as the input, 2) the **Document Visual REtriever** (DocVRE) to retrieve relevant documents, and 3) the **Document Visual Question Answering** (DocVQA) to extract answers from retrieved documents.

4.1 Universal Document Extractor

Different formats or sources of documents require different content extractors. For example, we need to write different crawlers and parsers to extract texts from documents on different websites. The content extractor for PDFs and Words are also not

universal across different sources and tasks. Moreover, contents from scanned documents and images can only be extracted by OCR⁵. To extract contents from various formats of documents in a more universal and scalable way, we directly convert all formats of documents into document images and adopt OCR to obtain the texts and layouts.

Given a document image I , we firstly parse the document image by an OCR engine to obtain the textual document $D = \{d_0, d_1, \dots, d_i, \dots, d_n\}$ and the rectangular bounding boxes $B = \{b_0, b_1, \dots, b_i, \dots, b_n\}$, where n is the document length, d_i is the i -th token in the document, and $b_i = (x_0^i, y_0^i, x_1^i, y_1^i)$ denoting the left, top, right, and bottom position of the i -th token boundary.

4.2 Document Visual Retriever

DocVRE aims to retrieve relevant documents from an extensive collection of documents. In this paper, we adopt the text contents in document images to build the retrieval library and use BM25 (Robertson and Zaragoza, 2009) to retrieve relevant documents.

4.3 Document Visual Question Answering

DocVQA aims to extract answers in the documents returned by DocVRE, that contains two challenges: long document understanding and multi-span answer extraction. For the long document understanding, we utilize a Hierarchical LayoutXLM (Xu et al., 2021) (Hi-LayoutXLM) to model the interaction within the documents by using text, layout, and visual information. Then, we extract multi-span answers by a sequence labeling method based on CRF (Conditional Random Fields). The model of DocVQA includes three stages: the paragraph encoder, the document encoder, and the answer extractor, as shown in Figure 3.

Paragraph Encoders: We take LayoutXLM as the paragraph encoder, which accepts text, layout, and visual features as inputs. Due to the input length limitation of LayoutXLM, we split the document and the bounding box (D, B) into m groups from (\bar{D}_1, \bar{B}_1) to (\bar{D}_m, \bar{B}_m) . Then, for each group (\bar{D}_j, \bar{B}_j) , we feed them along with the question Q and the whole document image I into the same LayoutXLM to get the hidden representations H_j of each token in the \bar{D}_j . Initially, the LayoutXLM encodes the concatenation of the question Q and

⁵OCR has been well-studied and been widely applied in many applications, which is not the bottleneck of our method.

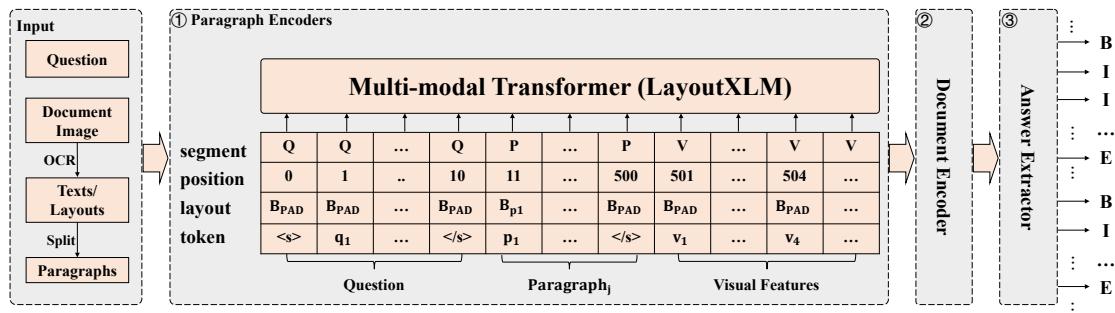


Figure 3: The stages of DocVQA. Taking a question and a document image, we utilize OCR to extract texts and layouts from the document image and split them into m paragraphs with an average length of 512. DocVQA contains three stages: 1) Paragraph encoders using LayoutXML to encode the input texts, layouts, and visual features, where the visual features are extracted by Mask-RCNN. All paragraph encoders share the same parameters; 2) Document encoder to further encode documents by combining all paragraph encodings in document images together; 3) Answer extractor applies a CRF layer to label multi-span answers with the BIOES label format.

the paragraph \bar{D}_j as the **text embedding** E_{T_j} , encodes the whole document image I by a visual encoder (Mask-RCNN (He et al., 2017)) as the **image embedding** E_I , encodes the sequence position of the text inputs as the **position embedding** E_{P_j} , and encodes the bounding boxes \bar{B}_j as the **layout embedding** E_{B_j} . All the embeddings are as the inputs of LayoutXML, as shown in Figure 3.

Document Encoder: We combine the hidden representations H_j of each paragraph \bar{D}_j together to get the document hidden representation H_D and then take one layer of the multi-modal Transformer as the document encoder to further encode the document for labeling answers.

Answer Extractor: Finally, we apply a CRF layer to label all the answer spans with the ‘‘BIOES’’ label format in the answer extractor. Similar to Named Entity Recognition (NER), ‘‘B’’ denotes the first token of the answer span, ‘‘I’’ denotes the subsequent tokens inside the answer span, ‘‘E’’ denotes the end of the answer span, and ‘‘O’’ denotes tokens outside answer spans. Besides, if there is only one token in the answer span, it will be labeled as ‘‘S’’. A Viterbi algorithm (Viterbi, 1967) is adopted to decode the tag sequence with the highest probability. There is an example of the multi-span answer shown in Appendix A.2 in Figure 5.

5 Experiment

In this section, we firstly describe the experiments we conduct on DuReader_{vis} dataset and then conduct further analysis and discussion. Case studies are shown in Appendix A.2.

5.1 Experimental Setup

In this subsection, we describe the experimental baselines and evaluation metrics.

5.1.1 Baselines

For DocVRE, we use BM25 to retrieve relevant document images as the baseline. For DocVQA, we apply two text-based pre-trained models (containing RobertaXML-base (Liu et al., 2019) and BERT-base-Chinese (Devlin et al., 2019)) into our proposed framework as the baseline, where we only use the textual contents as the input and replace the LayoutXML with text-based pre-trained models.

5.1.2 Evaluation Metrics

For DocVRE, we evaluate the retrieval results by Recall@5, Recall@10, and MRR. For DocVQA and Open-domain DocVQA, we concatenate all answer spans together and use Rouge-L and F1 to evaluate the answer extraction performance. The details are shown in Appendix A.1.

5.2 Implementation Details

We utilize Paddle-OCR⁶ to parse document images. The OCR results containing texts and bounding boxes are sorted by line. We evaluate PaddleOCR on our dataset and get F1 above 90, which shows that OCR is not the bottleneck of our task. For DocVQA baselines, we set the max paragraph number to 8, meaning the max document token length is about 4000. We truncate documents with more tokens. The Hi-LayoutXML has about 200M parameters. We train 10 epochs using the AdamW

⁶<https://github.com/PaddlePaddle/PaddleOCR>

Table 3: Experimental results on the DocVQA task in DuReader_{vis} (dev/test). “Text”, “List”, and “Table” are the answer types, and “All” denotes the whole set.

Model	All		Text (40%)		List (25%)		Table (35%)	
	Rouge-L	F1	Rouge-L	F1	Rouge-L	F1	Rouge-L	F1
Human	-/89.74	-/90.20	-/89.64	-/89.68	-/89.54	-/90.12	-/89.99	-/90.87
Hi-BERT	45.96/47.31	47.64/49.02	37.03/36.65	39.21/38.82	55.64/56.46	58.69/59.24	51.11/50.76	51.75/51.44
Hi-RoBERTaXLM	46.29/48.57	48.37/50.53	36.96/37.98	39.66/40.12	58.88/59.47	61.92/62.71	50.51/50.71	51.54/51.72
Hi-LayoutXLM	50.94/53.10	52.44/54.61	42.25/40.69	43.65/42.18	63.09/65.57	65.79/67.84	54.68/55.78	55.78/56.84

Table 4: Experimental results on the DocVRE task in DuReader_{vis} (dev/test).

Model	Recall@5	Recall@10	MRR
BM25	76.33/75.43	81.80/80.82	65.08/63.98

(Loshchilov and Hutter, 2017) optimizer with a 3e-5 learning-rate.

5.3 Main Results

The results of DocVRE, DocVQA and Open-domain DocVQA (DocVRE+DocVQA) are shown in Table 4, Table 3, Table 5 respectively.

DocVRE: From Table 4, we can see that BM25 obtains decent performance for retrieval, and the top 1 document will be used for DocVQA in the Open-domain QA setting.

DocVQA: As shown in Table 3, Hi-LayoutXLM performs best overall baselines, which denotes that the layout and visual features provide benefits for understanding document images. The results also show that there is still a performance gap between baseline models and human performance. Except for the overall performance, we also report the performance of the baselines on each answer type. All the models obtain better results on the list type since list items commonly have indicators like numbers and have similar layouts in the document, assisting models to gain further improvements (also shown in Figure 2). The performance gap between Hi-RoBERTaXLM and Hi-LayoutXLM in the table-type and list-type answers are bigger since lists and tables have rich layout and visual information, which is conducive to the list and table understanding. In comparison, the text-type answers focus more on understanding the text content, where the layout and visual information cannot provide too many benefits, thereby the performance gap is smaller.

Open-domain DocVQA: The experimental results of Open-domain DocVQA are shown in

Table 5: Experimental results on the Open-domain DocVQA task in DuReader_{vis} (dev/test).

	Rouge-L	F1
BM25+Hi-BERT	29.19/33.21	29.44/33.53
BM25+Hi-RoBERTaXLM	29.52/33.01	30.40/33.69
BM25+Hi-LayoutXLM	33.04/33.89	36.61/37.47

Table 5. Since DocVRE can not perform perfectly, the performance of our model on the open-domain DocVQA has decreased compared to that on the DocVQA. Compared to “BM25+Hi-BERT” and “BM25+Hi-RoBERTaXLM”, our method also performs better, with the same observations as DocVQA results.

5.4 Analysis and Discussion

In this subsection, we will perform more analysis to demonstrate the three challenges in DuReader_{vis}, the scalability of our approach, and the performance gap between our approach and open-domain QA. Finally, we will show the error case study and give some promising future directions.

5.4.1 Challenges in DuReader_{vis}

As mentioned above, our dataset has three main challenges: 1) long document understanding, 2) noisy texts, and 3) multi-span answer extraction. In this subsection, we will analyze their influence on DocVQA respectively.

Long Document Understanding: We conduct a statistical analysis of the relationship between the model performance and the document length on the development set of DuReader_{vis} using Hi-LayoutXLM. As shown in Figure 4, documents in DuReader_{vis} mainly have 500 to 3000 tokens. As the length of the document increases, the model performance gradually decreases. The increased document length makes documents harder to understand and makes models harder to extract valuable information to answer questions.

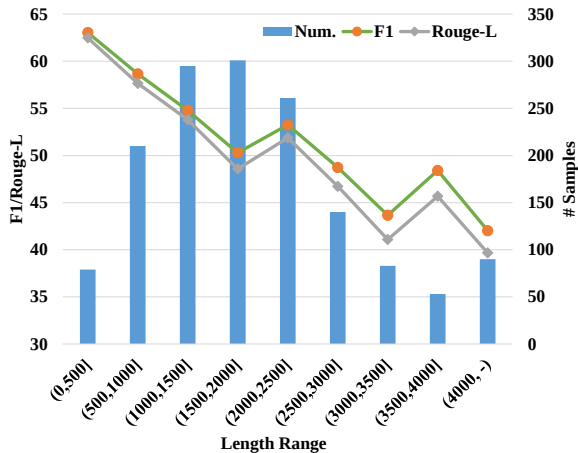


Figure 4: Length analysis on the development set of DuReader_{vis}. #Samples: the number of samples.

Noisy Texts: Documents from web pages commonly contain much noise, such as advertisements, relevant recommendations, etc. It is hard to distinguish between main contents and noise accurately, so we roughly remove noise through a heuristic algorithm. Then, we conduct a comparison experiment between whether denoising or not. After removing noise, there is a performance improvement (F1: 53.10 \rightarrow 57.24 (+4.14), and Rouge-L: 50.94 \rightarrow 54.83 (+3.89)). We attribute it to the fact that noisy texts increase the total amount of information in the document, and it is easier for models to focus more on the valuable information after removing noise.

Multi-span Answer Extraction: In this part, we perform experiments to verify that the task of the multi-span answer extraction is more challenging than that of the single-span answer extraction. We convert the multi-span extraction task to the single-span extraction task by concatenating all answer spans together and inserting the concatenated answers to the position of the first answer span. And then we make a comparison between the above two tasks. Compared with the multi-span answer extraction, the single-span answer extraction performs better (F1: 53.10 \rightarrow 59.38 (+6.28), and Rouge-L: 50.94 \rightarrow 58.08 (+7.14)), which indicates that the task format of the multi-span answer extraction is harder to model. If there is no multi-span answers, our task will be easier.

5.4.2 Zero Shot Study

In this subsection, we randomly select 100 questions (do not occur in DuReader_{vis}) from Baidu and obtain the most relevant documents from a large

Table 6: Comparisons between Open-domain QA and Open-domain DocVQA (dev/test). The metric for retrieval is MRR. The metric for reader is Rouge-L.

	Retrieval	Reader	Retrieval + Reader
Open-domain QA	67.67/68.82	62.45/63.49	40.28/42.40
Open-domain DocVQA	65.08/63.98	50.94/53.10	33.04/36.61

Chinese document collection website. Documents contain PDFs, Word documents, and scanned documents. We use Hi-LayoutXLM trained on DuReader_{vis} to test the performance on the selected 100 questions directly. The model gets decent performance (40.53 F1 and 36.71 Rouge-L) compared to that on the test dataset in DuReader_{vis}. The test procedure on the selected queries proves the scalability of our approach, and the results show the good transferable ability of the model trained on DuReader_{vis}.

5.4.3 Open-domain DocVQA v.s. Open-domain QA

The goal of the open-domain DocVQA is to develop a more scalable QA system that can be applied to diverse domains and document formats. In section 5.4.2, we have shown that our approach can be applied to various formats and achieve decent performance. In the future, we aim to achieve competitive (even better) performance compared to well-designed format-specific or task-specific QA systems.

In this subsection, we design an experiment to see the performance gap between our scalable open-domain DocVQA system (as shown in Figure 1(b)) and the well-designed format-specific open-domain QA system which extracts text contents from web pages with well-designed text extractors (Figure 1(a)). The results on DuReader_{vis} are shown in Table 6. Open-domain QA performs better for two reasons: 1) The textual contents are clean with little web-page noise. 2) The extracted contents only contain clean text, making the whole input shorter. The results show that there still leaves room for open-domain DocVQA to improve. It is of great value for researchers to push open-domain DocVQA to obtain competitive results with task-specific or format-specific methods to reduce task-specific or format-specific efforts.

5.4.4 Error Case Study and Future Directions

We randomly sample and manually analyze 100 error cases with Rouge-L lower than 0.5 from the prediction results. There are 55% wrong samples

in the retrieval stage, caused by the lousy string matching in the BM25 without understanding the semantics. In addition, about 15% of samples have multi-span answers, about 15% of samples make almost complete wrong predictions, and about 5% of samples output no answer. Furthermore, about 10% of samples have noisy texts.

From the results, we can give some promising directions to improve: 1) Utilize multi-modal information to model the long document images and reduce the impact of noises automatically; 2) Utilize dense retrieval methods (Karpukhin et al., 2020b; Qu et al., 2021b; Ren et al., 2021) to improve the document retrieval; 3) Pre-train multi-modal language models for long document images; and 4) Advanced methods to extract multi-span answers.

6 Conclusion

We propose an open-domain document visual question answering task to encourage scalable QA applications. We introduce DuReader_{vis} to move toward the open-domain DocVQA research. There are three challenges: long document understanding, noisy text, and multi-span answer extraction. We propose a baseline and the results show that there is still a huge gap compared to human performance. We show the scalability of our approach by a zero-shot study. Finally, we give error cases and future directions.

Acknowledgement

This work is supported by the Key Development Program of the Ministry of Science and Technology (No.2019YFF0303003), the National Key Research and Development Project of China (No.2018AAA0101900), the National Natural Science Foundation of China (No.61976068) and “Hundreds, Millions” Engineering Science and Technology Major Special Project of Heilongjiang Province (No.2020ZX14A02).

Ethic Consideration

This dataset should be ONLY used for NLP research purposes. All the instances in DuReader_{vis} are collected from public data and have been desensitized. All annotators and reviewers are formal employees and native speakers, thus we make sure that all workers are fairly compensated. Experienced reviewers have reviewed our dataset and the data collection process.

There are no copyright issues in our dataset. Firstly, the document images are crawled from the public data following the crawler protocol. Secondly, we strictly restrict the use of our dataset to academic research. If the websites think it needs to be removed, we will respond to removal at any time.

References

- Vittorio Castelli, Rishav Chakravarti, Saswati Dana, Anthony Ferritto, Radu Florian, Martin Franz, Dinesh Garg, Dinesh Khandelwal, Scott McCarley, Michael McCawley, Mohamed Nasr, Lin Pan, Cezar Pendus, John Pitrelli, Saurabh Pujar, Salim Roukos, Andrzej Sakrajda, Avi Sil, Rosario Uceda-Sosa, Todd Ward, and Rong Zhang. 2020. [The TechQA dataset](#). In [Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics](#), pages 1269–1278. Online. Association for Computational Linguistics.
- Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017a. [Reading wikipedia to answer open-domain questions](#). In [Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers](#), pages 1870–1879. Association for Computational Linguistics.
- Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017b. [Reading wikipedia to answer open-domain questions](#). In [Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics \(Volume 1: Long Papers\)](#), pages 1870–1879.
- Xingyu Chen, Zihan Zhao, Lu Chen, Jiabao Ji, Danyang Zhang, Ao Luo, Yuxuan Xiong, and Kai Yu. 2021. [Websrc: A dataset for web-based structural reading comprehension](#). In [Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021](#), pages 4173–4185. Association for Computational Linguistics.
- Pradeep Dasigi, Kyle Lo, Iz Beltagy, Arman Cohan, Noah A Smith, and Matt Gardner. 2021. [A dataset of information-seeking questions and answers anchored in research papers](#). [arXiv preprint arXiv:2105.03011](#).
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [Bert: Pre-training of deep bidirectional transformers for language understanding](#). In [NAACL-HLT \(1\)](#).
- Matthew Dunn, Levent Sagun, Mike Higgins, V Ugur Guney, Volkan Cirik, and Kyunghyun Cho. 2017. [Searchqa: A new q&a dataset augmented with](#)

- context from a search engine. [arXiv preprint arXiv:1704.05179](#).
- Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. 2017. Mask r-cnn. In [Proceedings of the IEEE international conference on computer vision](#), pages 2961–2969.
- Wei He, Kai Liu, Jing Liu, Yajuan Lyu, Shiqi Zhao, Xinyan Xiao, Yuan Liu, Yizhong Wang, Hua Wu, Qiaoqiao She, et al. 2018. Dureader: a chinese machine reading comprehension dataset from real-world applications. In [Proceedings of the Workshop on Machine Reading for Question Answering](#), pages 37–46.
- Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020a. Dense passage retrieval for open-domain question answering. [arXiv preprint arXiv:2004.04906](#).
- Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick S. H. Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020b. [Dense passage retrieval for open-domain question answering](#). In [Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020](#), pages 6769–6781. Association for Computational Linguistics.
- Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. 2019. Natural questions: A benchmark for question answering research. [Transactions of the Association for Computational Linguistics](#), 7:452–466.
- Sharon Levy, Kevin Mo, Wenhan Xiong, and William Yang Wang. 2021. Open-domain question-answering for covid-19 and other emergent domains. [arXiv preprint arXiv:2110.06962](#).
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. [arXiv preprint arXiv:1907.11692](#).
- Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. [arXiv preprint arXiv:1711.05101](#).
- Yuning Mao, Pengcheng He, Xiaodong Liu, Yelong Shen, Jianfeng Gao, Jiawei Han, and Weizhu Chen. 2020. Generation-augmented retrieval for open-domain question answering. [arXiv preprint arXiv:2009.08553](#).
- Minesh Mathew, Viraj Bagal, Rubèn Pérez Tito, Dimosthenis Karatzas, Ernest Valveny, and CV Jawahar. 2021a. Infographicvqa. [arXiv preprint arXiv:2104.12756](#).
- Minesh Mathew, Dimosthenis Karatzas, and CV Jawahar. 2021b. Docvqa: A dataset for vqa on document images. In [2021 IEEE Winter Conference on Applications of Computer Vision \(WACV\)](#), pages 2199–2208. IEEE.
- Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. Ms marco: A human generated machine reading comprehension dataset. In [CoCo@ NIPS](#).
- Yingqi Qu, Yuchen Ding, Jing Liu, Kai Liu, Ruiyang Ren, Wayne Xin Zhao, Daxiang Dong, Hua Wu, and Haifeng Wang. 2021a. Rocketqa: An optimized training approach to dense passage retrieval for open-domain question answering. In [Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies](#), pages 5835–5847.
- Yingqi Qu, Yuchen Ding, Jing Liu, Kai Liu, Ruiyang Ren, Wayne Xin Zhao, Daxiang Dong, Hua Wu, and Haifeng Wang. 2021b. [Rocketqa: An optimized training approach to dense passage retrieval for open-domain question answering](#). In [Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021](#), pages 5835–5847. Association for Computational Linguistics.
- Ruiyang Ren, Shangwen Lv, Yingqi Qu, Jing Liu, Wayne Xin Zhao, Qiaoqiao She, Hua Wu, Haifeng Wang, and Ji-Rong Wen. 2021. [PAIR: leveraging passage-centric similarity relation for improving dense passage retrieval](#). In [Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021, volume ACL/IJCNLP 2021 of Findings of ACL](#), pages 2173–2183. Association for Computational Linguistics.
- Stephen E. Robertson and Hugo Zaragoza. 2009. [The probabilistic relevance framework: BM25 and beyond](#). [Found. Trends Inf. Retr.](#), 3(4):333–389.
- Ryota Tanaka, Kyosuke Nishida, and Sen Yoshida. 2021. Visualmrc: Machine reading comprehension on document images. In [Proceedings of the AAAI Conference on Artificial Intelligence](#), volume 35, pages 13878–13888.
- Andrew Viterbi. 1967. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. [IEEE transactions on Information Theory](#), 13(2):260–269.
- Yiheng Xu, Tengchao Lv, Lei Cui, Guoxin Wang, Yijuan Lu, Dinei Florencio, Cha Zhang, and Furu Wei. 2021. Layoutxlm: Multimodal pre-training for multilingual visually-rich document understanding. [arXiv preprint arXiv:2104.08836](#).

A Appendix

A.1 Evaluation Metrics

We describe the metrics used in our experiments in details.

Recall@K: Recall@K is calculated as the proportion of questions where the top-k retrieved document images contain the answers.

MRR: Mean Reciprocal Rank (MRR) is the average of all question's reciprocal of the rank at the first retrieved relevant document image.

Rouge-L: Rouge-L uses LCS-based (Longest Common Subsequence-based) F-measure to estimate the similarity between the reference X of length m and the predication Y of length n .

F1: F1 measures the average overlap between the prediction and the reference, where we treat them as bags of tokens and compute their F1. We report the average over the F1 values of all questions.

A.2 Case Study

Here we give some results and show the performance of our proposed baseline, as shown in Figure 5, 6, 7.

In Figure 5, the answers are multi table cells which are not continuous. We can see that our model can predict the right answer by utilizing the layout information, while Hi-RoBERTaXLM cannot predict any answers since the information in the pages are difficult to model for Hi-RoBERTaXLM.

In Figure 6, the answer has highlight layout and visual information (the font is big and the color is red). It is easy to be captured by our model, while it is hard for Hi-RoBERTaXLM to predict the right answer.

In Figure 7, the answer is a single-span text, and the question is highlighted in the document image. Hi-RoBERTaXLM also predicts an answer, but the answer is not complete compared with the ground truth. The answer predicted by Hi-LayoutXLM is better. We can see that the layout can help locate the right answers.

From the above three cases, we can see that our approach can utilize layout and visual information to model multi-span answers much better than the baseline Hi-RoBERTaXLM.

登录 | 注册 网页无障碍 途牛首页 途牛商旅 旅游百货 企业旅游 会员俱乐部 我的订单 网站地图

途牛 tuniu.com 所有产品 毛里求斯 三亚 成都 高级搜索

自游 北京环球影城 海上假期 旅拍

首页 跟团游 自由行 酒+景 机票 酒店 **火车票** 用车 门票 特卖 主题游 邮轮游 出游服务 定制游 金融 攻略

您的位置: 火车票 > 火车票导航 > 火车票 > 时刻表

车型	出发站	终点站	运行时长	出发时间	到达时间
高铁					查看
站次	站名	到达时间	开车时间	停留时间	
1	襄阳东	16:32	16:32	10	
2	随州南	17:02	17:04	2	
3	孝感东	17:31	17:33	2	
4	汉口	17:59	17:59	0	

出发的车次

到达的车次

去旅游

- 跟团游
- 自由行
- 酒+景
- 公司旅游
- 当地活动
- 首团出发

寻优惠

- 特卖
- 订酒店 返现金
- 积分商城
- 银行特惠游

看攻略

- 攻略
- 途牛风向标
- 游记
- 达人玩法

查服务

- 帮助中心
- 会员俱乐部
- 阳光保障
- 火车时刻表
- 航班查询

途牛APP

扫描下载途牛APP

阳光行程 透明公开 **阳光价格** 明码实价 **阳光服务** 专属客服 客户服务热线 (免长途费) **4007-999-999**

品牌合作

旅游行业 强势品牌

途牛特卖

清仓秒杀 超值尾货

积分商城

小积分 大用途

旅游百货

高品质出行伴侣

途牛客服中心设立在江苏南京及江苏宿迁, 来电显示号码请查看: 途牛会员中心外呼电话号码汇总

北京途牛国际旅行社有限公司, 旅行社业务经营许可证编号: L-BJ-CJ00144 上海途牛国际旅行社有限公司, 旅行社业务经营许可证编号: L-SH-CJ00107

关于我们 Investor Relations 联系我们 投诉建议 广告服务 旅游券 途牛招聘 隐私保护 免责声明 旅游度假资质 主题旅游 平台服务协议 平台交易规则 网站地图 攻略地图 UEIP 帮助中心 网信办辟谣专栏

Copyright © 2006-2021 南京途牛科技有限公司 Tuniu.com | 营业执照 | ICP证: 苏B2-20130006 | 苏ICP备12009060号-4 | 苏网食备A32000000032 | 上海旅游网 全国旅游投诉热线12301

Question: g6874经过站点 (Which stations does g6874 pass through?)

Ground Truth: 襄阳东;随州南;孝感东;汉口 (Xiangyang East; Suizhou South; Xiaogan East; Hankou)

Hi-LayoutXLM: 襄阳东;随州南;孝感东;汉口 (Xiangyang East; Suizhou South; Xiaogan East; Hankou)

Hi-RobertaXLM: None.


Figure 5: A table-type answer example with multiple table cells as the answer. The red bounding box indicates the answer.

返回牛摩网 | 牛摩网VIP商家抢注>> 移动应用 欢迎来到牛摩网商城! [登录] [免费注册]

牛摩商城 mall.newmotor.com.cn

[首页](#)
[摩托车](#)
[电动车](#)
[骑士装备](#)
[装饰养护](#)
[零部件](#)
[实体服务](#)
[汽油机相关](#)
[众筹](#)
[惠买车](#)
[补拍余额](#)
[网购交流](#)

摩托车商城 > 摩托整车 > 商品浏览



豪爵铃木125.150-30F E
豪爵

价格: **¥9,280.00**

定金: ¥1,000.00

在线支付定金或全款由牛摩网担保







上架时间: 2020年10月08日

已售: 0 辆

快递: **不包邮**

数量: 辆(库存100辆)

商家热卖

- 1  豪爵铃木125.150-30F E **¥9280元**
- 2  豪爵铃木DR160V150 **¥12980元**
- 3  豪爵铃木VE125-26A **¥8680元**
- 4  豪爵VH125T-20A **¥8180元**
- 5  豪爵VF100T-8C **¥4980元**
- 6  豪爵EH150-25VA **¥6980元**

商品介绍 | 包装清单 | 商品评价 | 售后保障

豪爵! 摩托品质代名词

[了解更多本车信息>>](#)

购物指南

购物流程
常见问题
联系客服

配送方式

物流配送
快速配送

支付方式

货到付款
在线支付
汇款转账

售后服务

售后政策
退款说明
退换货
取消订单

特色服务

牛摩网担保
牛摩网服务商

牛摩网简介 | 业务介绍 | 联系我们 | 手机访问 | 关于牛摩联盟
Copyright © 2012.All Rights Reserved 版权所有 深圳市牛摩网科技商务有限公司
备案/许可证编号: 粤ICP备11020761号

Question: 豪爵150—30f价格 (What is the price of the Haojue 150-30f?)

Ground Truth: ¥9,280.00

Hi-LayoutXLM: ¥9,280.00

Hi-RoBERTaXLM: None.

Figure 6: A table-type answer example with a single table cell as the answer. The red bounding box indicates the answer.

西西软件园多重安全检测下载网站、值得信赖的软件下载站! 软件教程 | 最新更新 |

西西软件园 [电脑软件](#) [单机游戏](#) [安卓应用](#) [安卓游戏](#) [苹果电脑](#) [苹果手机](#) 软件 ▾

西西新闻 首页 → 西西教程 → 精选问答 → 微信支付怎么退款申请退款

微信支付怎么退款申请退款

来源: 西西整理 时间: 2020/10/17 10:57:57 字体大小: A- A+

作者: 西西 点击: 73 评论: 0次 标签: [微信](#) [支付](#)

微信长头发小表情生成app 1.0 立即下载

类型: 图形图像 大小: 3M 语言: 中文 评分: 10.0

标签:

微信支付进行支付后,不少小伙伴们想知道对商品不满意想要退款,不知道退款方法的,就让小编给大家详细的讲讲,一起来看看吧。

相关链接	版本说明	下载地址
微信Android版	安卓手机版	查看
微信电脑版	官方pc版不需要模拟器	查看
微信ipad版	平板高清版	查看
微信肾6版	for ios8\app watch	查看
微信网页版	浏览器版	查看
微信Mac版	苹果电脑版	查看

微信支付怎么退款申请退款

微信支付对商品或者支付金额有异议,需要退款,只能联系商家。商家同意后可以退款,退款金额直接返回到用户账户中。

怎么联系商家:

手机: iPhone11

系统: ios14

软件版本: [微信7.0.19](#)

打开微信,点击微信支付

推荐文章

相关下载

- 1 微信长头发小表情
- 2 微信DIY自拍表情
- 3 微信表情加头发辨
- 4 微信小辫子符号(可
- 5 微信QQ浏览器打开
- 6 微信隐私锁
- 7 微信读书eInk版
- 8 微信微商清理工具
- 9 微信PC内测版3.0
- 10 微信儿童版2020

最新文章

微
微信表情包行#88 2k

微
微信头像看不到好友动态,但是头像的头像看到 2k

冰
#超酷一个手机壳用1个小时 2k

支
微信支付 2k

Question: 微信支付在哪里申请退款 (Where can WeChat payment apply for refund?)

Ground Truth: 微信支付对商品或者支付金额有异议,需要退款,只能联系商家。商家同意后可以退款,退款金额直接返回到用户账户中。(When using WeChat payment, if there is any objection to the goods or payments, requiring to refund, you can only contact the merchant. Refunds can only be made after the merchant agrees, and the refund amount is directly returned to the user account.)

HiLayoutXLM: 微信支付对商品或者支付金额有异议,需要退款,只能联系商家。商家同意后可以退款,退款金额直接返回到用户账户中。(When using WeChat payment, if there is any objection to the goods or payments, requiring to refund, you can only contact the merchant. Refunds can only be made after the merchant agrees, and the refund amount is directly returned to the user account.)

HiRobertaXLM: 商家同意后可以退款,退款金额直接返回到用户账户中打开微信,点击微信支付 (Refunds can only be made after the merchant agrees, and the refund amount is directly returned to the user account. Open WeChat, and click WeChat Pay.)

Figure 7: A text-type answer example. The red bounding box indicates the answer.