

# “What is the value of {templates}?” Rethinking Document Information Extraction Datasets for LLMs

Ran Zmigrod\*, Pranav Shetty\*, Mathieu Sibue\*,  
Zhiqiang Ma, Armineh Nourbakhsh, Xiaomo Liu, Manuela Veloso  
JPMorgan AI Research  
first.last@jpmchase.com

## Abstract

The rise of large language models (LLMs) for visually rich document understanding (VRDU) has kindled a need for prompt-response, document-based datasets. As annotating new datasets from scratch is labor-intensive, the existing literature has generated prompt-response datasets from available resources using simple templates. For the case of key information extraction (KIE), one of the most common VRDU tasks, past work has typically employed the template “What is the value for the {key}?”. However, given the variety of questions encountered in the wild, simple and uniform templates are insufficient for creating robust models in research and industrial contexts. In this work, we present K2Q,<sup>1</sup> a diverse collection of five datasets converted from KIE to a prompt-response format using a plethora of bespoke templates. The questions in K2Q can span multiple entities and be extractive or boolean. We empirically compare the performance of seven baseline generative models on K2Q with zero-shot prompting. We further compare three of these models when training on K2Q versus training on simpler templates to motivate the need of our work. We find that creating diverse and intricate KIE questions enhances the performance and robustness of VRDU models. We hope this work encourages future studies on data quality for generative model training.

## 1 Introduction

Visually rich document understanding (VRDU) is a core research field at the intersection of natural language processing (NLP) and computer vision. The field aims to develop methods to process information from documents and perform natural language inference on them. VRDU is crucial for automation in industries such as finance, legal, healthcare, and

\*Equal Contribution.

<sup>1</sup>Full dataset available upon request at [airdata.requests@jpmorgan.com](mailto:airdata.requests@jpmorgan.com)

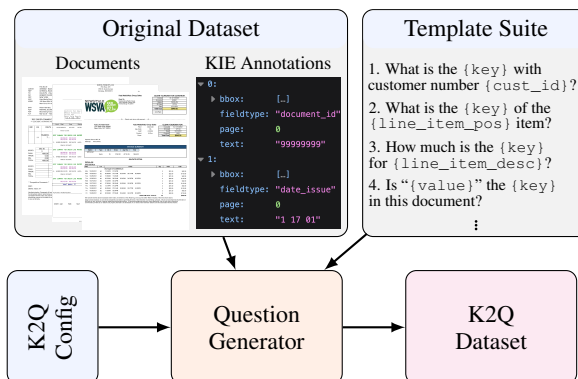


Figure 1: Generation pipeline of K2Q datasets. A suite of diverse templates is designed for each specific KIE dataset. These templates are populated in accordance to a configuration file that configures the dataset size and proportion of extractive and boolean questions.

government services. Naturally, diverse and high-quality datasets are essential for training and evaluating VRDU models. This is ever more important as generative models, such as large language models (LLMs), are becoming the mainstream state-of-the-art method for tackling VRDU problems (Bai et al., 2023; Wang et al., 2023a; Hu et al., 2024). These models require significant amounts of data for training as well as a diverse test set to evaluate whether robust document understanding is achieved.

VRDU covers an array of tasks that can be used to train generative models. While visual question answering (VQA) datasets (Mathew et al., 2021; Landeghem et al., 2023) can be directly used for instruction tuning, only a handful of Document VQA datasets are publicly available (Mathew et al., 2021; Landeghem et al., 2023). Thus, existing datasets for other tasks are generally transformed into a prompt-response style to increase training volume and document diversity.

Current works that leverage existing non-VQA VRDU datasets for generative models often populate uniform templates (Wang et al., 2023a; Ye et al., 2023b; Tanaka et al., 2024), resulting in

datasets that lack diversity and complexity. Specifically, for key information extraction (KIE), which aims to find important entities within a document, datasets tend to populate the template “What is the value for the  $\{key\}$ ?” (Ye et al., 2023a,b; Hu et al., 2024).<sup>2</sup> While such questions directly translate the KIE problem, they fail to capture the complexities of VRDU and the intricacies of real-world applications of generative models for documents. Such datasets are, therefore, inadequate to train and evaluate document understanding models robustly.

In this paper, we propose a new collection of transformed KIE datasets, **K2Q**, that aims to find a balance between templated and human-annotated VQA datasets. K2Q is derived from five datasets: CORD (Park et al., 2019), Docile (Simsa et al., 2023), Kleister Charity (Stanislawek et al., 2021), and VRDU Ad-Buy (Ad-Buy) and Registration Form (Reg. Form) (Wang et al., 2023c). We curate over 100 different templates that lead to a diverse set of over 300,000 questions across over 12,000 documents. The transformed datasets contain extractive questions as well as boolean (i.e., true or false) questions, where questions may consider multiple entities within a document. Our key contributions are given below:

- We introduce a new publicly available collection of datasets, K2Q, that converts five existing KIE datasets into rich and diverse prompt-response datasets. The creation pipeline for K2Q is illustrated in Figure 1.
- We show that K2Q exhibits closer characteristics to human-made VQA datasets than simple templates through substantially lower self-BLEU and perplexity scores.<sup>3</sup>
- We provide zero-shot and fine-tuned benchmarks for K2Q across seven models. A detailed performance breakdown is available in App. D.2.
- We conduct an in-depth analysis of the impact of diverse, dataset-specific templates on VRDU model performance and groundedness against simple templates. Training on diverse templates improves relative ANLS performance by 40% over simple templates.

<sup>2</sup>For example, finding an address on a form is converted into the question “What is the value for the address?”

<sup>3</sup>Perplexity of observing human-made questions from DocVQA and DUDE.

Dataset	Docs	Pages	Labels	Lines
Ad-Buy	641	1,598	14	✓
CORD	1,000	1,000	33	✓
Docile	5,680	7,372	55	✓
KLC	2,778	62,010	8	✗
Reg. Form	1,915	3,890	6	✗
BuDDIE	1,665	1,665	69	✗
DeepForm	1,100	4,720	5	✗
FUNSD	199	199	4	✗
KNDA	540	3,229	4	✗
SROIE	600	1,000	4	✗
WildReceipt	1,786	1,786	25	✗

Table 1: KIE Datasets. The first section contains the datasets we transform in K2Q. The second section contains other relevant KIE datasets. The *Lines* column indicates whether the dataset contains line items.

## 2 Key Information Extraction in VRDU

We focus on KIE in this work due to the abundance of existing datasets in the VRDU literature and the importance of information extraction in industry. We hope future work will examine other popular structured VRDU tasks such as document classification (Harley et al., 2015) or document structure prediction (Li et al., 2020; Xu et al., 2022; Zmigrod et al., 2024a). KIE is the task of identifying important entities within a document. As such, many datasets have naturally examined invoices and receipts (Huang et al., 2019; Park et al., 2019; Sun et al., 2021; Simsa et al., 2023), where entities are clearly defined and often connect through **line items** (e.g., the name and price of an item appear on the same “line”). Similarly, forms and legal documents have also been the focus of a plethora of datasets (Jaume et al., 2019; Borchmann et al., 2021; Stanislawek et al., 2021; Wang et al., 2023c; Zmigrod et al., 2024b). We provide details regarding available KIE datasets in Table 1.<sup>4</sup>

### 2.1 KIE for Generative Models

To train generative models for KIE, datasets must be formulated into a prompt-response type structure. So far, the VRDU and NLP literature has proposed three main strategies to obtain such formulations: populating simple templates based on existing annotations, manually curating questions, or generating questions using LLMs. We discuss the advantages and disadvantages of each method below to motivate our approach in Section 3.

<sup>4</sup>We only focus on English datasets in this work. Nevertheless, other non-English VRDU datasets exist (Wang et al., 2021; Qi et al., 2022; Ding et al., 2023).

Ad-Buy	Q: When does the advertisement start running on channel KPTH on program "People's Court"? A: 01/30/20 Q: Did the advertisement cost Teresa Tomlinson for Senate \$1,855.00? A: Yes
CORD	Q: How much did the order(s) of Combo 1 cost in total? A: 30.000 Q: Is "20.000" the number of servings/quantity of a menu item in this receipt? A: No
Docile	Q: How much is the total amount with tax of the 19th item? A: 130.00 Q: Is "Alyse For Alaska" the name of the customer that is being invoiced in this document? A: No
KLC	Q: How much did the charity with number 1154288 earn annually in British pounds? A: 36493079.00 Q: Is 287408.00 pounds the annual expenditure of Campaign For Learning? A: Yes
Reg. Form	Q: Who is the agent of the form? A: Hogan & Hartson LLP Q: Is manager the title of James A. Coppola? A: No

Figure 2: Examples of populated questions and answers from K2Q.

**Populating Simple Templates.** The most common method for designing large datasets of instructions for LLMs is templating. This is the case in both general NLP (Chung et al., 2022; Wei et al., 2022) as well as in VRDU (Tang et al., 2023; Ye et al., 2023b; Zhang et al., 2023; Wang et al., 2023c; Tanaka et al., 2024; Zmigrod et al., 2024b). Unfortunately, most templating approaches in VRDU rely on simplistic and dataset-agnostic templates that do not reflect a model’s true understanding of the task nor the real-life queries that would be submitted to a generative model. For instance, Ye et al. (2023a,b); Wang et al. (2023a); Hu et al. (2024) invoke a single simple template: “What is the value for the {key}?”. This can lead to under-specified questions, which our work aims to avoid. BuDDIE (Zmigrod et al., 2024b) is similar to these works but also includes boolean questions. In comparison, our work populates much richer templates thanks to the inclusion of multiple rephrasings per question, multiple entities per question, and multiple question types. Another approach to templating is that of the InstructDoc (ID) collection (Tanaka et al., 2024), which features five rephrased templates per dataset that ask the model to classify text snippets into one of the key entity types. Unfortunately, we believe this to be unrepresentative of the KIE task in practice, as the underlying goal of KIE is to *directly extract* entities from a large amount of text instead of simply classifying candidate extractions. Therefore, we do not include such templates here.

**Manually Constructing Questions.** Rather than relying on existing datasets from other tasks, researchers have also constructed prompt-response datasets from the ground up, such as DocVQA (Mathew et al., 2021; Tito et al., 2022) and DUDE

(Landeghem et al., 2023). Due to the lack of other VQA datasets truly focused on visually rich documents, the literature has also considered datasets from adjacent areas such as InfographicsVQA (Mathew et al., 2022) or ChartQA (Masry et al., 2022). The benefits of human-annotated datasets come from the quality, diversity, and depth of the questions that can be achieved. Unfortunately, curating VQA datasets is expensive, and thus, creating datasets of the scale needed to instruction-tune LLMs can be inaccessible to many practitioners. Our work addresses this by applying human intervention at the dataset level rather than at the document level. We use annotators to manually design templates, resulting in a collection of datasets over 6 and 7 times larger than DocVQA and DUDE respectively.<sup>5</sup>

**Generating Questions using LLMs.** Past work has demonstrated that utilizing LLMs to create fine-tuning datasets may be a promising research avenue (Honovich et al., 2023; Peng et al., 2023; Cheung et al., 2024; Taori et al., 2023; Wang et al., 2023b). This also includes work for general multi-modal LLMs (Liu et al., 2023; Li et al., 2023), suggesting this methodology can extend to VRDU. However, to the best of our knowledge, the VRDU literature has not investigated the use of LLMs to generate prompt-response pairs yet. While we do not leverage LLMs to generate data in this work, we believe a future iteration of K2Q could use an LLM to augment our set of templates.

<sup>5</sup>This size factor is after we sample our selection of templates. One can increase the sampling ratio to make a significantly larger dataset.

Dataset Name	Num. Temp.	Num. Ques.	% Extr. Ques.	% 1-Page Ques.	Ques. Length	Answer Length	Ent. per Ques.	Ques. per Doc.
Ad-Buy	50	15,119	57.8	96.8	9.5	2.3	1.5	23.6
CORD	22	39,575	49.6	100.0	12.6	1.4	1.4	39.6
Docile	17	185,557	68.1	99.9	11.6	3.6	1.1	32.7
KLC	31	44,813	43.2	88.3	9.8	1.6	1.8	16.1
Reg. Form	18	23,427	37.7	99.9	8.8	3.1	1.1	12.2

Table 2: K2Q statistics per dataset. Question length and answer length show the average number of tokens of questions and answers respectively. “Extr.” stands for extractive and “Ent.” for entities.

## 2.2 Modeling Spectrum for KIE

Models for VRDU (and therefore KIE) lie on a spectrum of those that only utilize text extracted from documents through optical character recognition (OCR) (Devlin et al., 2019; Liu et al., 2019) to those that ingest documents directly as image inputs (Lee et al., 2023). On this spectrum, some models are more focused on OCR and spatial features (Xu et al., 2020; Peng et al., 2022), while others are more focused on complementing a main visual channel with text (Kim et al., 2022; Davis et al., 2022; Tang et al., 2023). Multi-modal LLMs have also been developed along this spectrum (Wang et al., 2023a; Ye et al., 2023b,a; Hu et al., 2024; Tanaka et al., 2024).

## 3 The K2Q Dataset

We present **Key Information Extraction Transformed to Visual Question Answering Datasets**, or **K2Q** for short. This is a collection of five KIE datasets transformed into prompt-response type datasets (similar to VQA): Ad-Buy, CORD, Docile, Kleister Charity, and Reg. Form. These datasets were chosen as they cover a wide range of domains within the field of VRDU. K2Q contains more than 300,000 questions over 12,000 documents. Figure 2 provides several examples from K2Q along with their respective source dataset. The questions are designed to reflect the intricacies of document understanding, such as co-references (e.g., “How much did *the charity with number 1154288* earn annually in British pounds?”), disambiguation of similar entities (e.g., “How much is the total amount with tax of *the 19th item*?”), and questions spanning multiple pages. Table 2 shows a statistical breakdown of the questions generated for K2Q. Data split sizes are also provided in App. A.2.

### 3.1 Dataset Construction

Unlike other template-based approaches, we manually construct several templates *at the dataset level* for each entity in the KIE topology. Templates were

created by three VRDU researchers who were already familiar with the datasets. The datasets were split evenly among them for template creation and validation. When possible, questions are created in relation to other entities as shown in Figure 2 (e.g., “Is manager the title of James A. Coppola?” involves two distinct key entities from Reg. Form). In particular, CORD, Docile, and Ad-Buy contain line items that connect several different entities in an entry (similar to the rows of a table). Examples of such items are lines in a receipt and advertisement slots. For single-page questions in K2Q, we ensure that all entities in the question and answer are present in the input page fed to models. Furthermore, K2Q contains **extractive** and **boolean** questions; extractive questions have answers that are key entity values, while boolean questions have answers that are “Yes” or “No”. While K2Q is intended for training generative models, we ensure it can be used as a VQA training dataset for traditional VRDU models such as Xu et al. (2020) by providing OCR token span annotations. This required heuristics for Docile and KLC, which do not relate KIE entities to the OCR. To ensure the quality of the questions generated, we address several complexities.

**Cleaning OCR Output.** In the traditional KIE task, predictions are typically given as OCR token spans, which can be noisy. Thus, cleaning the OCR entity values is necessary to enable generative models trained on K2Q to produce more natural responses. We provide more details regarding our data cleaning in App. A.1. Note that the OCR output provided in the original datasets is used to train and test the OCR-based models in Section 4.

**Under-specified Questions.** It is sometimes possible for a question to have several correct answers within a document. K2Q handles such cases in two different ways. Firstly, if a document contains multiple entities of the same type that are not line items (e.g., several vendor names mentioned), we consider all entities to be allowed as an answer. This

Property	DUDE	ID	UReader	K2Q
Templated	✗	✓	✓	✓
Diverse	✓	✓	✗	✓
Extractive	✓	✗	✓	✓
Boolean	✓	✗	✗	✓
Unambiguous	✓	✓	✗	✓

Table 3: Properties of VRDU datasets for generative models. DUDE is provided here as a human-generated reference.

under-specificity only occurs for question-answer pairs from Docile. Secondly, if multiple line item rows contain repeated information, we avoid asking questions that may cause ambiguity when determining the row to consider. For example, if a receipt contains two items that both cost \$10, we do not include the question “Which item in the receipt cost \$10?” as it is unclear which line item is being referred to. While it may be desirable for models to learn to return all relevant line items in this case (multi-span answer), it complicates training and evaluation, so we omit these questions. Future work will look into incorporating such questions.

**Negative Boolean Question Generation.** Half of all boolean questions in K2Q are designed to be false. To achieve that, we devise a candidate set of false answers by drawing inspiration from the approach introduced in Zmigrod et al. (2024b). Firstly, we consider all unique values of a given entity type in the dataset. Secondly, for datasets with a hierarchical entity ontology, we consider unique values in the same document that share a parent entity. Lastly, we also consider other document values with the same format (e.g., string, integer, date, etc.). The false candidate is then sampled from one of these three sets.

**Question Sampling.** Due to the nature of templates, we do not generate all possible questions for each document to avoid introducing too much redundancy. For Ad-Buy, Kleister Charity, and Reg. Form, this was achieved by randomly sampling one extractive question per entity and then a fixed number of boolean questions. Due to the high number of entities for CORD and Docile, we generate all questions and sample after generation to select a fixed number of questions in both the extractive and boolean settings.

**Dataset Validation.** K2Q is generated systematically after manually curating templates; nevertheless, we applied various forms of human validation to ensure a *high-quality* dataset. Prior to question

Error Type	Error rate	Fleiss $\kappa$
Template	1.15%	0.53
Cleaning	1.38%	0.31
Annotation	2.76%	0.51
Other	0.23%	0.51
<b>Total</b>	5.52%	0.49

Table 4: K2Q validation results. We break down errors into four categories: (1) Template errors indicate a question design issue, (2) Cleaning errors indicate a data cleaning issue, (3) Annotation errors indicate issues with the original datasets, and (4) Other errors (e.g., OCR issues).

generation, at least one other template writer reviewed every template to verify grammatical correctness and variety. Post question generation, five documents from each dataset were randomly sampled. For each document, up to ten extractive and ten boolean questions are sampled, and three validators check questions for grammatical correctness and other data issues. An issue was considered if two out of three validators noted it. The validators were provided with guidelines and performed the validation exercises independently. The error types and the guidelines provided to validators are discussed in more detail in App. A.3. The overall Fleiss-kappa scores indicate moderate agreement between raters (Table 4). The disagreements demonstrate the complexity of assessing erroneous questions. For instance, cleaning errors may be subjective, as validators may disagree on whether or not specific words should be capitalized. Nevertheless, the low percentage of cleaning errors observed indicates this is not an issue. We also note that half of the errors are due to annotation issues with the original datasets, and so inevitably occur in K2Q.

### 3.2 Comparison of the Core Characteristics of K2Q and Related Datasets

Admittedly, K2Q requires more care and effort to collate than past template-based datasets. We empirically motivate why this additional work is worthwhile by comparing intrinsic and extrinsic characteristics of K2Q against those of simpler template approaches. This section analyzes the advantages of K2Q in terms of data volume, diversity, and resemblance to human data. Section 4 delves into the benefits of K2Q for model training and evaluation.

To facilitate the comparison between K2Q and previous work (Hu et al., 2024; Ye et al., 2023b; Wang et al., 2023a), we construct baseline datasets

Source Dataset	New Dataset	Num. Temp.	Num. Ques.	Ques. per Doc.
Ad-Buy	K2Q	50	15,119	23.6
	SD	1	4,986	7.8
CORD	K2Q	22	39,575	39.6
	SD	1	4,143	4.1
	ID	5	1,336	1.3
Docile	K2Q	17	185,557	32.7
	SD	1	53,547	9.4
	ID	5	56,369	9.9
KLC	K2Q	31	44,813	16.1
	SD	1	19,348	7.0
	ID	5	13,449	4.8
	URReader	1	27,664	8.0
Reg. Form	K2Q	18	23,427	12.2
	SD	1	8,826	4.6

Table 5: Comparison of variety of questions in K2Q compared to SD, ID, and URReader. Examples from each dataset are given in App. B.

Dataset	Perplexity ( $\downarrow$ )		Self-BLEU ( $\downarrow$ )	
	DUDE	DocVQA	2-gram	4-gram
DUDE	28.5	35.9	0.73	0.40
DocVQA	45.1	27.4	0.83	0.58
K2Q	229.6	228.6	0.92	0.83
SD	1592.8	928.5	1.00	1.00

Table 6: Perplexity and Self-BLEU scores. For the perplexity experiments, GPT2 is fine-tuned for one epoch on K2Q and on SD, two on DUDE, and three on DocVQA to match the number of training steps.

where we use the template “What is the value for the  $\{key\}$ ?” for all entities that are not line items.<sup>6</sup> We refer to this **simple** collection of **datasets** as **SD**. SD is representative of datasets used in past work, which are not used directly as prior work does not use all datasets comprising K2Q.

**High-level Comparison.** Table 3 summarizes different properties of related datasets. A comparison between the volume of questions generated with other template approaches is provided in Table 5 for the five source datasets featured in K2Q. We give example questions from ID and URReader in App. B. In Table 5, we see that K2Q has 3-6 times more templates than ID and URReader. The total number of questions and number of questions per document are also consequently higher.

**Diversity and Realism Comparison.** To quantitatively assess how closely K2Q resembles human-

<sup>6</sup>We do not create line item questions with this template as they would be ambiguous for models. For example, if a receipt has multiple items, the question “What is the value for the receipt item?” is too ambiguous to answer. This can be fixed by replacing “the” with “any”.

annotated datasets compared to SD, we conduct two studies. First, we measure the perplexity of the questions of the DocVQA and DUDE test sets with a small language model (GPT2) fine-tuned on the questions of K2Q and SD separately. Perplexity indicates how likely a language model is to generate new input, so a low perplexity in this experiment suggests that the unseen human-generated questions from DocVQA and DUDE align with the distribution of the fine-tuned GPT2. Secondly, following Ye et al. (2022), we compute the self-BLEU score of the questions of the K2Q and SD test sets to compare their diversity to that of DUDE and DocVQA questions. See App. C.2 for more details on the perplexity and self-BLEU experiments.

Table 6 gives the perplexity scores of GPT2 fine-tuned and tested on each combination of datasets and the self-BLEU scores. We observe that, even though DUDE and DocVQA generalize well to each other, K2Q and SD have a much higher perplexity. Similarly, the self-BLEU scores of K2Q and SD are higher than those of DUDE and DocVQA – highlighting the challenges of mimicking human-crafted questions using templates in general. However, fine-tuning GPT2 on K2Q questions did reduce the perplexity from fine-tuning on SD by a factor of 4 and 7 on DocVQA and DUDE respectively. Additionally, the self-BLEU score of K2Q is closer to that of DocVQA and DUDE than SD. These results demonstrate the benefits of K2Q over simpler templating methods in terms of similarity to human-curated data.

## 4 Modeling Experiments

### 4.1 Benchmark Models

To analyze the impact of training and testing on K2Q, we consider three non-LLM generative models in our experiments: Donut (200M parameters) (Kim et al., 2022) Pix2Struct base (282M parameters) (Lee et al., 2023), and UDOP (800M parameters) (Tang et al., 2023), which uses OCR-generated text input as well as the image of the document. Additionally, we consider four LLMs. Firstly, mPlugDocOwl 1.5 (8.1B parameters) and mPlugDocOwl 1.5-chat (8.1B parameters) (Hu et al., 2024), which are OCR-free LLMs and use the image of the document as input. Secondly, DocLLM (1.5B parameters) (Wang et al., 2023a) and the text-only variant of GPT-4<sup>7</sup> (OpenAI, 2023)

<sup>7</sup>We use gpt-4-0613 prompted with text tokens. Other models are prompted as described in their respective papers.

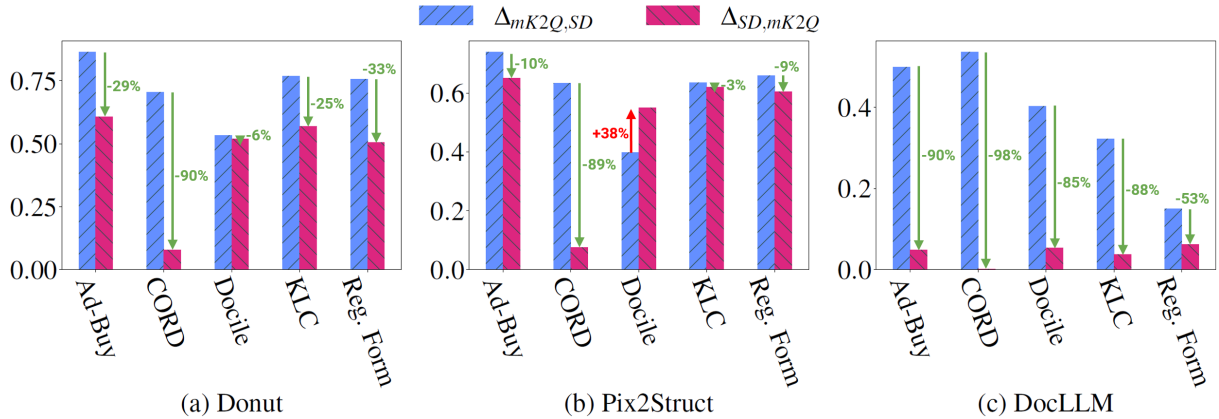


Figure 3: Comparison of training and evaluating on complex questions (mK2Q) and simple questions (SD).

Model	Ad-Buy	CORD	Docile	KLC	Reg-Form
Dnt <sub>ZS</sub>	1.4	18.0	7.2	3.8	5.9
P2S <sub>ZS</sub>	13.5	27.0	24.1	11.8	24.0
Doc <sub>ZS</sub>	23.9	43.0	48.0	<b>80.4</b>	27.6
UDOP <sub>ZS</sub>	29.1	29.7	35.1	30.0	39.1
mPD <sub>ZS</sub>	60.9	64.5	56.0	66.8	65.2
mPDC <sub>ZS</sub>	61.9	64.4	53.4	66.0	66.8
GPT-4 <sub>ZS</sub>	<b>72.7</b>	<b>85.3</b>	<b>61.3</b>	68.1	<b>76.5</b>
Dnt <sub>SD</sub>	7.7	24.4	22.3	15.6	19.7
P2S <sub>SD</sub>	18.0	31.0	35.5	28.0	29.7
Doc <sub>SD</sub>	<b>46.3</b>	<b>43.6</b>	<b>53.0</b>	<b>62.8</b>	<b>66.2</b>
Dnt <sub>mK2Q</sub>	56.4	82.7	47.7	67.2	81.4
P2S <sub>mK2Q</sub>	69.2	84.7	59.0	77.0	<b>87.0</b>
Doc <sub>mK2Q</sub>	<b>92.6</b>	<b>94.0</b>	<b>88.9</b>	<b>92.6</b>	78.0
Dnt <sub>K2Q</sub>	58.5	83.3	47.8	68.1	82.7
P2S <sub>K2Q</sub>	73.8	86.5	59.5	79.4	<b>88.7</b>
Doc <sub>K2Q</sub>	<b>93.9</b>	<b>96.5</b>	<b>90.0</b>	<b>93.6</b>	80.3

Table 7: ANLS ( $\uparrow$ ) results on K2Q test set using various training settings for models Donut (Dnt), Pix2Struct (P2S), DocLLM (Doc), UDOP, mPlugDocOwl 1.5 (mPD), mPlugDocOwl 1.5-Chat (mPDC), and GPT-4.

, which are OCR-based. GPT-4, UDOP, and the mPlugDocOwl models are only run in the zero-shot experiments due to resource constraints and data contamination considerations. Indeed, UDOP and the mPlugDocOwl models incorporate some of our original datasets in their pretraining; this may also be the case for GPT-4. We initialize all trainable models from DocVQA-fine-tuned checkpoints. Further model set-up details are given in App. C.1.

## 4.2 Evaluation Settings

We consider several training data settings for our modeling experiments. Firstly, we focus on the full K2Q dataset as presented in Section 3. Secondly, a baseline dataset, SD (as described in Section 3.2), is used to quantify the importance of having richer and more diverse templates. K2Q offers two main

advantages over SD: (1) the complexity of the questions and (2) the size of the dataset. To measure the impact of each advantage, we also train the models on a down-sampled version of K2Q, which we call **mini-K2Q (mK2Q)**. This dataset aims to reflect the same complexities as K2Q but with a comparable size to SD. Note that questions that span multiple pages are not used in our experiments, as not all models have multi-page capabilities.

We evaluate model performance using the Average Normalized Levenshtein Similarity (ANLS) metric (Biten et al., 2019). We define the ANLS score in App. C.2. We chose ANLS as the performance measure because it focuses on the surface similarity of the generated and true answer (as is intended in extraction) but does not suffer from the rigidity of exact matches.

## 4.3 Model Performance on K2Q

Table 7 gives the benchmark results for K2Q.<sup>8</sup> As expected, training on simple templates (SD) and testing on K2Q questions yields much lower performance than training directly on K2Q or mK2Q. We delve into the important effects of training and testing on different datasets in Section 4.4. In general, models trained on simple templates perform slightly better than their zero-shot counterparts, except DocLLM trained on KLC with simple templates and tested on K2Q templates. We also observe a consistently positive impact of training on K2Q versus its down-sampled counterpart mK2Q across the board (1.5% for Donut, 3% for Pix2Struct, and 1.9% for DocLLM on average). This highlights the diversity of questions contained in K2Q as well as the impact of greater training volume enabled by a template-based approach. These trends support our hypothesized benefits of K2Q.

<sup>8</sup>Results on SD are given in App. D.1 for comparison.

App. D.2 contains an ablation study on how the different training configurations impact performance on extractive and boolean questions and the number of entities present in each question.

We typically observe performance increases with model size on the same data setting. Indeed, DocLLM performs best among the models trained, followed by Pix2Struct and Donut. In the zero-shot setting, GPT-4 is the best model on all datasets except KLC. There is a large gap between the zero-shot and the fine-tuned results for Pix2Struct, Donut, and DocLLM – despite the original checkpoints we used being fine-tuned on DocVQA. This indicates sensitivity to distribution shifts in the type of documents and questions.

#### 4.4 Post-training Assessment of Model Robustness to Template Change

This work hypothesizes that using the same simple template for all datasets for training or evaluation does not reflect the intricacies of document understanding and may reduce model robustness at test time. To address this, we propose a suite of rich and diverse templates for transforming each dataset to create K2Q. To validate this hypothesis and motivate the need for our work, *we compare how models trained on simple templates perform when tested on K2Q templates, and vice versa*. To evaluate this experiment, we define the following metric that quantifies model robustness at test time to change in the templating approach used at train time:

$$\Delta_{D_1, D_2} \stackrel{\text{def}}{=} \frac{\text{ANLS}_{D_1/D_1} - \text{ANLS}_{D_2/D_1}}{\text{ANLS}_{D_1/D_1}} \quad (1)$$

where  $D_1$  and  $D_2$  are datasets drawn from different distributions, and  $\text{ANLS}_{D_2/D_1}$  is the ANLS score of a model trained on the train split of  $D_2$  and evaluated on the test split of  $D_1$ .  $\text{ANLS}_{D_1/D_1} \geq \text{ANLS}_{D_2/D_1}$  most often, as training and testing on datasets drawn from the same distribution almost always leads to better performance. Thus,  $\Delta_{D_1, D_2}$  measures the change in performance when swapping  $D_1$  and  $D_2$  for training while testing on  $D_1$ . A large value of  $\Delta_{D_1, D_2}$  indicates that training on  $D_2$  cannot generalize well to  $D_1$ . Conversely, a low value indicates that training on  $D_2$  or  $D_1$  generalizes well to  $D_1$ .

Figure 3 shows the difference between  $\Delta_{\text{mK2Q}, \text{SD}}$  and  $\Delta_{\text{SD}, \text{mK2Q}}$  for the three trainable models. We use mK2Q (the down-sampled version of K2Q introduced in Section 4.2) to enable a

fair comparison based on template style and not affected by training data volume. We observe in Figure 3 that the red bars corresponding to  $\Delta_{\text{SD}, \text{mK2Q}}$  are lower than the blue bars corresponding to  $\Delta_{\text{mK2Q}, \text{SD}}$  in 14 out of 15 cases. Consequently, training on mK2Q tends to yield better generalization to SD than the other way around. Indeed, a difference of 45% is observed when comparing  $\Delta_{\text{mK2Q}, \text{SD}}$  to  $\Delta_{\text{SD}, \text{mK2Q}}$  on average across all datasets and models (see green and red arrows in Figure 3). This observation corroborates the motivation behind K2Q that employing a rich and diverse set of templates for training models results in better robustness to new types of questions and formulations. We provide more metrics, including a comparison with the full K2Q, in App. D.3.

#### 4.5 Evaluation of Generated Errors

While ANLS is a useful metric for comparing extractive results from generative models, it fails to determine the cause of the errors, such as misreading the text in the document or not understanding the question. We use the notion of **groundedness** to determine this breakdown. A generated response is considered grounded if it can be identified in the OCR output of the document.<sup>9</sup> Correct generations are naturally grounded. We define incorrect generations that are grounded as **mis-extractions**. For ungrounded generations, we consider extracted strings that have an ANLS score of 0.8 or higher to be **misprints**. Any other errors we label as **other**. We provide examples in App. D.4.1.

Figure 4 gives the breakdown of groundedness and error types for the KLC extractive questions of K2Q using Donut, Pix2Struct, and DocLLM fine-tuned on K2Q. We choose KLC due to its large test set. The models tested on K2Q typically exhibit a higher level of grounding than those tested on SD, regardless of the training data. This could suggest that formulating templates for the dataset rather than using generic templates allows the models to contextualize the questions better. Grounded responses enable easier verification of KIE model outputs. Note that both Donut and Pix2Struct contain a large number of misprint errors compared to DocLLM. This could mean that using a more powerful OCR-free method may result in much stronger performance.

<sup>9</sup>We consider a response to be in the OCR if we can find a non-case-sensitive match. The match can be found within an OCR token or across OCR tokens.



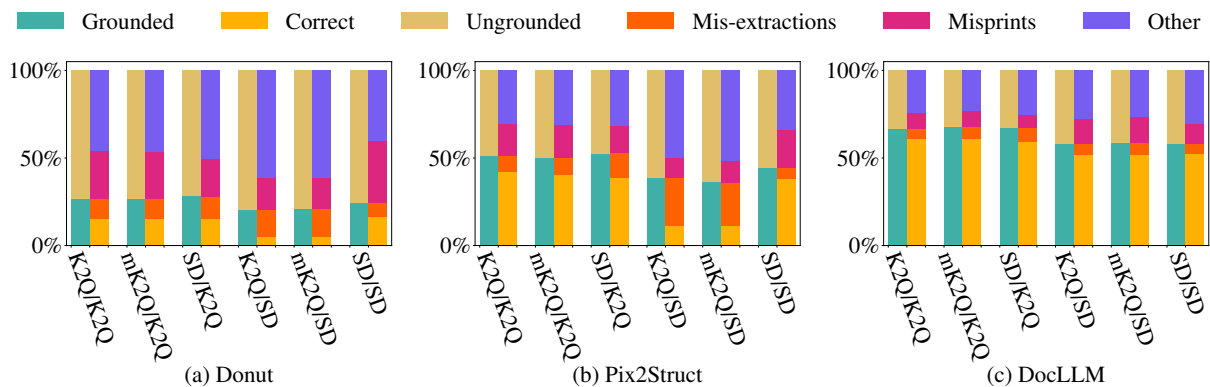


Figure 4: Detailed breakdown of groundedness and error types for KLC using different training/testing datasets.

## 5 Conclusion

This paper introduced K2Q, a new publicly available collection of five transformed KIE datasets for generative VRDU models. K2Q provides a large and diverse set of template-based questions that better capture the intricacies of KIE and the variety of questions that users can ask in real-world applications. We present a middle ground between LLM-generated and manually-curated questions for instruction tuning, which trades off the time to craft such data against diversity. Our approach can be extended to domains such as general VQA, multi-turn multimodal conversation, and video QA, which we leave to future work. Our experiments demonstrate that training generative models on K2Q instead of data from simple templates improves generalization to held-out types of instructions. In addition, our error analysis suggests that questions in K2Q provide enhanced contextualization compared to simple ones, resulting in more grounded answers from models, regardless of correctness. In future work, we plan to create templates covering a wider range of question types, few-shot instances, chain-of-thought answers with layout-informed explanations, and multi-round instructions. We hope this work encourages researchers to carefully consider the data used for generative modeling.

## Limitations

We conducted our experiments using three trainable models (Donut, Pix2Struct, and DocLLM) and four zero-shot models (UDOP, mPlugDocOwl 1.5, mPlugDocOwl 1.5-chat, and GPT-4). With additional resources, training state-of-the-art OCR-free models such as the mPlugDocOwl models would provide more complete results. Mainly due to resource limitations for training vision-based

LLMs, we left these experiments (along with training UDOP) for future work. We also note that mPlugDocOwl, UDOP, and possibly GPT-4 have already seen some of the datasets used in this work during the pretraining phase. Thus, data contamination could affect the zero-shot and fine-tuning performance of these models.

Additionally, while K2Q alleviates the burden of data collection by relying on existing KIE datasets, it still requires human intervention to curate high-quality, diverse templates manually. We will investigate using LLMs such as GPT-4 to generate templates for VRDU applications in future work.

## Disclaimer

This paper was prepared for information purposes by the Artificial Intelligence Research group of JPMorgan Chase & Co and its affiliates (“JP Morgan”) and is not a product of the Research Department of JP Morgan. J.P. Morgan makes no representation and warranty whatsoever and disclaims all liability for the completeness, accuracy, or reliability of the information contained herein. This document is not intended as investment research or investment advice, or a recommendation, offer, or solicitation for the purchase or sale of any security, financial instrument, financial product, or service, or to be used in any way for evaluating the merits of participating in any transaction, and shall not constitute a solicitation under any jurisdiction or to any person if such solicitation under such jurisdiction or to such person would be unlawful.

## References

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. 2023. [Qwen-VL: A frontier large](#)

- vision-language model with versatile abilities. *CoRR*, abs/2308.12966.
- Ali Furkan Biten, Rubèn Tito, Andrés Mafla, Lluís Gómez i Bigorda, Marçal Rusiñol, C. V. Jawahar, Ernest Valveny, and Dimosthenis Karatzas. 2019. [Scene text visual question answering](#). In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 4290–4300. IEEE.
- Lukasz Borchmann, Michal Pietruszka, Tomasz Stanislawek, Dawid Jurkiewicz, Michal Turski, Karolina Szyndler, and Filip Gralinski. 2021. [DUE: end-to-end document understanding benchmark](#). In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual*.
- Jerry Cheung, Yuchen Zhuang, Yinghao Li, Pranav Shetty, Wantian Zhao, Sanjeev Gramppurohit, Rampi Ramprasad, and Chao Zhang. 2024. [POLYIE: A dataset of information extraction from polymer material scientific literature](#). In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 2370–2385, Mexico City, Mexico. Association for Computational Linguistics.
- Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Y. Zhao, Yanping Huang, Andrew M. Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. [Scaling instruction-finetuned language models](#). *CoRR*, abs/2210.11416.
- Brian L. Davis, Bryan S. Morse, Brian L. Price, Chris Tensmeyer, Curtis Wigington, and Vlad I. Morariu. 2022. [End-to-end document recognition and understanding with dessurt](#). In *Computer Vision - ECCV 2022 Workshops - Tel Aviv, Israel, October 23-27, 2022, Proceedings, Part IV*, volume 13804 of *Lecture Notes in Computer Science*, pages 280–296. Springer.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [BERT: pre-training of deep bidirectional transformers for language understanding](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.
- Yihao Ding, Siqu Long, Jiabin Huang, Kaixuan Ren, Xingxiang Luo, Hyunsuk Chung, and Soyeon Caren Han. 2023. [Form-nlu: Dataset for the form natural language understanding](#). In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '23*, page 2807–2816, New York, NY, USA. Association for Computing Machinery.
- Adam W. Harley, Alex Ufkes, and Konstantinos G. Derpanis. 2015. [Evaluation of deep convolutional nets for document image classification and retrieval](#). In *13th International Conference on Document Analysis and Recognition, ICDAR 2015, Nancy, France, August 23-26, 2015*, pages 991–995. IEEE Computer Society.
- Or Honovich, Thomas Scialom, Omer Levy, and Timo Schick. 2023. [Unnatural instructions: Tuning language models with \(almost\) no human labor](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 14409–14428. Association for Computational Linguistics.
- Anwen Hu, Haiyang Xu, Jiabo Ye, Ming Yan, Liang Zhang, Bo Zhang, Chen Li, Ji Zhang, Qin Jin, Fei Huang, and Jingren Zhou. 2024. [mPLUG-DocOwl 1.5: Unified structure learning for OCR-free document understanding](#). *CoRR*, abs/2403.12895.
- Zheng Huang, Kai Chen, Jianhua He, Xiang Bai, Dimosthenis Karatzas, Shijian Lu, and C. V. Jawahar. 2019. [ICDAR2019 competition on scanned receipt OCR and information extraction](#). In *2019 International Conference on Document Analysis and Recognition, ICDAR 2019, Sydney, Australia, September 20-25, 2019*, pages 1516–1520. IEEE.
- Guillaume Jaume, Hazim Kemal Ekenel, and Jean-Philippe Thiran. 2019. [FUNSD: A dataset for form understanding in noisy scanned documents](#). In *2nd International Workshop on Open Services and Tools for Document Analysis, OST@ICDAR 2019, Sydney, Australia, September 22-25, 2019*, pages 1–6. IEEE.
- Geewook Kim, Teakgyu Hong, Moonbin Yim, Jeongyeon Nam, Jinyoung Park, Jinyeong Yim, Wonseok Hwang, Sangdoon Yun, Dongyoon Han, and Seunghyun Park. 2022. [OCR-free document understanding transformer](#). In *Computer Vision - ECCV 2022 - 17th European Conference, Tel Aviv, Israel, October 23-27, 2022, Proceedings, Part XXVIII*, volume 13688 of *Lecture Notes in Computer Science*, pages 498–517. Springer.
- Jordy Van Landeghem, Rubèn Tito, Lukasz Borchmann, Michal Pietruszka, Pawel Józiak, Rafal Powalski, Dawid Jurkiewicz, Mickaël Coustaty, Bertrand Anckaert, Ernest Valveny, Matthew B. Blaschko, Sien Moens, and Tomasz Stanislawek. 2023. [Document understanding dataset and evaluation \(DUDE\)](#). *CoRR*, abs/2305.08455.
- Kenton Lee, Mandar Joshi, Iulia Raluca Turc, Hexiang Hu, Fangyu Liu, Julian Martin Eisenschlos, Urvasi Khandelwal, Peter Shaw, Ming-Wei Chang,

- and Kristina Toutanova. 2023. [Pix2struct: Screenshot parsing as pretraining for visual language understanding](#). In *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 18893–18912. PMLR.
- Junnan Li, Dongxu Li, Silvio Savarese, and Steven C. H. Hoi. 2023. [BLIP-2: bootstrapping language-image pre-training with frozen image encoders and large language models](#). In *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 19730–19742. PMLR.
- Minghao Li, Yiheng Xu, Lei Cui, Shaohan Huang, Furu Wei, Zhoujun Li, and Ming Zhou. 2020. [Docbank: A benchmark dataset for document layout analysis](#).
- Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. [Visual instruction tuning](#). In *Advances in Neural Information Processing Systems*, volume 36, pages 34892–34916. Curran Associates, Inc.
- 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](#). *CoRR*, abs/1907.11692.
- Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq R. Joty, and Enamul Hoque. 2022. [Chartqa: A benchmark for question answering about charts with visual and logical reasoning](#). In *Findings of the Association for Computational Linguistics: ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 2263–2279. Association for Computational Linguistics.
- Minesh Mathew, Viraj Bagal, Rubèn Tito, Dimosthenis Karatzas, Ernest Valveny, and C. V. Jawahar. 2022. [Infographicvqa](#). In *IEEE/CVF Winter Conference on Applications of Computer Vision, WACV 2022, Waikoloa, HI, USA, January 3-8, 2022*, pages 2582–2591. IEEE.
- Minesh Mathew, Dimosthenis Karatzas, and C. V. Jawahar. 2021. [DocVQA: A dataset for VQA on document images](#). In *IEEE Winter Conference on Applications of Computer Vision, WACV 2021, Waikoloa, HI, USA, January 3-8, 2021*, pages 2199–2208. IEEE.
- OpenAI. 2023. [GPT-4 technical report](#). *CoRR*, abs/2303.08774.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. [Bleu: a method for automatic evaluation of machine translation](#). In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, July 6-12, 2002, Philadelphia, PA, USA*, pages 311–318. ACL.
- Seunghyun Park, Seung Shin, Bado Lee, Junyeop Lee, Jaeheung Surh, Minjoon Seo, and Hwalsuk Lee. 2019. [CORD: a consolidated receipt dataset for post-ocr parsing](#). In *Workshop on Document Intelligence at NeurIPS 2019*.
- Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. [Automatic differentiation in pytorch](#).
- Baolin Peng, Chunyuan Li, Pengcheng He, Michel Galley, and Jianfeng Gao. 2023. [Instruction tuning with GPT-4](#). *CoRR*, abs/2304.03277.
- Qiming Peng, Yinxu Pan, Wenjin Wang, Bin Luo, Zhenyu Zhang, Zhengjie Huang, Yuhui Cao, Weichong Yin, Yongfeng Chen, Yin Zhang, Shikun Feng, Yu Sun, Hao Tian, Hua Wu, and Haifeng Wang. 2022. [Ernie-layout: Layout knowledge enhanced pre-training for visually-rich document understanding](#). In *Findings of the Association for Computational Linguistics: EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 3744–3756. Association for Computational Linguistics.
- Le Qi, Shangwen Lv, Hongyu Li, Jing Liu, Yu Zhang, Qiaoqiao She, Hua Wu, Haifeng Wang, and Ting Liu. 2022. [DuReader<sub>vis</sub>: A Chinese dataset for open-domain document visual question answering](#). In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1338–1351, Dublin, Ireland. Association for Computational Linguistics.
- Stepán Simsa, Milan Sulc, Michal Uricár, Yash Patel, Ahmed Hamdi, Matej Kocián, Matyáš Skalický, Jirí Matas, Antoine Doucet, Mickaël Coustaty, and Dimosthenis Karatzas. 2023. [DocILE benchmark for document information localization and extraction](#). *CoRR*, abs/2302.05658.
- Tomasz Stanislawek, Filip Gralinski, Anna Wróblewska, Dawid Lipinski, Agnieszka Kaliska, Paulina Rosalska, Bartosz Topolski, and Przemyslaw Biecek. 2021. [Kleister: Key information extraction datasets involving long documents with complex layouts](#). In *16th International Conference on Document Analysis and Recognition, ICDAR 2021, Lausanne, Switzerland, September 5-10, 2021, Proceedings, Part I*, volume 12821 of *Lecture Notes in Computer Science*, pages 564–579. Springer.
- Hongbin Sun, Zhanghui Kuang, Xiaoyu Yue, Chenhao Lin, and Wayne Zhang. 2021. [Spatial dual-modality graph reasoning for key information extraction](#). *CoRR*, abs/2103.14470.
- Ryota Tanaka, Taichi Iki, Kyosuke Nishida, Kuniko Saito, and Jun Suzuki. 2024. [Instructdoc: A dataset for zero-shot generalization of visual document understanding with instructions](#). In *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, pages 19071–19079. AAAI Press.
- Zineng Tang, Ziyi Yang, Guoxin Wang, Yuwei Fang, Yang Liu, Chenguang Zhu, Michael Zeng, Cha Zhang, and Mohit Bansal. 2023. [Unifying vision](#),

- text, and layout for universal document processing. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pages 19254–19264. IEEE.
- Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. [https://github.com/tatsu-lab/stanford\\_alpaca](https://github.com/tatsu-lab/stanford_alpaca).
- Rubèn Tito, Dimosthenis Karatzas, and Ernest Valveny. 2022. Hierarchical multimodal transformers for multi-page docvqa. *CoRR*, abs/2212.05935.
- Dongsheng Wang, Natraj Raman, Mathieu Sibue, Zhiqiang Ma, Petr Babkin, Simerjot Kaur, Yulong Pei, Armineh Nourbakhsh, and Xiaomo Liu. 2023a. DocLLM: A layout-aware generative language model for multimodal document understanding.
- Jiapeng Wang, Chongyu Liu, Lianwen Jin, Guozhi Tang, Jiaxin Zhang, Shuaitao Zhang, Qianying Wang, Yaqiang Wu, and Mingxiang Cai. 2021. Towards robust visual information extraction in real world: New dataset and novel solution. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 2738–2745. AAAI Press.
- Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023b. Self-instruct: Aligning language models with self-generated instructions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 13484–13508. Association for Computational Linguistics.
- Zilong Wang, Yichao Zhou, Wei Wei, Chen-Yu Lee, and Sandeep Tata. 2023c. VRDU: A benchmark for visually-rich document understanding. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD '23*, page 5184–5193, New York, NY, USA. Association for Computing Machinery.
- Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. 2022. Finetuned language models are zero-shot learners. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.
- Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface’s transformers: State-of-the-art natural language processing. *CoRR*, abs/1910.03771.
- Yiheng Xu, Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, and Ming Zhou. 2020. LayoutLM: Pre-training of text and layout for document image understanding. In *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, pages 1192–1200. ACM.
- Yiheng Xu, Tengchao Lv, Lei Cui, Guoxin Wang, Yijuan Lu, Dinei Florencio, Cha Zhang, and Furu Wei. 2022. XFUND: A benchmark dataset for multilingual visually rich form understanding. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 3214–3224, Dublin, Ireland. Association for Computational Linguistics.
- Jiabo Ye, Anwen Hu, Haiyang Xu, Qinghao Ye, Ming Yan, Yuhao Dan, Chenlin Zhao, Guohai Xu, Chenliang Li, Junfeng Tian, Qian Qi, Ji Zhang, and Fei Huang. 2023a. mplug-docowl: Modularized multimodal large language model for document understanding. *CoRR*, abs/2307.02499.
- Jiabo Ye, Anwen Hu, Haiyang Xu, Qinghao Ye, Ming Yan, Guohai Xu, Chenliang Li, Junfeng Tian, Qi Qian, Ji Zhang, Qin Jin, Liang He, Xin Lin, and Fei Huang. 2023b. UReader: Universal OCR-free visually-situated language understanding with multimodal large language model. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 2841–2858, Singapore. Association for Computational Linguistics.
- Jiacheng Ye, Jiahui Gao, Qintong Li, Hang Xu, Jiangtao Feng, Zhiyong Wu, Tao Yu, and Lingpeng Kong. 2022. Zerogen: Efficient zero-shot learning via dataset generation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 11653–11669. Association for Computational Linguistics.
- Yanzhe Zhang, Ruiyi Zhang, Jiuxiang Gu, Yufan Zhou, Nedim Lipka, Diyi Yang, and Tong Sun. 2023. LLaVAR: Enhanced visual instruction tuning for text-rich image understanding. *CoRR*, abs/2306.17107.
- Yaoming Zhu, Sidi Lu, Lei Zheng, Jiaxian Guo, Weinan Zhang, Jun Wang, and Yong Yu. 2018. Texus: A benchmarking platform for text generation models. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR 2018, Ann Arbor, MI, USA, July 08-12, 2018*, pages 1097–1100. ACM.
- Ran Zmigrod, Zhiqiang Ma, Armineh Nourbakhsh, and Sameena Shah. 2024a. TreeForm: End-to-end annotation and evaluation for form document parsing. In *Proceedings of The 18th Linguistic Annotation Workshop (LAW-XVIII)*, pages 1–11, St. Julians, Malta. Association for Computational Linguistics.

Ran Zmigrod, Dongsheng Wang, Mathieu Sibue, Yulong Pei, Petr Babkin, Ivan Brugere, Xiaomo Liu, Nacho Navarro, Antony Papadimitriou, William Watson, Zhiqiang Ma, Armineh Nourbakhsh, and Sameena Shah. 2024b. [BuDDIE: A business document dataset for multi-task information extraction](#). *CoRR*, abs/2404.04003.

## A Dataset Construction

### A.1 Data Cleaning

To make questions and answers sound natural, we apply several data cleaning rules to the KIE annotations in the datasets instead of using the OCR text. We describe the data cleaning method for each dataset below.

**Ad-Buy.** For all entities, we replace all new line characters with spaces except for addresses where we insert a comma. Next, we check if the text is enclosed in brackets (parentheses, square, braces, etc.) and if so, we remove the brackets. In addition, we found that some entities were prefixed with “REMIT TO”, so we deleted any occurrence of this prefix. For the advertiser, agency, product, program\_desc, and tv\_address entities, we convert all multi-word, full upper case entities into title case. We have an additional caveat for tv\_address and do not modify words less than two characters as they are likely US states.

**CORD.** CORD contains too many entities to opt for an entity-type approach. However, two-thirds of all entities are numeric. For these entities, we remove all OCR tokens for which over half the characters are non-numeric. We also remove any leading or trailing punctuation symbols from the entities.

**Docile.** We take a similar approach to cleaning Docile entities as with CORD due to the similarities in their annotation schemes. For numeric entities, we consider only numerical and punctuation symbols. We treat numbers differently in Docile than in CORD due to our dataset-specific analyses of the original annotations. For non-numeric entities, we replace newline characters with a comma and a space. Additionally, we pre-process the text to the title case. We acknowledge that this may not always be the correct choice of data cleaning.

**KLC.** For KLC, we only title case entities that are address\_post\_town, address\_street\_line, and charity\_name. The other entities are identification type entities (e.g., charity\_name) or numbers and so data cleaning may compromise their values.

**Reg. Form.** We apply data cleaning to Reg. Form using the same methods described for Ad-Buy. In addition, we apply additional checks for

Dataset	Train	Dev	Test
Ad-Buy	11,362 (480)	— (—)	3,757 (161)
CORD	62,948 (800)	7,242 (100)	7,551 (100)
Docile	169,664 (5,180)	— (—)	15,893 (500)
KLC	27,180 (1,729)	7,294 (440)	10,339 (609)
Reg. Form	17,539 (1,436)	— (—)	5,888 (479)

Table 8: Number of questions and documents (in brackets) for K2Q splits.

words that should not be converted to title case. For each entity type, we find common acronyms used in the text (e.g., USA, LLC) and ensure these tokens remain in all caps.

### A.2 Data Splits

We follow the data splits provided by the original datasets. For Docile, the downloadable test set does not give the annotated entities; we thus use the dev set as its test set. For Ad-Buy and Reg. Form, multiple splits were given in Wang et al. (2023c), so we follow the split in Wang et al. (2023a). We provide the number of questions and documents for each dataset split in Table 8.

### A.3 Error type definitions in validation exercise

Table 4 reports the various types of errors we found in the template-generated questions. We define these errors in more detail below, along with guidelines provided to annotators.

1. **Template Error:** This is a grammatical mistake in the template. If the question doesn’t make sense after plugging in the specific values, then it is marked as a template error.
2. **Cleaning Error:** This is a mistake in post-processing the value, such as changing the case. The question would make sense if the original value from the document was used.
3. **Annotation Error:** This is a mistake in the original dataset where an entity should have been annotated differently. For example, “What was the address of the company named 123?” where 123 is an ID but was tagged as a name.

4. Other Error: The question doesn't make sense for a reason not included in the above errors.

## B K2Q Examples

In this section, we provide example documents and questions generated from each of the five datasets of K2Q. Figure 5, Figure 6, Figure 7, Figure 8, and Figure 9 compare questions generated in K2Q to those found in InstructDoc (Tanaka et al., 2024), UReader (Ye et al., 2023b), and SD. Note that InstructDoc and UReader do not contain most of the datasets used in this work. We thus use their templates to generate questions as described in the respective papers.

The key differences between the questions found in K2Q and those found in UReader and SD are the style and informativeness of the questions. Our templates are designed with the dataset in mind and thus sound more natural and resemble questions that one might encounter in the real world. Furthermore, our templates use other key entities within the questions to better contextualize queries as well as enable a wider selection of questions. We note that for datasets such as Docile which contain line items, UReader may generate ambiguous questions. For example, the question regarding the line item total price in Figure 7 has a variety of answers that could be correct as it is unclear which line was being referred to. Docile contains documents that have many lines, so such under-specified questions are avoided.

The primary difference between K2Q and InstructDoc lies in the kind of questions being asked. As explained in Section 3.2, InstructDoc transforms the information extraction problem into a classification problem. Therefore, the KIE portion of InstructDoc trains and tests a model's ability to characterize a key entity to its correct type, but not to find the entity itself. Other than formulating questions for a different task, InstructDoc has five dataset-agnostic templates it populates. Consequently, the variety of questions in K2Q is much wider.

## C Experimental Set-up

### C.1 Model Training Details

Each trainable model (Donut, Pix2Struct, and DocLLM) is then fine-tuned for ten epochs on each dataset of this study individually, with a learning rate of  $10^{-4}$ . Donut and Pix2Struct use the AdaFactor optimizer with a cosine scheduler with 500

warm-up steps and batch sizes of four and one, respectively. A weight decay of  $10^{-5}$  is used. DocLLM uses the AdamW optimizer with a cosine scheduler with warm-up and a batch size of 24 through gradient accumulation. We use available implementations of these models (Wolf et al., 2019) in PyTorch 1.13 (Paszke et al., 2017). Each model is fine-tuned on a single A10G GPU.

### C.2 Metric Definitions

In this section, we provide more details regarding the perplexity, Self-BLEU, and ANLS scores used in Section 3.2 and Section 4 respectively.

**Perplexity.** Perplexity measures how likely a language model is to generate a given sequence of text. Mathematically, it is the inverse of the joint probability of a sequence of text being drawn from a distribution. Alternatively, it is the exponentiated average of the negative log-likelihood of the input probability conditioned on all previous tokens. If we consider a tokenized string  $s = \{s_1, \dots, s_n\}$ , the perplexity of a model  $M$  is defined as

$$\text{PPL}(s) \stackrel{\text{def}}{=} \exp\left(-\frac{1}{n} \sum_{i=1}^n \log p_M(s_i | s_{<i})\right)$$

where  $p_M(s_i | s_{<i})$  is the probability of witnessing token  $s_i$  in the model, after observing all past tokens of  $s$ .

**Self-BLEU.** Self-BLEU (Zhu et al., 2018) measures the lexical diversity of a text corpus. It leverages the BLEU metric (Papineni et al., 2002) to evaluate how one sentence resembles the rest in a collection. It is obtained by averaging the BLEU score of each sentence of the corpus (hypothesis) with the rest of the collection (reference). A sample of the corpus is typically used as the computation is too resource-intensive otherwise; we use a sample size of 5,000 test questions per dataset.

**ANLS.** The Average Normalised Levenshtein Similarity is a string edit distance that measures the number of edits it takes to transform one string into another, normalized against string length. It is defined as

$$\text{ANLS}(s, t) \stackrel{\text{def}}{=} \begin{cases} 1 - \text{NL}(s, t) & \text{if } \text{NL}(s, t) < \tau \\ 0 & \text{otherwise} \end{cases}$$

where  $\text{NL}(s, t)$  is the normalized Levenshtein Distance between  $s$  and  $t$ , and  $\tau$  is a distance threshold (typically 0.5). The reported score over a

# INVOICE

Advertiser	POL/Michael Bloomberg/President/US/D	Invoice #	2335618-1
Product	BLOOMBERG 4 PRES	Invoice Date	01/26/20
Estimate Number	0116	Invoice Month	January 2020
		Invoice Period	12/30/19 - 01/26/20
Property	KRCW	Order #	2335618
Account Executive	Telerep Philadelphia	Alt Order #	09735556
Sales Office	Telerep/Philadelphia	Deal #	
Sales Region	National	Order Flight	01/11/20 - 01/17/20
Billing Calendar	Broadcast	Agency Code	9915458
Billing Type	Cash	Advertiser Code	MBLM
Special Handling		Product 1/2	MBLM

### K2Q Questions:

Q: From when until when is the contract in flight?  
A: 01/11/20 - 01/17/20  
Q: Did POL/Michael Bloomberg/President/US/D order the advertisement "Bloomberg 4 Pres"? A: Yes

### UReader and SD Questions:

Q: What is the value for the product? A: BLOOMBERG 4 PRES  
Q: What is the value for the flight start date? A: 01/11/20

### InstructDoc Questions:

Q: There are 15 categories for selection: "advertiser", ..., and "tv\_address". Please output the category corresponding to the text "BLOOMBERG 4 PRES".  
A: product

Figure 5: Excerpt of an Ad-Buy document with generated questions from K2Q, InstructDoc, UReader, and SD. The K2Q question "From when until when is the contract in flight?" uses jargon specific to the advertising domain. Applying such templates allows for creating domain-specific and diverse questions, which may differ from what is colloquially used. The generated question is thus grounded in the jargon used in the document.

CHOC BANANA	10.000
CHEESE BUN	10.500
SUPER CHEESE	13.000
SUPER CHEESE	13.000
CLS CHK CHIL	12.000
PLASTIC BAG ME	0.000
6.00 ITEMS	
TOTAL	58.500
CASH	100.000
CHANGE	41.500

### K2Q Questions:

Q: What is the menu item that cost a total of 10.000 in this bill?  
A: CHOC BANANA  
Q: How much did the order(s) of CHOC BANANA cost in total?  
A: 10.000  
Q: Is "58.500" the amount of change in cash in this receipt? A: No

### UReader and SD Questions:

Q: What is the value for the amount of change in cash? A: 41.500  
Q: What is the value for the menu item? A: CHOC BANANA

### InstructDoc Questions:

Q: Please tell me the category of the text "41.500" to select from following classes: "menu.nm", ..., and "total.menuqty\_cnt".  
A: total.changeprice

Figure 6: Excerpt of a CORD document with generated questions from K2Q, InstructDoc, UReader, and SD. Note that the second question for UReader and SD is not present in SD due to its ambiguity.



## Order Details

#	Market Station	Bind To	Start Date	End Date	No Of W.	On Air W.	Sch Days	Skip W.	M	T	W	Th	F	Sa	Su	Spots/Week	Spot Len.	Revenue Type	Rate	Ord. Spots	Ord. Cost	Make Good
1	Fort Collins KCOL-AM	06:00-10:00 Commercial	06/13/2022	06/16/2022	1	1	3	0	X	X	X	X	-	-	-	20	60	Local Agency-Political	\$44.00	20	\$880.00	
2	Fort Collins KCOL-AM	10:00-15:00 Commercial	06/13/2022	06/16/2022	1	1	3	0	X	X	X	X	-	-	-	20	60	Local Agency-Political	\$39.00	20	\$780.00	
3	Fort Collins KCOL-AM	15:00-19:00 Commercial	06/13/2022	06/16/2022	1	1	3	0	X	X	X	X	-	-	-	20	60	Local Agency-Political	\$31.00	20	\$620.00	

# Digital	Start Date	End Date	Description	Rev. Type	Impressions
1	06/13/2022	06/16/2022	iHeart Audience Network (iAN) - Streaming	POLITICAL	31250

<b>Number of Spots:</b>	60	Ordered Gross:	\$2,280.00
<b>Number of Miscellaneous Lines:</b>	0	Agency Commission:	\$342.00
<b>Number of Digital Impressions:</b>	31250	Ordered Net:	\$1,938.00
		Digital Assets Gross:	\$500.00
		Agency Commission:	\$75.00
		Digital Assets Net:	\$425.00
		<b>Total Net Due:</b>	<b>\$2,363.00</b>

### K2Q Questions:

- Q: How much is the **total amount with tax** of the 2nd item? A: 780.00  
 Q: What is the **total amount to be paid** in the document? A: 2,363.00  
 Q: Is **"60.48"** the total amount to be paid in this document? A: No

### UReader and SD Questions:

- Q: What is the value for the **total amount with tax**? A: 780.00  
 Q: What is the value for the **total amount to be paid**? A: 2,363.00

### InstructDoc Questions:

- Q: The document contains 36 key categories: "account\_num", ..., and "total". Kindly identify the category related to the text **"2,363.00"** mentioned in the provided document.  
 A: amount\_due

Figure 7: Excerpt of a Docile document with generated questions from K2Q, InstructDoc, UReader, and SD. Note that the first question for UReader and SD is not present in SD due to its ambiguity.

**HASLEMERE EDUCATIONAL MUSEUM**

**REPORT OF THE TRUSTEES**

**FOR THE YEAR ENDED 31ST DECEMBER 2014**

The Trustees who are also directors of the charity for the purposes of the Companies Act 2006 report with the financial statements of the charity for the year ended 31st December 2014 adopted the provisions of the Statement of Recommended Practice (SORP) 'Accruals Basis' issued in March 2005.

**REFERENCE AND ADMINISTRATIVE DETAILS**

**Registered Company number:**  
03203702 (England and Wales)

**Registered Charity number:**  
1071244

**Registered Office:** 78 High Street  
Haslemere  
Surrey  
GU27 2LA

**K2Q Questions:**  
Q: What is the number of the charity Haslemere Educational Museum? A: 1071244  
Q: What is the postcode for Haslemere Educational Museum? A: GU27 2LA  
Q: Is GU27 2LA the postcode of street 78 High Street? A: Yes

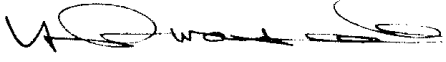
**UReader and SD Questions:**  
Q: What is the value for the charity number? A: 1071244  
Q: What is the value for the address postcode? A: GU27 2LA

**InstructDoc Questions:**  
Q: Categories: "address\_postcode", ..., and "spending\_annually\_in\_british\_pounds". Kindly provide me with the category of the text "78 High Street". A: address\_street\_line

Figure 8: Excerpt of a KLC document with generated questions from K2Q, InstructDoc, UReader, and SD.

14. Have two copies of this material been filed with the Department of Justice? Yes  No

15. Has this material been labeled as required by the act? Yes  No

Date of report 8/21/91	Name and title Yasumasa Iwasaki Executive Director	Signature 
---------------------------	--	---

**K2Q Questions:**  
Q: What position does the form signer hold? A: Executive Director  
Q: Did Yasumasa Iwasaki sign the form? A: Yes

**UReader and SD Questions:**  
Q: What is the value for the signer title? A: Executive Director  
Q: What is the value for the signer name? A: Yasumasa Iwasaki

**InstructDoc Questions:**  
Q: Options: "file\_date", ..., and "signer\_tital". Please select the category associated with the text "Yasumasa Iwasaki" in the given document.  
A: amount\_due

Figure 9: Excerpt of a Reg. Form document with generated questions from K2Q, InstructDoc, UReader, and SD.

dataset is the mean ANLS score over all answers in a dataset.

## D Additional Experimental Results

In this section, we provide the results of additional experiments that we could not provide in the main paper due to space constraints.

### D.1 Model Performance on SD

Table 9 gives the ANLS scores of our different modeling set-ups on the SD test set. As one would expect, training on SD yields the best outcomes. However, as observed in Section 4.3, DocLLM performs impressively when fine-tuned using the more diverse templates of K2Q. This suggests that the jump in model size from Donut and Pix2Struct to DocLLM enables better generalization with high-quality data.

### D.2 Ablation Studies

K2Q contains several interesting properties such as the existence of extractive and boolean questions and questions that contain multiple entities. We conduct an ablation study on how different training configurations and different models perform different question types and number of entities. Number of entities is counted as number of entities used within the question. Therefore, the minimum number of entities for extractive questions is zero while the minimum number of entities for boolean questions is one. These are given in Table 10, Table 11, Table 12, Table 13, and Table 14. Observe that Pix2Struct and Donut are unable to answer any boolean questions under the zero-shot setting or when trained on SD. This demonstrates the difficulty of generalizing beyond the types of questions seen during training. In general, the performance of the model decreases as the number of entities in the question increases.

### D.3 Extended Post-training Assessment of Model Robustness to Template Change

We give more results for the experiments described in Section 4.4. Figure 10 gives a complementary figure to Figure 3 but using K2Q rather than mK2Q with similar patterns observed. Since both Figure 3 and Figure 10 display relative differences, we illustrate the ANLS scores used to derive them in Figure 11 and Figure 12.

Model	Ad-Buy	CORD	Docile	KLC	Reg-Form
Dnt <sub>zs</sub>	4.8	61.1	8.9	7.6	15.5
P2S <sub>zs</sub>	19.1	63.8	25.2	12.8	30.0
Doc <sub>zs</sub>	1.6	72.9	28.1	<b>78.2</b>	2.8
UDOP <sub>zs</sub>	4.9	44.2	21.6	12.8	4.8
mPD <sub>zs</sub>	64.4	55.2	41.8	71.6	60.5
mPDC <sub>zs</sub>	<b>65.9</b>	42.5	41.2	71.8	59.8
GPT-4 <sub>zs</sub>	50.5	<b>82.4</b>	<b>48.0</b>	38.7	<b>70.2</b>
<hr/>					
Dnt <sub>sd</sub>	61.2	90.2	39.9	55.9	78.3
P2S <sub>sd</sub>	79.0	89.3	61.2	76.3	<b>85.6</b>
Doc <sub>sd</sub>	<b>95.7</b>	<b>94.4</b>	<b>86.7</b>	<b>88.9</b>	66.7
<hr/>					
Dnt <sub>mK2Q</sub>	24.0	80.3	19.2	24.1	38.8
P2S <sub>mK2Q</sub>	27.7	81.7	27.6	29.0	33.9
Doc <sub>mK2Q</sub>	<b>91.1</b>	<b>94.3</b>	<b>82.1</b>	<b>85.6</b>	<b>62.6</b>
<hr/>					
Dnt <sub>K2Q</sub>	24.2	81.7	19.3	24.3	35.9
P2S <sub>K2Q</sub>	28.2	81.6	27.6	29.9	28.2
Doc <sub>K2Q</sub>	<b>95.2</b>	<b>95.9</b>	<b>82.5</b>	<b>87.2</b>	<b>66.9</b>

Table 9: ANLS ( $\uparrow$ ) results on SD test set using various training settings for models Donut (Dnt), Pix2Struct (P2S), DocLLM (Doc), UDOP, mPlugDocOwl 1.5 (mPD), mPlugDocOwl 1.5-Chat (mPDC), and GPT-4.

### D.4 Extended Evaluation of Generated Errors

We give a breakdown of the groundedness and types of errors exhibited by our fine-tuned models on Ad-Buy, CORD, Docile, and Reg. Form in Figure 13, Figure 14, Figure 15, and Figure 16 respectively. We note that using K2Q does not always yield better groundedness than SD. This is likely due to the clarity of simple templates asking for an entity, specifically for CORD and Docile, for which there are many different entities. Nevertheless, we observe that for the majority of datasets, training using a version of K2Q leads to higher groundedness as well as a higher number of misprints. Misprints are a good error to observe as it indicates that the model was able to understand the question and answer, but could not generate the exact text.

We provide a few concrete examples of errors observed in Reg. Form below.

#### D.4.1 Error Examples.

We analyze some of the errors made by DocLLM (the best-performing model) on Reg. Form as a guide for future model development. Given that K2Q closely resembles the kind of questions users in the wild are likely to ask, such an error analysis provides insights as to what errors downstream applications can expect. An analysis using SD would not be able to provide such insights.

A representative example of an incorrect answer is from a question “Who is the signer of the form?”

Model	All	Extractive Questions					Boolean Questions		
		All	0 Entities	1 Entity	2 Entities	3 Entities	All	1 Entity	2 Entities
Donut <sub>ZS</sub>	1.4	2.5	2.9	1.7	0.6	19.0	0.0	0.0	0.0
Pix2Struct <sub>ZS</sub>	13.5	24.1	26.2	15.9	26.4	32.6	0.0	0.0	0.0
DocLLM <sub>ZS</sub>	23.9	5.3	5.6	2.2	7.4	16.8	47.6	47.3	48.1
UDOP <sub>ZS</sub>	29.1	22.0	16.4	23.4	36.1	42.4	38.2	42.8	27.7
mPlugDO <sub>ZS</sub>	60.9	47.2	52.2	31.0	54.0	25.8	78.4	82.2	69.6
mPlugDOC <sub>ZS</sub>	61.9	53.1	57.0	38.9	59.2	44.2	73.2	76.4	65.9
GPT-4 <sub>ZS</sub>	72.7	68.1	65.2	65.6	79.6	76.7	78.6	76.0	84.5
Donut <sub>SD</sub>	7.7	13.7	18.9	6.9	5.7	7.5	0.0	0.0	0.0
Pix2Struct <sub>SD</sub>	18.0	32.2	37.4	19.7	32.9	13.3	0.0	0.0	0.0
DocLLM <sub>SD</sub>	46.3	52.2	60.2	35.7	45.8	61.5	38.7	38.9	38.2
Donut <sub>mK2Q</sub>	56.4	48.2	44.6	44.6	62.8	65.1	66.9	69.7	60.5
Pix2Struct <sub>mK2Q</sub>	69.2	64.5	64.3	61.7	68.6	65.8	75.2	79.0	66.5
DocLLM <sub>mK2Q</sub>	92.6	89.3	93.6	83.1	84.2	77.5	96.7	97.6	94.8
Donut <sub>K2Q</sub>	58.5	49.1	45.9	44.8	63.4	63.1	70.5	73.9	62.6
Pix2Struct <sub>K2Q</sub>	73.8	68.6	69.6	63.2	72.1	69.2	80.3	84.7	70.2
DocLLM <sub>K2Q</sub>	93.9	91.5	96.2	85.6	85.3	77.4	96.9	97.6	95.2

Table 10: Ablation study of ANLS ( $\uparrow$ ) for K2Q Ad-Buy.

Model	All	Extractive Questions			Boolean Questions
		All	0 Entities	1 Entity	
Donut <sub>ZS</sub>	18.0	36.3	62.1	26.9	0.0
Pix2Struct <sub>ZS</sub>	27.0	54.3	68.7	49.1	0.0
DocLLM <sub>ZS</sub>	43.0	62.9	83.7	55.4	23.4
UDOP <sub>ZS</sub>	29.7	33.0	49.0	27.2	26.4
mPlugDO <sub>ZS</sub>	64.5	76.8	72.9	78.2	52.3
mPlugDOC <sub>ZS</sub>	64.4	76.2	74.0	77.0	52.8
GPT-4 <sub>ZS</sub>	85.3	87.7	88.0	87.6	82.9
Donut <sub>SD</sub>	24.4	49.0	86.1	35.5	0.0
Pix2Struct <sub>SD</sub>	31.0	62.4	89.3	52.6	0.0
DocLLM <sub>SD</sub>	43.6	70.8	95.6	61.9	16.6
Donut <sub>mK2Q</sub>	82.7	83.8	84.2	83.6	81.7
Pix2Struct <sub>mK2Q</sub>	84.7	88.4	86.6	89.0	81.1
DocLLM <sub>mK2Q</sub>	94.0	95.1	96.8	94.5	93.0
Donut <sub>K2Q</sub>	83.3	85.1	84.3	85.4	81.5
Pix2Struct <sub>K2Q</sub>	86.5	90.2	88.8	90.6	83.0
DocLLM <sub>K2Q</sub>	96.5	96.6	96.9	96.5	96.3

Table 11: Ablation study of ANLS ( $\uparrow$ ) for K2Q CORD.

in which the OCR output for one of the documents is “/ Chad Horrell” while the generated answer is “Chad Horrell”. We noticed incorrect answers are often subsequences of the ground truth answer or token sequences with a slight offset from the ground truth. In the above example, the solution may be to start with higher quality annotations as the “/” seems out of place. Another example of what seems to be a misprint error is the question “What is the registration ID of this form?” which produces the answer “6228” instead of the ground truth “6278”.

We also observe many errors that seem to be unrelated to the question or document. For instance, DocLLM generated the answer “Government Of

Japan–Japan External Trade Organization” instead of “Embassy Of The State Of Qatar” for the question, “Which foreign principal is this form about?”. The generated answer is not present in the document but is present in a different document of the training dataset. This could suggest that the models may suffer from overfitting or hallucination issues.

We note also that numbers and dates can be reformatted in the generated output which can lead to ungrounded outputs with low ANLS. We evaluated this case and found that this type of error occurs in less than 0.5% of cases across datasets and models in the fine-tuned setting. Therefore, we do not categorize such errors here and place them in **other**.

Model	All	Extractive Questions			Boolean Questions
		All	0 Entities	1 Entity	
Donutz <sub>zs</sub>	7.2	10.4	11.2	0.2	0.0
Pix2Struct <sub>zs</sub>	24.1	35.0	37.0	10.1	0.0
DocLLM <sub>zs</sub>	48.0	42.1	43.5	23.9	61.0
UDOP <sub>zs</sub>	35.1	33.2	34.7	14.0	39.2
mPlugDO <sub>zs</sub>	56.0	49.7	52.2	18.0	69.7
mPlugDOC <sub>zs</sub>	53.4	50.4	53.0	17.5	59.9
GPT-4 <sub>zs</sub>	61.2	54.0	54.5	47.0	77.3
Donut <sub>sd</sub>	22.3	32.4	34.5	5.2	0.0
Pix2Struct <sub>sd</sub>	35.5	51.6	53.0	33.1	0.0
DocLLM <sub>sd</sub>	53.0	76.4	76.9	70.0	1.5
Donut <sub>mK2Q</sub>	47.7	39.0	40.9	14.4	66.9
Pix2Struct <sub>mK2Q</sub>	59.0	60.1	60.7	51.8	56.7
DocLLM <sub>mK2Q</sub>	88.9	85.7	85.8	84.5	95.8
Donut <sub>K2Q</sub>	47.8	39.1	41.0	14.4	67.2
Pix2Struct <sub>K2Q</sub>	59.5	60.6	61.2	53.4	57.1
DocLLM <sub>K2Q</sub>	90.0	87.0	87.2	83.7	96.8

Table 12: Ablation study of ANLS ( $\uparrow$ ) for K2Q Docile.

Model	All	Extractive Questions			Boolean Questions		
		All	0 Entities	1 Entity	All	1 Entity	2 Entities
Donut <sub>zs</sub>	3.8	9.2	19.9	5.1	0.1	0.3	0.1
Pix2Struct <sub>zs</sub>	11.8	28.8	38.7	25.0	0.0	0.0	0.0
DocLLM <sub>zs</sub>	80.4	80.4	85.1	78.7	80.4	80.1	80.5
UDOP <sub>zs</sub>	30.0	22.2	31.5	18.7	35.4	50.7	30.6
mPlugDO <sub>zs</sub>	66.8	65.6	66.7	65.2	67.6	78.4	64.2
mPlugDOC <sub>zs</sub>	66.0	64.9	64.1	65.3	66.7	81.0	62.4
GPT-4 <sub>zs</sub>	68.1	65.7	53.4	70.3	69.8	62.1	72.2
Donut <sub>sd</sub>	15.6	37.9	66.7	27.0	0.0	0.1	0.0
Pix2Struct <sub>sd</sub>	28.0	68.4	89.2	60.5	0.0	0.0	0.0
DocLLM <sub>sd</sub>	62.8	88.7	95.8	86.0	44.8	47.2	44.0
Donut <sub>mK2Q</sub>	67.2	45.9	75.6	34.6	82.1	86.8	80.6
Pix2Struct <sub>mK2Q</sub>	77.0	72.4	90.7	65.5	80.2	81.3	79.9
DocLLM <sub>mK2Q</sub>	92.6	88.4	94.3	86.2	95.5	98.2	94.7
Donut <sub>K2Q</sub>	68.1	46.1	76.0	34.8	83.3	87.8	82.0
Pix2Struct <sub>K2Q</sub>	79.4	74.0	91.5	67.3	83.2	84.1	82.9
DocLLM <sub>K2Q</sub>	93.6	89.7	95.8	87.4	96.3	98.9	95.4

Table 13: Ablation study of ANLS ( $\uparrow$ ) for K2Q KLC.

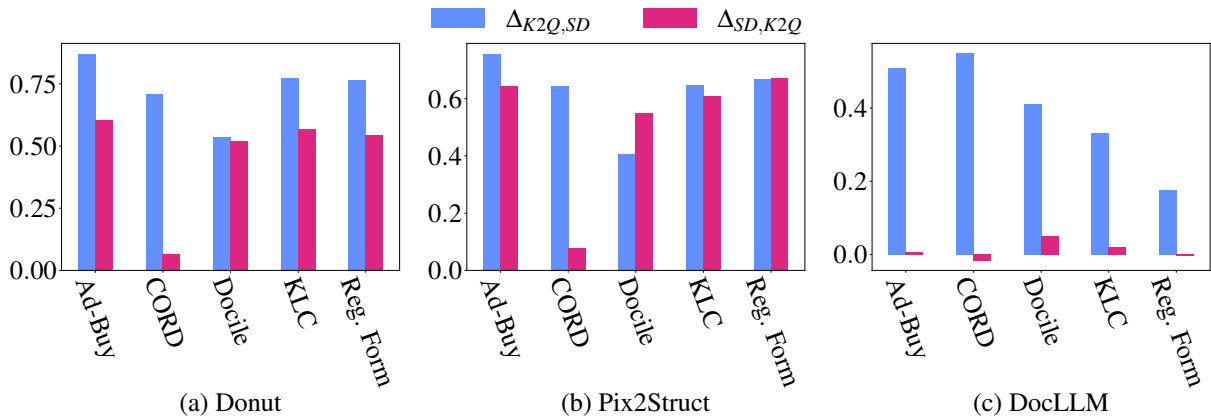


Figure 10: Comparison of training and evaluating on complex questions (K2Q) and simple questions (SD).

Model	All	Extractive Questions			Boolean Questions		
		All	0 Entities	1 Entity	All	1 Entity	2 Entities
Donut <sub>ZS</sub>	5.9	15.8	15.2	25.0	0.0	0.0	0.0
Pix2Struct <sub>ZS</sub>	24.0	63.7	62.1	91.3	0.0	0.0	0.0
DocLLM <sub>ZS</sub>	27.6	4.2	4.1	5.0	41.7	41.8	40.3
UDOP <sub>ZS</sub>	39.1	19.0	16.9	53.5	51.2	51.6	45.3
mPlugDO <sub>ZS</sub>	65.2	62.0	60.1	93.0	67.1	66.3	78.8
mPlugDOC <sub>ZS</sub>	66.8	65.5	63.9	92.2	67.6	67.4	71.2
GPT-4 <sub>ZS</sub>	76.5	74.6	73.2	97.1	77.7	78.4	66.1
Donut <sub>SD</sub>	19.7	52.4	51.0	75.3	0.0	0.0	0.0
Pix2Struct <sub>SD</sub>	29.7	78.8	78.3	87.0	0.0	0.0	0.0
DocLLM <sub>SD</sub>	66.2	64.2	62.6	91.1	67.5	68.1	58.5
Donut <sub>mK2Q</sub>	81.4	69.7	69.6	71.5	88.5	89.3	77.1
Pix2Struct <sub>mK2Q</sub>	87.0	81.1	80.9	84.3	90.7	91.1	84.7
DocLLM <sub>mK2Q</sub>	78.0	62.3	60.7	88.5	87.4	87.1	91.9
Donut <sub>K2Q</sub>	82.7	70.7	70.8	69.3	90.0	90.4	83.1
Pix2Struct <sub>K2Q</sub>	88.7	82.7	82.7	82.5	92.3	92.5	89.4
DocLLM <sub>K2Q</sub>	80.3	66.2	64.7	91.0	88.9	88.4	95.8

Table 14: Ablation study of ANLS ( $\uparrow$ ) for K2Q Reg. Form.

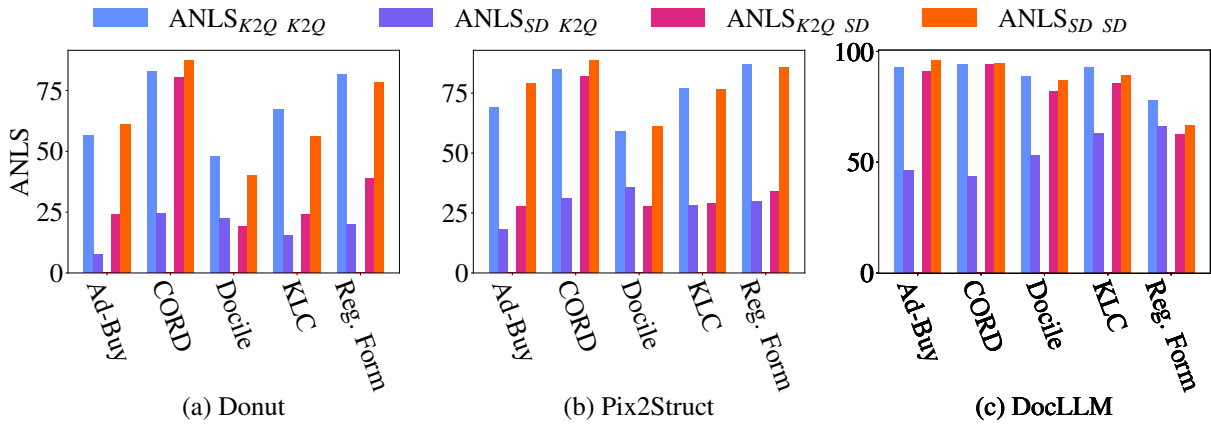


Figure 11: Comparison of ANLS scores for training and evaluating on sampled complex questions (mK2Q) and simple questions (SD).

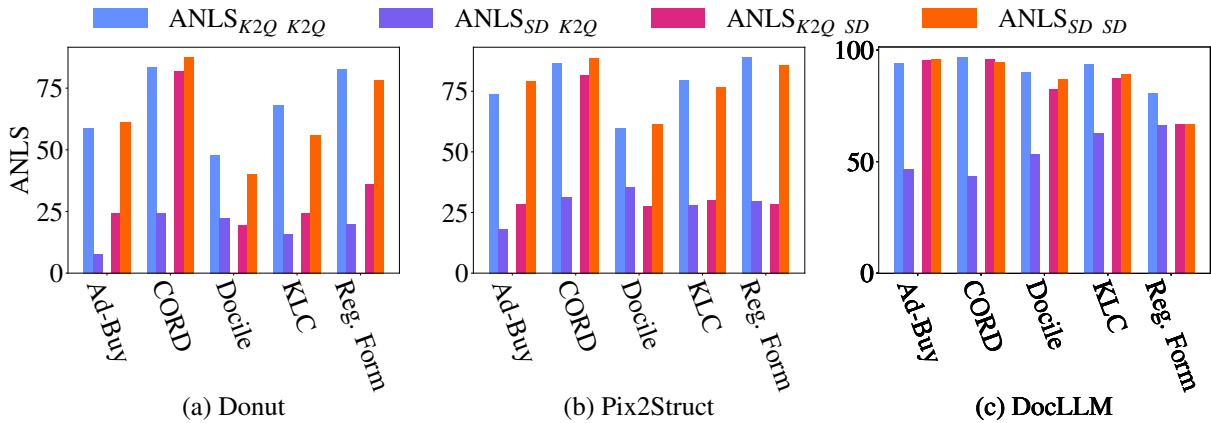


Figure 12: Comparison of ANLS scores for training and evaluating on complex questions (K2Q) and simple questions (SD).

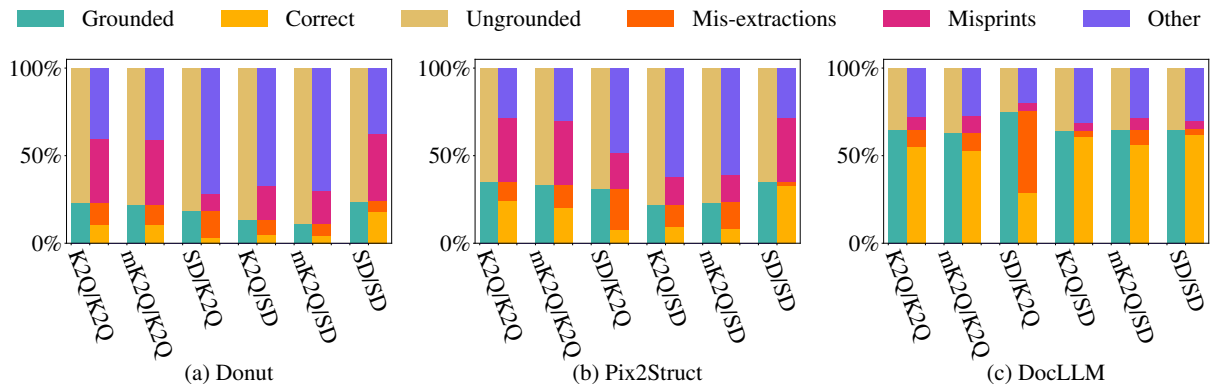


Figure 13: Detailed breakdown of groundedness and error types for Ad-Buy using different training / testing datasets.

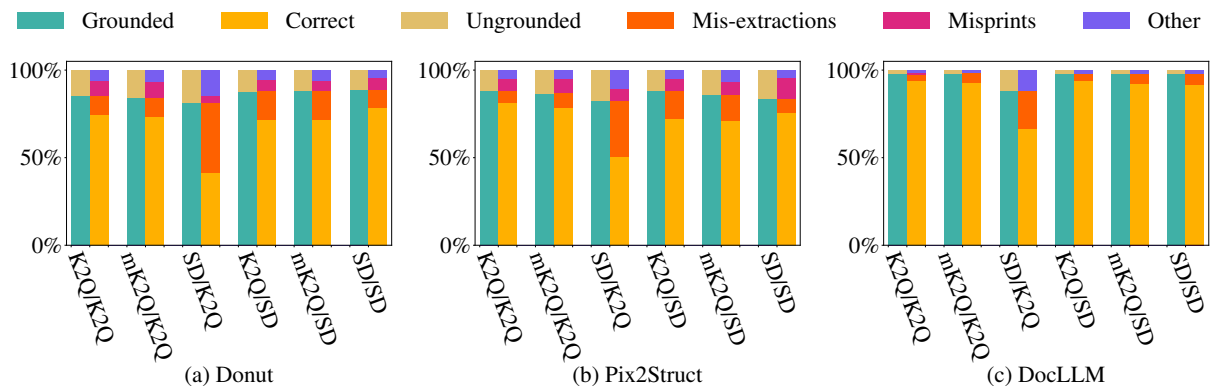


Figure 14: Detailed breakdown of groundedness and error types for CORD using different training / testing datasets.

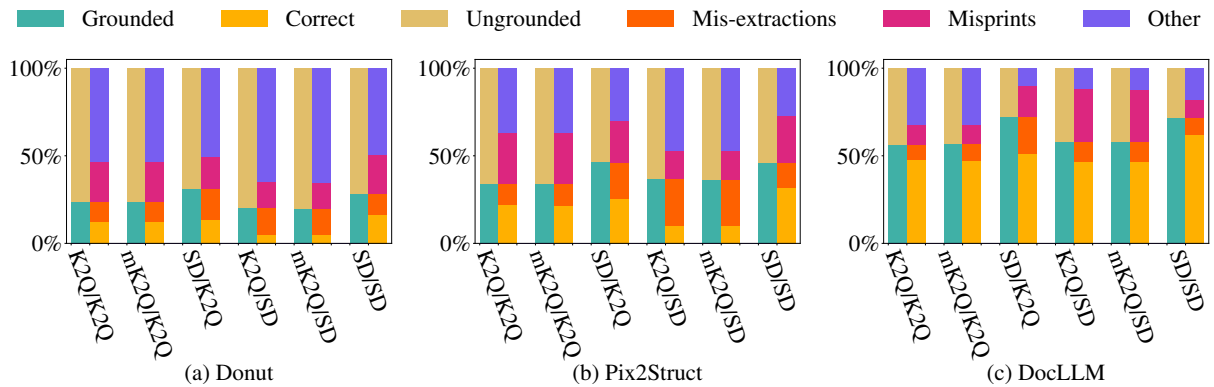


Figure 15: Detailed breakdown of groundedness and error types for Docile using different training / testing datasets.

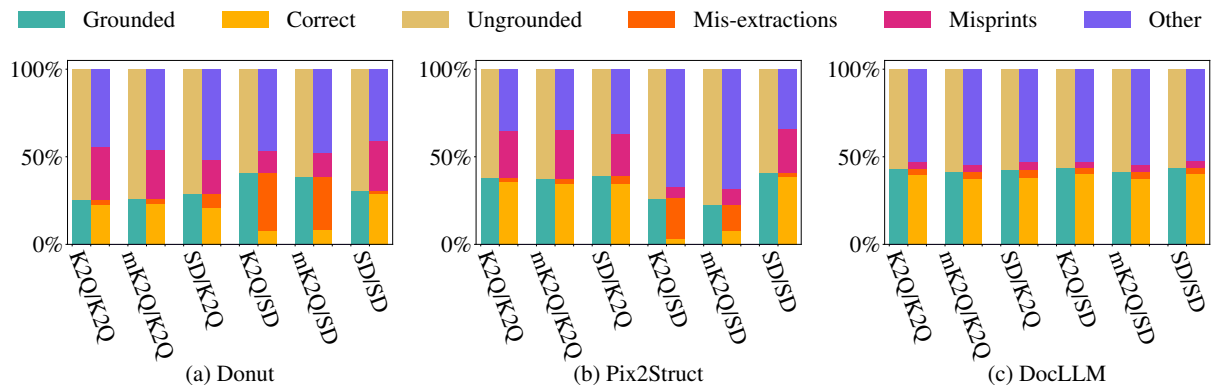


Figure 16: Detailed breakdown of groundedness and error types for Reg. Form using different training / testing datasets.